



(51) International Patent Classification:

G06F 3/01 (2006.01) G06F 3/0346 (2013.01)
G06F 3/03 (2006.01) G06F 3/04842 (2022.01)

(21) International Application Number:

PCT/US2024/013757

(22) International Filing Date:

31 January 2024 (31.01.2024)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

63/485,414 16 February 2023 (16.02.2023) US
18/310,622 02 May 2023 (02.05.2023) US

(71) Applicant: **META PLATFORMS TECHNOLOGIES, LLC** [US/US]; 1 Meta Way, Menlo Park, California 94025 (US).

(72) Inventors: **EMAMI, Ali**; 1 Meta Way, Menlo Park, California 94025 (US). **SELIG, Aaron**; 1 Meta Way, Menlo Park, California 94025 (US). **BASRAVI, Ahad Habib**; 1 Meta Way, Menlo Park, California 94025 (US). **O'DONNELL, James Michael-K**; 1 Meta Way, Menlo Park, California 94025 (US). **LANE, Nathaniel**; 1 Meta Way, Menlo Park, California 94025 (US). **HARRIS, Todd**; 1 Meta Way, Menlo Park, California 94025 (US). **MITTAL, Kimaya Vishal**; 1 Meta Way, Menlo Park, California 94025 (US).

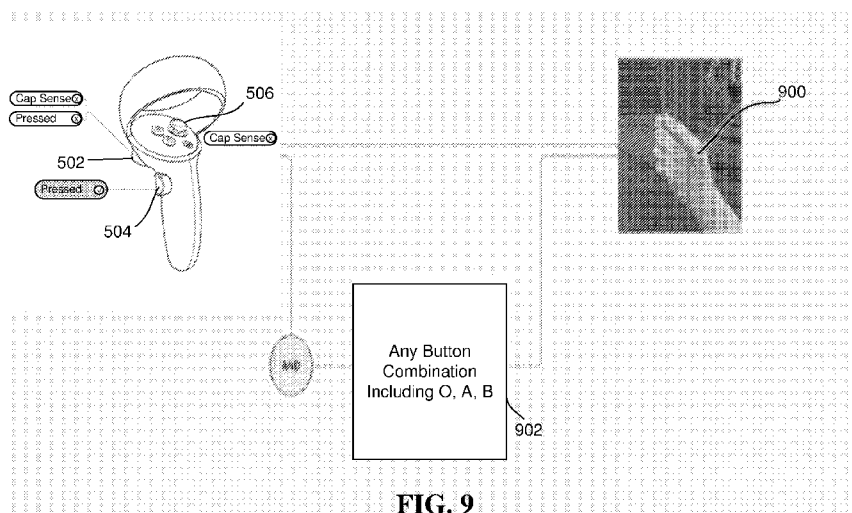
(74) Agent: **COLBY, Steven et al.**; Rimon Law, 420 West Main Street, Suite 101B, Boise, Idaho 83702 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG,

(54) Title: SIMULTANEOUS CONTROLLER AND TOUCH INTERACTIONS

Input State

Visually Rendered



(57) Abstract: An artificial reality (XR) system includes XR controllers and hand gestures and corresponding representation of the hand gestures in XR space. Example implementations allow a user to use controllers for distance interaction and interactions that require controller buttons, but also recognize when the user extends her finger and wants to manually touch things - using touch in conjunction with controllers. Thus, if the user has a controller in her hand and wants to touch a UI element, she can also use her finger. In particular, the system identifies the finger is not touching the controller (e.g., via capacitance sensors) and that a hand pose is present (e.g., finger extended - from camera), thereby processing hand input from the finger without the user having to put down the controller.

KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY,
MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA,
NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO,
RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH,
TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS,
ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

1 SIMULTANEOUS CONTROLLER AND TOUCH INTERACTIONS

2 TECHNICAL FIELD

3 **[0001]** The present disclosure is directed to artificial reality controllers and hand
4 gestures and corresponding representation of the hand gestures in artificial reality
5 space.

6 BACKGROUND

7 **[0002]** Virtual reality (VR) environments can be provided by a variety of
8 systems, such as projectors, head mounted displays, "cave" systems, etc. Users
9 can interact with an artificial reality environment, such as by selecting objects,
10 moving, rotating, resizing, actuating controls, changing colors or skins, defining
11 interactions between virtual objects, setting virtual forces to act on virtual objects,
12 or practically any other imaginable action. Various interaction modalities exist for
13 these taking such actions in a VR environment. For example, some systems can
14 employ one or more of gaze controls, hand-held hardware devices, gesture
15 controls, wearable devices (e.g., wrist bands), voice controls, etc.

16 SUMMARY

17 **[0003]** It is an object of the present invention to provide a method for providing
18 direct touch interaction with virtual objects while a user holds a controller, and a
19 computer-readable medium, and a computing system for providing direct touch
20 interaction with virtual objects while a user holds a controller. This object can be
21 achieved by the features as defined by the independent claims. Further
22 enhancements are characterized by the dependent claims. The invention is defined
23 by the claims.

24 **[0004]** In accordance with a first aspect of the disclosure, a method for providing
25 direct touch interaction with virtual objects while a user holds a controller is
26 disclosed. The method comprising: detecting a user gesture while the user holds
27 the controller; providing, in an artificial reality environment, a representation of a
28 user's hand corresponding to the gesture; and receiving an input, while the user
29 holds the controller, in relation to a virtual object based on an interaction of the
30 representation of the user's hand with the virtual object.

31 **[0005]** In some embodiments, the controller detecting the user gesture may
32 comprise processing contact data on one or more locations on the controller.

33 **[0006]** In some embodiments, the one or more locations may comprise a first

1 button of the controller.

2 **[0007]** In some embodiments, the contact data may be generated in response
3 to a transition of contact of a first finger of the user on the first button to the first
4 finger not contacting the first button.

5 **[0008]** In some embodiments, the first finger may comprise an index finger, and
6 the gesture may comprise a pointing gesture.

7 **[0009]** In some embodiments, the input may comprise a selection of the virtual
8 object.

9 **[0010]** In some embodiments, the controller may comprise at least one
10 capacitive sensor that generates the contact data used for determining whether the
11 user is contacting one or more locations on the controller.

12 **[0011]** In some embodiments, detecting the user gesture may further comprise:
13 collecting visual data of the user gesture using a camera; and determining, using
14 one or more classifiers, whether the user is contacting one or more locations on the
15 controller.

16 **[0012]** In some embodiments, providing, in the artificial reality environment, the
17 representation of a user's hand corresponding to the gesture may include providing
18 a representation of the controller.

19 **[0013]** In some embodiments, providing, in the artificial reality environment, the
20 representation of a user's hand corresponding to the gesture may comprise:
21 determining a distance between the representation of the user's hand and the
22 virtual object; in response to determining the distance is greater than a threshold
23 distance, rendering the representation of the user's hand in the artificial reality
24 environment as holding a virtual controller; and in response to determining the
25 distance is less than the threshold distance, rendering the representation of the
26 user's hand without the virtual controller.

27 **[0014]** In some embodiments, the gesture may comprise a neutral, pointing,
28 pinch, swipe or full hand grip.

29 **[0015]** In some embodiments, the detecting the user gesture may comprise
30 detecting a rotation of the controller.

31 **[0016]** In accordance with a second aspect of the disclosure, there is disclosed
32 a computer-readable medium comprising instructions that, when executed by a
33 computer, cause the computer to perform operations comprising: detecting a user

gesture while the user holds the controller; providing, in an artificial reality environment, a representation of a user's hand corresponding to the gesture; and receiving an input, while the user holds the controller, in relation to a virtual object based on an interaction of the representation of the user's hand with the virtual object.

[0017] In some embodiments, detecting the user gesture may comprise processing contact data on one or more locations on the controller.

[0018] In some embodiments, the one or more locations may comprise a first button of the controller.

[0019] In some embodiments, the contact data may be generated in response to a transition of contact of a first finger of the user on the first button to the first finger not contacting the first button.

[0020] In some embodiments, the first finger may comprise an index finger, and the gesture comprises a pointing gesture.

[0021] In some embodiments, the input may comprise a selection of the virtual object.

[0022] In some embodiments, the controller may comprise at least one capacitive sensor that generates the contact data used for determining whether the user is contacting one or more locations on the controller.

[0023] In accordance with a third aspect of the disclosure, there is disclosed a computing system for providing direct touch interaction with virtual objects while a user holds a controller, comprising: one or more processors; and one or more computer-readable medium storing instructions that, when executed by the one or more processors, configure the computing system to perform operations comprising: detecting a user gesture while the user holds the controller; providing, in an artificial reality environment, a representation of a user's hand corresponding to the gesture; and receiving an input, while the user holds the controller, in relation to a virtual object based on an interaction of the representation of the user's hand with the virtual object.

BRIEF DESCRIPTION OF THE DRAWINGS

[0024] Figure 1 is a block diagram illustrating an overview of devices on which some implementations of the present technology can operate.

[0025] Figure 2A is a wire diagram illustrating a virtual reality headset which can

1 be used in some implementations of the present technology.

2 **[0026]** Figure 2B is a wire diagram illustrating a mixed reality headset which can
3 be used in some implementations of the present technology.

4 **[0027]** Figure 2C is a wire diagram illustrating controllers which, in some
5 implementations, a user can hold in one or both hands to interact with an artificial
6 reality environment.

7 **[0028]** Figure 3 is a block diagram illustrating an overview of an environment in
8 which some implementations of the present technology can operate.

9 **[0029]** Figure 4 is a block diagram illustrating components which, in some
10 implementations, can be used in a system employing the disclosed technology.

11 **[0030]** Figures 5A and 5B illustrate two hand gestures in connection with a
12 controller for some implementations of the present technology.

13 **[0031]** Figure 6 illustrates an alternative avatar when the user's index finger is
14 no longer contacting a button of a controller for some implementations of the
15 present technology.

16 **[0032]** Figure 7A illustrates a "pinch" gesture detected via the controller for
17 some implementations of the present technology.

18 **[0033]** Figure 7B illustrates a "poke/press" gesture detected via the controller
19 for some implementations of the present technology.

20 **[0034]** Figure 8A illustrates a "neutral" gesture detected via the controller for
21 some implementations of the present technology.

22 **[0035]** Figure 8B illustrates a "full-hand grab" gesture detected via the controller
23 for some implementations of the present technology.

24 **[0036]** Figures 9-12 illustrate various different input states of the controller, and
25 the corresponding visually rendered avatar for some implementations of the present
26 technology.

27 **[0037]** Figures 13A and 13B illustrate, for some implementations of the present
28 technology, a transition between the avatar showing both the controller and hand
29 (Figure 13A) and then, as the controller gets closer to the UI, the hand avatar
30 without the controller (Figure 13B).

31 **[0038]** Figure 14 is a flow diagram illustrating a process used in some
32 implementations for facilitating simultaneous controller and touch interactions by
33 providing direct touch interaction with virtual objects while a user holds the

1 controller.

2 **[0039]** The techniques introduced here may be better understood by referring
3 to the following Detailed Description in conjunction with the accompanying
4 drawings, in which like reference numerals indicate identical or functionally similar
5 elements.

6 DETAILED DESCRIPTION

7 **[0040]** Aspects of the present disclosure are directed to artificial reality (XR)
8 controllers and hand gestures, and providing the ability to use touch interactions,
9 via hand gestures, while also holding the XR controller.

10 **[0041]** Example implementations allow a user to use controllers for distance
11 interaction and interactions that require controller buttons, but also recognize when
12 the user extends her finger and wants to manually touch things - using touch in
13 conjunction with controllers. Thus, if the user has a controller in her hand and wants
14 to touch a user interface (UI) element, she can also use her finger. In particular,
15 the system identifies the finger is not touching the controller (e.g., via capacitance
16 sensors) and/or that a hand pose is present (e.g., a finger is extended as detected
17 from a camera), thereby processing hand input from the finger without the user
18 having to put down the controller.

19 **[0042]** Example implementations can further automatically switch between an
20 avatar of a hand holding the controller to a hand only avatar without rendering the
21 controller (i.e., direct interaction hand visualization) based on distance from the UI
22 or a detection of a hand pose. The rendering of the hand only avatar without
23 controller can vary depending on the sensed position of the user's hand on the
24 controller.

25 **[0043]** Example implementations can further automatically switch between ray
26 casting using the controller for UI input to a finger for UI input based on a distance
27 between the controller and the UI.

28 **[0044]** Embodiments of the disclosed technology may include or be
29 implemented in conjunction with an artificial reality system. Artificial reality or extra
30 reality (XR) is a form of reality that has been adjusted in some manner before
31 presentation to a user, which may include, e.g., virtual reality (VR), augmented
32 reality (AR), mixed reality (MR), hybrid reality, or some combination and/or
33 derivatives thereof. Artificial reality content may include completely generated

1 content or generated content combined with captured content (e.g., real-world
2 photographs). The artificial reality content may include video, audio, haptic
3 feedback, or some combination thereof, any of which may be presented in a single
4 channel or in multiple channels (such as stereo video that produces a three-
5 dimensional effect to the viewer). Additionally, in some embodiments, artificial
6 reality may be associated with applications, products, accessories, services, or
7 some combination thereof, that are, e.g., used to create content in an artificial reality
8 and/or used in (e.g., perform activities in) an artificial reality. The artificial reality
9 system that provides the artificial reality content may be implemented on various
10 platforms, including a head-mounted display (HMD) connected to a host computer
11 system, a standalone HMD, a mobile device or computing system, a "cave"
12 environment or other projection system, or any other hardware platform capable of
13 providing artificial reality content to one or more viewers.

14 **[0045]** "Virtual reality" or "VR," as used herein, refers to an immersive
15 experience where a user's visual input is controlled by a computing system.
16 "Augmented reality" or "AR" refers to systems where a user views images of the
17 real world after they have passed through a computing system. For example, a
18 tablet with a camera on the back can capture images of the real world and then
19 display the images on the screen on the opposite side of the tablet from the camera.
20 The tablet can process and adjust or "augment" the images as they pass through
21 the system, such as by adding virtual objects. "Mixed reality" or "MR" refers to
22 systems where light entering a user's eye is partially generated by a computing
23 system and partially composes light reflected off objects in the real world. For
24 example, a MR headset could be shaped as a pair of glasses with a pass-through
25 display, which allows light from the real world to pass through a waveguide that
26 simultaneously emits light from a projector in the MR headset, allowing the MR
27 headset to present virtual objects intermixed with the real objects the user can see.
28 "Artificial reality," "extra reality," or "XR," as used herein, refers to any of VR, AR,
29 MR, or any combination or hybrid thereof.

30 **[0046]** Conventional systems allow interaction via controller, such as via an XR
31 controller or a conventional mouse. However, for some interaction tasks, such as
32 selecting an object, the user's hand/finger is more advantageous. In contrast,
33 example implementation can improve computing and/or computer system by

1 allowing a seamless transition between using a controller and using a hand for
2 interacting with objects, thus optimizing the type of interaction (e.g., controller or
3 hand) based on the environment and user interface scenario.

4 **[0047]** Several implementations are discussed below in more detail in reference
5 to the figures. Figure 1 is a block diagram illustrating an overview of devices on
6 which some implementations of the disclosed technology can operate. The devices
7 can comprise hardware components of a computing system 100 that provide
8 simultaneous controller and touch interactions. In various implementations,
9 computing system 100 can include a single computing device 103 or multiple
10 computing devices (e.g., computing device 101, computing device 102, and
11 computing device 103) that communicate over wired or wireless channels to
12 distribute processing and share input data. In some implementations, computing
13 system 100 can include a stand-alone headset capable of providing a computer
14 created or augmented experience for a user without the need for external
15 processing or sensors. In other implementations, computing system 100 can
16 include multiple computing devices such as a headset and a core processing
17 component (such as a console, mobile device, or server system) where some
18 processing operations are performed on the headset and others are offloaded to
19 the core processing component. Example headsets are described below in relation
20 to Figures 2A and 2B. In some implementations, position and environment data
21 can be gathered only by sensors incorporated in the headset device, while in other
22 implementations one or more of the non-headset computing devices can include
23 sensor components that can track environment or position data.

24 **[0048]** Computing system 100 can include one or more processor(s) 110 (e.g.,
25 central processing units (CPUs), graphical processing units (GPUs), holographic
26 processing units (HPUs), etc.) Processors 110 can be a single processing unit or
27 multiple processing units in a device or distributed across multiple devices (e.g.,
28 distributed across two or more of computing devices 101-103).

29 **[0049]** Computing system 100 can include one or more input devices 120 that
30 provide input to the processors 110, notifying them of actions. The actions can be
31 mediated by a hardware controller that interprets the signals received from the input
32 device and communicates the information to the processors 110 using a
33 communication protocol. Each input device 120 can include, for example, a mouse,

1 a keyboard, a touchscreen, a touchpad, a wearable input device (e.g., a haptics
2 glove, a bracelet, a ring, an earring, a necklace, a watch, etc.), a camera (or other
3 light-based input device, e.g., an infrared sensor), a microphone, or other user input
4 devices.

5 **[0050]** Processors 110 can be coupled to other hardware devices, for example,
6 with the use of an internal or external bus, such as a PCI bus, SCSI bus, or wireless
7 connection. The processors 110 can communicate with a hardware controller for
8 devices, such as for a display 130. Display 130 can be used to display text and
9 graphics. In some implementations, display 130 includes the input device as part
10 of the display, such as when the input device is a touchscreen or is equipped with
11 an eye direction monitoring system. In some implementations, the display is
12 separate from the input device. Examples of display devices are: an LCD display
13 screen, an LED display screen, a projected, holographic, or augmented reality
14 display (such as a heads-up display device or a head-mounted device), and so on.
15 Other I/O devices 140 can also be coupled to the processor, such as a network chip
16 or card, video chip or card, audio chip or card, USB, firewire or other external
17 device, camera, printer, speakers, CD-ROM drive, DVD drive, disk drive, etc.

18 **[0051]** In some implementations, input from the I/O devices 140, such as
19 cameras, depth sensors, IMU sensor, GPS units, LiDAR or other time-of-flights
20 sensors, etc. can be used by the computing system 100 to identify and map the
21 physical environment of the user while tracking the user's location within that
22 environment. This simultaneous localization and mapping (SLAM) system can
23 generate maps (e.g., topologies, grids, etc.) for an area (which may be a room,
24 building, outdoor space, etc.) and/or obtain maps previously generated by
25 computing system 100 or another computing system that had mapped the area.
26 The SLAM system can track the user within the area based on factors such as GPS
27 data, matching identified objects and structures to mapped objects and structures,
28 monitoring acceleration and other position changes, etc.

29 **[0052]** Computing system 100 can include a communication device capable of
30 communicating wirelessly or wire-based with other local computing devices or a
31 network node. The communication device can communicate with another device
32 or a server through a network using, for example, TCP/IP protocols. Computing
33 system 100 can utilize the communication device to distribute operations across

1 multiple network devices.

2 **[0053]** The processors 110 can have access to a memory 150, which can be
3 contained on one of the computing devices of computing system 100 or can be
4 distributed across of the multiple computing devices of computing system 100 or
5 other external devices. A memory includes one or more hardware devices for
6 volatile or non-volatile storage, and can include both read-only and writable
7 memory. For example, a memory can include one or more of random access
8 memory (RAM), various caches, CPU registers, read-only memory (ROM), and
9 writable non-volatile memory, such as flash memory, hard drives, floppy disks, CDs,
10 DVDs, magnetic storage devices, tape drives, and so forth. A memory is not a
11 propagating signal divorced from underlying hardware; a memory is thus non-
12 transitory. Memory 150 can include program memory 160 that stores programs and
13 software, such as an operating system 162, simultaneous controller and touch
14 interaction system 164, and other application programs 166. Memory 150 can also
15 include data memory 170 that can include, e.g., data used to determine hand
16 gestures, configuration data, settings, user options or preferences, etc., which can
17 be provided to the program memory 160 or any element of the computing system
18 100.

19 **[0054]** Some implementations can be operational with numerous other
20 computing system environments or configurations. Examples of computing
21 systems, environments, and/or configurations that may be suitable for use with the
22 technology include, but are not limited to, XR headsets, personal computers, server
23 computers, handheld or laptop devices, cellular telephones, wearable electronics,
24 gaming consoles, tablet devices, multiprocessor systems, microprocessor-based
25 systems, set-top boxes, programmable consumer electronics, network PCs,
26 minicomputers, mainframe computers, distributed computing environments that
27 include any of the above systems or devices, or the like.

28 **[0055]** Figure 2A is a wire diagram of a virtual reality head-mounted display
29 (HMD) 200, in accordance with some embodiments. The HMD 200 includes a front
30 rigid body 205 and a band 210. The front rigid body 205 includes one or more
31 electronic display elements of an electronic display 245, an inertial motion unit (IMU)
32 215, one or more position sensors 220, locators 225, and one or more compute
33 units 230. The position sensors 220, the IMU 215, and compute units 230 may be

1 internal to the HMD 200 and may not be visible to the user. In various
2 implementations, the IMU 215, position sensors 220, and locators 225 can track
3 movement and location of the HMD 200 in the real world and in an artificial reality
4 environment in three degrees of freedom (3DoF) or six degrees of freedom (6DoF).
5 For example, the locators 225 can emit infrared light beams which create light points
6 on real objects around the HMD 200. As another example, the IMU 215 can include
7 e.g., one or more accelerometers, gyroscopes, magnetometers, other non-camera-
8 based position, force, or orientation sensors, or combinations thereof. One or more
9 cameras (not shown) integrated with the HMD 200 can detect the light points.
10 Compute units 230 in the HMD 200 can use the detected light points to extrapolate
11 position and movement of the HMD 200 as well as to identify the shape and position
12 of the real objects surrounding the HMD 200.

13 **[0056]** The electronic display 245 can be integrated with the front rigid body 205
14 and can provide image light to a user as dictated by the compute units 230. In
15 various embodiments, the electronic display 245 can be a single electronic display
16 or multiple electronic displays (e.g., a display for each user eye). Examples of the
17 electronic display 245 include: a liquid crystal display (LCD), an organic light-
18 emitting diode (OLED) display, an active-matrix organic light-emitting diode display
19 (AMOLED), a display including one or more quantum dot light-emitting diode
20 (QOLED) sub-pixels, a projector unit (e.g., microLED, LASER, etc.), some other
21 display, or some combination thereof.

22 **[0057]** In some implementations, the HMD 200 can be coupled to a core
23 processing component such as a personal computer (PC) (not shown) and/or one
24 or more external sensors (not shown). The external sensors can monitor the HMD
25 200 (e.g., via light emitted from the HMD 200) which the PC can use, in combination
26 with output from the IMU 215 and position sensors 220, to determine the location
27 and movement of the HMD 200.

28 **[0058]** Figure 2B is a wire diagram of a mixed reality HMD system 250 which
29 includes a mixed reality HMD 252 and a core processing component 254. The
30 mixed reality HMD 252 and the core processing component 254 can communicate
31 via a wireless connection (e.g., a 60 GHz link) as indicated by link 256. In other
32 implementations, the mixed reality system 250 includes a headset only, without an
33 external compute device or includes other wired or wireless connections between

1 the mixed reality HMD 252 and the core processing component 254. The mixed
2 reality HMD 252 includes a pass-through display 258 and a frame 260. The frame
3 260 can house various electronic components (not shown) such as light projectors
4 (e.g., LASERs, LEDs, etc.), cameras, eye-tracking sensors, MEMS components,
5 networking components, etc.

6 **[0059]** The projectors can be coupled to the pass-through display 258, e.g., via
7 optical elements, to display media to a user. The optical elements can include one
8 or more waveguide assemblies, reflectors, lenses, mirrors, collimators, gratings,
9 etc., for directing light from the projectors to a user's eye. Image data can be
10 transmitted from the core processing component 254 via link 256 to HMD 252.
11 Controllers in the HMD 252 can convert the image data into light pulses from the
12 projectors, which can be transmitted via the optical elements as output light to the
13 user's eye. The output light can mix with light that passes through the display 258,
14 allowing the output light to present virtual objects that appear as if they exist in the
15 real world.

16 **[0060]** Similarly to the HMD 200, the HMD system 250 can also include motion
17 and position tracking units, cameras, light sources, etc., which allow the HMD
18 system 250 to, e.g., track itself in 3DoF or 6DoF, track portions of the user (e.g.,
19 hands, feet, head, or other body parts), map virtual objects to appear as stationary
20 as the HMD 252 moves, and have virtual objects react to gestures and other real-
21 world objects.

22 **[0061]** Figure 2C illustrates controllers 270 (including controller 276A and
23 276B), which, in some implementations, a user can hold in one or both hands to
24 interact with an artificial reality environment presented by the HMD 200 and/or HMD
25 250. The controllers 270 can be in communication with the HMDs, either directly
26 or via an external device (e.g., core processing component 254). The controllers
27 can have their own IMU units, position sensors, and/or can emit further light points.
28 The HMD 200 or 250, external sensors, or sensors in the controllers can track these
29 controller light points to determine the controller positions and/or orientations (e.g.,
30 to track the controllers in 3DoF or 6DoF). The compute units 230 in the HMD 200
31 or the core processing component 254 can use this tracking, in combination with
32 IMU and position output, to monitor hand positions and motions of the user. The
33 controllers can also include various buttons (e.g., buttons 272A-F) and/or joysticks

(e.g., joysticks 274A-B), which a user can actuate to provide input and interact with objects.

[0062] In various implementations, the HMD 200 or 250 can also include additional subsystems, such as an eye tracking unit, an audio system, various network components, etc., to monitor indications of user interactions and intentions. For example, in some implementations, instead of or in addition to controllers, one or more cameras included in the HMD 200 or 250, or from external cameras, can monitor the positions and poses of the user's hands to determine gestures and other hand and body motions. As another example, one or more light sources can illuminate either or both of the user's eyes and the HMD 200 or 250 can use eye-facing cameras to capture a reflection of this light to determine eye position (e.g., based on set of reflections around the user's cornea), modeling the user's eye and determining a gaze direction.

[0063] Figure 3 is a block diagram illustrating an overview of an environment 300 in which some implementations of the disclosed technology can operate. Environment 300 can include one or more client computing devices 305A-D, examples of which can include computing system 100. In some implementations, some of the client computing devices (e.g., client computing device 305B) can be the HMD 200 or the HMD system 250. Client computing devices 305 can operate in a networked environment using logical connections through network 330 to one or more remote computers, such as a server computing device.

[0064] In some implementations, server 310 can be an edge server which receives client requests and coordinates fulfillment of those requests through other servers, such as servers 320A-C. Server computing devices 310 and 320 can comprise computing systems, such as computing system 100. Though each server computing device 310 and 320 is displayed logically as a single server, server computing devices can each be a distributed computing environment encompassing multiple computing devices located at the same or at geographically disparate physical locations.

[0065] Client computing devices 305 and server computing devices 310 and 320 can each act as a server or client to other server/client device(s). Server 310 can connect to a database 315. Servers 320A-C can each connect to a corresponding database 325A-C. As discussed above, each server 310 or 320 can

correspond to a group of servers, and each of these servers can share a database or can have their own database. Though databases 315 and 325 are displayed logically as single units, databases 315 and 325 can each be a distributed computing environment encompassing multiple computing devices, can be located within their corresponding server, or can be located at the same or at geographically disparate physical locations.

[0066] Network 330 can be a local area network (LAN), a wide area network (WAN), a mesh network, a hybrid network, or other wired or wireless networks. Network 330 may be the Internet or some other public or private network. Client computing devices 305 can be connected to network 330 through a network interface, such as by wired or wireless communication. While the connections between server 310 and servers 320 are shown as separate connections, these connections can be any kind of local, wide area, wired, or wireless network, including network 330 or a separate public or private network.

[0067] Figure 4 is a block diagram illustrating components 400 which, in some implementations, can be used in a system employing the disclosed technology. Components 400 can be included in one device of computing system 100 or can be distributed across multiple of the devices of computing system 100. The components 400 include hardware 410, mediator 420, and specialized components 430. As discussed above, a system implementing the disclosed technology can use various hardware including processing units 412, working memory 414, input and output devices 416 (e.g., cameras, displays, IMU units, network connections, etc.), and storage memory 418. In various implementations, storage memory 418 can be one or more of: local devices, interfaces to remote storage devices, or combinations thereof. For example, storage memory 418 can be one or more hard drives or flash drives accessible through a system bus or can be a cloud storage provider (such as in storage 315 or 325) or other network storage accessible via one or more communications networks. In various implementations, components 400 can be implemented in a client computing device such as client computing devices 305 or on a server computing device, such as server computing device 310 or 320.

[0068] Mediator 420 can include components which mediate resources between hardware 410 and specialized components 430. For example, mediator 420 can

1 include an operating system, services, drivers, a basic input output system (BIOS),
2 controller circuits, or other hardware or software systems.

3 **[0069]** Specialized components 430 can include software or hardware
4 configured to perform operations for simultaneous controller and touch interactions.
5 Specialized components 430 can include a hand gesture identifier 434, a hand
6 gesture to avatar generator 436, an avatar interaction determiner 438, a controller
7 to hand only determiner 440, and components and APIs which can be used for
8 providing user interfaces, transferring data, and controlling the specialized
9 components, such as interfaces 432. In some implementations, components 400
10 can be in a computing system that is distributed across multiple computing devices
11 or can be an interface to a server-based application executing one or more of
12 specialized components 430. Although depicted as separate components,
13 specialized components 430 may be logical or other nonphysical differentiations of
14 functions and/or may be submodules or code-blocks of one or more applications.

15 **[0070]** Hand gesture identifier 434 identifies hand gestures of the user. The
16 hand gesture can be identified using sensors of controllers 270, using cameras of
17 HMD 200 or 250, using cameras of controllers 270, identifying a rotation of
18 controllers 270, or any combination. Additional details on identifying the hand
19 gestures of the user are provided below with reference to block 1402 of Figure 14.

20 **[0071]** Hand gesture to avatar generator 436 generates an avatar that includes
21 the hand gesture corresponding to the identified hand gesture from hand gesture
22 identifier 434. Additional details on generating the avatar from the identified hand
23 gesture are provided below with reference to block 1404 of Figure 14.

24 **[0072]** Avatar interaction determiner 438 processes an input based on an
25 interaction of the representation of the user's hand with a virtual object. Additional
26 details on processing the input are provided below with reference to block 1406 of
27 Figure 14.

28 **[0073]** Controller to hand only determiner 440 determines when an avatar that
29 includes a controller should be converted to a hand only avatar (i.e., rendering the
30 hand without rendering the controller), and vice versa, based on a threshold
31 distance from the UI or a virtual object or an output of the hand gesture identifier
32 434. Additional details on determining when the avatar should no longer include a
33 controller are provided below with reference to block 1406 of Figure 14.

1 **[0074]** Those skilled in the art will appreciate that the components illustrated in
2 Figures 1-4 described above, and in each of the flow diagrams discussed below,
3 may be altered in a variety of ways. For example, the order of the logic may be
4 rearranged, substeps may be performed in parallel, illustrated logic may be omitted,
5 other logic may be included, etc. In some implementations, one or more of the
6 components described above can execute one or more of the processes described
7 below.

8 **[0075]** Figures 5A and 5B illustrate two hand gestures in connection with a
9 controller for some implementations of the present technology. Figure 5A illustrates
10 a controller 500 (which can be implemented by one of controllers 270 of Fig. 2C)
11 that is intended to be gripped by a user's hand. In general, the user's index finger
12 will be contacting button 502 when controller 500 is fully gripped. A capacitive
13 sensor or other means can determine that the user's index finger is contacting (or
14 not contacting) button 502. In examples, the avatar representation on the UI when
15 controller 500 is fully gripped and the user's index finger is contacting button 502
16 (as indicated at 503) is shown at 510.

17 **[0076]** Figure 5B illustrates controller 500 that is gripped by the user where the
18 user has extended their index finger with the intent of initiating a pointing gesture.
19 The capacitive sensor in button 502 detects that the index finger is no longer
20 contacting button 502. In examples, the avatar representation on the UI when the
21 user's index finger is no longer contacting button 502 (as indicated at 503) is shown
22 at 512. The avatar's index finger 513, corresponding to the user's actual index
23 finger, which is assumed to be extended, or can be confirmed to be extended via a
24 camera, can be used to select an object on the user interface using a natural
25 pointing motion.

26 **[0077]** Any of the buttons, including button 504, or any other portions of
27 controller 500 can implement capacitive sensors that generate a signal or "contact
28 data" when the portion is contacted by the user's hand (e.g., by the user's finger,
29 palm, etc.). The contact data can indicate that the corresponding portion is being
30 touched, a button is being pressed (indicated as the presence of pressure at 504
31 using a pressure sensor, etc.), a joystick is being pushed, etc.

32 **[0078]** Figure 6 illustrates an alternative avatar when the user's index finger is
33 no longer contacting button 502 of controller 500 for some implementations of the

1 present technology. In contrast to avatars 510 and 512 of Figures 5A and 5B,
2 avatar 610 illustrates the user's hand without the controller the user is holding being
3 rendered (referred to as a "hand only" avatar). Avatar 610 is rendered based on
4 the gesture of the index finger being removed from the controller button and
5 extended so that it can be used for pointing and selecting. Avatar 610 can be
6 generated automatically in response to the user's finger ceasing contact with button
7 502, or may be transitioned to when a threshold distance from the controller to the
8 user interface is reached. As shown, the avatar's index finger 613, which
9 corresponds to the user's actual index finger, can be used to select any virtual
10 object or other area on UI 612.

11 **[0079]** Avatar 601 can be rendered in response to controller 500 receiving and
12 processing contact data, which is generated in response to a contact by the user
13 on button 502 (or another button or portion of controller 500 in other example
14 implementations). The contact data can be generated in response to a capacitive
15 sensor in button 502 sensing (or not sensing) contact with button 502. The contact
16 data can be in the form of a Boolean yes or no as to whether or not the user is
17 contacting button 502, or a capacitance range, where once a threshold is
18 surpassed, it is conclusively determined that the user is no longer contacting button
19 502 and avatar 610 is rendered. With the capacitance range, the rendering of
20 avatar 610 can change depending on the level of the range. For example, in an
21 interim period before the threshold is surpassed, avatar 610 may be rendered with
22 index finger 613 being partially extended, but not fully extended.

23 **[0080]** In addition to the index pointing gesture or pose, which is illustrated in
24 Figures 5 and 6, capacitive sensors or other means such as pressure sensors can
25 generate contact data of the user's hands to detect other types of gestures or poses.

26 **[0081]** Figure 7A illustrates a "pinch" gesture detected via controller 500 for
27 some implementations of the present technology. For example, the pinch gesture
28 can be shown when the virtual location of the controller is close to a location of a
29 virtual interface. For the pinch gesture input state, contact on button 502 is sensed,
30 without button 502 being pressed. A corresponding visually rendered avatar 700 is
31 generated.

32 **[0082]** Figure 7B illustrates a "poke/press" gesture detected via controller 500
33 for some implementations of the present technology. For the poke/press gesture

1 input state, contact is sensed on button 502, and button 502 being pressed is
2 sensed. A corresponding visually rendered avatar 702 is generated.

3 **[0083]** Figure 8A illustrates a “neutral” gesture detected via controller 500 for
4 some implementations of the present technology. For the neutral gesture input
5 state, contact is sensed on button 502, and button 502 being pressed is sensed and
6 button 504 being pressed is sensed. A corresponding visually rendered avatar 800
7 is generated.

8 **[0084]** Figure 8B illustrates a “full-hand grab” gesture detected via controller
9 500 for some implementations of the present technology. For the full-hand grab
10 gesture input state, contact is sensed on button 502, and button 502 being pressed
11 is sensed and button 504 being pressed is sensed. A corresponding visually
12 rendered avatar 802 is generated.

13 **[0085]** Figures 9-12 illustrate various different input states of the controller, and
14 the corresponding visually rendered avatar for some implementations of the present
15 technology. As shown in Figures 7-12, different input states may result in the same
16 corresponding avatar, and the same input states may result in different
17 corresponding avatars, depending on the type of state of the application, or other
18 factors.

19 **[0086]** As illustrated in Figure 9, for the input state, button 502 is not touched
20 or pressed, button 504 is pressed, joystick 506 is not touched, and any other button
21 combination, e.g., the O, A, or B buttons next to joystick 506, can be pressed at
22 902. In response, corresponding visually rendered avatar 900 is generated.

23 **[0087]** As illustrated in Figure 10, for the input state, button 502 is not touched
24 or pressed, button 504 can be pressed or not pressed and joystick 506 can be
25 touched or not touched. In response, corresponding visually rendered avatar 1000
26 is generated.

27 **[0088]** As illustrated in Figure 11, for the input state, button 502 is touched and
28 not pressed, button 504 is not pressed and joystick 506 can be touched or not
29 touched. In response, corresponding visually rendered avatar 1100 is generated.

30 **[0089]** As illustrated in Figure 12, for the input state, button 502 is touched and
31 pressed, button 504 is not pressed and joystick 506 can be touched or not touched.
32 In response, corresponding visually rendered avatar 1200 is generated.

33 **[0090]** Figures 13A and 13B illustrate, for some implementations of the present

1 technology, a transition between the avatar showing both the controller and hand
2 (Figure 13A) and then, as the controller gets closer to a target object on the UI, the
3 hand only avatar without the controller (Figure 13B). The threshold distance from
4 the target object to the user where the transition occurs can be predefined (e.g., 2
5 or 3 feet from the user) or variable depending on the content of the UI or other
6 factors. In some cases, the threshold distance is determined based on the arm
7 span (e.g., a percentage value of the user's arm length) of the user, such as whether
8 the target object is within arm's reach of the user. In some implementations, the
9 distance between the controller and the target object is calculated by a distance
10 between the coordinate of the controller in virtual space and the coordinate of the
11 target object in virtual space.

12 **[0091]** Figure 14 is a flow diagram illustrating a process 1400 used in some
13 implementations for facilitating the simultaneous controller and touch interactions
14 by providing direct touch interaction with virtual objects while a user holds the
15 controller. In some implementations, process 1400 can be performed "just in time,"
16 e.g., as a response to the UI displaying selectable objects.

17 **[0092]** At block 1402, process 1400 detects, at a controller, a user gesture while
18 the user holds the controller. The detection can include processing contact data on
19 one or more locations on the controller, where the contact data can be generated
20 by one or more capacitive sensors. The locations can be any portion of the
21 controller, including buttons, joysticks, switch, grip, etc. In one example, the contact
22 data is generated in response to a transition of contact of a finger of the user on a
23 button to the finger not contacting the first button, such as when extending an index
24 finger as shown in Figures 5A and 5B.

25 **[0093]** In other example implementations, detecting the user gesture or hand
26 "posture" at block 1402 includes, in combination with the contact data or by itself,
27 collecting visual data of the user gesture using a camera, that may be included in
28 the headset and/or the controller, and determining, using one or more machine
29 learning models such as a classifier, whether the user is contacting one or more
30 locations on the controller.

31 **[0094]** A hand posture refers to a hand location and/or pose. For example, a
32 hand posture can comprise one or both of where the hand is in an artificial reality
33 environment and/or the shape the hand is making. Certain hand postures can be

1 previously identified in training data as "gestures." For example, the training data
2 can include images or videos of a pinch gesture or a pointing gesture from multiple
3 angles. The classifier can be trained on the training data to output a hand pose
4 classification based on input hand postures. The processes described herein can
5 monitor hand postures in different manners. In some cases, hand postures can be
6 identified using input from external facing cameras that capture depictions of user
7 hands. In other cases, hand postures can be based on input from a wearable
8 device such as a glove or wristband that tracks aspects of the user's hands. In
9 some implementations, input can be interpreted as postures mapped as certain
10 gestures by applying the input to a machine learning model trained to identify hand
11 postures and/or gestures based on such input. In some implementations, heuristics
12 or rules can be used to analyze the input to identify hand postures and/or gestures.

13 **[0095]** In other example implementations, detecting the user gesture at block
14 1202 includes detecting a rotation of the controller via, for example, a gyroscope.
15 The rotation of the controller, using one or more classifiers, can establish a user
16 gesture.

17 **[0096]** The detected gestures in block 1402 can include a neutral gesture, a
18 pointing gesture, a pinch gesture, a swipe gesture, a full hand grip gesture, etc.

19 **[0097]** At block 1404, process 1400 provides, in an artificial reality environment,
20 a representation of a user's hand corresponding to the gesture. The representation
21 can also include a representation of the controller, such as shown in Figure 5B, or
22 be a hands only representation, such as shown in Figure 6. At least one gesture,
23 in response to the index finger being removed from a button and pointing forward,
24 can enhance the ability to select an object in the UI.

25 **[0098]** In example implementations, at block 1404, process 1400 determines a
26 distance between the representation of the user's hand and the virtual object when
27 providing the representation of the user's hand corresponding to the gesture.
28 Process 1400, in response to determining the distance is greater than a threshold
29 distance, renders the representation of the user's hand in the artificial reality
30 environment as holding a virtual controller. Further, process 1400, in response to
31 determining the distance is less than the threshold distance, renders the
32 representation of the user's hand without the virtual controller.

33 **[0099]** At block 1406, process 1400 receives an input in relation to a virtual

1 object based on an interaction of the representation of the user's hand with the
2 virtual object. In one example invention, the representation of the user's index
3 finger (or any other finger) is used to interact with the virtual object via a pointing
4 movement. The interaction can be a selection of the virtual object.

5 **[00100]** Example implementations allow the user to interact with the UI that is
6 very close (via pointing) while still holding the controller. Example inventions can
7 also fade out the controller in the representation, so the user only sees a hand,
8 without rendering the controller, when the index finger is used for pointing, as shown
9 in Figure 6.

10 **[00101]** Example implementations further enable a preferred input method (e.g.,
11 controller or hand) depending on the context. It provides seamless switching
12 between controller and hand. Example implementations can utilize digital tracking
13 of the controller, which in some examples works better and more accurately than
14 the tracking of a hand.

15 **[00102]** Example implementations can include haptics in the controller to enable
16 feedback to the user when using a finger to touch (i.e., assisted touch) as opposed
17 to using pure hands (i.e., no controller) for interaction. For example, when the
18 rendered finger "touches" or otherwise interacts with an object on a UI, an actuator
19 located on a portion of the controller, such as the grip, will generate vibrations to
20 confirm the touch to the user.

21 **[00103]** Example implementations can change hand visualization (e.g., of the
22 avatar) to reflect a hand gesture to show an intended action, which can be different
23 than the actual user's hand position. Example implementations can infer gestures
24 based on mappings of inputs/sensed grips on the controller to expected
25 corresponding gestures.

26 **[00104]** Example implementations can also use cameras and vision to determine
27 a hand gesture when gripping the controller in place of or in addition to the touch
28 sensing. In some implementations, this can be done by receiving input from a
29 camera that can be located on the controller or on the head mounted device, and
30 that captures images of the user's hand and determines, based on the images, a
31 posture such as a 3D model of the hand, a skeletal representation of the hand,
32 relative coordinates or distances for specified points on a hand (e.g., various joints
33 and fingertips), etc. In some implementations, hand postures can be monitored

1 based on other input in addition to cameras and/or sensing touch on the controller,
2 such as from sensors in a wearable glove or bracelet, sonar or ultrasound sensors,
3 etc. This multimodal tracking is helpful, for example, when switching between using
4 a controller and using pure hands for interaction.

5 **[00105]** Example implementations can also use a stylus as a controller with
6 capacitive sensing to determine which portion of the stylus the user is touching or
7 any other type of physical input device (e.g., keyboard, mouse, etc.).

8 **[00106]** Example implementations can transition visualizations from when the
9 avatar is a hand holding a controller to a hand only based on distance of the
10 hand/controller from UI elements. The transition to hand only can occur when the
11 distance is reduced below a threshold. Example implementations can have a mid-
12 transition to “ghost” opaque controller before transitioning to hand only. Example
13 implementations can show the controller with a finger off before fading away the
14 controller to the hand only representation.

15 **[00107]** Example implementations can also detect controller rotation using
16 gyroscopes of the controller or computer vision and use the rotation as another
17 trigger for determining direct hand input.

18 **[00108]** Capacitive sensors in example implementations can be on one or more
19 buttons of the controller and can also be on the grip. Example implementations can
20 also track how far finger is lifted off of a button (e.g., using a camera on the HMD
21 or controller).

22 **[00109]** Example implementations can detect hand position or infer gestures
23 based on buttons being touched or augmented or replaced by a computer vision
24 system. Any sensors can be used to infer user gestures, such as based on motions
25 as determined by an IMU.

26 **[00110]** In example implementations, when close to the UI and/or the user's
27 finger is lifted off the controller, the system can use finger touching for direct
28 interaction with the UI. When further from the UI and/or if the user's finger is not
29 lifted off the controller, example implementations can use ray casting from the
30 controller for input. If the user's hand is, for example, > 15cm from the UI that their
31 hand is aiming towards the UI, example implementations can show a line or ray
32 from the user's hand to the UI. When a user's hand is near a UI, she can directly
33 interact with the UI using her hand. The user can pinch to make a selection like

1 she normally would with her hand.

2 **[00111]** In some implementations, as long as the user is interacting with a ray
3 cast from a controller, example implementations may not switch to hand only mode
4 until the hand is within, for example, 8cm of the UI to avoid constant switching.
5 Example implementations can also look at heuristics such as the trajectory of the
6 hand and if it is moving closer to the UI. The distance used can be far enough
7 where the system does not accidentally show the user a ray when not intended but
8 also close enough to allow the user to hold their hand in a restful state to interact.

9 **[00112]** Reference in this specification to "implementations" (e.g., "some
10 implementations," "various implementations," "one implementation," "an
11 implementation," etc.) means that a particular feature, structure, or characteristic
12 described in connection with the implementation is included in at least one
13 implementation of the disclosure. The appearances of these phrases in various
14 places in the specification are not necessarily all referring to the same
15 implementation, nor are separate or alternative implementations mutually exclusive
16 of other implementations. Moreover, various features are described which may be
17 exhibited by some implementations and not by others. Similarly, various
18 requirements are described which may be requirements for some implementations
19 but not for other implementations.

20 **[00113]** As used herein, being above a threshold means that a value for an item
21 under comparison is above a specified other value, that an item under comparison
22 is among a certain specified number of items with the largest value, or that an item
23 under comparison has a value within a specified top percentage value. As used
24 herein, being below a threshold means that a value for an item under comparison
25 is below a specified other value, that an item under comparison is among a certain
26 specified number of items with the smallest value, or that an item under comparison
27 has a value within a specified bottom percentage value. As used herein, being
28 within a threshold means that a value for an item under comparison is between two
29 specified other values, that an item under comparison is among a middle-specified
30 number of items, or that an item under comparison has a value within a middle-
31 specified percentage range. Relative terms, such as high or unimportant, when not
32 otherwise defined, can be understood as assigning a value and determining how
33 that value compares to an established threshold. For example, the phrase

1 "selecting a fast connection" can be understood to mean selecting a connection that
2 has a value assigned corresponding to its connection speed that is above a
3 threshold.

4 **[00114]** As used herein, the word "or" refers to any possible permutation of a set
5 of items. For example, the phrase "A, B, or C" refers to at least one of A, B, C, or
6 any combination thereof, such as any of: A; B; C; A and B; A and C; B and C; A, B,
7 and C; or multiple of any item such as A and A; B, B, and C; A, A, B, C, and C; etc.

8

1 CLAIMS

- 2 1. A method for providing direct touch interaction with virtual objects while
3 a user holds a controller, the method comprising:
4 detecting a user gesture while the user holds the controller;
5 providing, in an artificial reality environment, a representation of a user's hand
6 corresponding to the gesture; and
7 receiving an input, while the user holds the controller, in relation to a virtual object
8 based on an interaction of the representation of the user's hand with the
9 virtual object.
- 10 2. The method of claim 1, wherein detecting the user gesture comprises
11 processing contact data on one or more locations on the controller.
- 12 3. The method of claim 2, wherein the one more locations comprise a first
13 button of the controller.
- 14 4. The method of claim 2 or 3, wherein the contact data is generated in
15 response to a transition of contact of a first finger of the user on the first button to
16 the first finger not contacting the first button.
- 17 5. The method of claim 4, wherein the first finger comprises an index
18 finger, and the gesture comprises a pointing gesture.
- 19 6. The method of any preceding claims 1 to 5, wherein the input
20 comprises a selection of the virtual object.
- 21 7. The method of any preceding claims 1 to 6, wherein the controller
22 comprises at least one capacitive sensor that generates the contact data used for
23 determining whether the user is contacting one or more locations on the controller.
- 24 8. The method of any preceding claims 1 to 7, wherein detecting the user
25 gesture comprises:
26 collecting visual data of the user gesture using a camera; and
27 determining, using one or more classifiers, whether the user is contacting one or
28 more locations on the controller.
- 29 9. The method of claim 1, wherein providing, in the artificial reality
30 environment, the representation of a user's hand corresponding to the gesture
31 includes providing a representation of the controller.
- 32 10. The method of claim 1, wherein providing, in the artificial reality
33 environment, the representation of a user's hand corresponding to the gesture

1 comprises:
2 determining a distance between the representation of the user's hand and the
3 virtual object;
4 in response to determining the distance is greater than a threshold distance,
5 rendering the representation of the user's hand in the artificial reality
6 environment as holding a virtual controller; and
7 in response to determining the distance is less than the threshold distance,
8 rendering the representation of the user's hand without the virtual controller.
9 11. The method of any preceding claims 1 to 10, wherein the gesture
10 comprises a neutral, pointing, pinch, swipe or full hand grip.
11 12. The method of any preceding claims 1 to 11, wherein the detecting the
12 user gesture comprises detecting a rotation of the controller.
13 13. A computer-readable medium comprising instructions that, when
14 executed by a computer, cause the computer to carry out the steps of any of claims
15 1 to 12.
16 14. A computing system for providing direct touch interaction with virtual
17 objects while a user holds a controller, the computing system comprising:
18 one or more processors; and
19 one or more computer-readable medium storing instructions that, when executed
20 by the one or more processors, configure the computing system to perform
21 operations comprising the steps of any of claims 1 to 12.

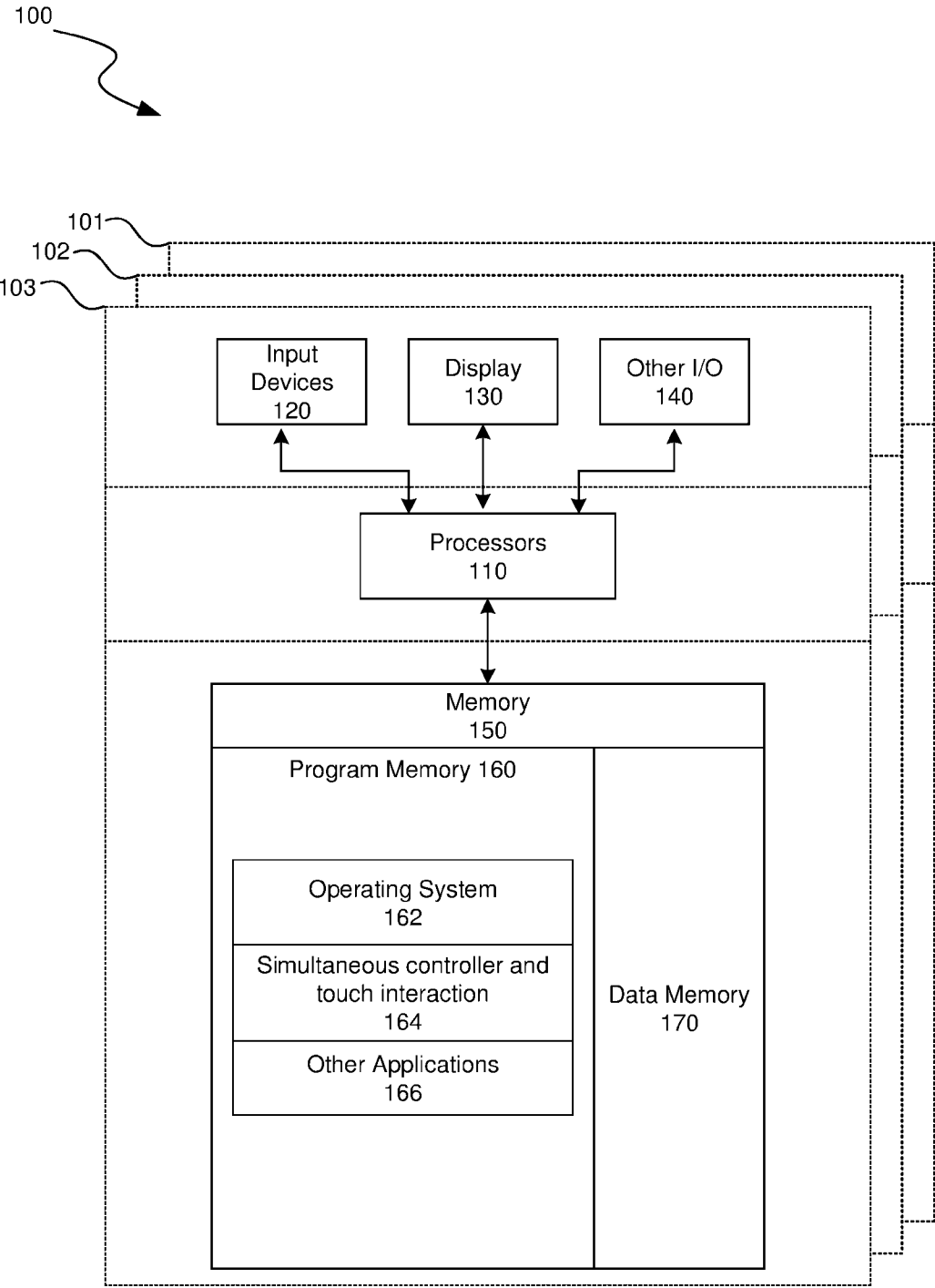


FIG. 1

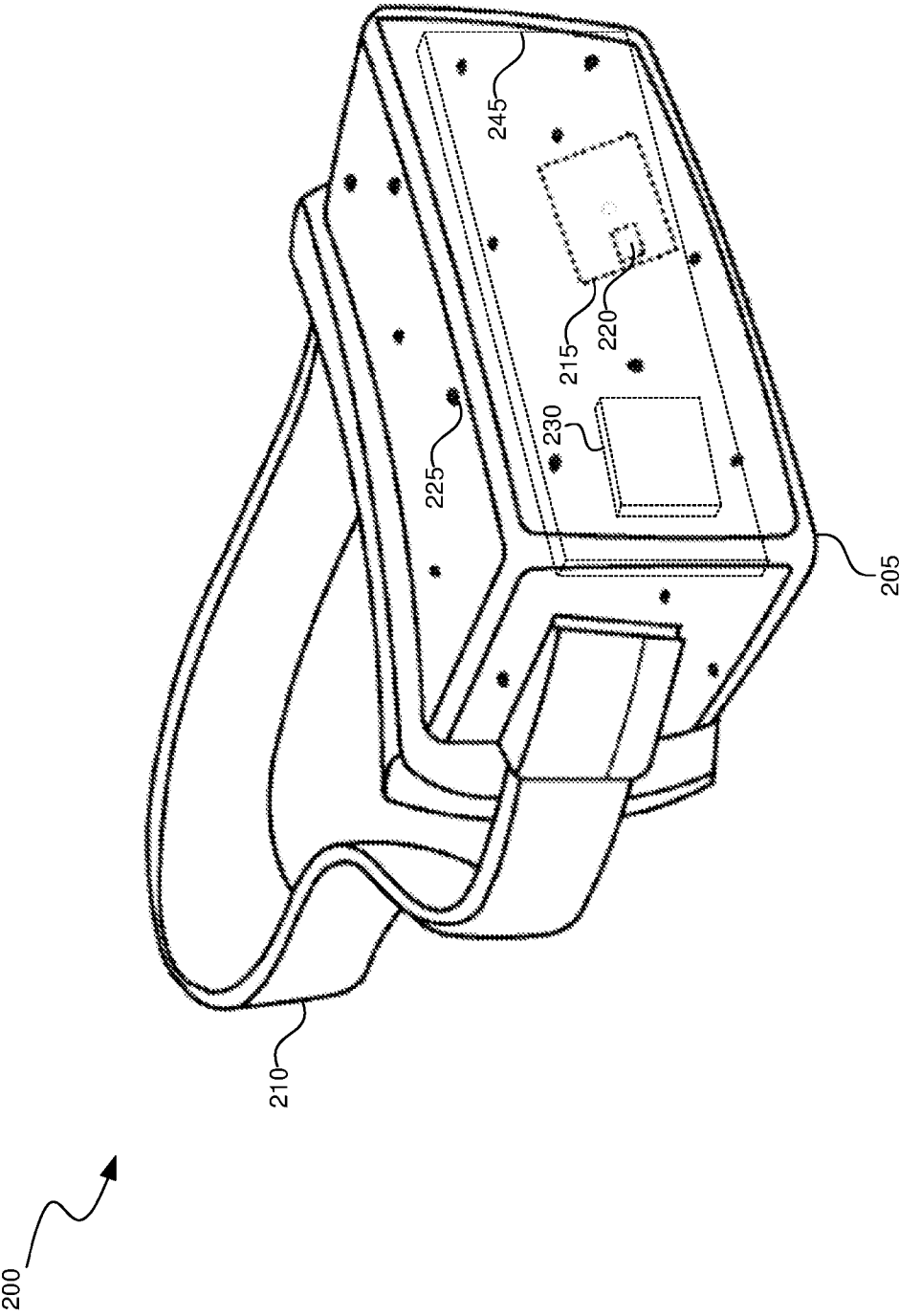


FIG. 2A

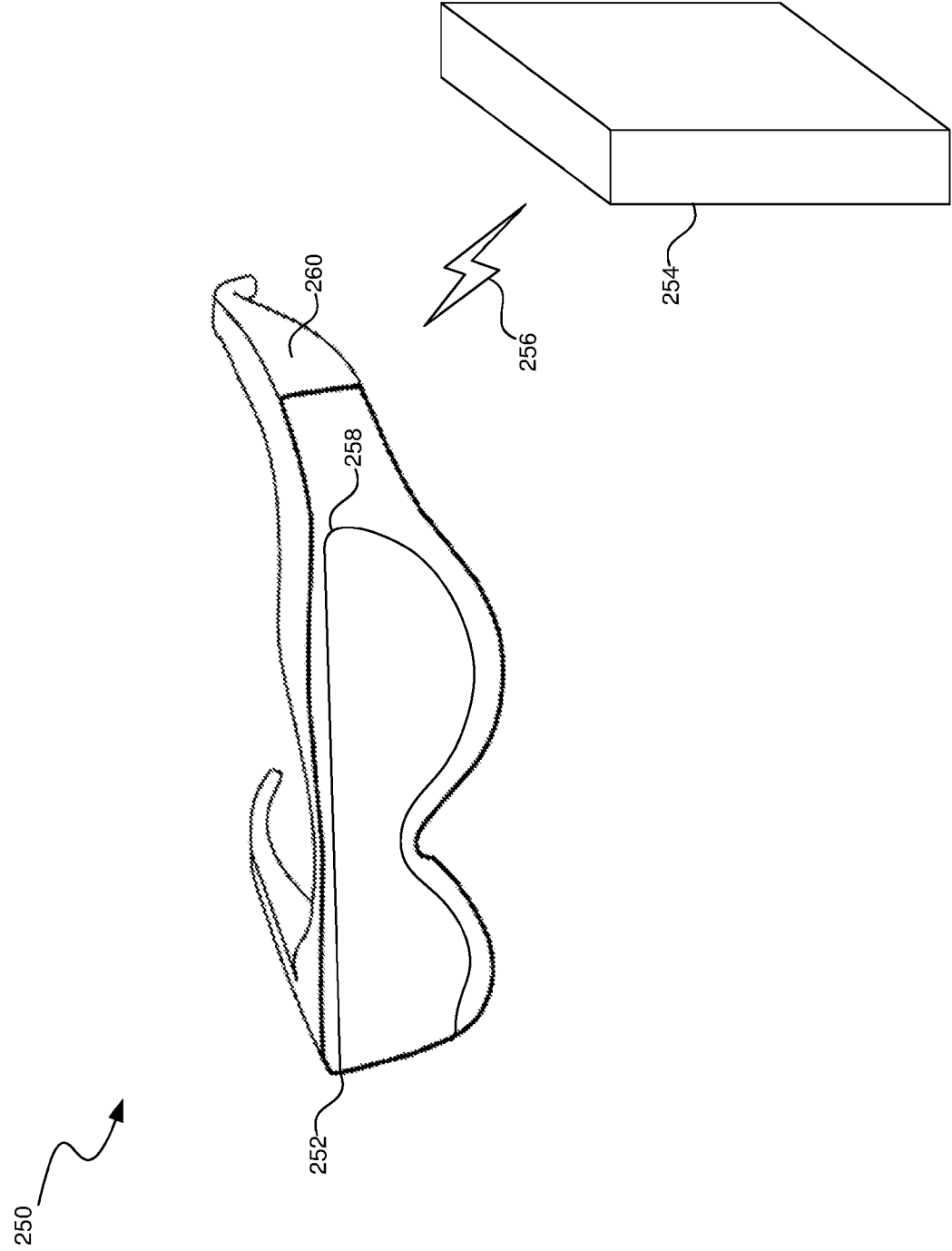


FIG. 2B

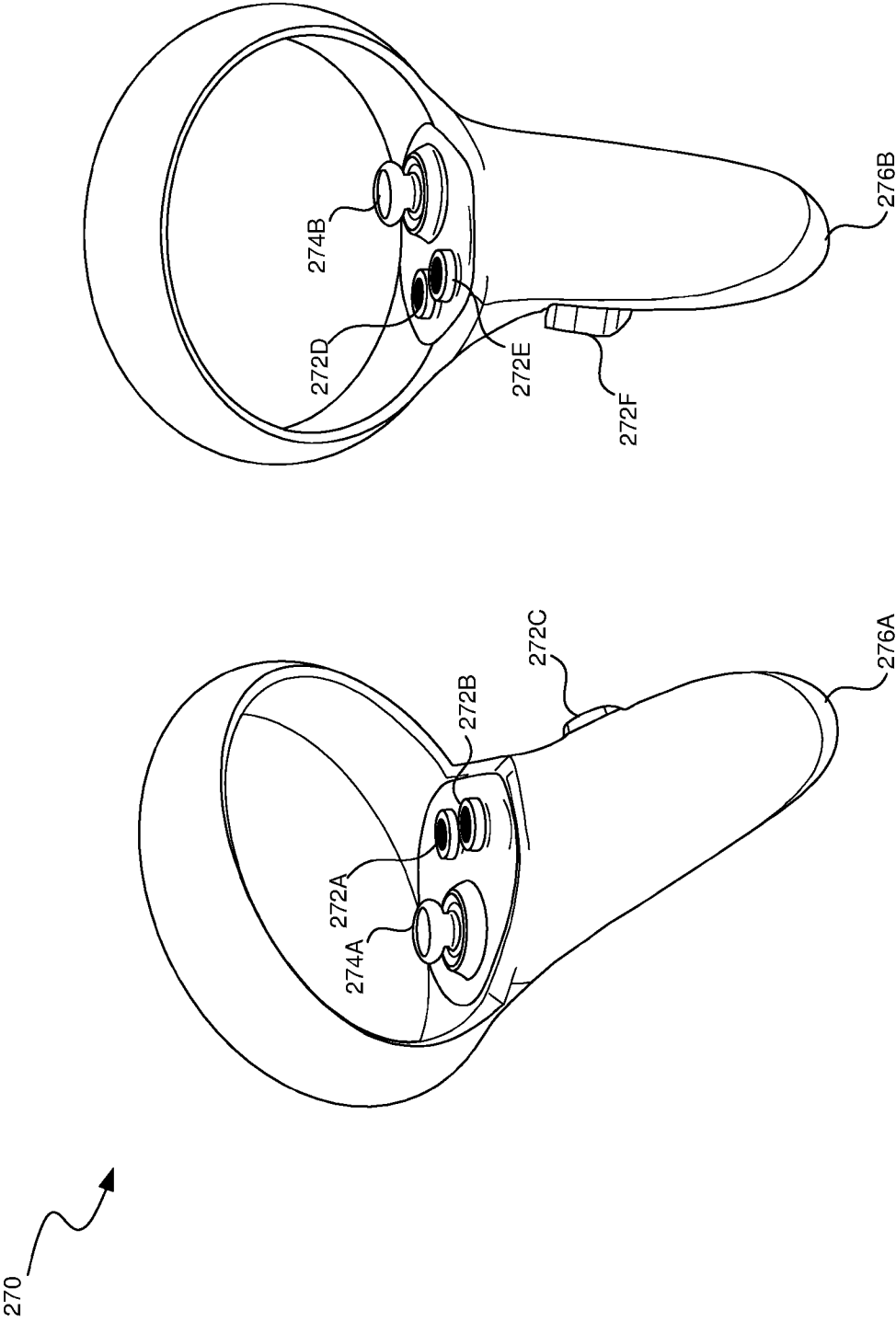


FIG. 2C

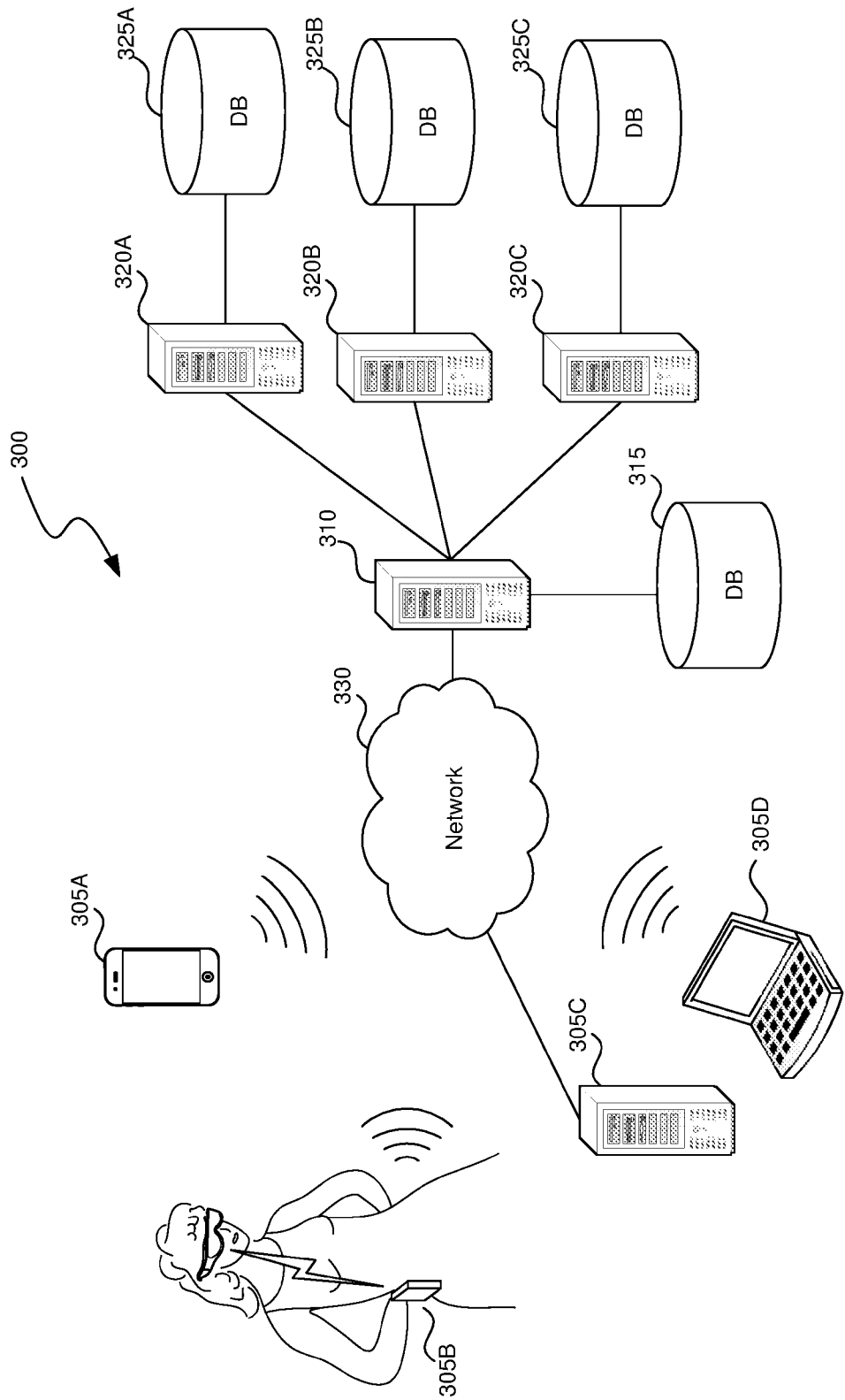
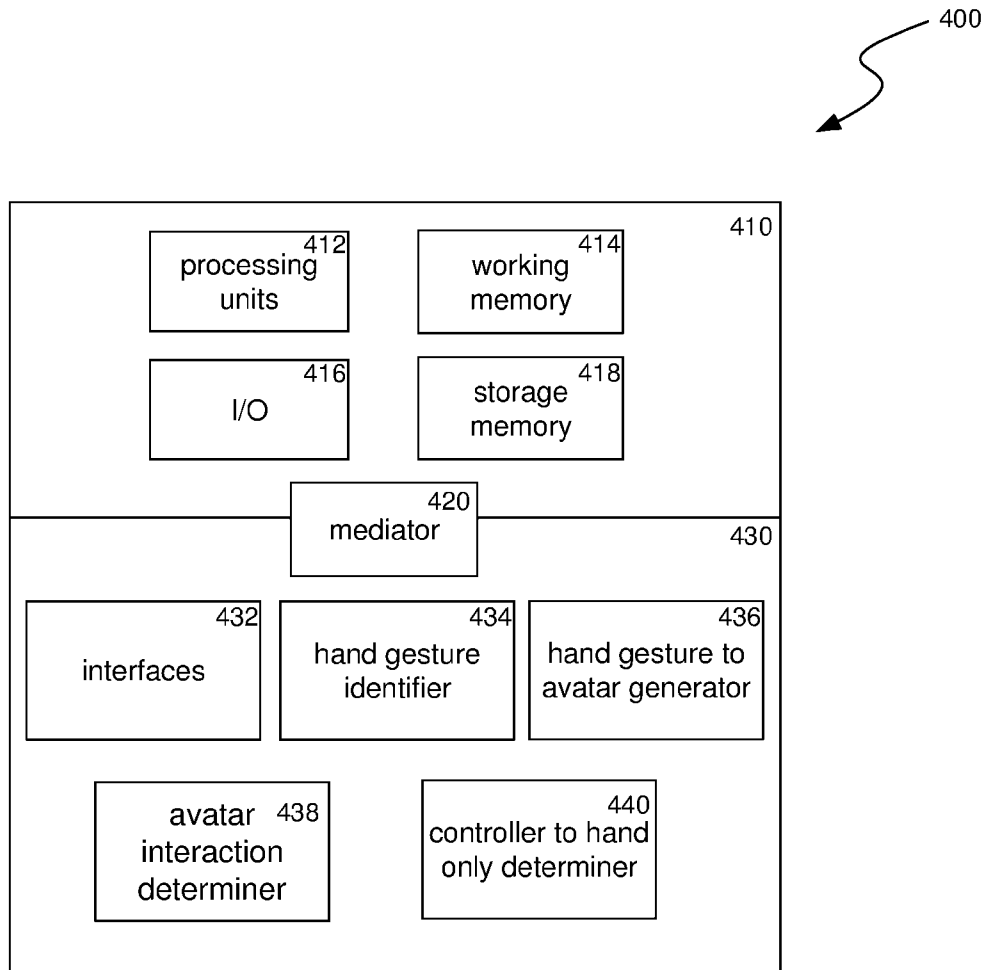


FIG. 3

**FIG. 4**

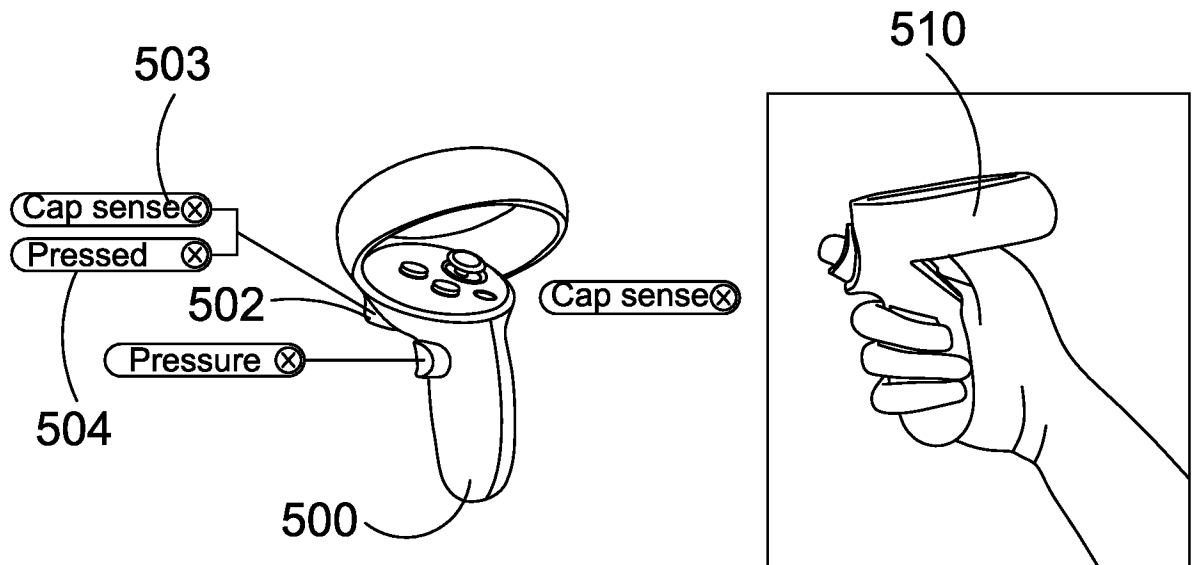


FIG. 5A

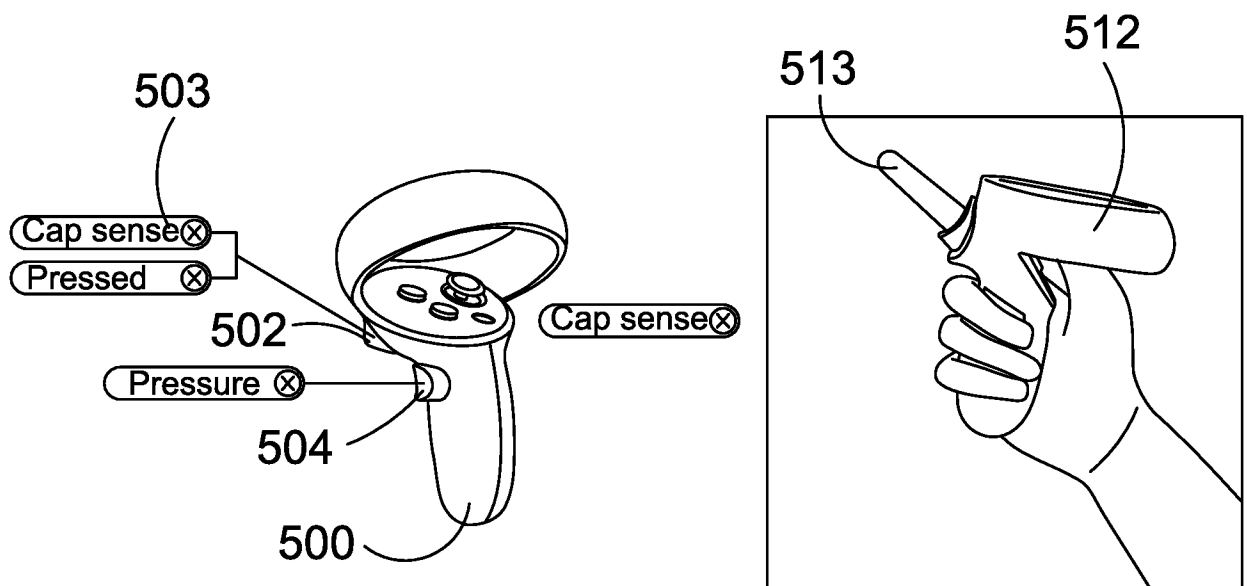


FIG. 5B

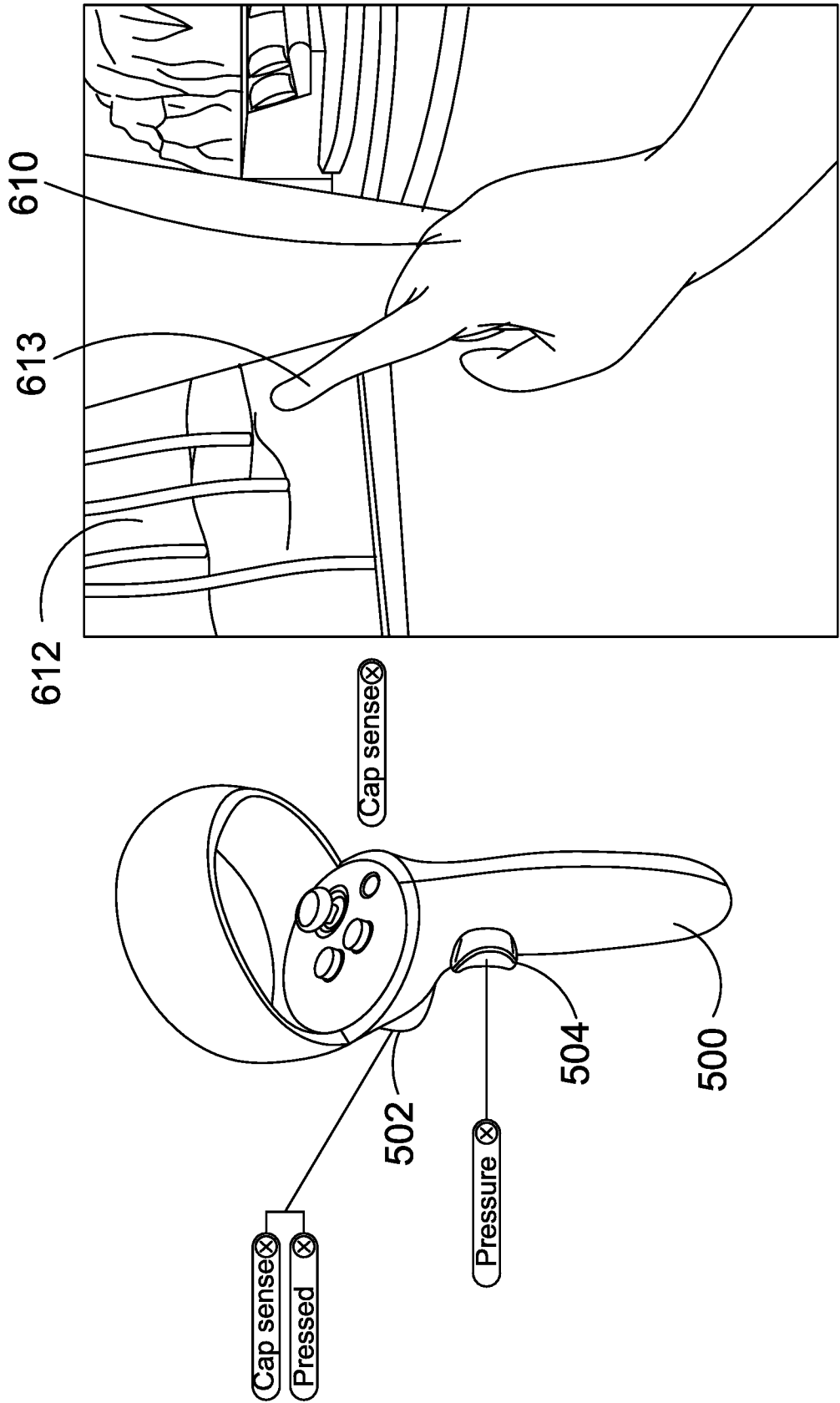


FIG. 6

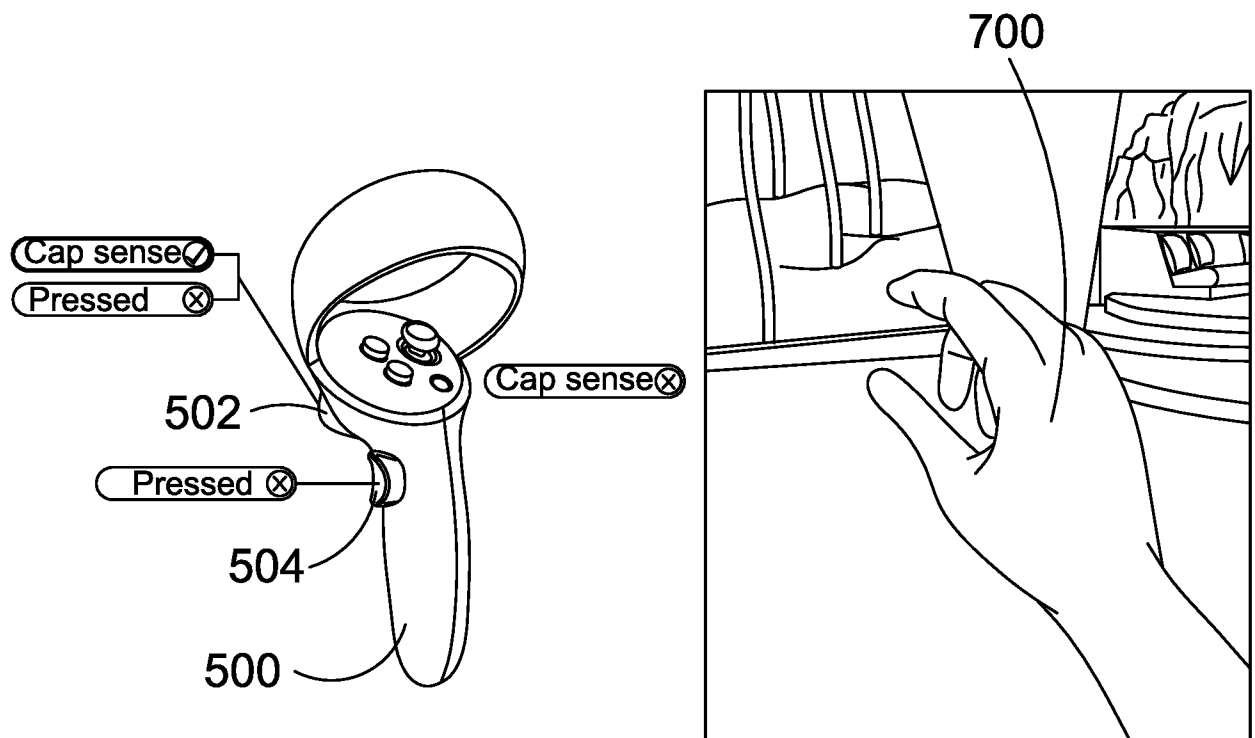


FIG. 7A

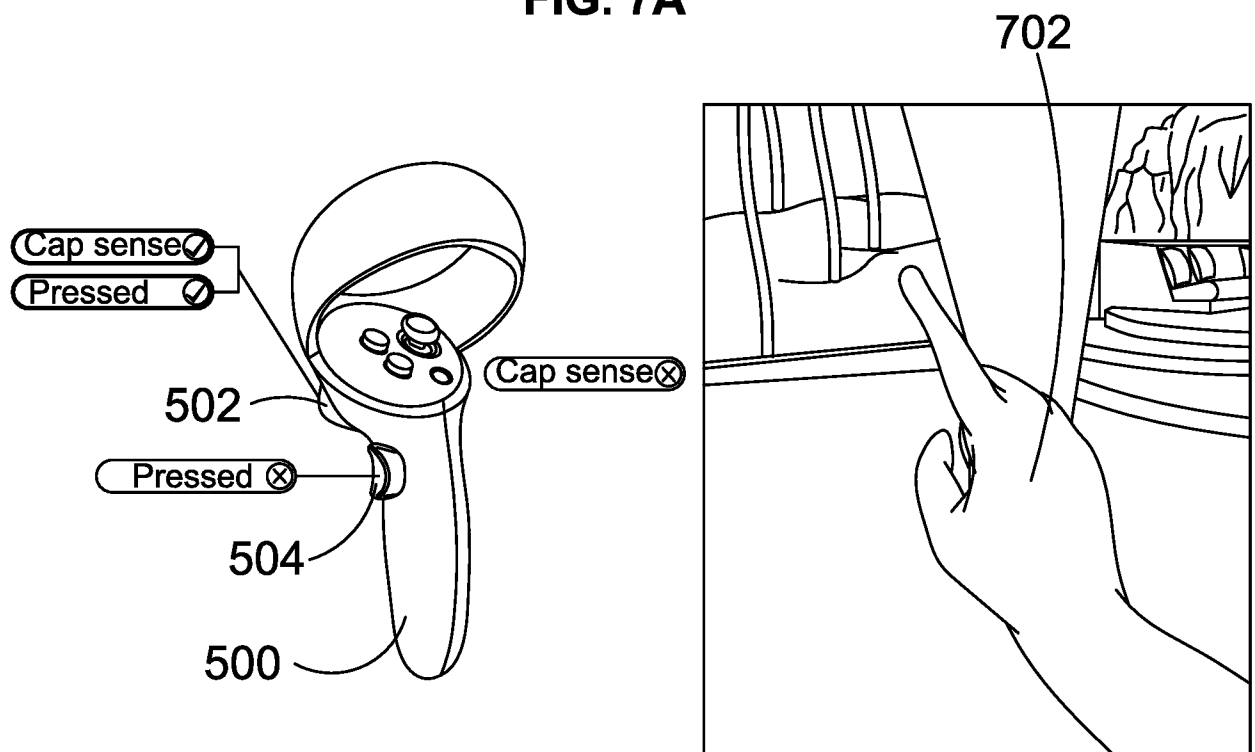


FIG. 7B

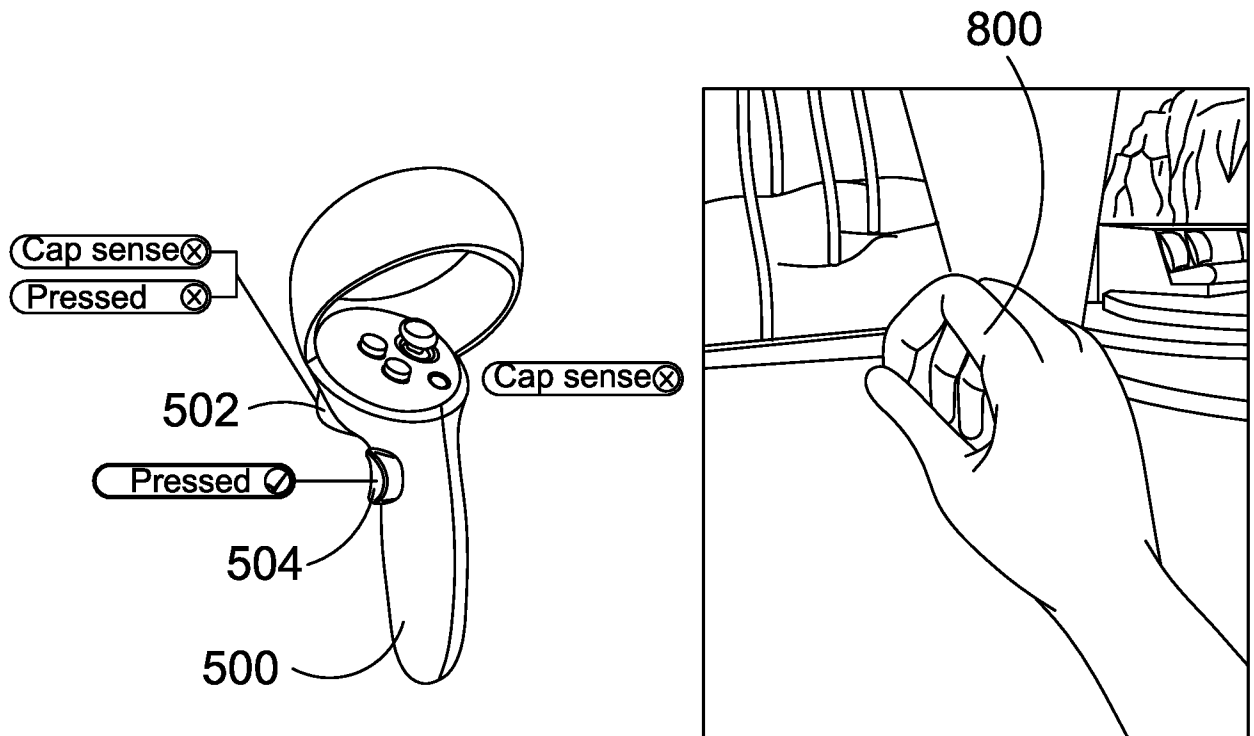


FIG. 8A

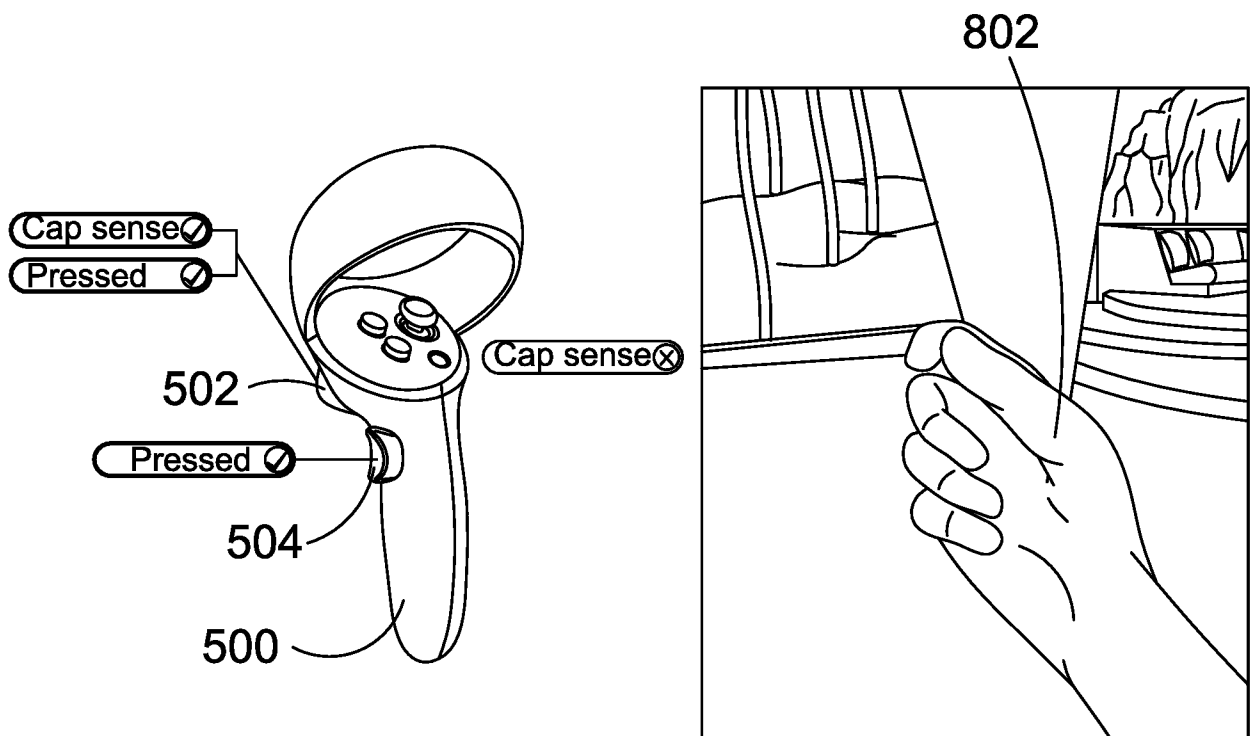


FIG. 8B

Input State

Visually Rendered

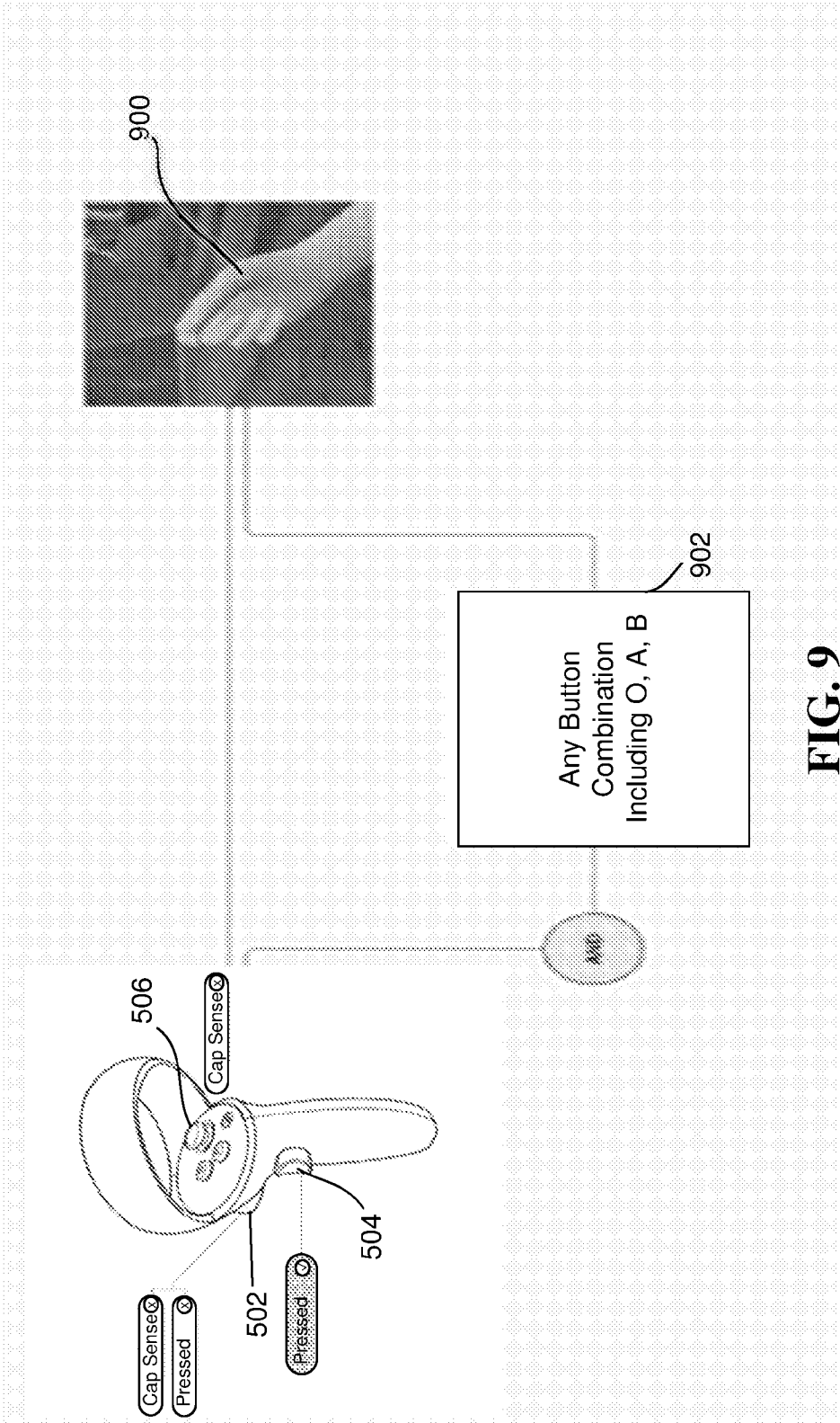


FIG. 9

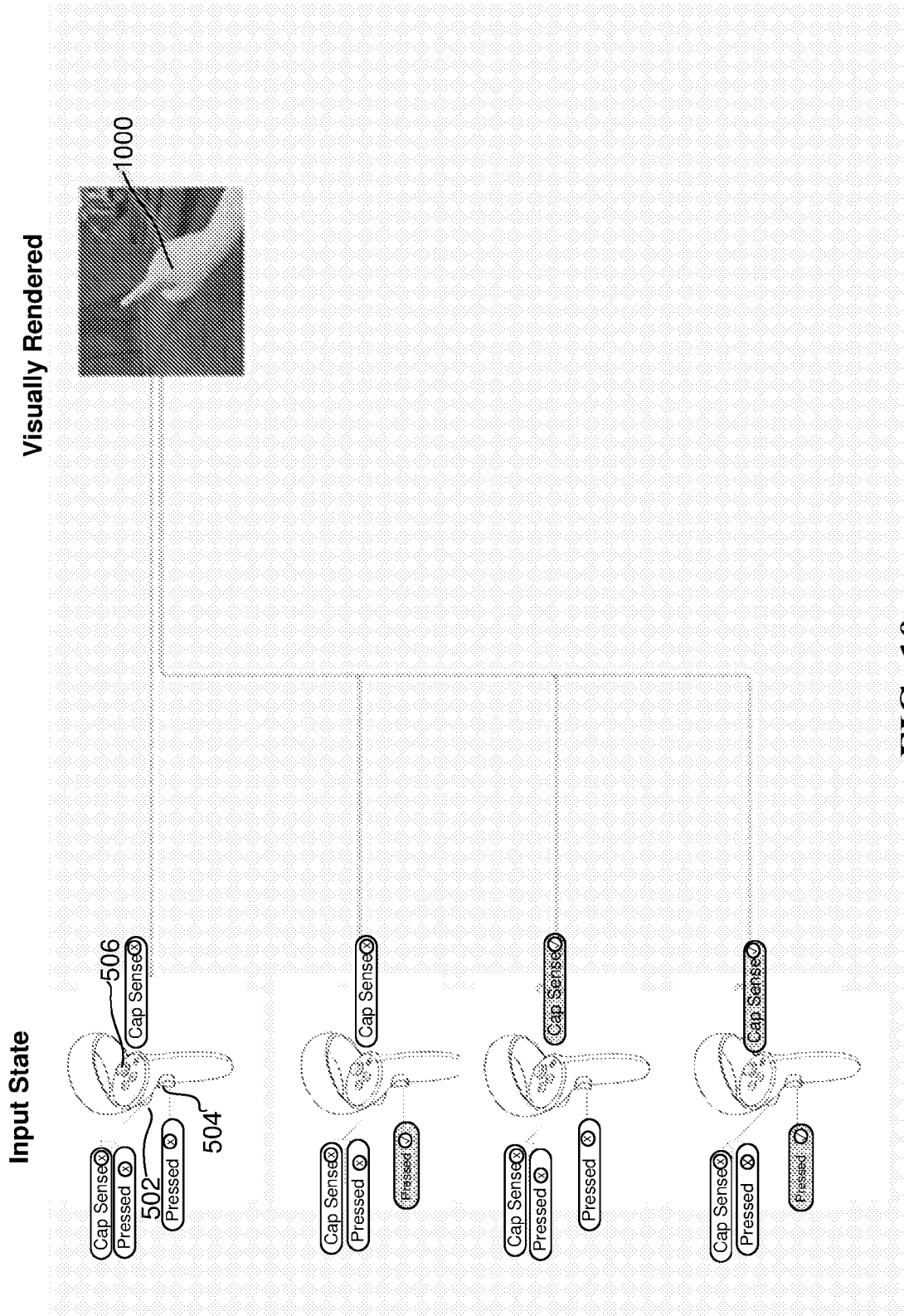


FIG. 10

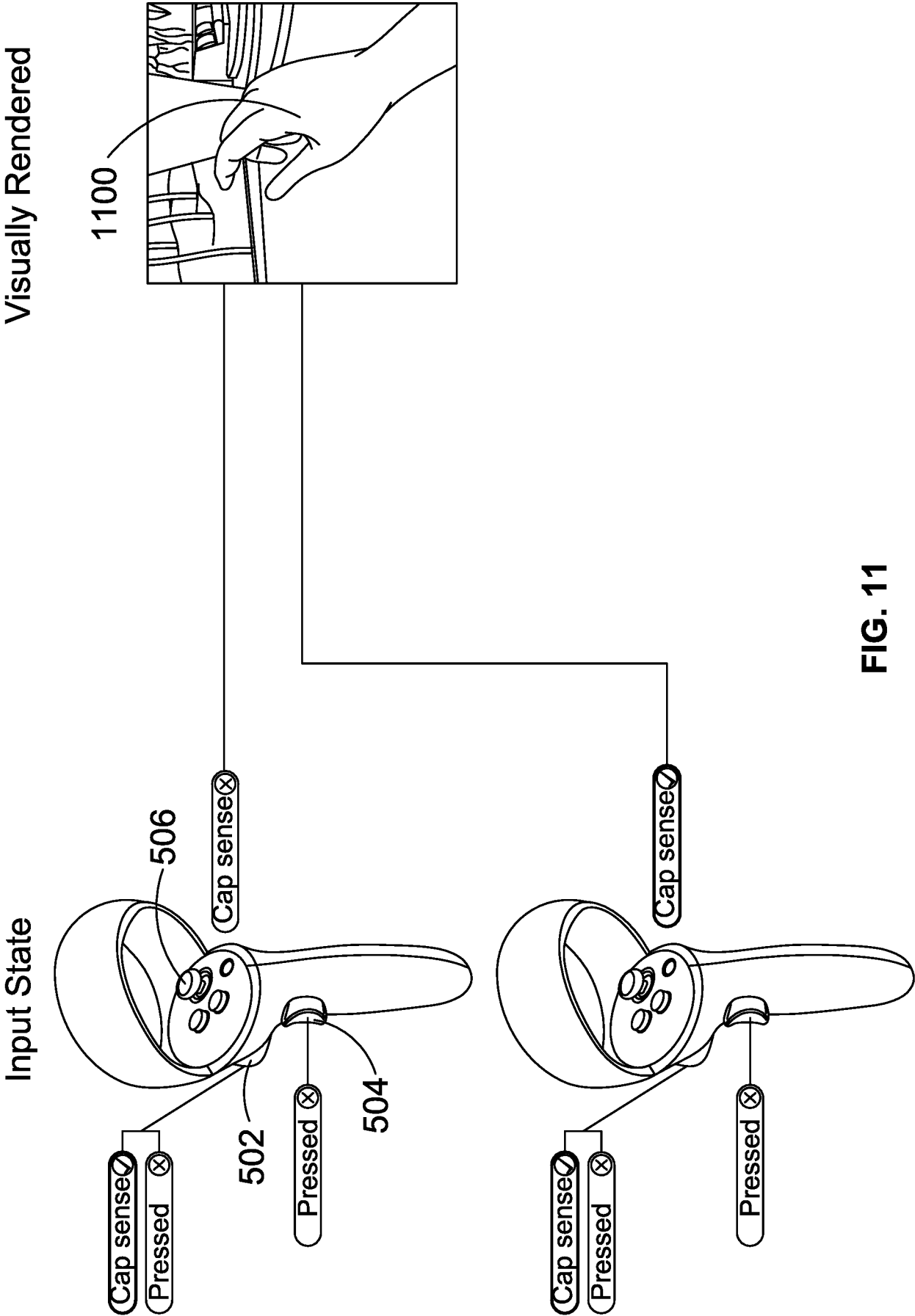


FIG. 11

Visually Rendered

Input State

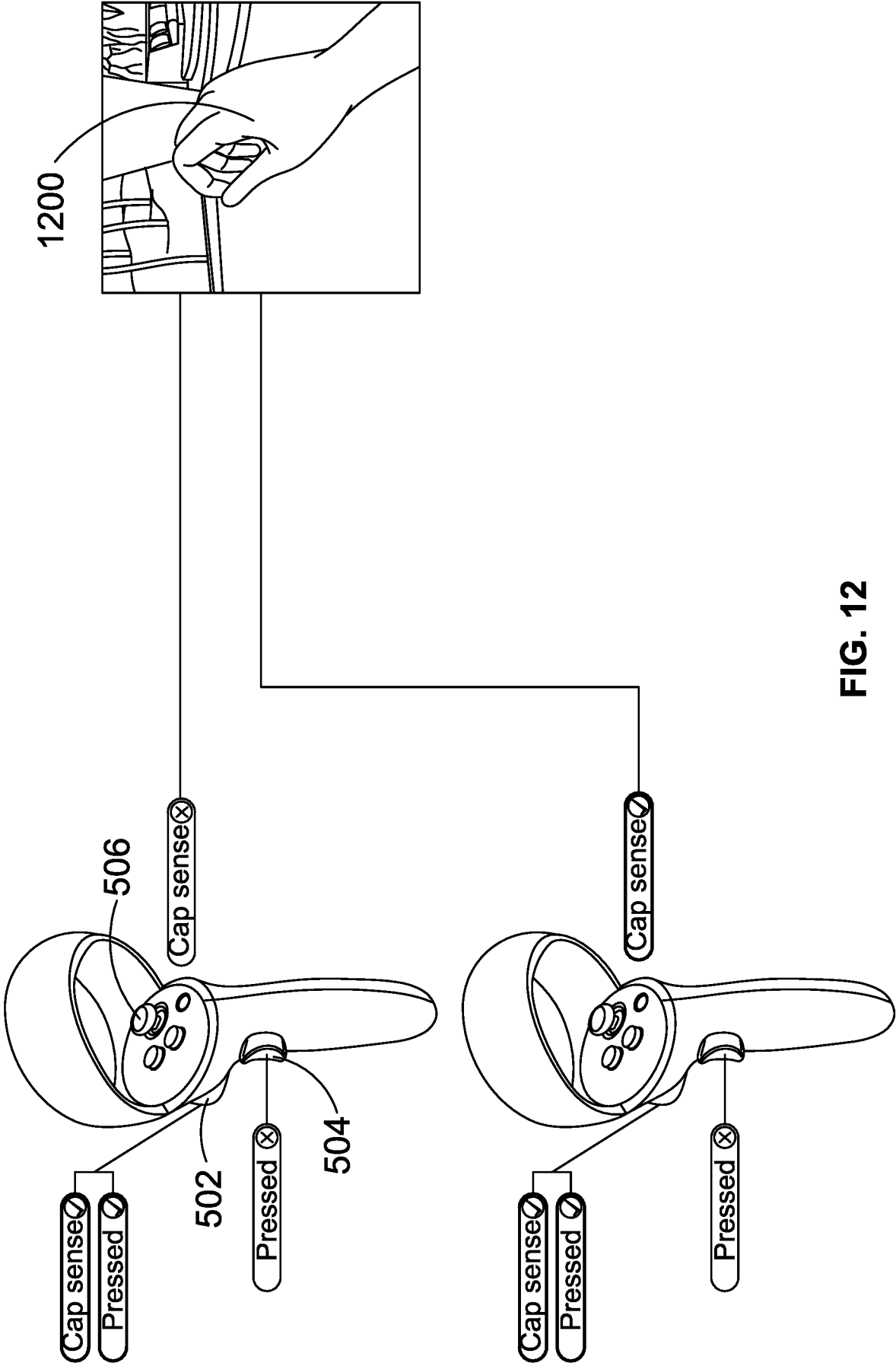


FIG. 12

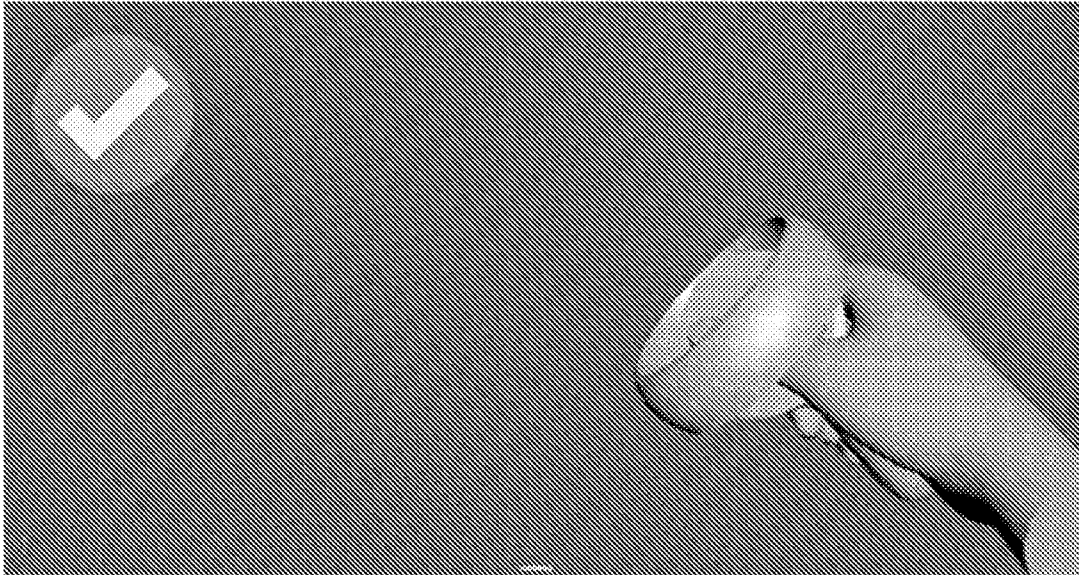


FIG. 13A

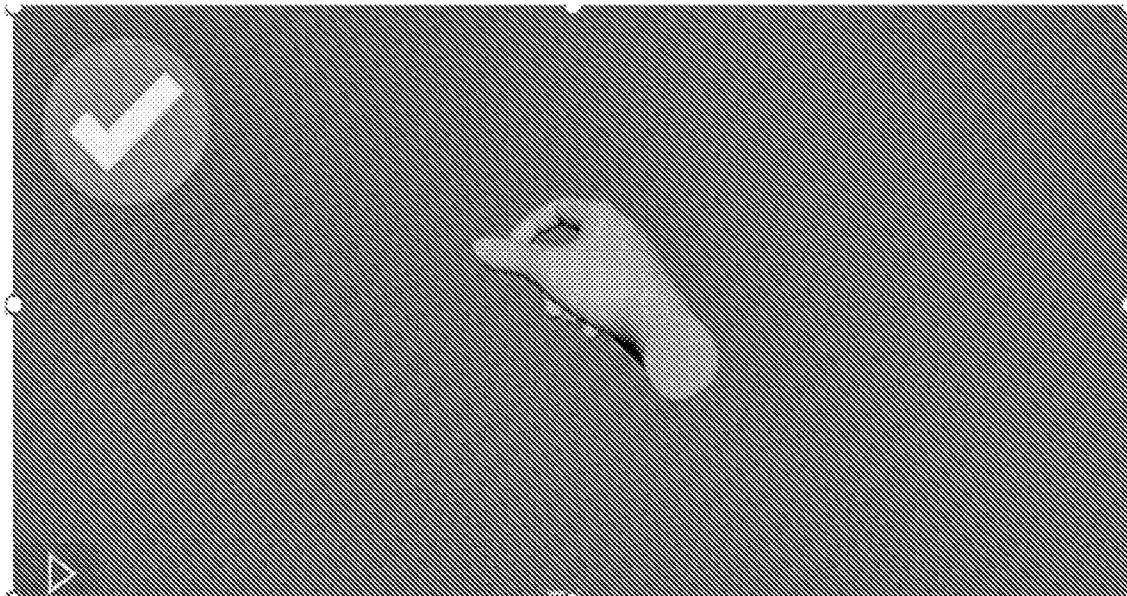
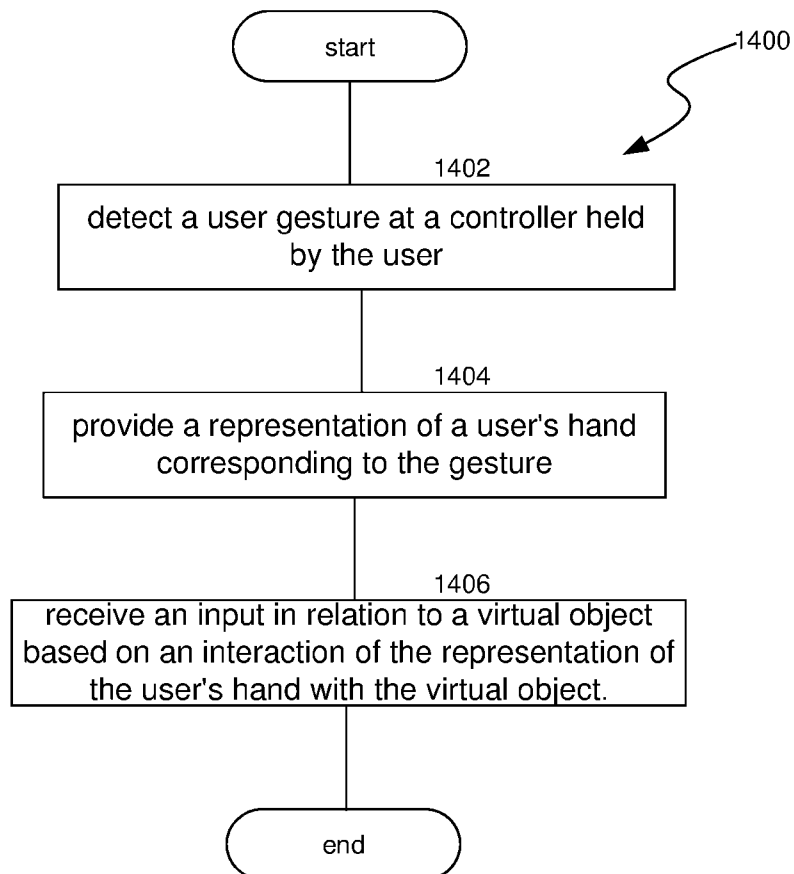


FIG. 13B

**FIG. 14**

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2024/013757

A. CLASSIFICATION OF SUBJECT MATTER		
INV.	G06F3/01	G06F3/03
	G06F3/0346	G06F3/04842
ADD.		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols)		
G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)		
EPO-Internal		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2022/150113 A1 (MICROSOFT TECHNOLOGY LICENSING LLC [US]) 14 July 2022 (2022-07-14) abstract figures 11A, 11B paragraph [0011] paragraph [0034] paragraph [0042] - paragraph [0047] paragraph [0071] - paragraph [0073] -----	1-4, 6-8, 11-14
X	Gamespot: "Oculus Touch Review", / 5 December 2016 (2016-12-05), XP093153438, Retrieved from the Internet: URL:https://www.youtube.com/watch?v=YO7Y0K EhUog Marks at [00:08]-[00:14], [00:19] ----- -/--	1-14
☒ Further documents are listed in the continuation of Box C.		
☒ See patent family annex.		
* Special categories of cited documents :		
"A" document defining the general state of the art which is not considered to be of particular relevance		
"E" earlier application or patent but published on or after the international filing date		
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)		
"O" document referring to an oral disclosure, use, exhibition or other means		
"P" document published prior to the international filing date but later than the priority date claimed		
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention		
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone		
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art		
"&" document member of the same patent family		
Date of the actual completion of the international search		Date of mailing of the international search report
24 April 2024		07/05/2024
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Fedrigo, Enrico

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2024/013757

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Meta Quest: "Oculus Touch - Hand Presence Technology", , 7 October 2016 (2016-10-07), XP093153437, Retrieved from the Internet: URL: https://www.youtube.com/watch?v=pppkQ4jIrMU [00:37], [00:48]–[00:55], [01:06]–[01:15], [02:23] -----	1–14

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2024/013757

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2022150113 A1	14-07-2022	CN 116783572 A	19-09-2023
		EP 4275106 A1	15-11-2023
		US 2022221928 A1	14-07-2022
		US 2023115959 A1	13-04-2023
		WO 2022150113 A1	14-07-2022
