

Filed on behalf of Meta Platforms, Inc.

Filed: August 26, 2026

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

META PLATFORMS, INC.,
Petitioner,

v.

CURIOXR, INC.,
Patent Owner.

Case No. IPR2026-00460
U.S. Patent No. 11,656,742

**Petition for *Inter Partes* Review of
U.S. Patent No. 11,656,742**

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
II.	MANDATORY NOTICES	1
	A. Real Parties-in-Interest.....	1
	B. Related Matters.....	1
	C. Notice of Counsel and Service Information.....	2
	D. Fee for <i>Inter Partes</i> Review	3
	E. Grounds for Standing	3
III.	IDENTIFICATION OF CHALLENGES.....	3
IV.	BACKGROUND	4
	A. Technology Overview	4
	B. '742 Patent.....	5
	C. Prosecution History	6
	1. The Parent '448 Patent.....	6
	2. '742 Patent	7
V.	PRIOR ART OVERVIEW	8
	A. Tang.....	8
	B. Ravasz.....	9
	C. Mine.....	10
VI.	LEVEL OF ORDINARY SKILL IN THE ART	11

VII. CLAIM CONSTRUCTION.....12

**VIII. GROUND 1: TANG WITH RAVASZ RENDERS OBVIOUS
CLAIMS 1-11 12**

A.	Motivation to Combine	12
B.	Reasonable Expectation of Success	14
C.	Independent Claim 1:	15
1.	1[pre]	15
2.	1[a.i.]	17
3.	1[a.ii]	19
4.	1[a.iii]	20
5.	1[a.iv]	21
6.	1[a.v]	23
7.	1[b.i]	25
8.	1[b.ii]	26
9.	1[b.iii]	28
D.	Dependent Claims 2-7:	31
1.	2	31
2.	3/4	33
3.	5	35
4.	6/7	35
E.	Independent Claim 8:	36
1.	8[pre]	36

2.	8[a.i]	36
3.	8[a.ii]	36
4.	8[a.iii]	40
5.	8[a.iv]	40
6.	8[a.v]	41
7.	8[b.i]	41
8.	8[b.ii]	42
9.	8[c]	43
F.	Dependent Claims 9-11:	43
1.	9	43
2.	10/11	44
IX.	GROUND 2: RAVASZ WITH TANG RENDERS OBVIOUS	
	CLAIMS 1-11	44
A.	Motivation to Combine	44
B.	Reasonable Expectation of Success	45
C.	Independent Claim 1	46
1.	1[pre]	46
2.	1[a.i.]	48
3.	1[a.ii]	48
4.	1[a.iii]	49
5.	1[a.iv]	51
6.	1[a.v]	53

7.	1[b.i]	57
8.	1[b.ii]	58
9.	1[b.iii]	59
D.	Dependent Claims 2-7:	61
1.	2	61
2.	3/4	62
3.	5	63
4.	6/7	63
E.	Independent Claim 8:	64
1.	8[pre]	64
2.	8[a.i]	64
3.	8[a.ii]	64
4.	8[a.iii]	67
5.	8[a.iv]	67
6.	8[a.v]	68
7.	8[b.i]	69
8.	8[b.ii]	70
9.	8[c]	71
F.	Dependent Claims 9-11:	71
1.	9	71
2.	10/11	72

X.	GROUND 3: RAVASZ WITH TANG AND MINE RENDERS	
	OBVIOUS CLAIMS 12-20	72
A.	Motivation to Combine	72
B.	Reasonable Expectation of Success	74
C.	Independent Claim 12:	75
1.	12[pre]	75
2.	12[a]	76
3.	12[b]	80
4.	12[c]	81
5.	12[d]	83
6.	12[e]	84
D.	Dependent Claims 13-20:.....	85
1.	13.....	85
2.	14.....	85
3.	15/19/20	86
4.	16.....	86
5.	17.....	86
6.	18.....	87
XI.	CONCLUSION	88

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>Caption Health, Inc. v. The Univ. of British Columbia</i> , IPR2025-01422, Paper 15 (Director Dec. 18, 2025)	12
<i>CurioXR, Inc. v Meta Platforms, Inc.</i> , 7:26-cv-00123-DC-DTG (W.D Tex)	1
<i>Dental Monitoring SAS v. Align Tech., Inc.</i> , No. 2025-1752, 2026 WL 2291181 (Fed. Cir. 2026)	11
<i>Dynamic Drinkware, LLC v. National Graphics, Inc.</i> , 800 F.3d 1375 (Fed. Cir. 2015)	11
<i>Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co. Ltd.</i> , 868 F.3d 1013 (Fed. Cir. 2017)	12
<i>Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.</i> , 200 F.3d 795 (Fed. Cir. 1999)	12, 14, 16
Statutes	
35 U.S.C. § 103	3
Other Authorities	
37 C.F.R. § 42.10(b)	3
37 C.F.R. §42.24(a).....	6
37 C.F.R. § 42.24(d)	6

TABLE OF EXHIBITS

Exhibit No.	Description
1001	U.S. Patent No. 11,656,742 to Fieldman (“the ’742 Patent”)
1002	File History of U.S. Patent No. 11,656,742
1003	Declaration of Gregory Welch (“Welch Decl.”)
1004	Curriculum Vitae of Gregory Welch
1005	U.S. Patent Application Publication No. 2020/0225830 to Tang et al. (“Tang”)
1006	U.S. Patent Application Publication No. 2025/0013302 to Mine et al. (“Mine”)
1007	U.S. Patent Application Publication No. 2020/0387228 to Ravasz et al. (“Ravasz”)
1008	RESERVED
1009	Mark R. Mine, Virtual Environment Interaction Techniques, UNC Chapel Hill Computer Science Technical Report TR95-018, May 5, 1995
1010	ASME, The Link Flight Trainer, June 10, 2000
1011	Royal Air Force Museum, Flight Simulation and Synthetic Trainers
1012	Irving C. Statler, NASA, Characteristics of Flight Simulator Visual Systems, April 1981

Exhibit No.	Description
1013	Dean Kocian, A Visually-Coupled Airborne System Simulator, 1977
1014	Aurale R. DeRuyck, An Electromagnetic Position Sensor, 1973
1015	Sutherland, A Head-Mounted Three Dimensional Display, 1968
1016	Bishop, The Self-Tracker: A Smart Optical Sensor on Silicon, 1984
1017	General Electric, Modifications to Interim Visual Spaceflight Simulator, 1966
1018	Parisi, Learning Virtual Reality, November, 2015
1019	U.S. Patent No. 2,955,156 to Heilig ("Heilig")
1020	TechCrunch, HoloLens Hands-On: How We Built An App for Microsoft's Augmented Reality Headset, May 1, 2015
1021	Paul James, RoadtoVR, Watch Halo Played on Microsoft HoloLens in Augmented Reality, December 7, 2015
1022	Hayden Dingman, Five Years of VR: A Look at the Greatest Moments from Oculus Rift to Quest 2, March 29, 2021
1023	Brian Crecente, Oculus Rift S, Oculus Quest Hit May 21, Pre-Orders Open, April 30, 2019
1024	Oculus Quest Basics Tutorial, https://www.youtube.com/watch?v=pVFcsedWsKE , May 20, 2019 (last accessed Aug. 18, 2026)

Exhibit No.	Description
1025	Jason Jerald, The VR Book: Human-Centered Design for Virtual Reality, 2016
1026	NSA, Virtual Reality, 1992
1027	Billinghurst, A Survey of Augmented Reality, March, 2015
1028	Mark R. Mine, Exploiting Proprioception in Virtual-Environment Interaction, 1997
1029	Mark R. Mine, ISAAC: A Meta-CAD System for Virtual Environments, 1997
1030	Mark R. Mine, Working in a Virtual World: Interaction Techniques Used in the Chapel Hill Immersive Modeling Program, 1996
1031	Ben Schneiderman, Designing Menu Selection Systems, 1986
1032	Github, MixedRealityToolkit-Unity, August 21, 2019
1033	Github, MixedRealityToolkit-Unity, README.md
1034	Github, MixedRealityToolkit-Unity, README_Button.md
1035	Github, MixedRealityToolkit-Unity, Pointers
1036	Frederick P. Brooks, Jr., Interactive Graphics for Molecular Graphics System, UNC Chapel Hill Department of Computer Science, March 1980
1037	Frederick P. Brooks, Jr., Project GROPE – Haptic Displays for Scientific Visualization, SIGGraph Comput. Graph., 1990

Exhibit No.	Description
1038	RESERVED
1039	David M. Ewalt, In Pictures: The Nintendo Wii, Forbes, November 14, 2006
1040	Nintendo, Wii Operations Manual: System Setup, 2009
1041	RESERVED
1042	Brian Barrett, How Oculus Plans to Fix One of VR's Biggest Problems, WIRED, June 2015
1043	Myron W. Krueger, Responsive Environments, Proceedings of the June 13-16, 1977, National Computer Conference
1044	Myron W. Krueger, VIDEOPLACE - An Artificial Reality, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1985
1045	Sam Machkovech, Surprise! Oculus Quest Becomes First VR Set with Native Hand Tracking--This Week, Ars Technica, December 2019
1046	Ben Lang, Oculus Quest Hand-tracking Starts Rolling Out This Week, Developer SDK Next Week, RoadtoVR, December 2019
1047	JP Patent Application No. 2022-049042 Certified Translation
1048	Declaration of Stephanie Box, Certification of English Translation of JP Patent Application No. 2022-049042
1049	U.S. Patent No. 11,531,448 to Fieldman (“the ’448 Patent”)
1050	File History of U.S. Patent No. 11,531,448

Exhibit No.	Description
1051	Hyunyoung Song et al., Grips and Gestures on a Multi-Touch Pen, CHI 2011. May 7-12, 2011, Vancouver, BC, Canada
1052	Munehiko Sato et al., Touché: Enhancing Touch Interaction on Humans, Screens, Liquids, and Everyday Objects, CHI 2012, May 5-10, 2012, Austin, Texas, USA
1053	HyeonBeon Yi et al., DexController : Designing a VR Controller with Grasp Recognition for Enriching Natural Game Experience, VRST 2019, November 12-15, 2019, Parramatta, NSW, Australia
1054	James P. Williams and Greg Welch, The Pressure Sensitive Touch-pad, Purdue University, E.E.T. 454 Project Report, April 30, 1985
1055	Greg Welch and James P. Williams, The Easy Chair, May 5, 1986
1056	Chris Kohler, Review: Oculus Touch, WIRED, Dec. 5, 2016
1057	GitHub – Initial Commit ValveSoftware/IndexHardware, June 25, 2019
1058	Matt Gottsacker, From One World to Another: Interfaces for Efficiently Transitioning Between Virtual Environments, CHI 2026, April 13-17, 2026, Barcelona, Spain

I. INTRODUCTION

Meta Platforms, Inc. (“Petitioner”) respectfully requests *inter partes* review of claims 1-20 of U.S. Patent No. 11,656,742 (EX1001, “’742 Patent”).

By the 2022 priority date, providing separate selection techniques for near and far objects were basic and well-known. For example, Tang (EX1005) described a “near interaction mode” and “far interaction mode” for selecting and interacting with objects. Yet the Examiner allowed the challenged claims based on this very feature.

Accordingly, the ’742 Patent should not have issued. Meta respectfully requests the Board correct this Examiner error and find all the challenged claims unpatentable.

II. MANDATORY NOTICES

A. Real Parties-in-Interest

The real parties-in-interest are Meta Platforms, Inc. and Meta Platforms Technologies, LLC.

B. Related Matters

U.S. Patent Office records indicate that the ’742 Patent is assigned to CurioXR, Inc. (“Patent Owner”), which is currently asserting the ’742 Patent in the following concurrent litigation filed on April 2, 2026: *CurioXR, Inc. v Meta Platforms, Inc.*, 7:26-cv-00123-DC-DTG (W.D Tex).

There are two patents asserted in the concurrent litigation. Petitioner has filed a petition for *inter partes* review against U.S. Patent No. 11,531,448 (IPR2026-00459).

C. Notice of Counsel and Service Information

Petitioner provides the following designation of counsel:

Lead Counsel	Lisa K. Nguyen (Reg. No. 58,018) Paul Hastings LLP 1117 S. California Avenue Palo Alto, CA 94303 Telephone: 650.320.1800 Fax: 650.320.1900 Email: PH-META-Curio@paulhastings.com
Back-Up Counsel	David Tennant (Reg No. 48,362) Paul Hastings LLP 2050 M. Street NW Washington, D.C. 20036 Telephone: 202.551.1997 Email: PH-META-Curio@paulhastings.com
Back-Up Counsel	Kamilah Alexander (Reg. No. 57,210) Paul Hastings LLP 4655 Executive Drive, Suite 350 San Diego, CA 92121 Telephone: 858.458.3000 Fax: 858.458.3107 Email: PH-META-Curio@paulhastings.com

Back-Up Counsel	Michelle Zhang (<i>pro hac vice</i> to be filed) Paul Hastings LLP 1117 S. California Avenue Palo Alto, CA 94303 Telephone: 650.320.1800 Fax: 650.320.1900 Email: PH-META-Curio@paulhastings.com
Back-Up Counsel	Allison Greubel (Reg. No. 82,758) Paul Hastings LLP 1117 S. California Avenue Palo Alto, CA 94303 Telephone: 650.320.1800 Fax: 650.320.1900 Email: PH-META-Curio@paulhastings.com

Pursuant to 37 C.F.R. § 42.10(b), a Power of Attorney accompanies this petition. Petitioner consents to electronic service by e-mail.

D. Fee for *Inter Partes* Review

The Director is authorized to charge the fee due to Deposit Account 50-2613.

E. Grounds for Standing

The '742 Patent is available for *inter partes* review and no bars or estoppels apply.

III. IDENTIFICATION OF CHALLENGES

Ground	Claim	Statute	Prior Art
1	1-11	35 U.S.C. § 103	Tang in view of Ravasz
2	1-11	35 U.S.C. § 103	Ravasz in view of Tang
3	12-20	35 U.S.C. § 103	Ravasz in view of Tang and Mine

IV. BACKGROUND

A. Technology Overview

Wearable virtual reality (“VR”) systems date back to the 1960s, when early pioneers invented HMDs to provide immersive and realistic simulations. EX1003, ¶¶33-40. As technology developed, VR became more immersive, allowing users to interact with virtual objects. *Id.*, ¶¶41-69. In 1985, Myron Krueger developed Videoplace which used video cameras to detect and respond to user movements including gestures, providing the foundation for modern body-tracking. *Id.*, ¶¶68-69. Concurrently, physical controllers that allowed users to control and interact with virtual objects with finer precision developed. *Id.*, ¶¶54-68.

By the 1990s, it was well-known to provide a user with different interaction modalities to select or activate buttons, including hand gestures and controllers. EX1030, 4-12; EX1003, ¶¶70-91. In 1995, Mark Mine described fundamental forms of interactions in a virtual world including object selection. Notably, Dr. Mine recognized:

There are two primary selection techniques: *local* and *at-a-distance*. In a local technique, the desired object is within reach, and the user can interact with it directly. When the user cannot reach an object, action-at-a-distance selection is required.

EX1009. In 1996, Dr. Mine taught the use of a controller to select menu items and emphasized that this controller approach could be advantageous over the use of a fingertip when the menu is beyond the reach of the user. EX1030, 9; EX1003, ¶80.

B. '742 Patent

The '742 Patent is directed to well-known interfaces for selecting applications in VR environments, describing the claimed interfaces and methods in just over four columns. EX1001, 1:61-67.

The '742 Patent complains “[i]n conventional VR systems, such as the Meta Quest 2 platform, users can use either a physical controller or their hands to select and activate applications...in an application selection interface.” *Id.*, 1:27-31. In such systems, “when the user is using a physical controller they move the controller or use buttons, joystick, scrolling or similar input controls on the controller to select and launch an application.” *Id.*, 1:31-34. In contrast, “[i]f a user is using their hands to interact with the application interface...the user can make a pinch or similar pointing gesture to simulate laser pointing from the fingertips of the hand...for selecting an application for launch from the display.” *Id.*, 1:38-45.

An alleged problem with these conventional systems is that when selecting an application with a hand, “the laser pointing effect can be slow to be recognized or not even recognized, difficult to quickly and accurately choose an application for launch, and be less intuitive for moving between application selection,” particularly

when an icon to be selected appears to be far away from the hand. *Id.*, 1:51-55. The '742 Patent addresses this problem by using a hand to interact with icons that appear closer and a physical controller to interact with icons that appear farther.

“When a user initially does not use or switches away from using a physical controller, to use a user’s hands..., [an] application selection interface that is better and more intuitively adapted for hand interaction and control is displayed to the user.” *Id.*, 2:15-20. The “different application selection interface for use with a user’s hands in a VR environment...may be presented to appear to be directly acted upon by the user hands, such as pushing or pulling the selected application with a hand to launch the application.” *Id.*, 2:21-28.

C. Prosecution History

1. The Parent '448 Patent

The original claim 1 of the '448 Patent was generally directed to application selection interface comprising a group of icons in which one icon launches a software application by pushing, pulling, tapping, pressing or sliding action of a body part of the user directly on the icon. EX1050, 24. On July 26, 2022, the Examiner issued a non-final office action, finding claim 1 anticipated and claims 1-20 obvious. *Id.*, 41-65.

On October 18, 2022, the Applicant responded by amending claim 1 to require “a first grouping of a plurality of software application icons...appears closer...than

another second grouping.” *Id.*, 284. The Applicant emphasized “[t]he proposed amendment...highlights that the invention of claim 1 is directed to the interface of the first grouping of icons...**appearing closer** to the user’s body part that is tracked by a camera, as compared to...the second grouping of icons displayed as an application interface when a physical controller is used by the user.” *Id.*, 289.

On November 2, 2022, the Examiner issued a Notice of Allowance. *Id.*, 308. The PE2E Prior Art Search History confirms that the Examiner did not perform any searches targeted at this additional feature with no searches including the terms “closer,” “near,” “far,” “local,” “distant,” or equivalent terms. *Id.*, 315-16.

2. ’742 Patent

There was no substantive review of the ’742 Patent during prosecution. The PE2E Prior Art Search History confirms that the Examiner, again, did not perform any searches targeted at this additional feature.

The Examiner allowed the claims after the Applicant agreed to amend “the first application software” to “the first application software icon” in claim 1 and to file a terminal disclaimer to overcome a double patenting rejection. EX1002, 229, 314.

V. PRIOR ART OVERVIEW

A. Tang

Prior art under §§102(a)(1) and 102(a)(2), Tang discloses every limitation of independent claims 1 and 8 except a *first software application icon operatively linked to launch a respective software application*. Like the '742 Patent, Tang recognizes that virtual objects “beyond the reach of the user...may be difficult or tiresome for the user to control.” EX1005, [0023]. In contrast, a user has “greater ability to make fine movements/selections with respect to a virtual interaction object **58**...in proximity to the user’s hand.” *Id.*, [0029]. Thus, Tang discloses a VR system implemented through an HMD that allows the user to interact with virtual objects by switching between a “far interaction mode” and “near interaction mode.” *Id.*, Abstract. A user can interact with the virtual objects using “user-input devices such as a...game controller” in far interaction mode, and can use a “gesture, such as a...tap gesture” or “a pressing motion by a finger” in near interaction mode. *Id.*, [0049], [0026], [0035].

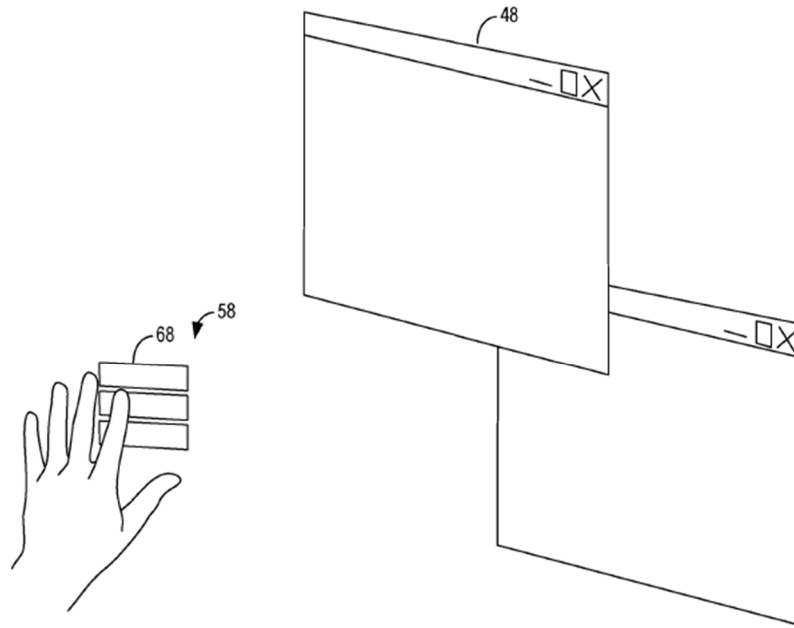


FIG. 7

EX1005, FIG. 7

B. Ravasz

Prior art under §§102(a)(1) and 102(a)(2), Ravasz discloses every limitation of independent claims 1 and 8 except *a first software application icon appears closer to a body part of the user...than a second software application icon*. Like the '742 Patent and Tang, Ravasz discloses “an artificial reality environment” implemented through an HMD. EX1007, Abstract, [0008]. A user can interact with the UI elements within a UI icon array using controllers 114, or the UI elements within a “UI menu 124 [that]...appear[s] to be larger, and thus closer to the user” using pre-defined gestures. *Id.*, [0087].

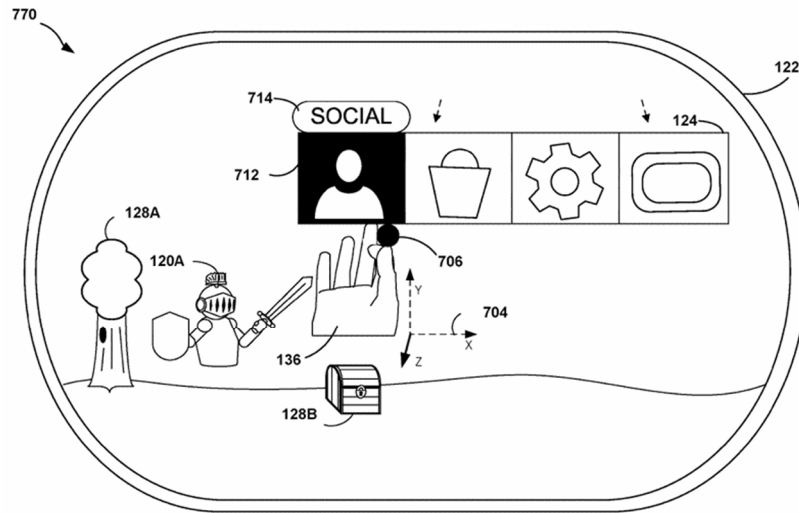


FIG. 7E

EX1007, FIG. 7E

C. Mine

Mine discloses detecting the controller is not in use as required by independent claim 12. Like Ravasz and Tang, Mine discloses “a virtual reality (VR) technique, which make a user feel a space different from a real space using a head-mounted display (HMD).” EX1006, [0003], [0024]; EX1047, [0002], [0013]. A user can “control [] selecting CG [computer graphics] or moving the position of CG in the composite image displayed on the HMD” using controller 120 or gestures. EX1006, [0043]; EX1047 [0032]. Mine discloses that the controller 120 “includes a button enclosing an optical trackpad ([] ‘OTP’)” and can “determine[] whether the hand of the user is in contact with the OTP” and therefore whether the controller 120 is in use. EX1006, [0028], [0072]; EX1047, [0017], [0061]; FIG. 6.

Mine is prior art under §102(a)(2). Mine was “effectively filed” on March 24, 2022 (see EX1047), which is before the “effective filing date,” June 1, 2022, of the ’742 patent. Mine’s Japanese Priority Application fully supports claim 1 of Mine shown by the exemplary citations below. *See* EX1003,¶113; *see also Dynamic Drinkware, LLC v. National Graphics, Inc.*, 800 F.3d 1375, 1382-83 (Fed. Cir. 2015); *Dental Monitoring SAS v. Align Tech., Inc.*, No. 2025-1752, 2026 WL 2291181, *4 (Fed. Cir. 2026).

Mine, Claim 1 (EX1006)	Mine Priority (EX1047)
[1.pre]	[0007], [0013], [0014], Fig. 1, claim 1
[1.a]	[0044], Fig. 5
[1.a.1]	[0007], [0044], Fig. 5, claim 1
[1.a.2]	[0079]
[1.a.2.i]	[0007], [0039], [0055], [0058], Fig. 5, claim 1
[1.a.2.ii]	[0007], [0033], claim 1
[1.b.1]	[0015], [0017], [0033], Fig. 1, claim 12
[1.b.2]	[0015], [0016], claim 12

VI. LEVEL OF ORDINARY SKILL IN THE ART

A person of ordinary skill in the art (“POSITA”) in 2022 would have had at least a bachelor’s degree in computer science, computer engineering, or a similar technical field, and relevant experience working with VR and AR systems. Additional experience could substitute for less education, and additional education could likewise substitute for less experience. EX1003,¶¶18-20.

This Petition does not turn on this precise definition. *Id.*

VII. CLAIM CONSTRUCTION

This Petition establishes the prior art meets each of the claim limitations under its plain and ordinary meaning.¹ Thus, the claim terms require no express construction. *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co. Ltd.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017); *Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999).

VIII. GROUND 1: TANG WITH RAVASZ RENDERS OBVIOUS CLAIMS 1-11

A. Motivation to Combine

A POSITA would have been motivated to combine Tang's virtual object interface with Ravasz's application-launching and configurable-settings menu functionality. EX1003, ¶¶117-131.

First, both are directed to the user experience in VR and address the same user control design problem. EX1005, [0016], [0021]-[0026], [0049], [0052], [0059]; EX1007, [0004]. A POSITA implementing Tang's VR system would naturally look

¹ Petitioners reserve the right to provide an express construction consistent with that applied in this proceeding and the Parallel Litigation. *See Caption Health, Inc. v. The Univ. of British Columbia*, IPR2025-01422, Paper 15, 3 (Director Dec. 18, 2025).

at other VR systems that focus on the user experience, like Ravasz, to improve the implementation. EX1003, ¶¶118, 121.

Second, both disclose systems that include similar components. EX1003, ¶¶122-131. Tang discloses the ability to display objects closer to the user for ease of interaction, particularly with gestures. EX1005, [0016]. Ravasz similarly discloses “artificial reality systems...that generate and render graphical user interface elements for display to a user in response to...pre-defined gestures by the user” that would be easier to use on closer UI elements. EX1007, [0005]. Therefore, a POSITA implementing Tang would look to other references that disclose virtual objects, such as Ravasz, to improve gesture interaction with the virtual objects. EX1003, ¶122.

Third, a POSITA implementing the launching of application from another application and buttons or context menu of Tang, would naturally look at similar references like Ravasz that provide detail on what actions are performed when the virtual objects in the buttons or context menu are tapped or pressed. EX1003, ¶¶119-120, 123-130. Tang discloses that “different programs may be instantiated from the same application.” EX1005, [0047]; *see also* [0017], [0041]. As shown in Figures 5C and 5F of Tang, the virtual objects in the grouping of buttons and context menu are literal black boxes. Ravasz discloses that the UI menu can include modifying system settings or launching applications (EX1007, FIGs. 7A-

7G), including a “social” software application, which Tang and Ravasz both share (EX1005, FIG. 6; EX1007, FIG. 7E). The combination would have increased versatility, lowered friction, improved workflow, and enhanced the user experience, as well as supported a user’s ability to multitask. EX1003, ¶129; *see also* EX1007, [0049].

B. Reasonable Expectation of Success

A POSITA would have had a reasonable expectation of success in modifying Tang’s virtual objects with Ravasz’s application-launching and configurable-settings menu functionality. EX1003, ¶¶132-137.

First, both use similar VR hardware components, making the combination straightforward. *Id.*, ¶¶133-135. Tang discloses outward-facing cameras, gesture recognition module, and inertial measurement units to determine gestures. EX1005, [0020], [0017], [0019]. Ravasz discloses image capture devices, gesture detector, and motion sensors to determine user gestures. EX1007, [0008], [0045]. Because both utilize similar components in a similar way to detect interaction, the systems would use similar inputs and outputs. EX1003, ¶¶133-135. Specifically, Tang includes all the hardware components needed to render and interact with virtual objects to launch applications and modify settings as taught by Ravasz. *Id.*

Second, both disclose similar gestures to interact with objects/elements, making the combination just a matter of programming. *Id.*, ¶136. Tang discloses the

gesture recognition module can detect predetermined gestures to interact with objects. EX1005, [0018], [0026], [0029]. Similarly, Ravasz discloses the gesture detector can detect pre-defined gestures that can be used to interact with the UI menu. EX1007, [0063], [0064]. Associating Tang’s virtual objects with Ravasz’s application-launching or configurable settings menu functionality would have already been within Tang’s software structure, and a POSITA’s skillset. EX1003, ¶136.

Therefore, a POSITA would have understood that Tang’s system already included all the necessary hardware components, inputs and outputs, and software structure to implement Ravasz’s ability to launch applications and modify settings. *Id.*, ¶137.

C. Independent Claim 1:

1. 1[pre]

To the extent limiting, Tang discloses the preamble. EX1003, ¶¶138-145. Tang discloses a *virtual reality environment* that displays virtual objects to allow a user to select those objects (*selection interface*).

Tang’s system provides a “virtual reality (VR) environment [that] may be populated with an array of virtual objects with which a user may interact.” EX1005, [0002]. Tang describes “a computing system **10**” that “include[s] a head mounted display (HMD) device **14**” further “configured to display a virtual object **48**...on the

display 16” (*interface*). EX1005, [0017], [0021]; *see also* [0038]; EX1003, ¶¶139-140.

Tang discloses “a plurality of control points associated with the virtual object.” EX1005, Abstract. When “the control points associated with the virtual object are further than a predetermined threshold distance from the user,” Tang’s system will “invoke a far interaction mode for the virtual object.” *Id.* Tang discloses that “in the far interaction mode the available interactions with the virtual object 48 may be limited, and in some situations may only include...display of a far context menu due to the distance at which the virtual object is displayed.” *Id.*, [0022]. A user can “*select* the virtual object 48” (*selection interface*). *Id.*, [0026]; *see also* [0022]-[0025], [0029], FIG. 6; EX1003, ¶¶141-143.

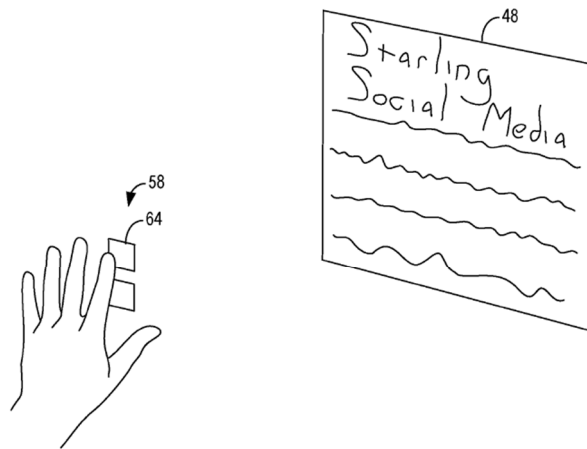


FIG. 6

EX1005, FIG. 6

Tang also discloses “a near interaction mode,” where the “virtual interaction object [is] within the predetermined threshold distance from the user.” EX1005, [0003]. A “processor 12 may be configured to execute control of the virtual object 48 according to gestures of the hand of the user.” *Id.*, [0026]; *see also* [0027]. “The predetermined gesture may be a...tap gesture.” *Id.*, [0026]. “The plurality of gestures may further include a pressing motion by a finger, a circular motion by the hand or a finger, a wrist flick, a rotation of the hand and/or arm, and/or a snap, to name a few additional examples.” *Id.*, [0028]; EX1003, ¶144.

2. 1[a.i.]

Tang with Ravasz renders obvious this limitation. EX1003, ¶¶146-152. Tang’s virtual object (virtual objects 48 and virtual interaction objects 58) among virtual buttons 64 or context menu 68, when combined with Ravasz’s application-launching menu functionality, is *a first software application icon*. *Id.*, ¶147.

Tang discloses a “**plurality** of virtual interaction objects 58” that are displayed to the user including “virtual button[s] 64” or “context menu 68” (*see* EX1005, [0028]), depicted as a *grouping* in Figures 5C and 5F. *See also* EX1005, [0053], FIG. 7; EX1003, ¶¶148-149.



FIG. 5C



FIG. 5F

EX1005, FIGs. 5C, 5F

Ravasz discloses that menu items (akin to Tang’s grouping of virtual buttons 64/context menu 68) can “correspond to applications” and that “icons representing applications available or running in the artificial reality system” (*software applications icons*) may be displayed to the user. EX1007, [0089]. Ravasz further discloses that upon detecting a selection gesture, the system may “cause an application to be instantiated, or can cause a currently running application to be brought into the foreground.” EX1007, [0076]. EX1003, ¶150. Specifically, Ravasz teaches that “in response to gesture detector 424 detecting one of the pre-defined gestures” on a UI element, “launch[ing] *applications*.” EX1007, [0063].

A POSITA would have modified Tang’s virtual objects to be software applications icons to launch respective applications as taught by Ravasz, resulting in *a first software application icon*. EX1003, ¶151; *see also* Sections VIII.A-B.

3. 1[a.ii]

Tang discloses this limitation. EX1003,¶¶153-157. The application-launching virtual objects, among virtual buttons 64 or context menu 68 (discussed in Section VIII.C.2), are *displayed in a virtual reality environment to a user using* Tang's HMD 14 (*virtual reality hardware device*). EX1005, [0020]; EX1003,¶154.

Tang's system provides a “virtual reality (VR) environment [that] may be populated with an array of virtual objects with which a user may interact.” EX1005, [0002]. This system “may take the form of one or more...wearable computing devices such as...head mounted augmented/virtual reality devices,” such as “head mounted display (HMD) device 14.” *Id.*, [0039], [0017], [0021], [0025], FIG. 2; EX1003,¶¶155-156.

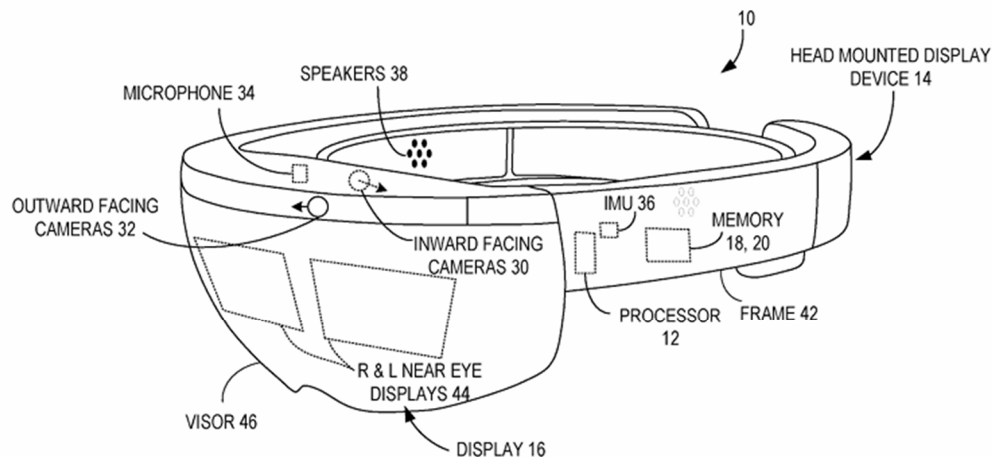


FIG. 2

EX1005, FIG. 2

4. 1[a.iii]

Tang discloses this limitation. EX1003, ¶¶158-167. The application-launching virtual objects, among virtual buttons 64 and context menu 68 (discussed in Section VIII.C.2) (*wherein the first software application icon*) are displayed within a predetermined threshold distance (*appears closer*) from the user, including the user's hand (*to a body part of the user*). EX1003, ¶159.

Tang discloses a near interaction mode in which the virtual objects among the virtual buttons or context menu are displayed within a predetermined threshold distance from the user, and a far interaction mode in which virtual objects in a far context menu are displayed further than the predetermined threshold distance from the user. EX1005, [0025] (“configured to invoke a near interaction mode and display a virtual interaction object 58 *within* the predetermined threshold distance 54 from the user”), [0033] (in the “far interaction mode,” “one or more of the control points 52 associated with the virtual object 48 are *further* than a predetermined threshold distance 54 from the user.”); EX1003, ¶¶160-161.

Tang discloses that the predetermined threshold distance is “*with respect to various parts of the user*,” and “measured from [] [any] point on the body of the user” (*body part of the user*). EX1005, [0021], [0031]. A POSITA would have understood the virtual objects (*the first software application icons* discussed in Section VIII.C.2) are displayed “*within* the predetermined threshold distance” and

would *appear closer to the body part of the user* than virtual objects displayed “further” from the user’s body as in Figure 7. EX1003, ¶¶162-163.

In Figure 7, Tang describes a VR environment in which a user invokes the near interaction mode to display virtual interaction objects 58 “*near the user’s hand*” (*closer to a body part of the user*). EX1005, [0029], FIG. 7. Tang explains that the “advantage” of near interaction mode includes “improved readability” and “greater ability to make fine movements/selections with respect to a virtual interaction object” “*in proximity to the user’s hand.*” EX1005, [0029]; *see also* [0031]; EX1003, ¶¶164-166.

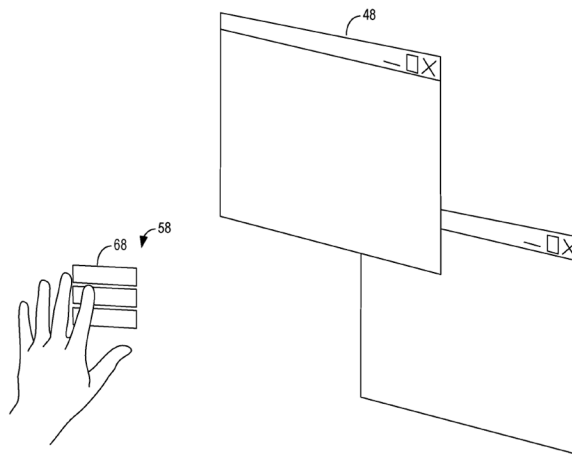


FIG. 7

EX1005, FIG. 7

5. 1[a.iv]

Tang discloses this limitation. EX1003, ¶¶168-177. The user’s hand (*a body part of the user*) is tracked by outward-facing cameras 32 (*a camera*) in HMD 14 (*of*

the virtual reality hardware device) to recognize gestures (*to interact*) associated with the virtual objects, among virtual buttons 64 and context menus 68 (*first software application icon*). *See id.*, ¶169; Section VIII.C.2.

Tang discloses cameras incorporated into the HMD that track the user's hands to allow the user to interact with the virtual objects. EX1005, [0018] (“The gesture recognition module **24** may be configured to identify poses and gestures expressed by the user by processing image data of the user collected by outward-facing cameras **32** when these cameras capture images of the user's body or portions thereof, such as the user's hands.”); *see also* [0016]-[0020], FIG. 2; EX1003, ¶¶170-173.

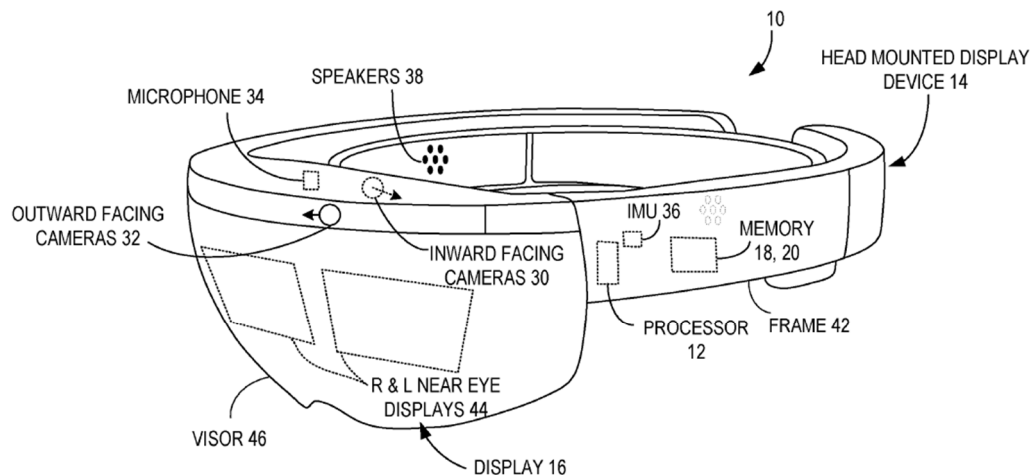


FIG. 2

EX1005, FIG. 2

The cameras and gesture recognition module would have been used to detect and recognize user interaction with the virtual objects (*to interact with the first software application icons*) to allow the user to select a virtual object among virtual buttons 64 or context menu 68 as in FIGs. 5 and 7. *See* EX1005, [0029]; *see* Section VIII.C.2; EX1003, ¶¶174-176.

6. 1[a.v]

Tang with Ravasz renders obvious this limitation. EX1003, ¶¶178-188. Tang's virtual object in a far context menu in the far interaction mode with Ravasz's application-launching functionality (*another second software application icon*) is *displayed to the user when the user is using a user-input device (physical controller) in the virtual reality environment*. EX1003, ¶179.

Tang teaches “in the far interaction mode the available interactions with the virtual object 48 may be limited, and in some situation may only include selection...and display of a far context menu.” EX1005, [0022]. As Tang explains, this is “due to the distance at which the virtual object is displayed.” *Id.* As discussed, Tang does not expressly teach that the virtual object in the far context menu launches an application, but Ravasz renders this obvious. EX1007, [0089] (menu items “correspond to applications” and UI menu includes “icons representing applications available or running in the artificial reality system”); *see also* Section VIII.C.2. A POSITA would have been motivated to implement Tang's virtual object

in a far context menu to include Ravasz's application-launching functionality (*second software application icon*) because Ravasz provides the details about menu actions that Tang leaves to a POSITA to implement. *See* Sections VIII.A-B; EX1003, ¶¶180-184.

Tang discloses an “input subsystem 310” that interfaces with “one or more user-input devices such as a keyboard, mouse, touch screen, or game controller” (*physical controllers*). EX1005, [0049]. Using these user-input devices in the “far context menu” (*id.*, [0022]) (*when the user is using a physical controller in the virtual reality environment*), as opposed to their hands, is necessary because “a significant number of control points 52 [of the virtual object 58] are beyond the reach of the user and thus may be difficult or tiresome for the user to control.” EX1005, [0023]; EX1003, ¶¶185-186. Some gestures used to “access[] a context menu, may even be impossible in situations where virtual content is displayed at a distance beyond the reach of a user.” EX1005, [0016]; *see also* [0022].

Tang discloses [*first software application icon appears closer*] *than another second software application icon* because the virtual object among the buttons or context menu displayed to the user in a near interaction mode (*first software application icon*) is within a predetermined threshold distance from the user's body part whereas the virtual object in the far context menu (*second software application icon*) is further than the predetermined threshold distance. *See* Section VIII.C.4;

EX1003,¶187.

7. 1[b.i]

Tang with Ravasz renders obvious this limitation. EX1003,¶¶189-196. Tang’s virtual object among the group of buttons 64 or context menu 68, with Ravasz’s application-launching functionality is the *first software application icon operatively linked to launch a respective software application*. See Section VIII.C.2; EX1003,¶190.

Like Tang’s virtual object, Ravasz discloses a user interface (UI) element. Ravasz teaches that “in response to gesture detector 424 detecting one of the pre-defined gestures,” such as a “selection gesture” on a UI element in a UI menu, “user interface engine 428 and/or application engine 440 may...**launch applications**.” EX1007, [0063]-[0064]; *see also* [0046]; [0056]. Ravasz further discloses that menu items can “correspond to applications” and that detecting a selection gesture may “cause an application to be instantiated, or can cause a currently running application to be brought into the foreground.” EX1007, [0089], [0076]; EX1003,¶¶191-192.

A POSITA would have been motivated to combine Tang’s virtual object with Ravasz’s functionality because this would have broadened the capabilities of Tang’s system in a natural way. See Section VIII.A; EX1003,¶¶193-194. Tang discloses that “[o]bjects that respond to user interaction in an intuitive and user-friendly manner may provide an improved experience for the user of the AR or V[R] system,”

and introducing application-launching functionality is the type of functionality that a user would expect from a system that allows for user manipulation of virtual objects. EX1005, [0002], [0016]; EX1003, ¶¶193-194.

A POSITA would have had a reasonable expectation of success in making this combination, because both Tang and Ravasz disclose HMDs for use in a virtual reality environment that rely on interactions (gestures or controllers) with virtual objects to manipulate the environment. EX1005, [0003]; EX1007, [0005]; *see also* Section VIII.B; EX1003, ¶195. A POSITA would have understood that the modification to Tang's virtual objects to incorporate application-launching functionality would be a straightforward software modification within the abilities of a POSITA. *Id.*

8. 1[b.ii]

Tang discloses this limitation. EX1003, ¶¶197-206. The software application associated with the virtual object is launched (discussed in Section VIII.C.2) *when* the user's hand (*the body part of the user*) performs a *tapping* or *pressing* action (*interacted with*) *directly* on virtual object among the virtual button 64 or context menu 68 (*first software application icon*). EX1003, ¶198.

Tang's Figures 5-7 depict a user interacting with virtual objects, including virtual buttons 64 and context menus 68, by *pushing*, *tapping*, or *pressing* with their hand (*body part of the user*). EX1005, [0028]-[0029]; EX1003, ¶¶199-202.



FIG. 5C



FIG. 5F

EX1005, FIGs. 5C, 5F

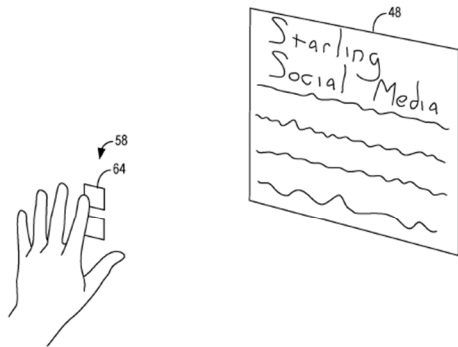


FIG. 6

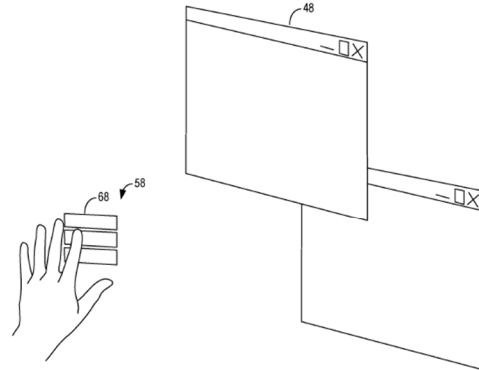


FIG. 7

EX1005, FIG. 6

EX1005, FIG. 7

Tang discloses “an array of virtual objects with which a user may interact.” EX1005, [0002]. Tang explains that “[t]he predetermined gesture [recognized by the gesture recognition module] may be a...tap gesture.” *Id.*, [0026]. “The plurality of gestures may further include a pressing motion by a finger.” *Id.*, [0028]. Tang thus discloses that gestures performed directly on the virtual objects within the predetermined threshold distance include both tapping and pressing actions by the user’s hand (body part of the user). EX1005, [0026], [0028]; EX1003, ¶¶203-205.

A POSITA would have understood that upon a user *tapping* or *pressing* the virtual objects (including *the first software application icon*) in the grouping of buttons or context menu as disclosed by Tang, it would have launched the associated application as disclosed by Ravasz (*launch the software application*). EX1003,¶205.

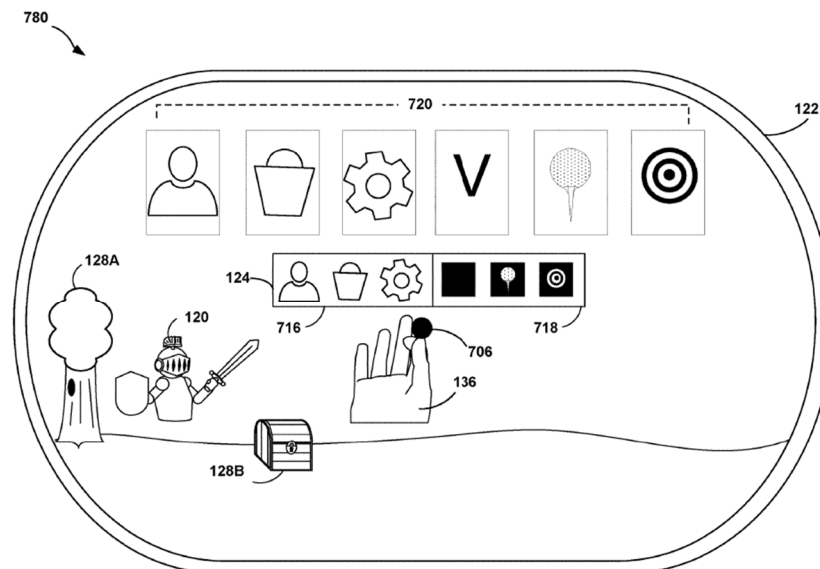
9. 1[b.iii]

Tang alone or with Ravasz renders obvious this limitation. EX1003,¶¶207-214.

Tang discloses that the virtual object among the virtual buttons or context menu in the near interaction mode (discussed in Section VIII.C.2) (*first software application icon*) and the virtual object in the far context menu in far interaction mode (the combination discussed in Section VIII.C.6) (*second software application icon*) can be the same (*see* EX1005, FIGs. 3, 8), and therefore *operatively linked to launch the same respective software application*. EX1003,¶208. Tang teaches that “[a] potential advantage is that once the far interaction mode is invoked, the large virtual object 48 may be selected by the user to become a virtual interaction object 58 and thus be reduced in size for the convenience of the user.” EX1005, [0030]. A POSITA would have understood the large and reduced size versions of the virtual object correspond to the same object or function, because the virtual object is merely changing in size and distance to the user as a result of the user moving from far interaction mode to near interaction mode where the system brings the virtual object

closer to the user for ease of interaction. EX1003, ¶209.

To the extent that Tang does not disclose this limitation, it would have been obvious with Ravasz. EX1003, ¶¶210-213. In Figure 7F, Ravasz illustrates “an example HIVID display 780 illustrating a UI menu 124 and UI icon array 720 in accordance with aspects of the disclosure....The images on the individual icons in icon array 720 can represent a current display of the corresponding application, or an image associated with the application.” EX1007, [0089], FIG. 7F. A POSITA would have understood that the icons in UI menu 124 and UI icon array 720 are two software application icons (*a first and second software application icon*) that are *operatively linked to launch the same application*, because their icon images are the same. EX1003, ¶210.



EX1007, FIG. 7F

A POSITA would have been motivated to combine Tang’s resizing of virtual objects based on a far and near interaction mode with Ravasz’s software application icons. EX1003, ¶211. A POSITA would have been motivated to make this combination because this would have been more convenient for the user, allowing them to interact with the system in the most user-friendly way for a near and far distance—namely, gestures and user-input devices, respectively. EX1005, [0030]; *see* Sections VIII.A, VIII.C.6; EX1003, ¶211.

A POSITA would have had a reasonable expectation of success in making this combination because both references contemplate displaying the same objects at two different distances from the user. EX1005, [0025], [0030], FIGs. 3, 8; EX1007, FIG. 7F, [0089]; EX1003, ¶212. This modification would have been a simple implementation of specific distance values and object size rendering, a mere programming change. EX1003, ¶212.

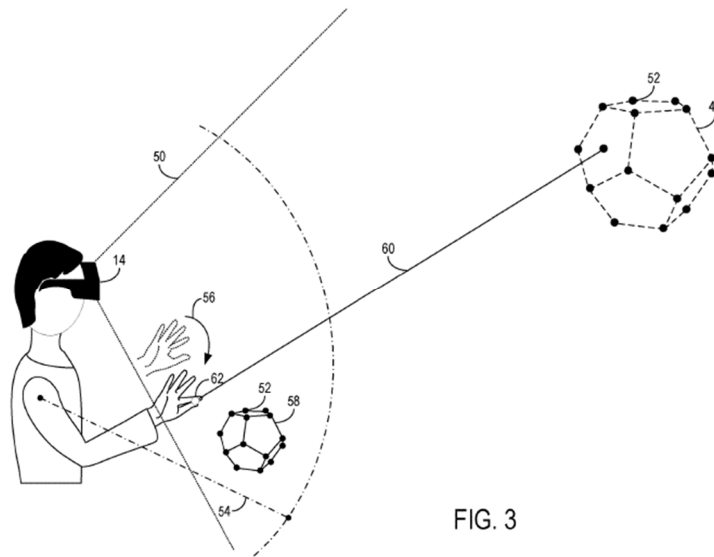


FIG. 3

EX1005, FIG. 3

D. Dependent Claims 2-7:

1. 2

Tang alone or with Ravasz renders obvious this limitation. EX1003, ¶¶215-226. The software application associated with the virtual object launches (discussed in Section VIII.C.2) when the user's hand performs a pressing action (*when pressed by the user's hand or finger*) on a virtual object (*at least one software application icon*). EX1003, ¶216.

Tang depicts and describes the manipulation of three-dimensional virtual objects in FIG. 5, including *pinchable object* 62, handles 70 associated with the *virtual object* 48, and *control widget* 66. Ex. 1005, [0028].



FIG. 5A

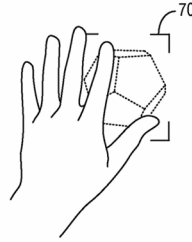


FIG. 5B

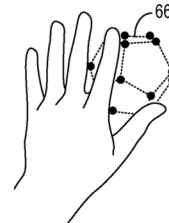


FIG. 5E

EX1003, FIGs. 5A, 5B, 5E

Because Tang discloses three-dimensional virtual objects, a POSITA would have understood and found it obvious to make the virtual objects among the virtual buttons and context menu three-dimensional. EX1003, ¶218. A POSITA would have been motivated to make the buttons three-dimensional because this would have enhanced the user experience by further immersing the user in a virtual world. *Id.*, ¶219. A POSITA would have had a reasonable expectation of success since Tang already discloses that virtual objects could be rendered as three-dimensional, and rendering three dimensional squares and rectangles would have been well within the skill of a POSITA as Tang describes. *Id.*, ¶220.

To the extent that Tang does not explicitly disclose this limitation, this would have been obvious with Ravasz. EX1003, ¶¶221-225.

Ravasz discloses “UI element 126” can be depicted as “two-dimensional or three-dimensional shapes” (*a 3-dimensional button*). EX1007, [0026].

A POSITA would have been motivated to render Tang’s virtual objects as three-dimensional shapes as taught by Ravasz because this would have increased the

type and variety of virtual interaction objects that Tang supports, thus enhancing the user experience. EX1003,¶222. Further, a POSITA would have understood that three-dimensional objects would have increased visual contrast with the background. EX1003,¶¶223-224. A POSITA would have had a reasonable expectation of success in making this combination. EX1003,¶225. Rendering the virtual object as a three-dimensional button would be a simple programming modification within the skills of a POSITA, because Tang already discloses rendering three-dimensional shapes. *Id.*

2. 3/4

Tang alone or with Ravasz renders obvious these limitations. EX1003,¶¶227-233. Tang's virtual objects in a grouping of buttons or context menu that launch applications or modify settings is *a plurality of one or both of software application and virtual reality platform system setting icons that includes the first software application icon*. See Sections VIII.C.2 and VIII.E.3; EX1003,¶228.

Tang further discloses *up and down arrows displayed as buttons adjacent a grouping...wherein the up and down arrows are operatively linked to scrolling the plurality of one or both of software application and virtual reality platform system setting icons in the grouping when directly interacted with by a user's hand or finger* because Tang discloses that virtual content, like the context menu, can be scrolled using buttons as shown in Figure 6. EX1005, [0029], FIG. 6; EX1003,¶229.

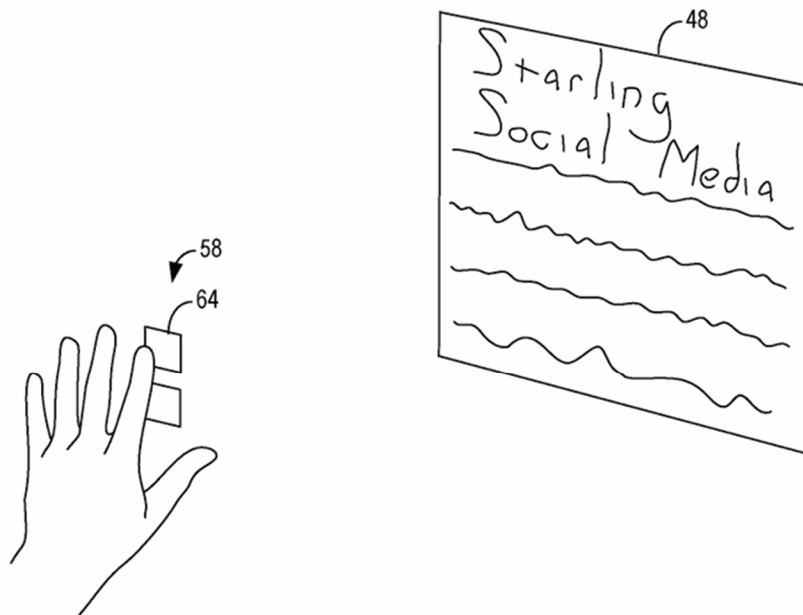


FIG. 6

EX1005, FIG. 6

Tang explains “generat[ing] display of the scrolling mechanism in proximity to the user’s hand...eas[es] the operation of scrolling the virtual content.” *Id.* [0029]. A POSITA would have understood that the *up and down arrows* are *displayed as buttons*, specifically as the top and bottom *buttons* respectively as this is common way to display scrolling buttons to scroll up and down the content in user interfaces. EX1003, ¶230.

To the extent Tang does not disclose *up and down arrows*, this would have been obvious with Ravasz. EX1003, ¶¶231-232. Ravasz discloses “arrows may be used to indicate the direction that the user’s index finger and thumb should be

moved” as a menu prompt. EX1007, [0069]. A POSITA would have been motivated to render Tang’s scrolling buttons as Ravasz’s arrows to provide a concise way for the user to understand how to scroll up and down. EX1003, ¶231. A POSITA would have had a reasonable expectation of success since the use of arrows to indicate which direction to scroll was common and could have been implemented with a simple programming change to render a virtual button as an arrow. EX1003, ¶232.

3. 5

Tang discloses this limitation. EX1003, ¶¶234-238. Tang discloses that the virtual objects among the grouping of buttons or context menu (including those that launch applications or modify settings, the combination discussed in Section VIII.D.2) are displayed within a predetermined threshold distance 54 from the user (*within an apparent distance...to the user*). EX1005, [0031]. Tang discloses that this distance “may be in a range of 460 mm to 580 mm,” representing an “average distance from the shoulder of an average user to the grip center of a hand” (*of a user’s arm length to the user*). *Id.*; EX1003, ¶237.

4. 6/7

Tang with Ravasz renders obvious these limitations. EX1003, ¶¶239-241. Tang’s virtual object with Ravasz’s application-launch functionality among the buttons or context menu in near interaction mode (*first software application icon*) (the combination discussed in Section VIII.C.2) is displayed within a predetermined

threshold distance representing an “average distance from the shoulder of an average user to the grip center of a hand (*displayed within an apparent distance of a user's arm length to the user*). EX1005, [0031]; EX1003, ¶240; *see also* Section VIII.D.3.

E. Independent Claim 8:

1. 8[pre]

Tang discloses this limitation. *See* Section VIII.C.1; EX1003, ¶¶242-244. Tang discloses a *virtual reality environment* that displays virtual objects to a user and allows the user to select those objects (*selection interface*). *Id.*

2. 8[a.i]

Tang with Ravasz renders obvious this limitation. *See* Section VIII.C.2; EX1003, ¶¶245-247. A virtual object among Tang’s virtual buttons 64 or context menu 68 with Ravasz’s application-launching functionality is *a first software application icon*. *Id.*

3. 8[a.ii]

Tang with Ravasz renders obvious this limitation. EX1003, ¶¶248-257. A virtual object among Tang’s buttons 64 or context menu 68 with Ravasz’s configurable settings functionality is *a first virtual reality platform system setting icon*. *Id.*, ¶249.

Ravasz discloses that a user interacts with “a graphical user interface, such as a menu or sub-menu...to operate the *artificial reality system*.” EX1007, [0026].

Thus, the user has the option to control or operate the VR system (*virtual reality platform system*) or software applications within the VR environment. *Id.*; EX1003,¶252.

Ravasz further discloses “configurable” settings, which are *virtual reality platform system settings*. For instance, “the artificial reality system can be configurable (for example, by the user) to determine a configuration of the left hand or the right hand.” EX1007, [0067]; EX1003,¶253. Also, the “user interface engine 428 dynamically generates a user interface...allowing the user 110 to...configur[e] HIVID 112” (EX1007, [0063]), which is a further alteration of the VR system settings. EX1003,¶253. Moreover, the “user interface engine 428 and/or application engine 440” can themselves “modify *configurable settings*” upon detection of particular pre-defined gestures by the user. EX1007, [0063].

Further in FIG. 7B, Ravasz depicts menu item 708 labeled as a “Settings” icon. EX1007, FIG. 7B, [0076]; EX1003,¶254; *see also*, FIG. 7C (menu item illustrated using a traditional settings icon); FIG. 7G (same).

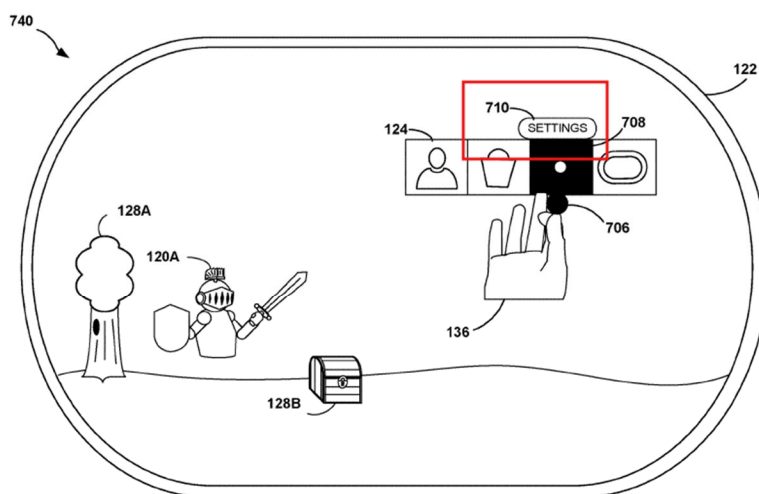


FIG. 7B

EX1007, FIG. 7B (annotated)

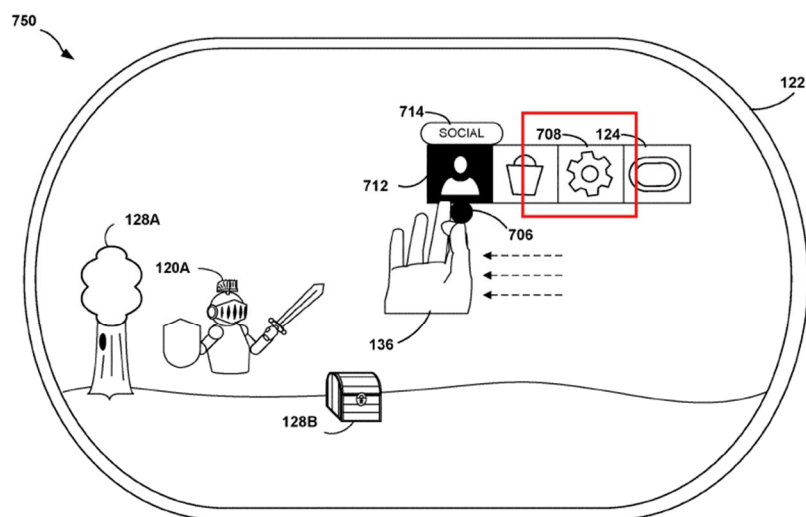


FIG. 7C

EX1007, FIG. 7C (annotated)

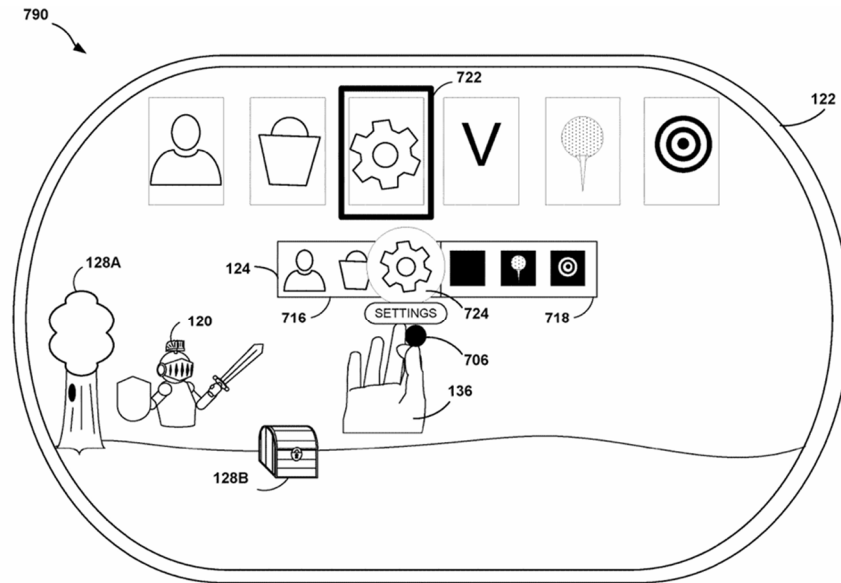


FIG. 7G

EX1007, FIG. 7G

A POSITA would be motivated to incorporate Ravasz's setting icons as one of Tang's virtual objects in a grouping of buttons or context menu because it would have increased the functionality of Tang's buttons or context menu. EX1003, ¶255. This would have benefited the VR environment system to allow the user to have more control over the operation of the system in an intuitive way (through a group of buttons or context menu). *Id.*; *see also* Section VIII.A.

A POSITA would have had a reasonable expectation of success in incorporating Ravasz's system setting icons because this would have amounted to a simple programming change, within the abilities of a POSITA. *See* Section VIII.B; EX1003, ¶256. It was well-known to include a settings button as part of group of buttons or context menu as an option for the user in an easy-to-find place. *Id.*

4. 8[a.iii]

Tang discloses this limitation. EX1003,¶¶258-260. The application-launching virtual object and configurable-settings virtual object are among the grouping of buttons 64 or context menu 68, and therefore *displayed together*. See Section VIII.C.2; EX1003,¶259. The virtual buttons 64 or context menus 68 are *displayed in a virtual reality environment to a user using Tang's HMD 14 (virtual reality hardware device)*. See Section VIII.C.3; EX1003,¶259.

5. 8[a.iv]

Tang discloses this limitation. See Sections VIII.C.4-5, VIII.E.3; EX1003,¶¶261-264. The application-launching virtual object (*first software application icon*) and configurable-settings virtual object (*first virtual reality platform system setting icon*) are among the grouping of buttons 64 or context menu 68 in near interaction mode that are displayed within a predetermined threshold distance (*appear closer*) from the user, including the user's hand (*to a body part of the user*). See Section VIII.C.4; EX1003,¶262. The user's hand (*a body part of the user*) is tracked by outward-facing cameras 32 (*a camera*) in HMD 14 (*of the virtual reality hardware device*) to recognize gestures (*to interact*) associated with the application-launching virtual object (*first software application icon*) and configurable-settings virtual object (*first virtual reality platform system setting icon*) that are among the grouping of virtual buttons 64 or context menus 68. See Section

VIII.C.5; EX1003,¶263.

6. 8[a.v]

Tang with Ravasz renders obvious this limitation. *See* Sections VIII.C.6, VIII.E.2-3; EX1003,¶¶265-270. Two of Tang’s virtual objects in a far context menu in far interaction mode with Ravasz’s application-launching and configurable settings functionality, respectively (*a second software application icon and second virtual reality platform system setting icon*) is displayed to the user when the user is using a user-input device (*physical controller*) in the virtual reality environment. *See* Section VIII.C.6; EX1003,¶¶266-267. A POSITA implementing Tang’s far context menu would naturally look to other similar VR systems like Ravasz that provide more detail on what actions could be performed through a menu. EX1003,¶¶268-269; *see also* Sections VIII.E.2-3.

7. 8[b.i]

Tang with Ravasz renders obvious this limitation. EX1003,¶¶271-275. Tang’s virtual object among the buttons or context menu with Ravasz’s configurable-settings functionality (*first virtual reality platform system setting icon*) is *operatively linked to activate a setting*. EX1003,¶¶273-274.

Ravasz discloses the ability to “modify configurable settings, [and] start or stop processes or perform other actions.” EX1007, [0063]. Therefore, Ravasz’s system has “settings” and “processes” (*setting*) that can be started or stopped.

EX1007, [0063]. A POSITA would have understood that a setting or a process is “started,” is one that is *activated* when engaged. EX1003, ¶273. Ravasz discloses that “in response to gesture detector 424 detecting one of the pre-defined gestures” on an element (*virtual reality platform system setting icon*), “***modify[ing]/configurable settings***” (*first virtual reality platform system setting icon*). EX1007, [0063]; Section VIII.E.3; *see also* EX1007, EX1007, [0003]. A POSITA would also have understood modifying a configurable setting encompasses “activating” it. EX1003, ¶273.

Interacting with a system setting icon to *activate* it would have been obvious to a POSITA. EX1003, ¶274; Sections VIII.A, VIII.E.3.

8. 8[b.ii]

Tang with Ravasz renders obvious this limitation. *See* Section VIII.C.8; EX1003, ¶¶276-279. An action is performed when the user taps or presses the virtual object. *Id.*, ¶277. Ravasz teaches “in response to gesture detector 424 detecting one of the pre-defined gestures,...modify configurable settings.” *Id.*, [0063]; *see also* [0076]. It would have been obvious to a POSITA that this pre-defined gesture could be tap or press to activate a setting, since taps and presses were already used to interact with other icons including launching applications. EX1003, ¶278; *see also* Section VIII.C.8.

9. 8[c]

Tang with Ravasz renders obvious this limitation. See Sections VIII.C.7-8 and VIII.E.2-3. EX1003, ¶¶280-282. Tang's virtual object among the buttons or context menu with Ravasz's application-launching functionality will launch an application when the user taps or presses the user's hand directly on the virtual object (*operatively linked to launch a software application when interacted with a...tapping [or] pressing...action of the body part of the user directly on the first software application icon*). See Section VIII.C.7-8.

F. Dependent Claims 9-11:

1. 9

Tang with Ravasz renders obvious this limitation. EX1003, ¶¶283-287. As discussed, a virtual object that *is displayed as a 3-dimensional button that launches a software application when pressed by the user's hand or finger*. See Section VIII.D.1; EX1003, ¶284. It would have also been obvious to render the virtual object that modifies the settings (*the first virtual reality platform system setting icon...that activates the setting when pressed by the user's hand or finger*) as a 3-dimensional button. EX1005, [0028], FIGs. 5A, 5B, 5E; EX1007, [0026] ("UI element 126 may be a...three-dimensional shape."); EX1003, ¶285. A POSITA would have understood the benefits of rendering all icons in the same grouping of buttons or

context menu in three-dimensional form because it would be more aesthetically pleasing and therefore enhance the user experience. EX1003,¶286.

2. 10/11

Tang with Ravasz renders obvious these limitations. *See* Section VIII.D.3; EX1003,¶¶288-290. The virtual objects in the grouping of buttons or context menu in near interaction mode (*first grouping of the plurality of icons/first software application icon and the first virtual reality platform system setting icon*) are displayed within a predetermined threshold distance 54 which “may be in a range of 460 mm to 580 mm,” representing an “average distance from the shoulder of an average user to the grip center of a hand” (*an apparent distance of a user’s arm length to the user*). EX1005, [0031]; EX1003,¶289; *see* Section VIII.D.3.

IX. GROUND 2: RAVASZ WITH TANG RENDERS OBVIOUS CLAIMS 1-11

A. Motivation to Combine

A POSITA would have been motivated to combine Ravasz’s interface with Tang’s system that switches between near and far interaction modes. EX1003,¶¶291-297.

To begin with, a POSITA would have been motivated for the reasons discussed previously (*see* Section VIII.A)—Ravasz and Tang are both directed to improving the user experience with the same or complementary hardware and software components. *See* EX1007, [0004], [0016]; EX1003,¶¶291-293.

Tang recognizes a “challenge” with VR systems is to accurately and comfortably interact with and/or manipulate virtual objects located further from the user. *See* EX1005, [0016]; EX1003, ¶¶292, 294. Tang solves this problem through a distance interaction mode system, whereby a user can interact with virtual objects in near and/or far interaction modes. *See* EX1005, [0003], [0016]; EX1003, ¶294.

A POSITA would have been motivated to implement Tang’s distance interaction mode system in Ravasz’s virtual user interface system to manage elements close to and further from the user. EX1003, ¶295. Tang’s distance interaction mode system provides a mechanism that ensures virtual elements can be displayed closer to the user when using gestures, allowing the user to more easily interact with the virtual elements. *Id.* A POSITA would have understood that when the user is not using the physical controllers, it would have been preferable to display icons closer to the user in a near interaction mode. *Id.*, ¶296.

B. Reasonable Expectation of Success

A POSITA would have had a reasonable expectation of success in implementing Tang’s distance interaction modes in Ravasz’s artificial reality system because the modification would have been a simple change of programming. EX1003, ¶¶298-299. Both disclose VR systems that employ HMDs with processors that detect images of user movement using cameras and display interface elements. *See* Sections V.A-B. Both HMDs also communicate with and receives input from

the controllers. EX1007, [0040].

Moreover, Ravasz already contemplates displaying virtual content further from a user. EX1003, ¶299. For example, using the “known orientation and position of [the] display” relative to the HMD, the processor tracks the HMD position and “render[s] artificial reality content according to a current viewing perspective” of the HMD and the user. EX1007, [0044]. Ravasz discloses that such a tracking system could be used, for example, to display virtual images “representative of the physical environment” (e.g., mixed reality), which would have been displayed further away than user interface elements. EX1007, [0045]. Thus, programming Ravasz’s processor to display UI elements both close to and further from a user would have been a straightforward application, well-known to the POSITA. EX1003, ¶299; EX1029, 551; EX1030, 9; EX1034 (depicting icons both near and far).

C. Independent Claim 1

1. 1[pre]

To the extent limiting, Ravasz discloses the preamble. EX1003, ¶¶300-308. Ravasz discloses selectable UI menu elements that launch applications (*selection interface*) in an artificial reality environment (*virtual reality environment*). *Id.*, ¶¶301-302.

Ravasz discloses “user interface elements” such as a “*graphical user*

interface” displayed as “a menu or sub-menu with which user 110 interacts to operate the *artificial reality* system, or individual user interface elements *selectable* and manipulatable by user 110, such as icon elements...” (*virtual reality environment selection interface*). EX1007, [0026], [0046]; EX1003, ¶¶301-303.

Ravasz also describes UI icon arrays from which a user is able to select various icons (*see, e.g., FIG. 7F*), including an array of “icons representing applications available or running in the artificial reality system” (*virtual reality environment selection interface*). EX1007, [0089], 7E, 7F; EX1003, ¶¶304-307.

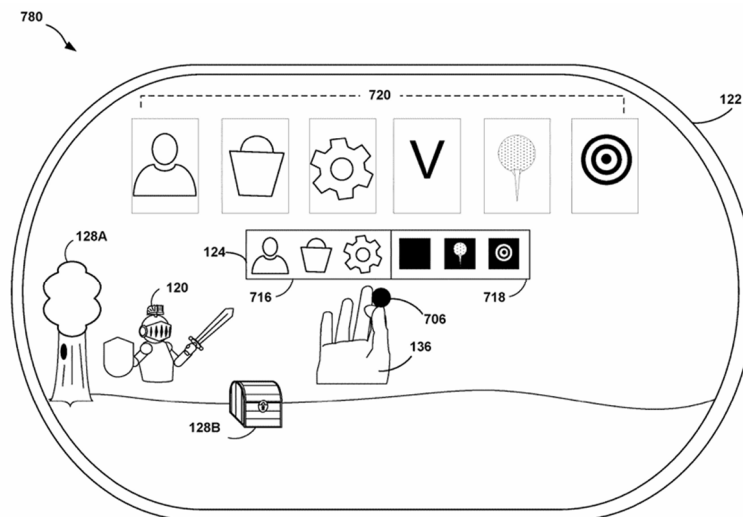


FIG. 7F

EX1007, FIG. 7F

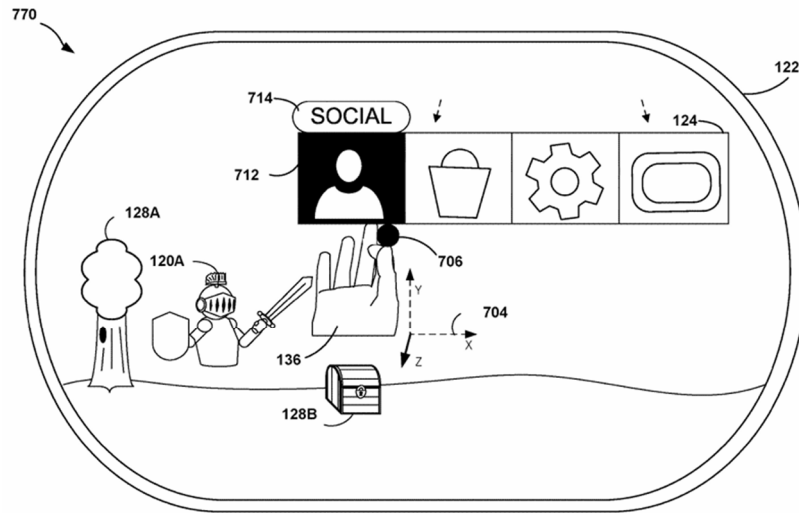


FIG. 7E

EX1007, FIG. 7E

2. 1[a.i.]

Ravasz discloses this limitation. EX1003, ¶¶309-314. Ravasz discloses “UI elements 126 that are arrayed along a dimension of UI menu 124.” EX1007, [0073]. The UI menu items “*correspond to applications*,” including “favorite applications” or “applications currently running within the artificial reality system” (*software application icons*) EX1007, [0089]; *see also* [0026], [0046], [0073], [0089], [0090]; *see also* FIGs. 7A-7E (depicting application icons in various UI menus); EX1003, ¶¶310-313.

3. 1[a.ii]

Ravasz discloses this limitation. EX1003, ¶¶315-320. Ravasz discloses the UI menu element launching an application is *displayed in* an artificial reality

environment (*a virtual reality environment*) to a user wearing HMD 112 (*using a virtual reality hardware device*). *Id.*, ¶¶316.

Ravasz teaches “graphical user interface elements and techniques for presenting and controlling the user interface elements ***within an artificial reality environment.***” EX1007, [0004]. Ravasz explains that “artificial reality” includes “***virtual reality*** (VR).” *See id.*, [0002]; EX1003, ¶¶317-318.

Ravasz discloses a head mounted display device, or HMD/HIVID, (*virtual reality hardware device*) “configured to output artificial reality content” (*displayed in a virtual reality environment*). EX1007, [0008]-[0009], [0023], [0040], [0043], [0044], FIGs. 7A-7E (depicting various HMDs displaying UI menus); EX1003, ¶319.

4. 1[a.iii]

Ravasz with Tang renders obvious this limitation. EX1003, ¶¶321-333.

Ravasz describes a UI application menu element (*first software application icon*). EX1007, [0072], [0073]. The system “renders the portions of hand 132 of user 110 that are within field of view 130 as a virtual hand 136 within artificial reality content 122” such that “user 110 is ***able to view the portions of their hand 132 and/or arm 134*** that are ***within field of view 130*** as objects within artificial reality content 122.” *Id.*, [0028]; *see also id.*, [0068], [0072], [0073]; EX1003, ¶¶322-323, 332.

The UI menu is then rendered “*in proximity* to the virtual hand” that has been positioned at the location of the user’s actual hand. EX1007, [0072], [0073]; [0088]; *see also* FIG. 7E. For instance, the system “can detect the back-to-front motion” where the user has “moved their hand closer to themselves along the Z axis” and can translate the position such that the UI menu “*appear[s] to be larger, and thus closer to the user*” (*closer to a body part of the user*). *Id.*, [0087], FIG. 7E; EX1003, ¶¶324-325.

To the extent Ravasz does not explicitly disclose this limitation, Tang renders it obvious. EX1003, ¶326. Tang discloses a virtual reality system that allows users to interact with virtual objects (like Ravasz’s UI elements) displayed both near to (near interaction mode) and at a distance from (far interaction mode) the user. EX1005, [0016], [0025]; EX1003, ¶327; *see also* Section VIII.C.4.

In near interaction mode, virtual objects like “virtual buttons” or “context menu” are displayed to *appear closer to a body part of a user*. *See* Section VIII.C.4; EX1005, [0021] (interactions within a “predetermined threshold distance” measured “with respect to various *parts of the user*” like “the *shoulder of the user*” or “from another *point on the body of the user*”), [0028]-[0029]; EX1003, ¶328.

A POSITA would have implemented the combination because it would have allowed users to more easily select an UI element. *See* Section IX.A; EX1003, ¶330.

A POSITA would have had a reasonable expectation of success because it was a simple change of programming. *See* Section IX.B; EX1003,¶331.

5. 1[a.iv]

Ravasz discloses this limitation. EX1003,¶¶334-340. The user's hand's *is tracked by* image capture devices 138 (*a camera*) of the HMD 112 (*virtual reality hardware device*) to select the UI menu element that launches applications (*interact with the first software application icon*). *Id.*,¶335.

Ravasz's HMD/HIVID 112 includes "one or more image capture devices 138, e.g., ***cameras***...for capturing image data of the surrounding [] environment." EX1007, [0023], [0045]; FIG. 1A. The gesture detector uses the images obtained by the image capture devices (*cameras*) of the HMD to "identify the hand and/or arm of user 110, and ***track movements of the hand and/or arm*** relative to HIVID 112 ***to identify gestures performed by user 110***." *Id.*, [0055], [0061], FIG. 2; EX1003,¶¶336-337.

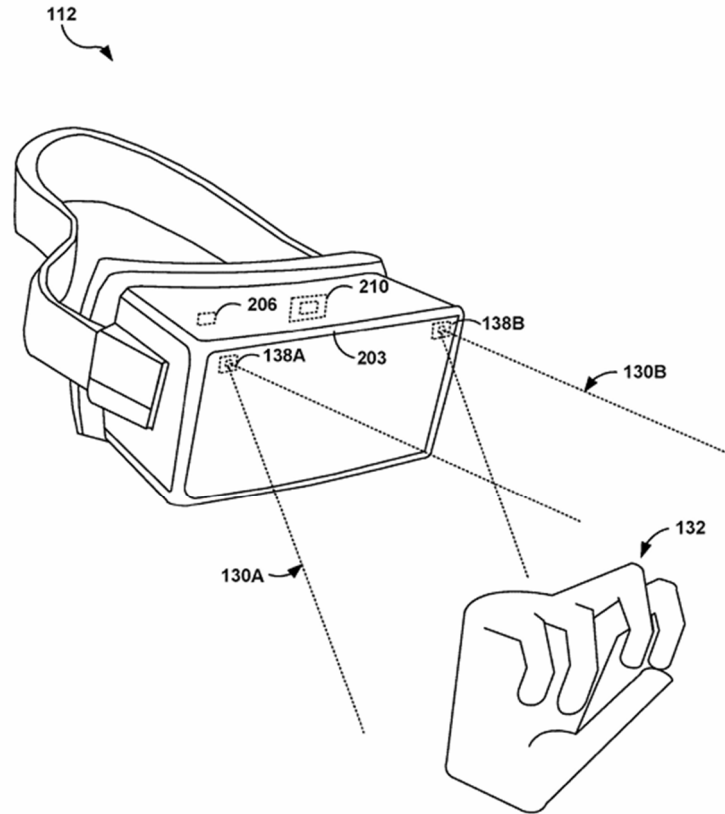


FIG. 2

EX1007, FIG. 2

Ravasz explains to identify a “selection gesture,” the gesture detector “utilize[s] image data from image capture devices 138” (*cameras*) “to analyze configurations, positions, and/or orientations of a user’s hand.” *Id.*, [0064]. A selection gesture can be “movement of a finger of the hand” or “movement of the hand in a direction that is substantially normal to the plane of the UI menu” (*id.*, [0076]) to “select the menu element” (*interact with the first software application icon*). *Id.*, [0077], FIG. 7B; EX1003, ¶¶338-339.

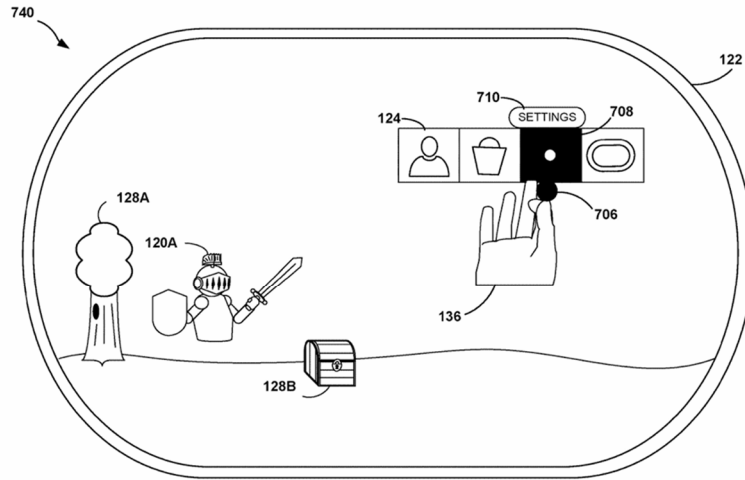


FIG. 7B

EX1007, FIG. 7B

6. 1[a.v]

Ravasz with Tang renders obvious this limitation. EX1003, ¶¶341-352.

Ravasz discloses displaying both “a UI menu 124” comprised of UI icon elements (*first software application icon*) and a “UI icon array 720” comprised of UI icon elements (*second software application icon*). EX1007, FIG. 7F; EX1003, ¶342. The icons in UI icon array 720 have a “corresponding application, or an image associated with the application.” *Id.*, [0089]; *see also* FIGs. 7F, 7G. The individual UI menu 124 elements and UI icon array 720 elements are depicted in FIGs. 7F and 7G.

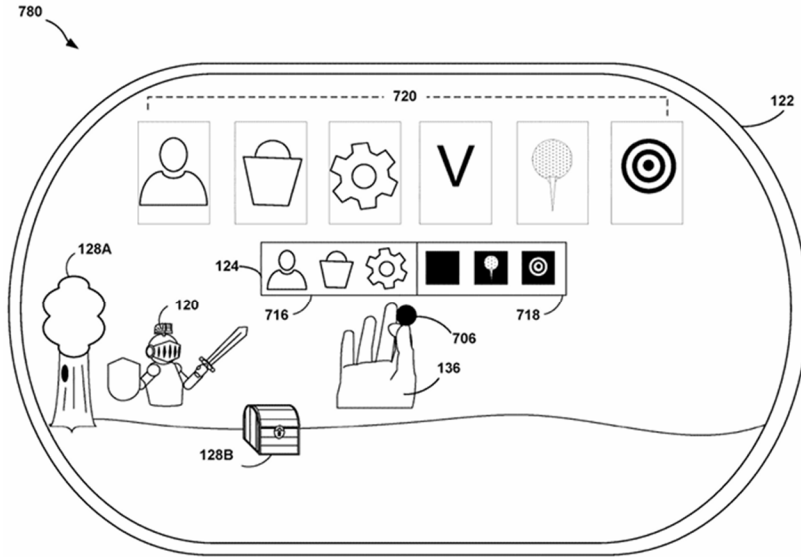


FIG. 7F

EX1007, FIG. 7F

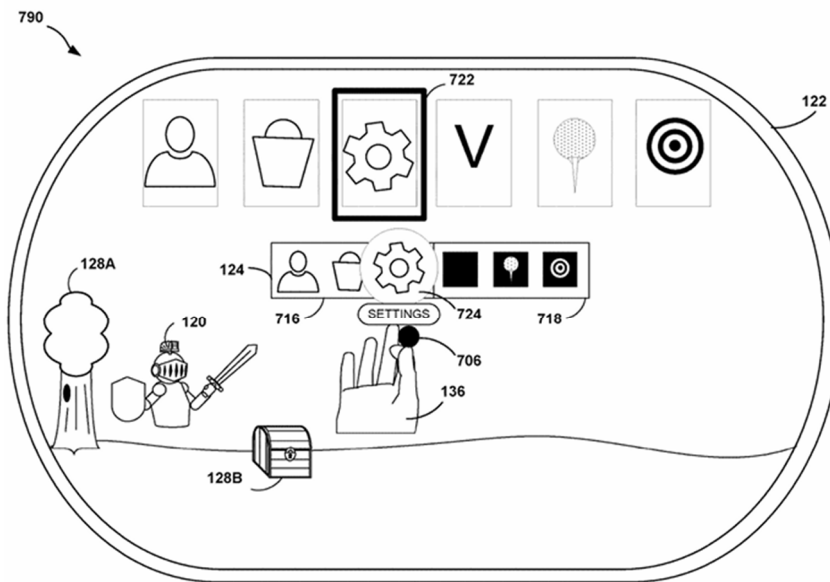


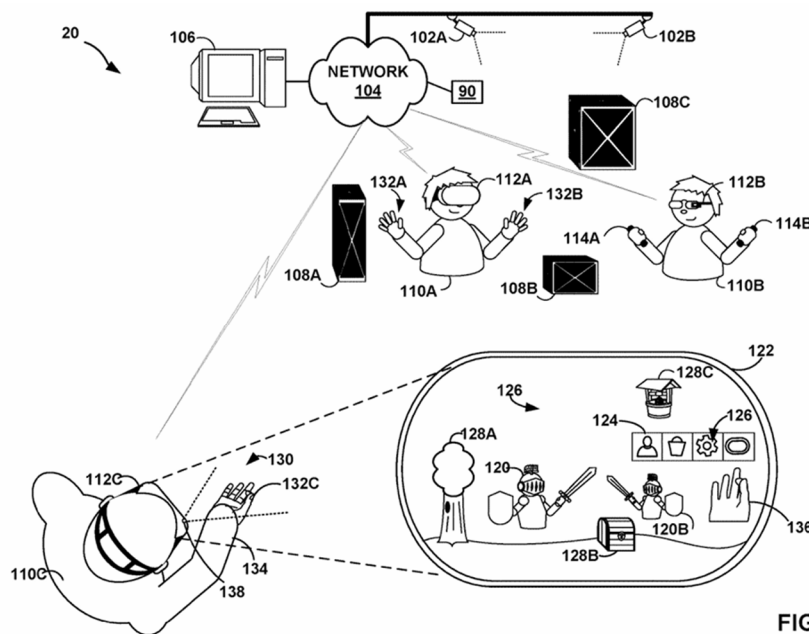
FIG. 7G

EX1007, FIG. 7G

The UI element 722 “appear[s] larger” and thus, closer, to the user than those in the UI element 724 in the UI icon array. EX1007, [0090]; *see* FIG. 7G (menu

icon 724 enlarged to appear closer to the hand). As such, the UI menu element (*first software application icon*) appear closer to the user's hand than UI icon array elements (*second software application icon that is displayed to the user*). EX1003, ¶¶342-344.

The UI icon array elements can be *displayed to the user when the user is using a physical controller in the virtual reality environment* when the user uses controllers 114 to communicate user inputs to the HMD/HIVID. See EX1007, [0040], [0037], [0039]-[0040], FIG. 1B; EX1003, ¶¶345-346.



EX1007, FIG. 1B

To the extent that Ravasz does not explicitly disclose this limitation, Tang renders it obvious. EX1003, ¶347.

Tang discloses a distance interaction mode system that a POSITA would have

implemented to display Ravasz's UI menu element (*first software application icon*) closer to the user's hand (*appears closer to a body part of a user*) than the UI icon array to display another UI element (*another second software application icon*). See Sections VIII.C.4, VIII.C.6. EX1003,¶348. Recognizing that virtual objects "beyond the reach of the user...may be difficult or tiresome for the user to control," Tang taught a system that could recognize whether a virtual object is within or outside a predetermined threshold distance to invoke a near or far interaction mode. EX1005, [0023].

As Tang explains, "a user interacting with a distant object may experience muscle fatigue if the interaction involves raised and/or extended hands and/or arms. Some types of interactions, such as...accessing a context menu, may even be impossible in situations where virtual content is displayed at a distance beyond the reach of the user." *Id.*, [0016]. Using near and far interactions modes, "may facilitate ease of interaction between a user and a virtual object... to make fine controls of the virtual object 48 and reduced muscle fatigue of the user." *Id.*, [0037]. Further, using gestures when a context menu is in proximity to the user's hand, "may provide greater intuitive and continuous interaction between the user and the virtual object." *Id.*; EX1003,¶349.

A POSITA improving Ravasz's artificial reality system would have been motivated to implement Tang's distance interaction mode system so that the user

employs gestures to interact with an element of a UI menu in near interaction mode. EX1003,¶350. As Tang recognized, gestures are difficult to use when the virtual object (or UI element in Ravasz’s system) is beyond the reach of the user. Accordingly, in near interaction mode, the user can use gestures when the virtual objects (such as a UI element in the UI menu) is within a predetermined threshold distance. In contrast, when Ravasz’s physical controllers are used to interact with a UI element located further than a predetermined threshold distance (e.g., UI icon array, far context menu) the user may not be fatigued because the interaction does not “involve[] raised and/or extended hands and/or arms.” EX1005, [0016]; EX1003,¶350. Thus, in far interaction mode, the user can use controllers to interact with UI elements located further than a predetermined threshold distance. *Id.*

Implementing Tang’s distance interaction mode in Ravasz’s artificial reality system would have realized the benefits disclosed in Tang. EX1003,¶351. Such implementation would have provided “greater ability to make fine movements/selections with respect to a virtual object (buttons or context menu) that is in proximity to the user’s hand and to “respond to user interaction in an intuitive and user-friendly manner [to] provide an improved experience for the user.” *Id.*, [0002], [0029]; *see also* Sections IX.A-B.

7. 1[b.i]

Ravasz discloses this limitation. EX1003,¶¶353-358.

One of the UI elements/menu item in the UI menu is the *first software application icon*. See Section IX.C.2. Ravasz explains “the menu items of a UI menu 124 can correspond to applications.” EX1007, [0089]. The menu item is *operatively linked to launch a respective software application* because “[d]etection of [a] selection gesture may cause the artificial reality system to perform some action,” including “an application to be instantiated.” *Id.*, [0076]; EX1003, ¶¶354-355.

Ravasz explains that “in response to gesture detector 424 detecting one of the pre-defined gestures,” such as a “selection gesture” on a menu item, “user interface engine 428 and/or application engine 440 may...***launch applications***” (*operatively linked to launch a respective software application*). EX1007, [0063]; see also *id.*, [0046], [0056]; EX1003, ¶¶356-357.

8. 1[b.ii]

Ravasz alone or with Tang renders obvious this limitation. EX1003, ¶¶359-366.

Ravasz discloses that the user “perform[s] a selection gesture,” to select “a particular menu item of the UI menu” using movement of a finger or a hand. EX1007, [0034], [0076], [0077]. Hand movement can be “in a direction that is substantially normal to the plane of the UI menu (e.g., between 0 and 30 degrees normal to the plane). *Id.*, [0076]; see also cl. 8. A POSITA would have understood

movement substantially normal (i.e. perpendicular) to the UI menu plane to select (*interact with*) is a *pushing, pulling, tapping, [or] pressing...action* on the icon. EX1003,¶¶360-362.

To the extent Ravasz does not explicitly disclose this limitation, Tang renders it obvious. *Id.*,¶363. Like Ravasz’s gesture detector, Tang discloses a “gesture recognition module 24 [that] may be configured to identify poses and gestures expressed by the user by processing image data of the user collected by outward-facing cameras 32 when these cameras capture images of the user’s body or portions thereof, such as the user’s hands.” EX1005, [0018]; EX1003,¶364. The “predetermined gesture may be a gesture...such as a tap gesture.” EX1005, [0026]. “The plurality of gestures may further include a pressing motion by finger.” *Id.*, [0028].

A POSITA would have been motivated to implement Tang’s tap or press gestures as one of Ravasz’s selection gestures with a reasonable expectation of success because taps and presses directly on an element were well-known gestures for selection and execution of an action. EX1003,¶365; *see also* Sections IX.A-B.

9. 1[b.iii]

Ravasz discloses this limitation. EX1003,¶¶367-371.

The application-launching elements of Ravasz’s UI menu and UI icon array are *the first software application icon and second software application icon*,

respectively, *operatively linked to launch the same respective software application*.
See Section VIII.C.9; EX1003,¶368.

Ravasz teaches that “in response to gesture detector 424 detecting one of the pre-defined gestures,” such as a “selection gesture” on a UI element in a UI menu, “may...***launch applications***.” EX1007, [0063]; *see also* [0046], [0056]. Ravasz further discloses that menu items can “correspond to applications” and detecting a selection gesture may “cause an application to be instantiated, or can cause a currently running application to be brought into the foreground.” EX1007, [0089], [0076]; EX1003,¶369.

Figure 7F shows “an example HIVID display 780 illustrating a UI menu 124 and UI icon array 720 in accordance with aspects of the disclosure....The images on the individual icons in icon array 720 can represent a current display of the corresponding application, or an image associated with the application.” EX1007, [0089], FIG. 7F. A POSITA would have understood that the icons in UI menu 124 and UI icon array 720 are two software application icons (*a first and second software application icon*) that are *operatively linked to launch the same application*, because their icon images are the same. EX1003,¶370.

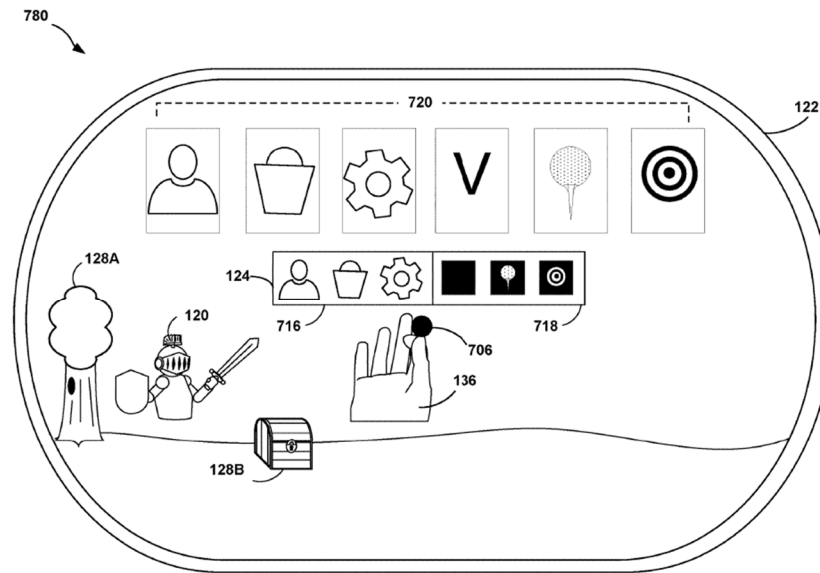


FIG. 7F

EX1007, FIG. 7F

D. Dependent Claims 2-7:

1. 2

Ravasz with Tang renders obvious this limitation. EX1003, ¶¶372-378. Ravasz's UI element in the UI menu (*at least one software application icon*) that *launches* an application when the selection gesture is Tang's press (*a respective software application when pressed*) by the user's hand or finger. See Sections IX.C.7-8; EX1003, ¶373.

Ravasz discloses "UI element 126 may be a ... three-dimensional shape that is manipulatable by a user." EX1007, [0026]; *see also* [0006]. A POSITA would have understood that the icon is a virtual "button." EX1003, ¶¶374-377; *see also* Section IX.A-B. Virtual "buttons" well-known as demonstrated by Tang. EX1005,

[0028] (describing “virtual buttons”); EX1003, ¶377.

2. 3/4

Ravasz with Tang renders obvious these limitations. EX1003, ¶¶379-387.

A POSITA would have implemented Tang’s scrolling functionality to navigate Ravasz’s UI menu. EX1005, [0029], FIG. 6; EX1003, ¶¶380-381. Ravasz further teaches rendering on Tang’s buttons scrolling “arrows [that]...indicate the direction the user’s index figure...should be moved.” EX1007, [0069]; EX1003, ¶¶382-384; *see also* Section VIII.D.2. The buttons would be *adjacent* the UI menu (*grouping*). *See* Section VIII.D.2; EX1003, ¶385.

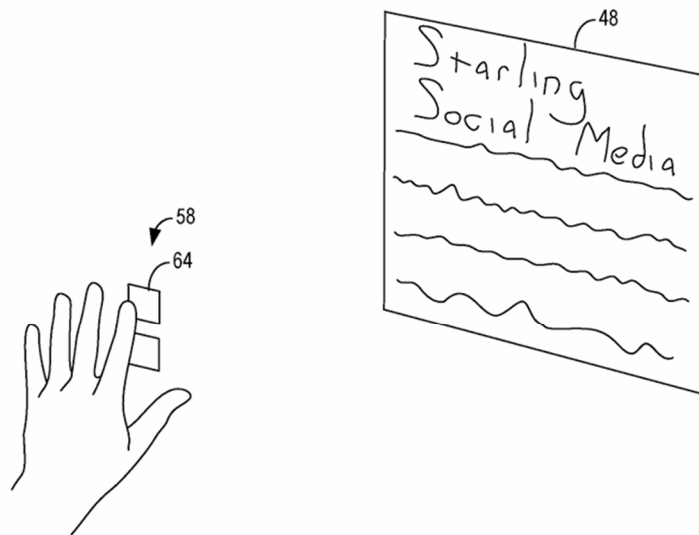


FIG. 6

EX1005, FIG. 6

A POSITA would have been motivated to implement Tang’s scrolling functionality to navigate Ravasz’s UI menu with a reasonable expectation of success because it was a well-known, intuitive UI function to navigate long lists. EX1003,¶386; *see also* Sections IX.A-B.

3. 5

Ravasz with Tang renders obvious this limitation. EX1003,¶¶388-390. Ravasz’s elements in the UI menu (*the plurality of one or both of software application and virtual reality platform setting icons in the grouping*) are displayed *within* Tang’s predetermined threshold distance 54. *See* Section IX.D.3; EX1003,¶389. Tang discloses the predetermined threshold distance “in a range of 460 mm to 580 mm” or the “average distance from the shoulder of an average user to the grip center of a hand” (*within an apparent distance of a user’s arm length to the user*). EX1005, [0031]; *see also* Section VIII.D.3.

4. 6/7

Ravasz with Tang renders obvious these limitations. EX1003,¶¶391-393. Ravasz’s application-launching element in the UI menu (*first software application icon*) is displayed *within* Tang’s predetermined threshold distance 54. *See* Section IX.C.4; EX1003,¶392. Tang discloses the predetermined threshold distance “in a range of 460 mm to 580 mm” or the “average distance from the shoulder of an

average user to the grip center of a hand” (*within an apparent distance of a user’s arm length to the user*). EX1005, [0031]; *see also* Section VIII.D.3.

E. Independent Claim 8:

1. 8[pre]

To the extent limiting, Ravasz discloses the preamble. *See* Section IX.C.1; EX1003, ¶¶394-396. Ravasz discloses menus of selectable UI elements that launch applications (*selection interface*) in an artificial reality environment (*virtual reality environment*). *Id.*

2. 8[a.i]

Ravasz discloses this limitation. *See* Section IX.C.2; EX1003, ¶¶397-399. Ravasz discloses a UI element in a UI menu that can launch an application (*first software application icon*). *Id.*

3. 8[a.ii]

Ravasz discloses this limitation. EX1003, ¶¶400-404. The settings element in the UI menu is *a first virtual reality platform system setting icon*. *Id.*

Ravasz discloses that a user interacts with “a graphical user interface, such as a menu or sub-menu...to operate the ***artificial reality system***” itself, or selectable and modifiable “individual user interface elements ... such as [application] icon elements.” EX1007, [0026]. Thus, the user has the option to control or operate the VR system (*virtual reality platform system*) or software applications within the VR

environment. *Id.*; EX1003, ¶401.

Ravasz further discloses “configurable” settings, which are *virtual reality platform system settings*. For instance, “the artificial reality system can be configurable (for example, by the user) to determine a configuration of the left hand or the right hand.” EX1007, [0067]. Such a configuration represents a modification of a system setting (e.g., left or right-handedness of the system). EX1003, ¶402. The user can also configure the HMD itself when gesture detector 424 detects a pre-defined gesture, such that the “user interface engine 428 dynamically generates a user interface...allowing the user 110 to...configur[e] HIVID 112” (EX1007, [0063]), which is a further alteration of the VR system settings. EX1003, ¶402. Moreover, the “user interface engine 428 and/or application engine 440” can themselves “modify *configurable settings*” upon detection of particular pre-defined gestures by the user. EX1007, [0063].

A POSITA would have understood a “configurable” setting to be a system setting because configuring the HMD/HIVID 112, for instance, would encompass altering/operating the VR system settings. EX1003, ¶403. Ravasz depicts “example HIVID displays illustrating positioning and interacting with UI menus,” in FIGs. 7A-7G, including, system settings icons. EX1007, [0019]. For instance, in FIG. 7B, Ravasz depicts menu item 708 labeled as a “Settings” icon. EX1007, FIG. 7B, [0076]; EX1003, ¶403; *see also*, FIG. 7C (menu item illustrated using a traditional

settings icon); FIG. 7G (same).

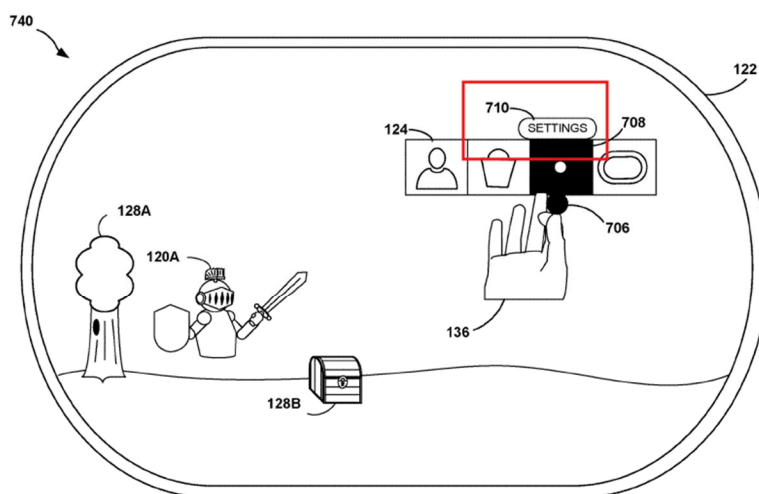


FIG. 7B

EX1007, FIG. 7B (annotated)

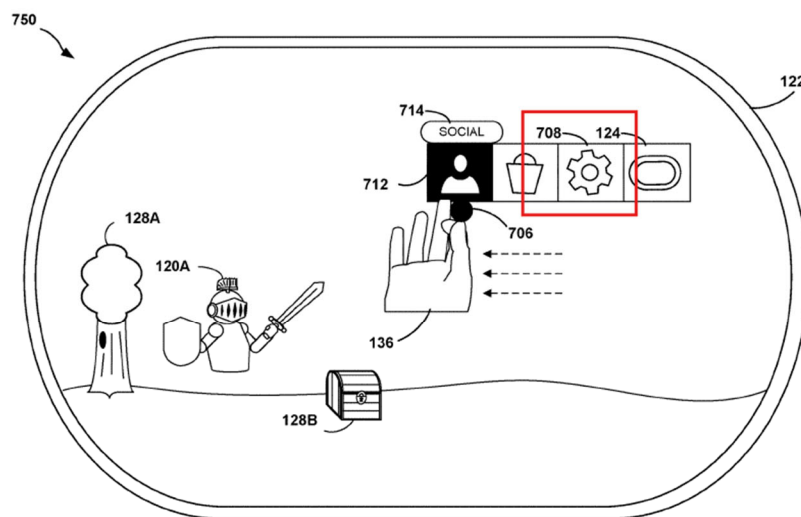


FIG. 7C

EX1007, FIG. 7C (annotated)

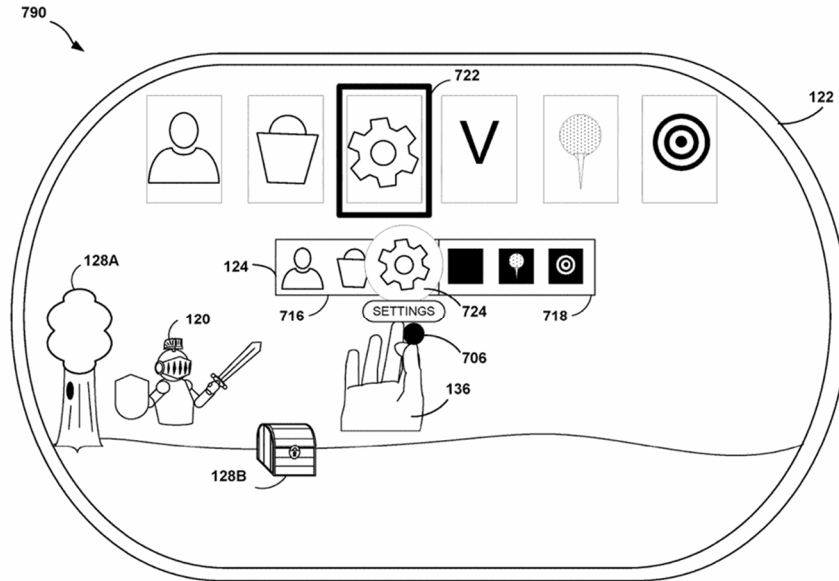


FIG. 7G

EX1007, FIG. 7G

4. 8[a.iii]

Ravasz discloses this limitation. EX1003, ¶¶405-407. The application-launching element and configurable-settings element are displayed in the UI menu, and therefore *displayed together*. See Section IX.C.2; EX1003, ¶406. The UI menu is *displayed in* an artificial reality environment (*virtual reality environment*) to a user using Tang's HMD 14 (*virtual reality hardware device*). See Section IX.C.3.

5. 8[a.iv]

Ravasz with Tang discloses this limitation. See Sections IX.C.4-5, IX.E.3; EX1003, ¶¶408-410. The application-launching element (*first software application icon*) and configurable-settings element (*first virtual reality platform system setting icon*) in the UI menu in near interaction mode are displayed within a predetermined

threshold distance (*appear closer*) to the user's hand (*to a body part of the user*) that *is tracked by* image capture device 138 (*a camera*) in HMD 112 (*of the virtual reality hardware device*) to recognize gestures (*to interact*) associated with the application-launching element (*first software application icon*) and configurable-settings element (*first virtual reality platform system setting icon*). EX1003,¶409.

6. 8[a.v]

Ravasz with Tang renders obvious this limitation. *See* Sections IX.C.6, IX.E.2-3; EX1003,¶¶411-414. The UI icon array in far interaction mode (*than a second software application icon and a second virtual reality platform system setting icon*) that is displayed to the user when the user is using controller 114 (*physical controller*) in the artificial reality environment (*virtual reality environment*). *See* Section IX.C.6; EX1003,¶412. A UI element in the UI icon array modifies the system settings (*/setting icon*). Ex. 1007, [0063], FIG. 7G. EX1003,¶413.

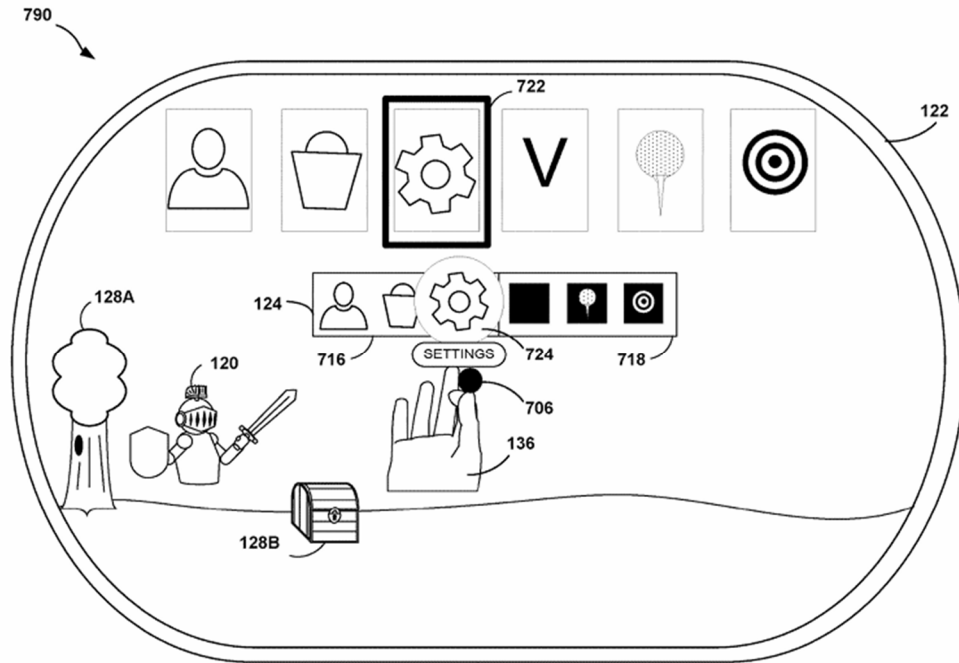


FIG. 7G

EX1007, FIG. 7G

7. 8[b.i]

Ravasz discloses this limitation. EX1003, ¶¶415-419. Ravasz’s configurable-settings element (*first virtual reality platform system setting icon*) is *operatively linked to activate a setting*. *Id.* ¶416..

Ravasz discloses that the “user interface engine 428 and/or application engine 440” have an ability to “modify configurable settings, [and] start or stop processes or perform other actions.” EX1007, [0063]. Therefore, Ravasz’s system has “settings” and “processes” (*setting*) that can be started or stopped. EX1007, [0063]. A POSITA would have understood that a setting or a process is “started,” is one that is *activated* when engaged. EX1003, ¶417. Ravasz discloses that “in response to

gesture detector 424 detecting one of the pre-defined gestures” on, for instance, a user interface element (*virtual reality platform system setting icon*), “user interface engine 428 and/or application engine 440 may...***modify configurable settings***” (of which there is one *first virtual reality platform system setting icon*). EX1007, [0063]; Section VIII.E.3; *see also* EX1007, [0003] (“During operation, the user typically interacts with the artificial reality system to...configure the system”). A POSITA would also have understood modifying a configurable setting encompasses “activating” it. EX1003, ¶417.

Moreover, interacting with a system setting icon to *activate* it would have been obvious to a POSITA. EX1003, ¶418; Section IX.E.3.

8. 8[b.ii]

Ravasz alone or with Tang renders obvious this limitation. *See* Section IX.C.8; EX1003, ¶¶420-422. An action is performed when the user taps or presses the configurable-setting element using the user’s hand. *Id.* ¶421. Ravasz teaches that “[d]etection of the selection gesture may cause the artificial reality system to perform some action.” EX1007, [0076]. Notably, “in response to gesture detector 424 detecting one of the pre-defined gestures,...modify configurable settings.” *Id.*, [0063]. It would have been obvious to a POSITA that this pre-defined gesture could be tap or press of the virtual object representing the configurable settings to activate a setting, particularly since taps and presses were already being used to interact with

other icons including launching applications. EX1003,¶¶421; *see also* Section IX.C.8.

9. 8[c]

Ravasz with Tang renders obvious this limitation. *See* Sections IX.C.8 and IX.E.2. EX1003,¶¶423-425. Ravasz's application-launching element will launch an application when the user taps or presses the user's hand directly on the element (*operatively linked to launch a software application when interacted with a...tapping [or] pressing...action of the body part of the user directly on the first software application icon*). *See* Section IX.C.7-8; EX1003,¶¶424.

F. Dependent Claims 9-11:

1. 9

Ravasz discloses this limitation. EX1003,¶¶426-428. As discussed, the combination discloses an application-launching element that *is displayed as a 3-dimensional button that launches a software application when pressed by the user's hand or finger*. *See* Section IX.D.1; EX1003,¶427. It would have also been obvious to render configurable-settings UI element (*the first virtual reality platform system setting icon...that activates the setting when pressed by the user's hand or finger*) as *a 3-dimensional button*. EX1005, [0028], FIGs. 5A, 5B, 5E; EX1007, [0026] ("UI element 126 may be a...three-dimensional shape."); EX1003,¶427. A POSITA would have understood the benefits of rendering all icons in the same UI menu in

three-dimensions because it would be more aesthetically pleasing and enhance the user experience. *Id.*

2. 10/11

Ravasz with Tang renders obvious these limitations. *See* Section IX.D.3; EX1003, ¶¶429-431. The UI menu items in near interaction mode (*first grouping of the plurality of icons/first software application icon and the first virtual reality platform system setting icon*) are displayed within a predetermined threshold distance 54 which “may be in a range of 460 mm to 580 mm,” representing an “average distance from the shoulder of an average user to the grip center of a hand” (*an apparent distance of a user’s arm length to the user*). EX1005, [0031]; EX1003, ¶430; *see* Section IX.D.3.

X. GROUND 3: RAVASZ WITH TANG AND MINE RENDERS OBVIOUS CLAIMS 12-20

A. Motivation to Combine

A POSITA would have been motivated to modify Ravasz-Tang’s method for providing a VR interactive interface with distance-based interaction modes to incorporate Mine’s method for detecting contact with a controller. EX1003, ¶¶432-437.

First, all three are directed to interactive interfaces in a VR environment. EX1005, [0016]; EX1006, [0003], [0008]; EX1047, [0002], [0007]; EX1007, [0004]; *see also* Section V.A-C. A POSITA implementing the Ravasz-Tang

interactive interface would naturally look at other VR systems that focus on the interactive interfaces like that in Mine. EX1003,¶434.

Second, the references specifically address the problems identified in the other respective references. EX1003,¶434. Tang explains that its depiction of objects near and far addresses issues with user reach and fatigue. EX1005, [0016]. Mine discloses a similar concern, regarding the need for a user to keep their hand at a certain height to be included in the image capturing range in gesture modes. EX1006, [0052]-[0053]; EX1047, [0041]-[0042]. A POSITA implementing Ravasz-Tang's interface would have implicated the concerns and solutions detailed in Mine. EX1007, [0007]; EX1003,¶434. Mine's controller contact detection method would allow the interface to determine and automatically invoke a far detection mode if the user has become fatigued and taken the opportunity to rest their hands and use a different input method. EX1003,¶434.

Third, all include complementary components. EX1003,¶435. Ravasz discloses "artificial reality systems...that generate and render graphical user interface elements for display to a user in response to...pre-defined gestures by the user, such as particular motions, configurations, positions, and/or orientations of the user's hands, fingers, thumbs or arms, or a combination of pre-defined gestures." EX1007, [0005]. Tang also discloses virtual objects that may be displayed to the user, and allows those objects to be displayed at varying distances relative to the

user, incorporating a distance component to Ravasz. EX1005, [0016]. Mine similarly discloses computer graphical elements that may be displayed to the user, and allows interaction with those elements, with either the user's hand or controller, merely adding functionality that Ravasz already contemplates. EX1006, [0044]-[0051]; EX1047, [0033]-[0040].

B. Reasonable Expectation of Success

A POSITA would have had a reasonable expectation of success in modifying Ravasz-Tang's method for providing an interactive interface in a VR environment to incorporate Mine's method for detecting contact with a controller. EX1003, ¶¶438-441.

First, all use similar components in VR devices, making the combination more straightforward. EX1003, ¶439. Ravasz discloses image capture devices 238, gesture detector 324, and motion sensors 206 to detect gestures. EX1007, [0008], [0045]. Tang discloses inward and outward-facing cameras (30/32), gesture recognition modules 24, and inertial measurement units 36. EX1005, [0020], [0017], [0019]. Mine discloses an imaging unit 202, control unit 211, and attitude sensor 204 including an inertial measurement unit. EX1006, [0029], [0033], [0045]-[0046]; EX1047, [0018], [0022], [0034]-[0035]. Therefore, combining these references would have been a simple modification of similar components. EX1003, ¶439.

Second, a POSITA would have been aware of multiple ways to implement Mine's method of detecting controller contact and controller use in general. EX1003, ¶440. A POSITA would also have been aware of capacitive and other electrical means for determining whether a user's hand is in contact with the surface of a controller. *Id.*

Third, all disclose a controller or user-input device, such that implementing Mine's controller contact detection in Ravasz-Tang would be straightforward for a POSITA. EX1007, [0039]; EX1005, [0049] ; EX1003, ¶441. Therefore, a POSITA would have known that the benefits of Mine's controller detection could have been integrated into Ravasz-Tang's system. EX1003, ¶441.

C. Independent Claim 12:

1. 12[pre]

To the extent limiting, Ravasz discloses the preamble. EX1003, ¶¶442-445. Ravasz discloses a method for providing menus of selectable UI elements that launch applications and modify configurable settings (*method for providing an interactive control interface*) in an artificial reality environment (*in a virtual reality environment*). *Id.*

Ravasz discloses “*techniques for presenting* and controlling the *user interface* elements within an *artificial reality environment*” (*method for providing an interactive control interface in a virtual reality environment*). EX1007, [0004].

Ravasz further discloses “user interface elements specifically designed for user interaction and manipulation within an *artificial reality environment*.” *Id.*, [0022]. Specifically, Ravasz discloses “methods for activating menu prompts and menus” (user interface (UI) elements) and “for determining positioning and *user interaction* with menus.” EX1007, [0065]. Therefore, Ravasz discloses *a method for providing an interactive control interface in a virtual reality environment*. EX1003, ¶¶442-445.

2. 12[a]

Ravasz alone or with Mine renders obvious this limitation. EX1003, ¶¶446-458. The user is *using* an artificial reality environment (*a virtual reality environment*) displayed through HMD 112 (*via a virtual reality hardware device*). See Section IX.C.3.

Ravasz further discloses that the user can interact with virtual UI elements using a controller 114 (*physical device controller used to interact with the virtual reality environment*) held by the user. See EX1007, [0040]. “Controllers 114 may be in communication with HIVID 112B using near-field communication of short-range wireless communication such as Bluetooth, using wired communication links, or using other types of communication links.” *Id.* A POSITA would have understood that a user *is not using or has discontinued use of* the controller 114 if there is no signal received by HMD 112 from controller 114. EX1003, ¶447.

To the extent Ravasz does not explicitly disclose *detecting that a user...is not using or has discontinued use of* controller 114, Mine renders it obvious. EX1003, ¶448.

Mine discloses an “information processing system 1 includes an HMD 100, an image processing device 110, and a controller 120” for interacting with computer graphic (CG) content in a virtual reality environment. EX1006, [0003], [0023], [0024]; EX1047, [0002], [0012], [0013]. Controller 120 “perform[s] various controls of the HMD.” EX1006, [0026]; EX1047, [0015]. The controller is “holdable by hand or wearable on the hand of the user” and operable by “physical contact” (*physical device controller*). EX1006, [0027], [0028]; EX1047, [0016], [0017]. Mine discloses that the controller 120 “includes a button enclosing an optical trackpad ([] ‘OTP’)” and other physical control components. EX1006, [0026]-[0028]; EX1047, [0015]-[0017]. To interact with CG elements, the user can cause “the HMD 100 to display a menu, including a pointer, by a lengthy pressing of the button of the OTP” or can “select[ing] [] [an] item” “by pushing the button of the OTP down.” EX1006, [0027]; EX1047, [0016].

Mine discloses a method for determining “whether the hand of the user is in contact with the OTP of the controller 120,” that is, whether the user is making “physical contact” with the controller (*detecting that a user...is not using or has discontinued use of a physical device controller*). EX1006, [0028], [0072]; EX1047,

[0017], [0061]; *see also* EX1006, [0078] (“The state of the hand of the user (***whether or not the hand of the user is in contact*** with the OTP ...) tends to indicate the intention of the user on how the user wants to ***control the HMD*** 100.”); EX1047, [0067]. In Figure 6, Mine outlines the mode selection processing, in which a control unit 211 determines whether the user is in one of three modes, including two hand control modes (ray control mode, tracking control mode), and a physical controller mode (OTP control mode). EX1006, FIG.6; EX1047, FIG. 6. During this processing, “[i]f ***it is determined that the hand of the user is not in contact with the OTP***, processing advances” to determine which hand-controlled modes are being used (i.e., ray control mode or tracking control mode). EX1006, [0072]-[0073]; EX1047, [0061]-[0062]. Because the controller unit 211 determines that the hand of the user is not in contact with the controller, the controller unit is *detecting that a user...is not using or has discontinued use of a physical device controller*. EX1003, ¶¶450-451.

A POSITA would have been motivated to implement Ravasz’s interface method to include Mine’s controller contact detection method because it would have improved user interaction. EX1003, ¶452. Ravasz’s system receives user inputs from controllers 114A and 114B, as well as user hand gestures. EX1007, [0040]. Because Ravasz’s artificial reality system receives user inputs from both gestures and controllers (*see* Sections IX.E.6, IX.E.9), a POSITA would have incorporated a

method to detect when a user had discontinued use of the controllers to switch to a hand gesture mode. EX1003,¶452.

A POSITA would have also understood that such a method could be more accurate. EX1003,¶453. Detecting whether there is physical contact on the controller would be more accurate. *Id.* For example, a controller could still be connected to the HMD 112 by Bluetooth or through a wired link, but the user could have switched to hand gestures. A POSITA would have understood such a detection method would have benefited a system that allowed for multiple user input methods. *Id.*,¶454. If a user stopped using a controller and began using their hands, efficiently detecting the switch would have prevented errant control inputs from the controller, thereby making selection more precise and improving the user's experience. *Id.*

A POSITA would have had a reasonable expectation of success in making this combination because both Ravasz and Mine disclose physical controllers as user input methods for their artificial and virtual reality systems. EX1007, [0040]; EX1006, [0023], [0026]; EX1047, [0012], [0015]; EX1003,¶455. Incorporating a method to determine whether the user was in contact with Ravasz's controllers to better select/switch control modes would have been a simple mechanical (adding an OTP to Ravasz's controller) and programming change (recognizing OTP input and adding an algorithm to switch modes) that would have been well within the abilities of a POSITA. EX1003,¶¶456-457.

3. 12[b]

Ravasz with Tang renders obvious this limitation. EX1003,¶¶459-464.

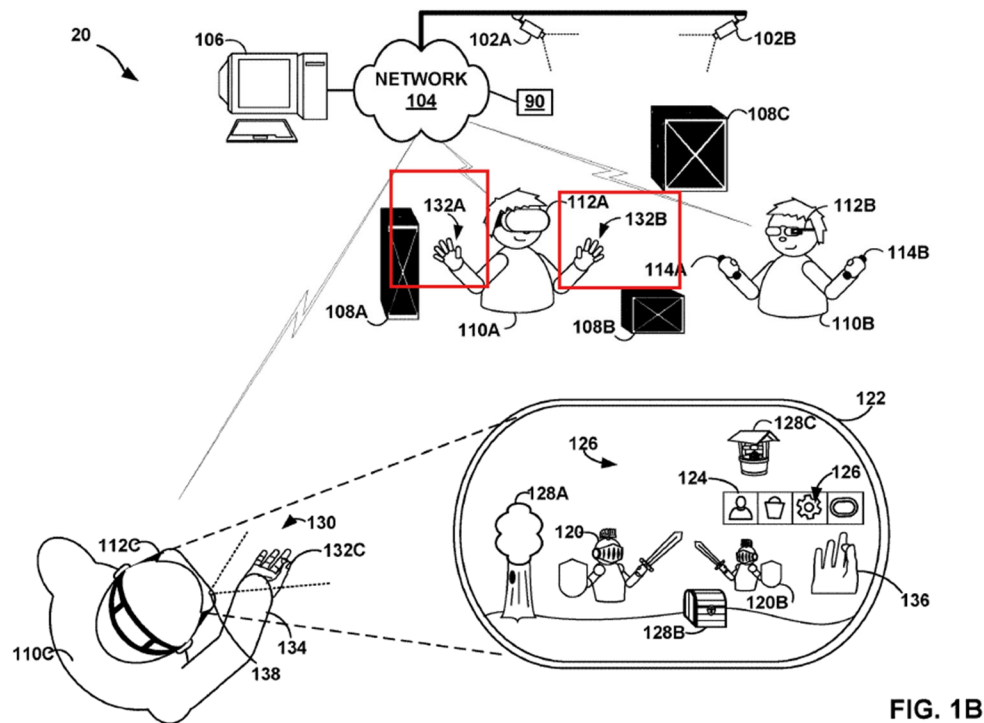
As discussed, in Ravasz-Tang, the HMD displays to a user UI elements (also called menu items) in a UI menu, including an application-launching and configurable-settings elements (*displaying to the user...a software application and a virtuality reality platform setting*). See Sections IX.E.2-4; EX1003,¶460. The HMD would be *displaying...a body interactive icon* so that the user can interact with these elements using gestures. See Section IX.E.5; EX1003,¶460. Further, these gesture-interaction elements in the UI menu displayed in near interaction mode is within a predetermined threshold distance to the user's hand that is tracked by image capture devices 138 (*at least one body interactive icon is displayed to appear closer to a body part of the user that is tracked by a camera*). See Sections IX.E.5-6; EX1003,¶461. The user's hand interacts with the UI menu using pre-defined gestures including movements normal to the UI menu such as tapping and pressing (*to interact with the first body interactive icon*). See Section IX.E.8; *Id.* The UI icon display is displayed further than the predetermined threshold distance in the far interaction mode (*than...a software application and virtual reality platform system setting displayed to the user*) when the user is using controller 114 in the artificial reality environment (*when the user was using the physical device controller in the virtual reality environment*). See Section IX.E.6-7; EX1003,¶462. The HMD would

be displaying a *physical controller icon* so that the user can interact with these elements using a controller. *See* Section IX.E.6; EX1003,¶462. HMD 112 would display the UI menu in Tang’s near interaction mode after determining that the user’s hand is not in contact with the controller (discuss in Section X.C.3) (*after said detecting*) so that the system can switch to gestures in near interaction mode. *See* Section X.C.2; EX1003,¶463.

4. 12[c]

Ravasz discloses this limitation. EX1003,¶¶465-470.

Ravasz discloses a system that “receive[s] user inputs in the form of gestures by [sic] hands 132A, 132B of user 110A” (*receiving a control input*). EX1007, [0040]; *see also* FIG. 1B, [0035]; EX1003,¶466.



EX1007, FIG. 1B (annotated)

Further, Ravasz discloses “selection of UI element 126” (*selecting the body interactive icon*) “of UI menu 124”. EX1007, [0032]; *see also* Section X.C.3.

Ravasz discloses that the user “perform[s] a selection gesture,” to select “a particular menu item of the UI menu” using movement of a finger or a hand (*that appears as a direct body interaction*) *Id.*, [0034], [0041], [0076]. Therefore, the user “selects” the “UI element 126” (*body interactive icon*) as an *input* to the system. *See* Section X.C.3; EX1003, ¶¶466-468.

Ravasz teaches that the “selection gesture can be a movement of the hand in a direction that is substantially normal to the plane of the UI menu.” EX1007,

[0076]. A POSITA would understand that such a movement *appears as a direct touching body interaction from the user to the* UI element because it is used to simulate pushing, pulling, tapping, or pressing. EX1003,¶469.

5. 12[d]

Ravasz discloses this limitation. *See* Sections IX.E.7-8; EX1003,¶¶471-473. As discussed, Ravasz discloses that if the UI element is associated with a software application (*if the body interactive icon represents a software application*), the system will *launch* the application associated with the element (*a respective software application that is linked to the body interactive icon*) in response to receiving the selection gesture (*the control input*). *See* Sections IX.E.7-8; *see also* EX1007, FIG. 7E.

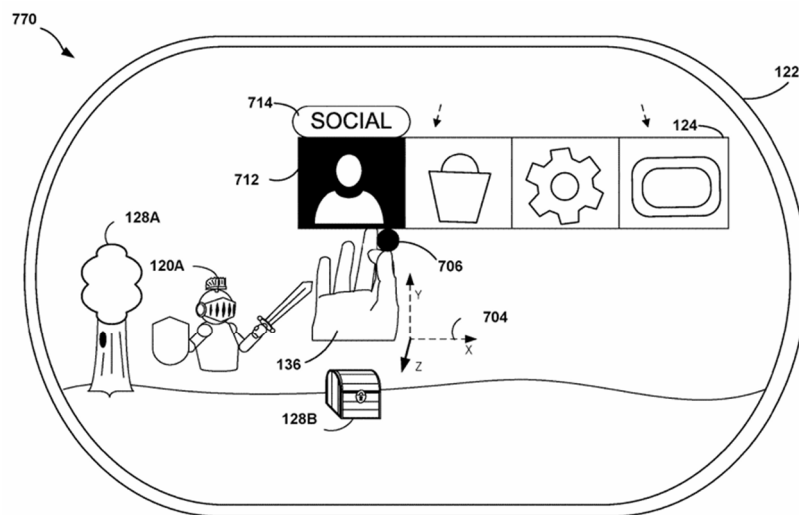


FIG. 7E

EX1007, FIG. 7E

6. 12[e]

Ravasz discloses this limitation. *See* Sections IX.E.9, X.C.3; EX1003, ¶¶474-476. As discussed, Ravasz discloses that if the UI element is associated with settings to configure the artificial reality environment (*if the body interactive icon represents a virtual reality platform system setting*), modifying (*activating*) the configurable settings associated with the UI menu item (*a virtual reality platform system setting that is linked to the body interactive icon*) in response to receiving the selection gesture (*the control input*). *See* Sections IX.E.9, X.C.3; EX1003, ¶475; *see also* EX1007, FIG. 7G.

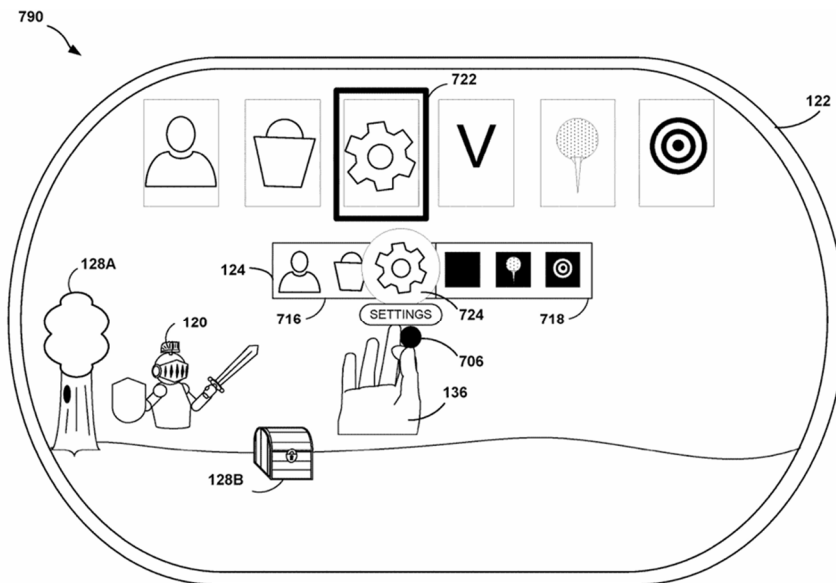


FIG. 7G

EX1007, FIG. 7G

D. Dependent Claims 13-20:

1. 13

Ravasz with Tang renders obvious this limitation. EX1003, ¶¶477-481. As discussed, Ravasz discloses pre-defined gestures performed directly on the UI element to modify configurable settings or launch an application (*the body interactive icon...that (i) activates a virtual reality platform system setting...and (ii) launches a software application*). See Sections X.C.5-6; EX1003, ¶478. The icon would have been displayed as a three-dimensional shape such as a button. See Section IX.F.1; EX1003, ¶479; *see also* EX1007, [0026] (“UI element 126 may be a...three-dimensional shape that is manipulatable by a user”); EX1005, [0028] (“The plurality of virtual interaction objects 58 may include...at least one virtual button 64.”). To the extent that Ravasz does not explicitly teach pressing (*pressed by the user's hand or finger if the body interactive icon represents a software application*), Tang renders this obvious. See Section IX.E.8-9; EX1003, ¶480; *see also* EX1005, [0035] (“The plurality of gestures may include a pressing motion by a finger...”).

2. 14

Ravasz with Tang renders obvious this limitation. See Section IX.D.2; EX1003, ¶¶482-484. A POSITA would have implemented Tang’s scrolling functionality to navigate Ravasz’s UI menu. EX1005, [0029], FIG. 6; EX1003, ¶483. Ravasz further teaches rendering on Tang’s buttons 64 for scrolling “arrows

[that]...indicate the direction that the user's index figure...should be moved.”
EX1007, [0069]; EX1003, ¶483.

3. 15/19/20

Ravasz with Tang renders obvious these limitations. *See* Section IX.D.3; EX1003, ¶¶485-488. The UI menu items are within a predetermined threshold distance from the user's hand (*determining an apparent distance relative to the user*). *See* Section IX.D.3; EX1003, ¶486. The predetermined threshold distance is “in a range of 460 mm to 580 mm,” representing an “average distance from the shoulder of an average user to the grip center of a hand” (*touching distance of the user*). EX1005, [0031]; EX1003, ¶487; *see also* Section IX.D.3.

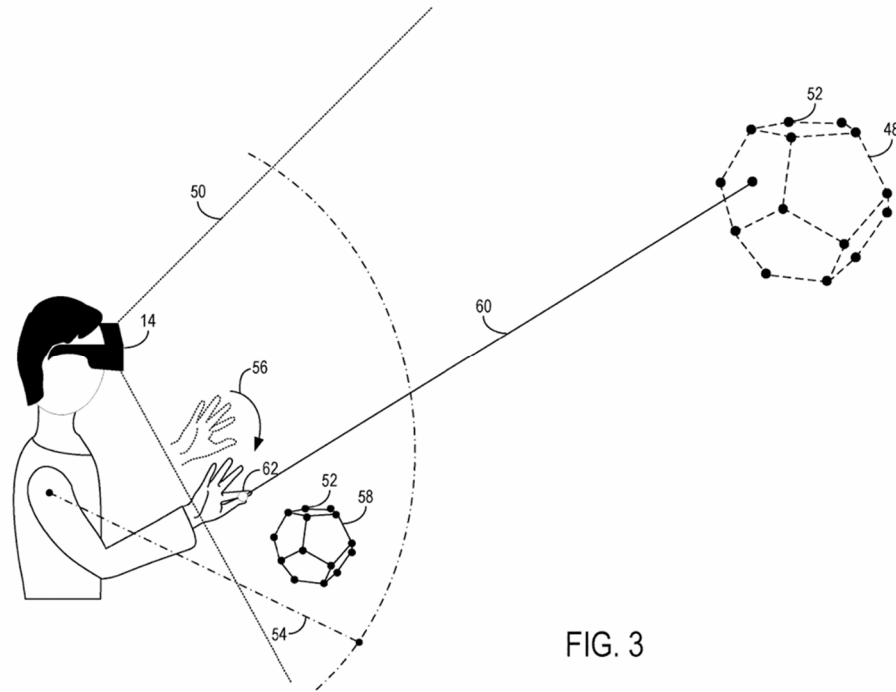
4. 16

Ravasz with Tang renders obvious this limitation. *See* Section IX.D.3; EX1003, ¶¶489-490. Because predetermined threshold distance represents the “average distance from the shoulder of an average user to the grip center of a hand,” EX1005, [0031], a POSITA would have understood that length to be a user's arm's length. EX1003, ¶489.

5. 17

Ravasz with Tang renders obvious this limitation. EX1003, ¶¶491-494. Tang discloses predetermined threshold distance 54 is also limited by the user's field of view. EX1005, [0021], FIG. 3. A POSITA would have understood that the user's

field of view is based upon the user's height, and therefore the predetermined threshold distance 54 will be limited by the user's height (*apparent distance is determined based on a height of the user*). EX1003, ¶¶492-493. Additionally, a person's height is approximately their arm span. See Section X.D.6.



EX1005, FIG. 3

6. 18

Ravasz with Tang renders obvious this limitation. EX1003, ¶¶495-498. The predetermined threshold distance (*apparent distance*) is “in a range of 460 mm to 580 mm,” (*determined based on a predetermined value*) representing an “average distance from the shoulder...to the grip center of a hand” (*average arms' length*) of “an average user.” EX1005, [0031]; EX1003, ¶497; see also Section IX.D.3. A

POSITA would have understood that “an average user” is based on the average of a *predetermined group of persons*. EX1003,¶497.

XI. CONCLUSION

For these reasons, Petitioner respectfully requests institution.

Dated : August 26, 2026

Respectfully submitted,

By: / Lisa K. Nguyen /
Lisa K. Nguyen (Reg. No. 58,018)
PAUL HASTINGS LLP
1117 S. California Avenue
Palo Alto, CA 94304
Telephone: 650-320-1800
lisanguyen@paulhastings.com

Counsel for Petitioner
META PLATFORMS, INC.

APPENDIX: LISTING OF CLAIMS FROM U.S. PATENT NO. 11,656,742

1[pre]. A virtual reality environment selection interface comprising:

[a.i] a first software application icon [a.ii] displayed in a virtual reality environment to a user using a virtual reality hardware device, [a.iii] wherein the first software application icon appears closer to a body part of the user [a.iv] that is tracked by a camera of the virtual reality hardware device to interact with the first software application icon [a.v] than another second software application icon that is displayed to the user when the user is using a physical controller in the virtual reality environment; and

[b.i] the first software application icon operatively linked to launch a respective software application [b.ii] when interacted with by a pushing, pulling, tapping, pressing or sliding action of the body part of the user directly on the first software application icon, [b.iii] wherein the first software application icon and the second software application icon are operatively linked to launch the same respective software application.

2. The virtual reality environment selection interface of claim 1, wherein the first software application icon is displayed as a 3-dimensional button that launches the respective software application when pressed by the user's hand or finger.

3. The virtual reality environment selection interface of claim 2, further comprising up and down arrows displayed as buttons adjacent a grouping of a plurality of one or both of software application and virtual reality platform system setting icons that includes the first software application icon, wherein the up and down arrows are operatively linked to scrolling the plurality of one or both of software application and virtual reality platform system setting icons in the grouping when directly interacted with by a user's hand or finger.

4. The virtual reality environment selection interface of claim 1, further comprising up and down arrows displayed as buttons adjacent a grouping of a plurality of one or both of software application and virtual reality platform system setting icons that includes the first software application icon, wherein the up and down arrows are operatively linked

to scrolling the plurality of one or both of software application and virtual reality platform system setting icons in the grouping when directly interacted with by a user's hand or finger.

5. The virtual reality environment selection interface of claim 4, wherein the plurality of one or both of software application and virtual reality platform system setting icons in the grouping are displayed within an apparent distance of a user's arm length to the user.

6. The virtual reality environment selection interface of claim 1, wherein the first software application icon is displayed within an apparent distance of a user's arm length to the user.

7. The virtual reality environment selection interface of claim 2, wherein the first software application icon is displayed within an apparent distance of a user's arm length to the user.

8[pre] A virtual reality environment selection interface comprising:

[a.i] a first software application icon and [a.ii] a first virtual reality platform system setting icon [a.iii] displayed together in a virtual reality environment to a user using a virtual reality hardware device, [a.iv] wherein the first software application icon and the first virtual reality platform system setting icon appear closer to a body part of the user that is tracked by a camera of the virtual reality hardware device to interact with the first software application icon and the first virtual reality platform system setting icon [a.v] than a second software application icon and a second virtual reality platform system setting icon that is displayed to the user when the user is using a physical controller in the virtual reality environment;

[b.i] the first virtual reality platform system setting icon being operatively linked to activate a setting [b.ii] when interacted with by a pushing, pulling, tapping, pressing or sliding action of the body part of the user directly on the first virtual reality platform system setting icon; and

[c] the first software application icon being operatively linked to launch a software application when interacted with by a pushing, pulling, tapping, pressing or sliding action of the body part of the user directly on the first software application icon.

9. The virtual reality environment selection interface of claim 8, wherein the first virtual reality platform system setting icon is displayed as a 3-dimensional button that activates the setting when pressed by the user's hand or finger and the first software application icon is displayed as a 3-dimensional button that launches the software application when pressed by the user's hand or finger.

10. The virtual reality environment selection interface of claim 9, wherein the first grouping of the plurality of icons are displayed within an apparent distance of a user's arm length to the user.

11. The virtual reality environment selection interface of claim 8, wherein the first software application icon and the first virtual reality platform system setting icon are displayed within an apparent distance of a user's arm length to the user.

12 [pre] A method for providing an interactive control interface in a virtual reality environment comprising:

[a] detecting that a user using a virtual reality environment via a virtual reality hardware device is not using or has discontinued use of a physical device controller used to interact with the virtual reality environment;

[b] displaying to the user a body interactive icon of a software application and a virtual reality platform system setting after said detecting, wherein the body interactive icon is displayed to appear closer to a body part of the user that is tracked by a camera to interact with the body interactive icon than a physical controller icon of a software application and virtual reality platform system setting displayed to the user when the user was using the physical device controller in the virtual reality environment, wherein the body interactive icon and the physical

controller icon both are linked to launch the same software application or virtual reality platform setting;

[c] receiving a control input selecting the body interactive icon from the user that appears as a direct touching body interaction from the user to the body interactive icon;

[d] launching, in response to receiving the control input, a virtual reality platform system setting that is linked to the body interactive icon if the body interactive icon represents a software application; and

[e] activating, in response to receiving the control input, a virtual reality platform system setting that is linked to the body interactive icon if the body interactive icon represents a virtual reality platform system setting.

13. The method of claim 12, wherein the body interactive icon is displayed as a 3-dimensional button that (i) activates a virtual reality platform system setting when pressed by the user's hand or finger if the body interactive icon represents a virtual reality platform system setting and (ii) launches a software application when pressed by the user's hand or finger if the body interactive icon represents a software application.

14. The method of claim 13, further comprising up and down arrows displayed as buttons adjacent a plurality of body interactive icons that include the body interactive icon, wherein the up and down arrows are operatively linked to scrolling the plurality of body interactive icons when directly interacted with by a user's hand or finger.

15. The method of claim 14, further comprising determining an apparent distance relative to the user that the body interactive icon should be displayed to be within a touching distance of the user.

16. The method of claim 15, wherein the apparent distance is determined based on an arm's length of a user.

17. The method of claim 15, wherein the apparent distance is determined based on a height of the user.

18. The method of claim 15, wherein the apparent distance is determined based on a predetermined value representing average arms' length for a predetermined group of persons.

19. The method of claim 12, further comprising determining an apparent distance relative to the user that the body interactive icon should be displayed to be within a touching distance of the user.

20. The method of claim 13, further comprising determining an apparent distance relative to the user that the body interactive icon should be displayed to be within a touching distance of the user.

CERTIFICATE OF COMPLIANCE

Pursuant to 37 C.F.R. § 42.24(d), the undersigned certifies that foregoing
**PETITION FOR INTER PARTES REVIEW OF U.S. PATENT NO.
11,656,742** exclusive of the parts exempted as provided in 37 C.F.R. §42.24(a),
contains 13,997 words and therefore complies with the type-volume limitations of
37 C.F.R. §42.24(a).

Dated : August 26, 2026

Respectfully submitted,

By: / Lisa K. Nguyen /
Lisa K. Nguyen (Reg. No. 58,018)
PAUL HASTINGS LLP
1117 S. California Avenue
Palo Alto, CA 94304
Telephone: 650-320-1800
lisanguyen@paulhastings.com

Counsel for Petitioner
META PLATFORMS, INC.

CERTIFICATE OF SERVICE

I hereby certify that a true and correct copy of the foregoing **Petition for *Inter Partes* Review of U.S. Patent No. 11,656,742 and Exhibits** are being served on August 26, 2026 via Express overnight mail on counsel of record for U.S. Patent No. 11,656,742 as addressed below:

**POLSINELLI PC
PO Box 140310
KANSAS CITY, MO 64114-0310**

A courtesy copy is also being served on counsel for the patent holder in the pending litigation:

Rajeev Gupta, raj.gupta@finnegan.com
Karthik Kumar, karthik.kumar@finnegan.com
Daniel F. Roland, daniel.roland@finnegan.com
Daniel F. Klodowski, daniel.klodowski@finnegan.com
C. Collette Corser, collette.corser@finnegan.com
Victor M. Palace, victor.palace@finnegan.com
Jordan M. Cowger, jordan.cowger@finnegan.com **FINNEGAN, HENDERSON,
FARABOW, GARRETT & DUNNER, LLP
901 New York Avenue, N.W., Washington, D.C. 20001-4413**

Eric H. Findlay, efindlay@findlaycraft.com
**FINDLAY CRAFT, P.C.
7270 Crosswater Avenue, Suite B, Tyler, TX 75703**

Dated : August 26, 2026

Respectfully submitted,

By: / Lisa K. Nguyen/
Lisa K. Nguyen (Reg. No. 58,018)
PAUL HASTINGS LLP
1117 S. California Avenue
Palo Alto, CA 94304

U.S. Patent No. 11,656,742
Petition for *Inter Partes* Review

Telephone: 650-320-1800
lisanguyen@paulhastings.com

Counsel for Petitioner
META PLATFORMS, INC.