



- (51) International Patent Classification: G06F 3/01 (2006.01)
- (21) International Application Number: PCT/US2018/046223
- (22) International Filing Date: 10 August 2018 (10.08.2018)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data: 62/543,775 10 August 2017 (10.08.2017) US
- (71) Applicant: GOOGLE LLC [US/US]; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US).
- (72) Inventors: CHEN, Shiqi; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US). TOMPSON, Jonathan; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US). GARG, Rahul; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US).
- (74) Agent: WALLENFELT, Brian P. et al.; Brake Hughes Bellermann LLP, c/o CPA Global, 900 2nd Ave South, Suite 600, Minneapolis, Minnesota 55402 (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, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, 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, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, 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, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) Title: CONTEXT-SENSITIVE HAND INTERACTION

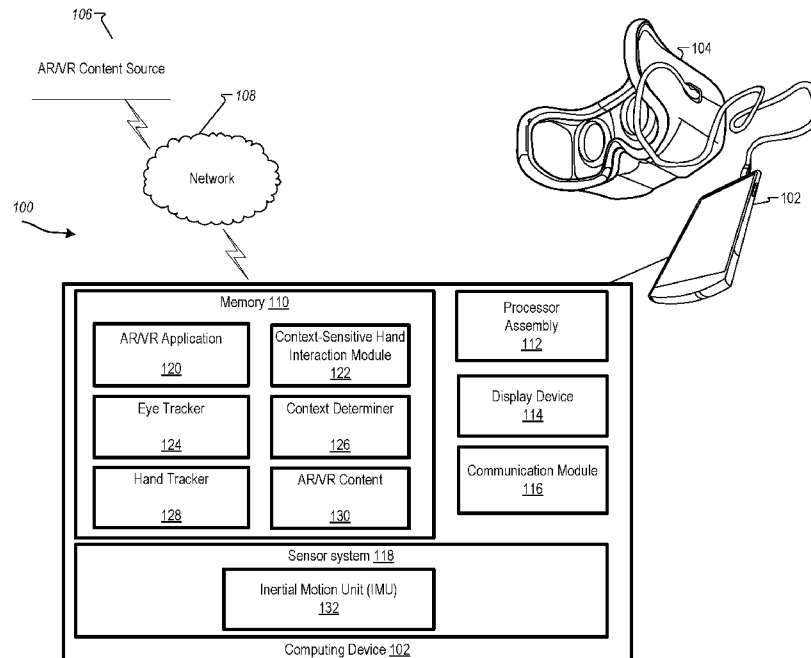


FIG. 1

(57) Abstract: Systems and methods for context-sensitive hand interaction with an immersive environment are provided. An example method includes determining a contextual factor for a user and selecting an interaction mode based on the contextual factor. The example method may also include monitoring a hand of the user to determine a hand property and determining an interaction with an immersive environment based on the interaction mode and the hand property.



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))*
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

# CONTEXT-SENSITIVE HAND INTERACTION

## CROSS-REFERENCE TO RELATED APPLICATION

**[0001]** This application claims priority to U.S. Application No. 62/543,775, filed on August 10, 2017, the disclosure of which is incorporated herein by reference in its entirety.

## BACKGROUND

**[0002]** Various types of systems can generate immersive environments for users. For example, a virtual reality (VR) system generates an immersive virtual environment for a user. The immersive virtual environment can be three-dimensional (3D) and can include multiple virtual objects with which the user may interact. As another example, an augmented reality (AR) system generates an immersive augmented environment for a user. The immersive augmented environment can be generated by superimposing computer-generated graphical content, such as images, on a user's field of view of the real world.

**[0003]** The user can experience these immersive environments via various display devices such as, for example, a helmet or other head mounted device including a display, glasses, or goggles that a user looks through when viewing a display device.

**[0004]** The user may interact with the immersive environment in a variety of ways. For example, the user may use hand movements to interact with the immersive environment. The user may also manipulate a controller to interact with the immersive environment.

## SUMMARY

**[0005]** This document relates, generally, to context-sensitive hand interaction in an immersive environment rendering. In some implementations, the foveated rendering is smoothly varying.

**[0006]** One aspect is a method comprising: determining a contextual factor for a user; selecting an interaction mode based on the contextual factor; monitoring a hand of the user to determine a hand property; and determining an interaction with an immersive environment based on the interaction mode and the hand property.

**[0007]** Another aspect is a system comprising: at least one processor; and memory storing instructions that, when executed by the at least one processor, cause the system to: determine a contextual factor for a user; select an interaction mode based on the contextual

factor; monitor the user's hand to determine a hand property; and determine an interaction with an immersive environment based on the interaction mode and the hand property.

**[0008]** Another aspect is a method comprising: generating an immersive environment that includes a three-dimensional virtual canvas; determining an orientation of a handheld electronic device held in a first hand of a user; based on the orientation of the handheld electronic device indicating the first hand is in a first state, selecting a positional interaction mode; using a hand tracker to determine a position of a second hand of the user; mapping the position of the second hand to a virtual hand position in the immersive environment; generating a visual entity in the three-dimensional virtual canvas based on the virtual hand position; monitoring for a change to the orientation of the handheld electronic device; and responsive to the orientation of the handheld device changing to a changed orientation: determining that the changed orientation indicates the first hand is in a second state; and responsive to determining that the first hand is in the second state, selecting a ray-based collision mode as the interaction mode and generating a virtual ray that extends in a pointing direction from the virtual hand position.

**[0009]** The details of one or more implementations are set forth in the accompanying drawings and the description below. Other features will be apparent from the description and drawings, and from the claims.

#### **BRIEF DESCRIPTION OF THE DRAWINGS**

**[0010]** FIG. 1 is a block diagram illustrating a system according to an example implementation.

**[0011]** FIG. 2 is a flowchart of an example method of providing context-sensitive hand interaction, in accordance with implementations as described herein.

**[0012]** FIGS. 3A, 3B, and 3C are diagrams depicting an example head-mounted display device and controller, in accordance with implementations as described herein.

**[0013]** FIG. 4 is a third-person view of a physical space in which a user is experiencing an immersive environment through the HMD of FIG. 1.

**[0014]** FIG. 5 is another third-person view of the physical space of FIG. 4 in which the user is experiencing the immersive environment of FIG. 4.

**[0015]** FIG. 6 is a third-person view of a physical space in which a user is experiencing an immersive environment through the HMD of FIG. 1.

**[0016]** FIG. 7 is another third-person view of the physical space of FIG. 6 in which

the user is experiencing the immersive environment of FIG. 6.

**[0017]** FIGS. 8 and 9 show third-person views of a physical space in which a user is experiencing an immersive environment through the HMD of FIG. 1.

**[0018]** FIG. 10 shows an example of a computer device and a mobile computer device that can be used to implement the techniques described herein.

### DETAILED DESCRIPTION

**[0019]** Reference will now be made in detail to non-limiting examples of this disclosure, examples of which are illustrated in the accompanying drawings. The examples are described below by referring to the drawings, wherein like reference numerals refer to like elements. When like reference numerals are shown, corresponding description(s) are not repeated and the interested reader is referred to the previously discussed figure(s) for a description of the like element(s).

**[0020]** At least some implementations of VR systems and AR systems include a head-mounted display device (HMD) that can be worn by a user to generate an immersive environment for the user. Examples of an immersive environment include a VR environment generated by a VR system and an AR environment generated by an AR system. In at least some implementations, the HMD includes a stereoscopic display in which different images and/or videos are shown to each of the user's eyes to convey depth. The HMD may display images that cover some (AR) or all (VR) of a user's field of view. The HMD may also track the movement of the user's head and/or pupil location. As the user's head moves, the HMD may display updated images that correspond to the user's changing orientation and/or position within the AR or VR environment.

**[0021]** Some implementations of VR and/or AR systems include a hand tracker that can track a user's hands. The hand tracker may track one of the user's hands or both of the user's hands. The hand tracker may be a component of the HMD or may be separate from the HMD. In some implementations, the hand tracker performs key point detection to track one or more points of a user's hand, such as finger tips, knuckles, etc.

**[0022]** The hand tracker may determine various hand properties of the user's hand. In some implementations, a hand property may correspond to an action performed by a hand of a user to provide user input to the immersive environment. For example, the hand action may control a virtual entity in the immersive environment or interact with the immersive environment. The hand action may include positioning the hand in a specific place, moving the hand in a specific motion, or arranging the fingers of the hand in a specific configuration.

**[0023]** In some implementations, a hand property is a measurement of one or more of a position, relative position, orientation, relative orientation, motion, or configuration (e.g., arrangement of fingers) of a user's hand. The relative position and relative orientation may be determined with respect to a direction of gravitational force, the position or orientation of an HMD worn by the user, or a position or orientation of a portion of the user's body. In some implementations, a hand property is determined based on measurements of one or more of a position, relative position, orientation, relative orientation, motion, or arrangement of a user's hand. For example, a hand property may be a gesture (e.g., a continuous motion) performed by a user's hand that may be determined based on a sequence of measurements of the position, orientation, motion, and arrangement of the user's hand.

**[0024]** An example hand property is a position of the user's hand. Another example hand property is a pointing direction of a finger of the user. Another example hand property is a position of a finger of the user. Another example hand property is a gesture performed by the hand. Another example hand property is an arrangement (or hand sign/signal) of the user's hand (e.g., the positions and orientations of at least some of the fingers relative to the hand). Another example hand property is a movement of the user's hand from a first position to a second position.

**[0025]** In some implementations, hand properties are determined based on measurements (or sequences of measurements) of the position, motion, orientation, or six-degrees-of-freedom pose of a handheld controller. These measurements may be used to infer one or more of a position, relative position, orientation, relative orientation, motion, or arrangement of a user's hand and, in turn, a hand property.

**[0026]** Some implementations of VR and/or AR systems also determine an interaction mode. In some implementations, the interaction mode defines how user inputs interact with the immersive environment. For example, the interaction mode may define which hand properties are determined and how the hand properties impact the immersive environment. The interaction mode may be stored as variable or setting in the system. Based on the determined interaction mode and the determined hand property, an interaction with the immersive environment is performed. Non-limiting examples of interaction modes include a positional interaction mode, a ray-based collision mode, and a point-based collision mode.

**[0027]** In at least some implementations, the interaction mode is selected based on a contextual factor. For example, the VR or AR system may determine a contextual factor for the user and then select an interaction mode based on the contextual factor.

**[0028]** In some implementations, the contextual factor is a property of the physical

environment, the immersive environment, or the user. For example, the contextual factor may be a property of the user's body position, the user's head position, or the user's eye position. In some implementations, the contextual factor is determined based on a position, orientation, or six-degree-of-freedom pose of one or more of handheld controllers or HMDs. The contextual factor may also be a state of a user input control of the system.

**[0029]** In some implementations, the contextual factors include properties of the user in a physical environment, properties of a representation of the user in an immersive environment, or values of settings in the system that a user may be able to modify. In some implementations, a contextual factor is a property of the physical environment, the immersive environment, the application environment, the user, or a representation of the user in the immersive environment. Non-limiting examples of contextual factors include a mode setting, an actuation state of a user-input element, a physical position of a user's hand, a position or orientation of a handheld electronic device, a proximity between the user's hands, a virtual position of the user's hand in the immersive environment, an orientation of the user's head, and a focus point determined based on the user's gaze.

**[0030]** In some implementations, the interaction mode may be selected based on various determined hand properties, such as the pose of the user's hand. The pose may, for example, be a six degrees of freedom pose of the user's hand that includes the position and orientation of the user's hand. For example, the interaction mode may be selected based on the pose of the user's hand and the interaction may be determined based on another hand property, such as the position of the user's hand or a finger of the user's hand.

**[0031]** In one example, the VR or AR system generates an immersive painting environment. For example, the immersive painting environment may allow the user to paint within a three-dimensional canvas volume. In some implementations, the three-dimensional canvas volume occupies a portion of the immersive environment. In other implementations, the three-dimensional canvas volume occupies the entire immersive volume (i.e., the user can paint anywhere). The immersive painting environment may also include a user interface entity through which the user can perform various actions (e.g., cut-and-paste, save the canvas volume, etc.) and adjust various settings, such as paint color, brush type/style, etc.

**[0032]** The VR or AR system may determine an interaction mode for the user within the immersive painting environment based on a contextual factor for a user. For instance, in a first interaction mode that provides for positional interaction, the position of the user's hand may be tracked and mapped to a virtual hand position. Visual entities may then be generated in the immersive environment at the virtual hand positions to, for example, represent paint

from a paint brush. The visual entities may include vertices, polygonal meshes, sprites, voxels, and/or point clouds. In a second interaction mode that provides for ray-based collisions, both the position of the user's hand and a pointing direction of a finger of the user's hand may be tracked and used to generate a virtual ray that extends in the pointing direction from the finger of the user's hand in the immersive environment. The VR or AR system may then identify entities in the immersive environment that intersect with the virtual ray. For example, the user may perform an action using the user-interface entity by directing the virtual ray at a portion of the user-interface entity in the immersive environment.

**[0033]** In some implementations, the VR or AR system uses properties of a first hand of the user to interact in the immersive environment (e.g., to generate visual entities or to generate a virtual ray as described above) and determines an interaction mode based on contextual factors determined about a second hand of the user. For example, if the second hand is in an inactive position (e.g., behind the user or at the user's side), the VR or AR system may select the first interaction mode in which the user can paint by generating visual entities in the canvas volume based on the position/movements of the first hand. However, when, the second hand is moved to an active position (e.g., in front of the user, near the first hand, raised), the VR or AR system may switch to the second interaction mode in which a virtual ray is generated from the user's finger in the immersive environment. The position of the second hand may be determined using the hand tracker. Additionally, the position of the second hand may be determined using a handheld electronic device that is held by the user. For example, the handheld electronic device may include an inertial motion unit (IMU) that may be able to determine the orientation, acceleration, and/or motion of the handheld electronic device. The position/orientation of the user's hand may be inferred from one or more of the determined orientation, acceleration, and/or motion of the handheld electronic device.

**[0034]** Additionally, the VR or AR system may switch between the first interaction mode and the second interaction mode based on a user input via a user-input component of the handheld electronic device, such as a button, touchscreen, switch, dial, or other type of user-input component. In some implementations, the VR or AR system switches between interaction modes based on the orientation of the user's head, an action by the user's head (e.g., a head nod), a direction of the user's gaze, or a speech input.

**[0035]** FIG. 1 is a block diagram illustrating a system 100 according to an example implementation. The system 100 generates an immersive environment, such as an AR environment or VR environment, for a user of the system 100. In some implementations, the

system 100 includes a computing device 102, a head-mounted display device (HMD) 104, and an AR/VR content source 106. Also shown is a network 108 over which the computing device 102 may communicate with the AR/VR content source 106.

**[0036]** In some implementations, the computing device 102 is a mobile device (e.g., a smart phone) which may be configured to provide or output immersive AR/VR content to a user. The computing device 102 may include a memory 110, a processor assembly 112, a display device 114, a communication module 116, and a sensor system 118. The memory 110 may include an AR/VR application 120, a context-sensitive hand interaction module 122, an eye tracker 124, a context determiner 126, a hand tracker 128, and AR/VR content 130. The computing device 102 may also include various user input components (not shown) such as a handheld electronic device that communicates with the computing device 102 using a wireless communications protocol.

**[0037]** The memory 110 can include one or more non-transitory computer-readable storage media. The memory 110 may store instructions and data that are usable to generate an immersive environment for a user.

**[0038]** The processor assembly 112 includes one or more devices that are capable of executing instructions, such as instructions stored by the memory 110, to perform various tasks, such as generating an immersive environment, determining contextual factors, selecting an interaction mode based on contextual factors, and managing hand-based user interaction. For example, the processor assembly 112 may include a central processing unit (CPU) and/or a graphics processor unit (GPU). For example, if a GPU is present, some image or video rendering tasks associated with generating the immersive environment may be offloaded from the CPU to the GPU.

**[0039]** The display device 114 may, for example, include an LCD (liquid crystal display) screen, an OLED (organic light emitting diode) screen, a touchscreen, or any other screen or display for displaying images or information to a user. In some implementations, the display device 114 includes a light projector arranged to project light onto a portion of a user's eye when the HMD 104 is worn.

**[0040]** The communication module 116 includes one or more devices for communicating with other computing devices, such as the AR/VR content source 106. The communication module 116 may communicate via wireless or wired networks.

**[0041]** The sensor system 118 may include various sensors, including an inertial motion unit (IMU) 132. Implementations of the sensor system 118 may also include different types of sensors, including, for example, a light sensor, an audio sensor, an image sensor, a

distance sensor, a proximity sensor, a contact sensor such as a capacitive sensor, a timer, and/or other sensors and/or different combination(s) of sensors.

**[0042]** The IMU 132 detects motion, movement, and/or acceleration of the computing device 102 and/or the HMD 104. The IMU 132 may include various types of sensors such as, for example, an accelerometer, a gyroscope, a magnetometer, and other such sensors. A position and orientation of the HMD 104 may be detected and tracked based on data provided by the sensors included in the IMU 132. The IMU 132 may include various different types of sensors such as, for example, an accelerometer, a gyroscope, a magnetometer, and other such sensors. The detected position and orientation of the HMD 104 may allow the system to detect and track the user's gaze direction and/or head movement.

**[0043]** The AR/VR application 120 may generate and present an immersive environment to a user via one or more output devices of the computing device 102 such as the display device 114, a speaker(s) (not shown), and/or other output devices. In some implementations, the AR/VR application 120 includes instructions stored in the memory 110 that, when executed by the processor assembly 112, cause the processor assembly 112 to perform the operations described herein. For example, the AR/VR application 120 may generate and present an immersive environment to the user based on, for example, AR/VR content, such as the AR/VR content 130 and/or AR/VR content received from the AR/VR content source 106. The AR/VR content 130 may include 3D scenes that can be rendered as images or videos for display on the display device 114. For example, the 3D scene can include one or more objects represented as polygonal meshes. The polygonal meshes may be associated with various surface textures, such as colors and images. The 3D scene may also include other information such as, for example, light sources that are used in rendering the 3D scene.

**[0044]** The AR/VR application 120 may use the context-sensitive hand interaction module 122 to select an interaction mode. In some implementations, the context-sensitive hand interaction module 122 includes instructions stored in the memory 110 that, when executed by the processor assembly 112, cause the processor assembly 112 to perform the operations described herein. The interaction mode may be used to determine how interaction hand properties as interactions within the immersive environment. As described above, non-limiting examples of interaction modes include a positional interaction mode, a ray-based collision mode, and a point-based collision mode. For example, the context-sensitive hand interaction module 122 may determine the interaction mode based on one or more contextual factors determined by the context determiner 126.

**[0045]** The context determiner 126 may determine one or more contextual factors about the user, the immersive environment, and/or the AR/VR application using, for example, the eye tracker 124, the hand tracker 128, the sensor system 118, and/or a handheld electronic device (not shown). Non-limiting examples of contextual factors determined by implementations of the context determiner 126 include a mode chosen by a user, a user input via a user-input component of the handheld electronic device, a position or orientation of at least one of the user's hands in a physical space, a position or orientation of at least one of the user's hands in the immersive environment, an arrangement of the user's hand/s, a speech input, an orientation of the user's head, a movement of the user's head, a direction of the user's gaze, and/or a focus position of the user's gaze within the immersive environment.

**[0046]** In some implementations, the eye tracker 124 includes instructions stored in the memory 110 that, when executed by the processor assembly 112, cause the processor assembly 112 to perform the operations described herein. For example, the eye tracker 124 may determine a location on the display device 114 at which the user's gaze is directed. The eye tracker 124 may make this determination based on identifying and tracking the location of the user's pupils in images captured by an imaging device of the sensor system 118. Based on the identified location of the user's pupils, a gaze direction for the user may be determined and/or a focus position of the user's gaze within the immersive environment.

**[0047]** In some implementations, the hand tracker 128 includes instructions stored in the memory 110 that, when executed by the processor assembly 112, cause the processor assembly 112 to perform the operations described herein. In some implementations, the hand tracker 128 also includes one or more imaging systems, such as visible light cameras or infrared cameras. In some implementations, the hand tracker 128 includes a wearable component such as visible markers, gloves, gesture control armbands, etc. The hand tracker 128 may determine a location in physical space of one or more key points on one or both of the user's hands. For example, a virtual position and orientation of the user's hand in the immersive environment may then be determined.

**[0048]** The AR/VR application 120 may update the AR/VR environment based on input received from the IMU 132 and/or other components of the sensor system 118. For example, the IMU 132 may detect motion, movement, and/or acceleration of the computing device 102 and/or the display device 114. A position and orientation of the HMD 104 may be detected and tracked based on data provided by the sensors included in the IMU 132. The detected position and orientation of the HMD 104 may allow the system to, in turn, detect and track the user's gaze direction and head movement. Based on the detected gaze direction

and head movement, the AR/VR application 120 may update the AR/VR environment to reflect a changed orientation and/or position of the user within the environment.

**[0049]** Although the computing device 102 and the HMD 104 are shown as separate devices in FIG. 1, in some implementations, the computing device 102 may include the HMD 104. In some implementations, the computing device 102 communicates with the HMD 104 via a cable, as shown in FIG. 1. For example, the computing device 102 may transmit audio and video signals to the HMD 104 for display for the user, and the HMD 104 may transmit motion, position, and/or orientation information to the computing device 102. In some implementations, the HMD 104 includes a chamber in which the computing device 102 may be placed. In some implementations, the user is able to view the display device 114 of the computing device 102 while wearing the HMD 104 (e.g., through lenses or apertures within the HMD 104). For example, the computing device 102 and the HMD 104 can together function as a stereoscopic viewer by partitioning a screen of the display device 114 into a first image that is viewable by only the left eye of the user when viewed through the HMD and a second image that is viewable by only the right eye of the user when viewed through the HMD.

**[0050]** The AR/VR content source 106 may generate and output AR/VR content, which may be distributed or sent to one or more computing devices, such as the computing device 102, via the network 108. In an example implementation, the AR/VR content includes three-dimensional scenes and/or images. Additionally, the AR/VR content may include audio/video signals that are streamed or distributed to one or more computing devices. The AR/VR content may also include an AR/VR application that runs on the computing device 102 to generate 3D scenes, audio signals, and/or video signals and to process hand interactions with an immersive environment. According to an illustrative example implementation, virtual reality (VR) may generate an immersive environment by, at least in some cases, replicating or simulating, to varying degrees, an environment or physical presence in places in the real world or imagined worlds or environments. Augmented reality (AR) may generate an immersive environment by, at least in some cases, overlaying computer generated graphical content, such as images, on a user's field of view of the real world.

**[0051]** The network 108 may be the Internet, a local area network (LAN), a wireless local area network (WLAN), and/or any other network. The computing device 102, for example, may receive the audio/video signals via the network 108, which may be provided as part of VR content in an illustrative example implementation.

**[0052]** FIG. 2 is a flowchart of an example method 200 of providing for context-based hand interaction. For example, the method 200 may be performed by implementations of the context-sensitive hand interaction module 122 to generate interactions within an immersive environment generated by the AR/VR application 120.

**[0053]** At operation 202, a contextual factor is determined for a user. In some implementations, multiple contextual factors are determined for the user. The contextual factors may be determined by, for example, the context determiner 126. At least some of the contextual factors may be determined, at least in part, using the eye tracker 124 and/or the hand tracker 128. In some implementations, contextual factors are determined from head motions and/or speech inputs.

**[0054]** At operation 204, an interaction mode is selected based on the contextual factor. In some implementations, the interaction mode is selected based on multiple contextual factors. The interaction mode can, for example, control how hand properties, such as positions, orientations, arrangements, poses, gestures, and motions, are converted to interactions within the immersive environment. Non-limiting examples of interaction modes include a positional interaction mode, a ray-based collision mode, and a point-based collision mode.

**[0055]** At operation 206, the user's hand is monitored to determine a hand property. In some implementations, multiple hand properties are determined. The hand properties may be determined using the hand tracker 128. Additionally, different hand properties may be determined based on the interaction mode determined at operation 204. In some implementations, a pointing direction of a finger of the user's hand may be determined only when in the ray-based collision mode in some implementations. In this manner, fewer processor cycles may be used when in the other interaction modes because the pointing direction is not determined. However, in some implementations, the same hand properties are determined in multiple interaction modes.

**[0056]** At operation 208, an interaction is determined with the immersive environment based on the interaction mode determined at operation 204 and the hand property determined at operation 206. For example, a visual entity (e.g., to represent an artistic stroke with a paintbrush) may be generated in some interaction modes. While in other interaction modes, a user interface element may be actuated to select a menu option or adjust the value of a setting.

**[0057]** FIGS. 3A and 3B are perspective views of an example HMD 300, such as, for example, the HMD 104 of FIG. 1, and FIG. 3C illustrates an example handheld electronic

device 302 that is usable with the HMD 300.

**[0058]** The handheld electronic device 302 may include a housing 303 in which internal components of the handheld electronic device 302 are received, and a user interface 304 on an outside of the housing 303, accessible to the user. The user interface 304 may include a touch sensitive surface 306 configured to receive user touch inputs. The user interface 304 may also include other components for manipulation by the user such as, for example, actuation buttons, knobs, joysticks and the like. In some implementations, at least a portion of the user interface 304 may be configured as a touchscreen, with that portion of the user interface 304 being configured to display user interface items to the user, and also to receive touch inputs from the user on the touch sensitive surface 306. The handheld electronic device 302 may also include a light source 308 configured to selectively emit light, for example, a beam or ray, through a port in the housing 303, for example, in response to a user input received at the user interface 304.

**[0059]** The HMD 300 may include a housing 310 coupled to a frame 320, with an audio output device 330 including, for example, speakers mounted in headphones, also being coupled to the frame 320. In FIG. 3B, a front portion 310a of the housing 310 is rotated away from a base portion 310b of the housing 310 so that some of the components received in the housing 310 are visible. A display 340 may be mounted on an interior facing side of the front portion 310a of the housing 310. In some implementations, the display 340 is a display device from a computing device, such as the computing device 102 of FIG. 1, that is inserted and secured between the front portion 310a and the base portion 310b.

**[0060]** Lenses 350 may be mounted in the housing 310, between the user's eyes and the display 340 when the front portion 310a is in the closed position against the base portion 310b of the housing 310. In some implementations, the HMD 300 may include a sensing system 360 including various sensors and a control system 370 including a processor 390 and various control system devices to facilitate operation of the HMD 300.

**[0061]** In some implementations, the HMD 300 may include a camera 380 or multiple cameras to capture still and moving images. The images captured by the camera 380 may be used to help track a physical position of the user, the user's hand or hands, and/or the handheld electronic device 302 in the real world, or physical environment relative to the immersive environment, and/or may be displayed to the user on the display 340 in a pass-through mode, allowing the generation of an augmented reality environment that includes a combination of images from the real world and computer generated imagery. In some implementations, the pass-through mode is used to allow the user to temporarily leave the

immersive environment and return to the physical environment without removing the HMD 300 or otherwise changing the configuration of the HMD 300 to move the housing 310 out of the line of sight of the user.

**[0062]** For example, in some implementations, the sensing system 360 may include an inertial measurement unit (IMU) 362 including various different types of sensors such as, for example, an accelerometer, a gyroscope, a magnetometer, and other such sensors. A position and orientation of the HMD 300 may be detected and tracked based on data provided by the sensors included in the IMU 362. The detected position and orientation of the HMD 300 may allow the system to, in turn, detect and track the user's head gaze direction and movement.

**[0063]** In some implementations, the HMD 300 may include a gaze tracking device 365 to detect and track an eye gaze of the user. The gaze tracking device 365 may include, for example, an image sensor 365A, or multiple image sensors 365A, to capture images of the user's eyes, for example, a particular portion of the user's eyes, such as, for example, the pupil, to detect, and track direction and movement of, the user's gaze. In some implementations, the HMD 300 may be configured so that the detected gaze is processed as a user input to be translated into a corresponding interaction in the AR experience or the immersive VR experience. In some implementations, the HMD 300 is configured to use the detected gaze of the user to determine a focus point in the immersive environment, which can then be used to determine an interaction mode.

**[0064]** FIG. 4 is a third-person view of a physical space 400, in which a user U is experiencing an immersive environment 402 through the HMD 104. In this example, the computing device 102 is disposed within the HMD 104 so that the user can see the display device 114 while wearing the HMD 104. The immersive environment 402 is generated by the computing device 102 and displayed on the display device 114 of the computing device 102.

**[0065]** In this example, the immersive environment 402 provided by the AR/VR application 120 is an immersive painting environment. The immersive environment 402 includes a three-dimensional canvas volume in which the user U can virtually paint.

**[0066]** As shown in the figure, an interaction mode is selected based on contextual factors of the user. In this case, the position of a handheld electronic device 302 in the user's left hand (not shown) is determined as a contextual factor and used to select the interaction mode. Because the left hand is down at the user's side, a positional interaction mode is selected.

**[0067]** In this example positional interaction mode, a position of the user's right hand

(RH) is tracked in the real world using, for example, the hand tracker 128. A corresponding virtual position for a virtual paint brush entity 406 within the immersive environment 402 is determined. As the user's right hand moves around, the virtual paint brush entity 406 moves in a corresponding manner within the immersive environment 402 and visual entities 408 are added to the canvas 404 to represent the user's paint strokes. Although the examples herein often relate to painting and may include a virtual paint brush entity, other implementations relate to other activities or applications and may include other types of virtual entities that are controlled by a user.

**[0068]** FIG. 5 is a third-person view of the physical space 400 of FIG. 4, in which the user U is experiencing the immersive environment 402 through the HMD 104. In FIG. 5, the user U continues to interact with an immersive painting environment. However, in this example, the user U is now holding the handheld electronic device 302 up in front of the user. The position of the handheld electronic device 302 causes the computing device 102 to select a ray-based intersection mode as the interaction mode. In this example, multiple hand properties are determined in the ray-based intersection mode. First, the position of the user's right hand (RH) is determined and mapped to a virtual hand position in the immersive environment 402. Additionally, because the ray-based intersection mode has been selected, a pointing direction of a finger of the user's right hand is determined. A virtual ray 502 is generated within the immersive environment 402 extending in the determined pointing direction from the virtual hand position. As the user's hand moves in the physical world, the virtual ray 502 will move in the immersive environment 402 allowing the user to point to and interact with various user interface elements of a user interface entity 504. In some implementations, the user interface entity 504 is a virtual entity that is displayed in the immersive environment 402 based on a determined interaction mode, when the user performs a specific action (e.g., holding up the handheld electronic device 302), or when the user orients toward a specific position in the immersive environment 402.

**[0069]** FIG. 6 is a third-person view of a physical space 600, in which a user U is experiencing an immersive environment 602 through the HMD 104. In this example, the computing device 102 is disposed within the HMD 104 so that the user can see the display device 114 while wearing the HMD 104. The immersive environment 602 is generated by the computing device 102 and displayed on the display device 114 of the computing device 102.

**[0070]** In this example, the immersive environment 602 provided by the AR/VR application 120 is an immersive nature scene and includes a virtual keyboard 604. The immersive nature scene may include various natural entities with which the user U can

interact.

**[0071]** As shown in the figure, an interaction mode is selected based on contextual factors of the user. In this case, a virtual hand position 606 of the user's right hand (RH) is used to select an interaction mode. For example, based on the distance between the virtual hand position 606 and the position of the virtual keyboard 604 exceeding a threshold, a point-based collision mode is selected as the interaction mode. In the point-based collision mode, a region 608 around a point of the virtual hand position 606 is used for interactions with entities in the immersive environment. For example, if an entity from the immersive environment 602 (e.g., the peak of the mountain) is within the region 608, an interaction with that entity may occur. As the user's right hand moves around in the physical space 600, the virtual hand position 606 will move around correspondingly in the immersive environment 602.

**[0072]** FIG. 7 is a third-person view of the physical space 600 of FIG. 6, in which the user U is experiencing the immersive environment 602 through the HMD 104. In FIG. 7, the user U continues to interact within the immersive nature scene. However, in this example, the right hand of the user U has moved so as to cause the virtual hand position 606 to be positioned closer to the virtual keyboard 604. For example, based on the distance between the virtual hand position 606 and the position of the virtual keyboard 604 being within a threshold, a ray-based collision mode is selected as the interaction mode. In the ray-based collision mode, a virtual ray 610 extends from the virtual hand position 606 toward the virtual keyboard 604. Beneficially, the ray-based collision mode may be more intuitive for the user U to interact with a user-interface entity such as the virtual keyboard 604.

**[0073]** FIGS. 8 and 9 show third-person views of a physical space 800, in which a user U is experiencing an immersive environment 802 through the HMD 104. In this example, the computing device 102 is disposed within the HMD 104 so that the user can see the display device 114 while wearing the HMD 104. The immersive environment 802 is generated by the computing device 102 and displayed on the display device 114 of the computing device 102.

**[0074]** In this example, the immersive environment 802 provided by the AR/VR application 120 is an immersive nature scene and includes a virtual keyboard 804. The immersive nature scene may include various natural entities with which the user U can interact. FIGS. 8 and 9 are similar to the examples of FIGS. 6 and 7 except that the interaction mode is selected based on a focus position in the immersive environment 802 rather than a virtual hand position.

**[0075]** As shown in FIG. 8, the interaction mode is selected based on contextual factors of the user. In this case, a focus position 806 within the immersive environment 802 is determined based on the user's gaze direction as determined by, for example, the eye tracker 124. For example, based on the distance between the focus position 806 and the position of the virtual keyboard 804 exceeding a threshold, a point-based collision mode is selected as the interaction mode. As described previously in the point-based collision mode, a virtual hand position 808 is determined from the position and/or movement of the user's right hand (RH) in the physical space 800. In some implementations, a region 810 around a point of the virtual hand position 808 is then used for interactions with entities in the immersive environment.

**[0076]** As shown in FIG. 9, the user U continues to interact within the immersive nature scene of the immersive environment 802. However, in this figure, the user's gaze has shifted and the focus position 806 is closer to the virtual keyboard 804. In some implementations, based on the distance between the focus position 806 and the position of the virtual keyboard 804 being within a threshold, a ray-based collision mode is selected as the interaction mode. In the ray-based collision mode, a virtual ray 812 extends from the virtual hand position 808 toward the virtual keyboard 804. As described before, the ray-based collision mode may be more intuitive for the user U to interact with a user-interface entity such as the virtual keyboard 804.

**[0077]** FIG. 10 shows an example of a computer device 1000 and a mobile computer device 1050, which may be used with the techniques described here. Computing device 1000 includes a processor 1002, memory 1004, a storage device 1006, a high-speed interface 1008 connecting to memory 1004 and high-speed expansion ports 1010, and a low speed interface 1012 connecting to low speed bus 1014 and storage device 1006. Each of the components 1002, 1004, 1006, 1008, 1010, and 1012, are interconnected using various busses, and may be mounted on a common motherboard or in other manners as appropriate. The processor 1002 can process instructions for execution within the computing device 1000, including instructions stored in the memory 1004 or on the storage device 1006 to display graphical information for a GUI on an external input/output device, such as display 1016 coupled to high speed interface 1008. In other implementations, multiple processors and/or multiple buses may be used, as appropriate, along with multiple memories and types of memory. Also, multiple computing devices 1000 may be connected, with each device providing portions of the necessary operations (e.g., as a server bank, a group of blade servers, or a multi-processor system).

**[0078]** The memory 1004 stores information within the computing device 1000. In one implementation, the memory 1004 is a volatile memory unit or units. In another implementation, the memory 1004 is a non-volatile memory unit or units. The memory 1004 may also be another form of computer-readable medium, such as a magnetic or optical disk.

**[0079]** The storage device 1006 is capable of providing mass storage for the computing device 1000. In one implementation, the storage device 1006 may be or contain a computer-readable medium, such as a floppy disk device, a hard disk device, an optical disk device, or a tape device, a flash memory or other similar solid state memory device, or an array of devices, including devices in a storage area network or other configurations. A computer program product can be tangibly embodied in an information carrier. The computer program product may also contain instructions that, when executed, perform one or more methods, such as those described above. The information carrier is a computer- or machine-readable medium, such as the memory 1004, the storage device 1006, or memory on processor 1002.

**[0080]** The high speed controller 1008 manages bandwidth-intensive operations for the computing device 1000, while the low speed controller 1012 manages lower bandwidth-intensive operations. Such allocation of functions is exemplary only. In one implementation, the high-speed controller 1008 is coupled to memory 1004, display 1016 (e.g., through a graphics processor or accelerator), and to high-speed expansion ports 1010, which may accept various expansion cards (not shown). In the implementation, low-speed controller 1012 is coupled to storage device 1006 and low-speed expansion port 1014. The low-speed expansion port, which may include various communication ports (e.g., USB, Bluetooth, Ethernet, wireless Ethernet) may be coupled to one or more input/output devices, such as a keyboard, a pointing device, a scanner, or a networking device such as a switch or router, e.g., through a network adapter.

**[0081]** The computing device 1000 may be implemented in a number of different forms, as shown in the figure. For example, it may be implemented as a standard server 1020, or multiple times in a group of such servers. It may also be implemented as part of a rack server system 1024. In addition, it may be implemented in a personal computer such as a laptop computer 1022. Alternatively, components from computing device 1000 may be combined with other components in a mobile device (not shown), such as device 1050. Each of such devices may contain one or more of computing device 1000, 1050, and an entire system may be made up of multiple computing devices 1000, 1050 communicating with each other.

**[0082]** Computing device 1020 includes a processor 1052, memory 1064, an input/output device such as a display 1054, a communication interface 1066, and a transceiver 1068, among other components. The device 1050 may also be provided with a storage device, such as a microdrive or other device, to provide additional storage. Each of the components 1050, 1052, 1064, 1054, 1066, and 1068, are interconnected using various buses, and several of the components may be mounted on a common motherboard or in other manners as appropriate.

**[0083]** The processor 1052 can execute instructions within the computing device 1020, including instructions stored in the memory 1064. The processor may be implemented as a chipset of chips that include separate and multiple analog and digital processors. The processor may provide, for example, for coordination of the other components of the device 1050, such as control of user interfaces, applications run by device 1050, and wireless communication by device 1050.

**[0084]** Processor 1052 may communicate with a user through control interface 1058 and display interface 1056 coupled to a display 1054. The display 1054 may be, for example, a TFT LCD (Thin-Film-Transistor Liquid Crystal Display) or an OLED (Organic Light Emitting Diode) display, or other appropriate display technology. The display interface 1056 may include appropriate circuitry for driving the display 1054 to present graphical and other information to a user. The control interface 1058 may receive commands from a user and convert them for submission to the processor 1052. In addition, an external interface 1062 may be provide in communication with processor 1052, so as to enable near area communication of device 1050 with other devices. External interface 1062 may provide, for example, for wired communication in some implementations, or for wireless communication in other implementations, and multiple interfaces may also be used.

**[0085]** The memory 1064 stores information within the computing device 1020. The memory 1064 can be implemented as one or more of a computer-readable medium or media, a volatile memory unit or units, or a non-volatile memory unit or units. Expansion memory 1074 may also be provided and connected to device 1050 through expansion interface 1072, which may include, for example, a SIMM (Single In Line Memory Module) card interface. Such expansion memory 1074 may provide extra storage space for device 1050, or may also store applications or other information for device 1050. Specifically, expansion memory 1074 may include instructions to carry out or supplement the processes described above, and may include secure information also. Thus, for example, expansion memory 1074 may be provided as a security module for device 1050, and may be programmed with instructions

that permit secure use of device 1050. In addition, secure applications may be provided via the SIMM cards, along with additional information, such as placing identifying information on the SIMM card in a non-hackable manner.

**[0086]** The memory may include, for example, flash memory and/or NVRAM memory, as discussed below. In one implementation, a computer program product is tangibly embodied in an information carrier. The computer program product contains instructions that, when executed, perform one or more methods, such as those described above. The information carrier is a computer- or machine-readable medium, such as the memory 1064, expansion memory 1074, or memory on processor 1052, that may be received, for example, over transceiver 1068 or external interface 1062.

**[0087]** Device 1050 may communicate wirelessly through communication interface 1066, which may include digital signal processing circuitry where necessary. Communication interface 1066 may provide for communications under various modes or protocols, such as GSM voice calls, SMS, EMS, or MMS messaging, CDMA, TDMA, PDC, WCDMA, CDMA2000, or GPRS, among others. Such communication may occur, for example, through radio-frequency transceiver 1068. In addition, short-range communication may occur, such as using a Bluetooth, Wi-Fi, or other such transceiver (not shown). In addition, GPS (Global Positioning System) receiver module 1070 may provide additional navigation- and location-related wireless data to device 1050, which may be used as appropriate by applications running on device 1050.

**[0088]** Device 1050 may also communicate audibly using audio codec 1060, which may receive spoken information from a user and convert it to usable digital information. Audio codec 1060 may likewise generate audible sound for a user, such as through a speaker, e.g., in a handset of device 1050. Such sound may include sound from voice telephone calls, may include recorded sound (e.g., voice messages, music files, etc.) and may also include sound generated by applications operating on device 1050.

**[0089]** The computing device 1020 may be implemented in a number of different forms, as shown in the figure. For example, it may be implemented as a cellular telephone 1080. It may also be implemented as part of a smart phone 1082, personal digital assistant, or other similar mobile device.

**[0090]** Various implementations of the systems and techniques described here can be implemented in digital electronic circuitry, integrated circuitry, specially designed application specific integrated circuits (ASICs), computer hardware, firmware, software, and/or combinations thereof. These various implementations can include implementation in one or

more computer programs that are executable and/or interpretable on a programmable system including at least one programmable processor, which may be special or general purpose, coupled to receive data and instructions from, and to transmit data and instructions to, a storage system, at least one input device, and at least one output device.

**[0091]** These computer programs (also known as programs, applications, apps, software, software applications or code) include machine instructions for a programmable processor, and can be implemented in a high-level procedural and/or object-oriented programming language, and/or in assembly/machine language. As used herein, the terms “machine-readable medium” and “computer-readable medium” refer to any computer program product, apparatus and/or device (e.g., magnetic discs, optical disks, memory, Programmable Logic Devices (PLDs)) used to provide machine instructions and/or data to a programmable processor, including a machine-readable medium that receives machine instructions as a machine-readable signal. The term “machine-readable signal” refers to any signal used to provide machine instructions and/or data to a programmable processor.

**[0092]** To provide for interaction with a user, the systems and techniques described here can be implemented on a computer having a display device (e.g., an LCD (liquid crystal display) screen, an OLED (organic light emitting diode)) for displaying information to the user and a keyboard, touchscreen, and a pointing device (e.g., a touchpad, mouse, or trackball) by which the user can provide input to the computer. Additionally, the user can provide input via hand or body actions, arrangements, poses, or gestures. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback (e.g., visual feedback, auditory feedback, or tactile feedback); and input from the user can be received in any form, including acoustic, speech, or tactile input.

**[0093]** The systems and techniques described here can be implemented in a computing system that includes a back end component (e.g., as a data server), or that includes a middleware component (e.g., an application server), or that includes a front end component (e.g., a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the systems and techniques described here), or any combination of such back end, middleware, or front end components. The components of the system can be interconnected by any form or medium of digital data communication (e.g., a communication network). Examples of communication networks include a local area network (“LAN”), a wide area network (“WAN”), and the Internet.

**[0094]** The computing system can include clients and servers. A client and server are

generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

**[0095]** In some implementations, the computing devices depicted in FIG. 1 can include sensors that interface with a virtual reality (VR headset/HMD device 1090) to generate an AR or VR environment with context-based hand interactions to improve the interface between the user and the immersive environment and to reduce the number of processor cycles required to provide the immersive environment, increase framerate, and/or reduce the processor cycles required for rendering. For example, one or more sensors included on a computing device 1020 or other computing device depicted in FIG. 1, can provide input to VR headset 1090 or in general, provide input to a VR space. The sensors can include, but are not limited to, a touchscreen, accelerometers, gyroscopes, pressure sensors, biometric sensors, temperature sensors, humidity sensors, and ambient light sensors. The computing device 1020 can use the sensors to determine an absolute position and/or a detected rotation of the computing device in the VR space that can then be used as input to the VR space. For example, the computing device 1020 may be incorporated into the VR space as a virtual object, such as a controller, a laser pointer, a keyboard, a weapon, etc. Positioning of the computing device/virtual object by the user when incorporated into the VR space can allow the user to position the computing device so as to view the virtual object in certain manners in the VR space. For example, if the virtual object represents a laser pointer, the user can manipulate the computing device as if it were an actual laser pointer. The user can move the computing device left and right, up and down, in a circle, etc., and use the device in a similar fashion to using a laser pointer.

**[0096]** In some implementations, one or more input devices included on, or connected to, the computing device 1020 can be used as input to the VR space. The input devices can include, but are not limited to, a touchscreen, a keyboard, one or more buttons, a trackpad, a touchpad, a pointing device, a mouse, a trackball, a joystick, a camera, a microphone, earphones or buds with input functionality, a gaming controller, or other connectable input device. A user interacting with an input device included on the computing device 1020 when the computing device is incorporated into the VR space can cause a particular action to occur in the VR space.

**[0097]** In some implementations, a touchscreen of the computing device 1020 can be rendered as a touchpad in VR space. A user can interact with the touchscreen of the computing device 1020. The interactions are rendered, in the VR headset 1090 for example,

as movements on the rendered touchpad in the VR space. The rendered movements can control virtual objects in the VR space.

**[0098]** In some implementations, one or more output devices included on the computing device 1020 can provide output and/or feedback to a user of the VR headset 1090 in the VR space. The output and feedback can be visual, tactical, or audio. The output and/or feedback can include, but is not limited to, vibrations, turning on and off or blinking and/or flashing of one or more lights or strobes, sounding an alarm, playing a chime, playing a song, and playing of an audio file. The output devices can include, but are not limited to, vibration motors, vibration coils, piezoelectric devices, electrostatic devices, light emitting diodes (LEDs), strobes, and speakers.

**[0099]** In some implementations, the computing device 1020 may appear as another object in a computer-generated, 3D immersive environment. Interactions by the user with the computing device 1020 (e.g., rotating, shaking, touching a touchscreen, swiping a finger across a touch screen) can be interpreted as interactions with the object in the VR space. In the example of the laser pointer in a VR space, the computing device 1020 appears as a virtual laser pointer in the computer-generated, 3D environment. As the user manipulates the computing device 1020, the user in the VR space sees movement of the laser pointer. The user receives feedback from interactions with the computing device 1020 in the VR environment on the computing device 1020 or on the VR headset 1090.

**[0100]** In some implementations, a computing device 1020 may include a touchscreen. For example, a user can interact with the touchscreen in a particular manner that causes the system to mimic what happens on the touchscreen with what happens in the VR space. For example, a user may use a pinching-type motion to zoom content displayed on the touchscreen. This pinching-type motion on the touchscreen can cause information provided in the VR space to be zoomed. In another example, the computing device may be rendered as a virtual book in a computer-generated, 3D environment. In the VR space, the pages of the book can be displayed in the VR space and the swiping of a finger of the user across the touchscreen can be interpreted as turning/flipping a page of the virtual book. As each page is turned/flipped, in addition to seeing the page contents change, the user may be provided with audio feedback, such as the sound of the turning of a page in a book.

**[0101]** In some implementations, one or more input devices in addition to the computing device (e.g., a mouse, a keyboard) can be rendered in a n immersive environment such as a computer-generated, 3D environment. The rendered input devices (e.g., the rendered mouse, the rendered keyboard) can be used as rendered in the VR space to control

objects in the VR space.

**[0102]** Computing device 1000 is intended to represent various forms of digital computers and devices, including, but not limited to laptops, desktops, workstations, personal digital assistants, servers, blade servers, mainframes, and other appropriate computers. Computing device 1020 is intended to represent various forms of mobile devices, such as personal digital assistants, cellular telephones, smart phones, and other similar computing devices. The components shown here, their connections and relationships, and their functions, are meant to be exemplary only, and are not meant to limit implementations of the inventions described and/or claimed in this document.

**[0103]** A number of embodiments have been described. Nevertheless, it will be understood that various modifications may be made without departing from the spirit and scope of the specification.

**[0104]** In addition, the logic flows depicted in the figures do not require the particular order shown, or sequential order, to achieve desirable results. In addition, other steps may be provided, or steps may be eliminated, from the described flows, and other components may be added to, or removed from, the described systems. Accordingly, other embodiments are within the scope of the following claims.

**[0105]** While certain features of the described implementations have been illustrated as described herein, many modifications, substitutions, changes and equivalents will now occur to those skilled in the art. It is, therefore, to be understood that the appended claims are intended to cover all such modifications and changes as fall within the scope of the implementations. It should be understood that they have been presented by way of example only, not limitation, and various changes in form and details may be made. Any portion of the apparatus and/or methods described herein may be combined in any combination, except mutually exclusive combinations. The implementations described herein can include various combinations and/or sub-combinations of the functions, components and/or features of the different implementations described.

**[0106]** In the following some examples are given.

**[0107]** Example 1: A method for context-sensitive hand interaction with an immersive environment comprising: determining a contextual factor for a user; selecting an interaction mode based on the contextual factor; monitoring a hand of the user to determine a hand property; and determining an interaction with an immersive environment based on the interaction mode and the hand property.

**[0108]** Example 2: The method of example 1, wherein the monitoring the hand of the

user to determine a hand property includes monitoring the hand to determine a pointing direction of a finger of the user's hand.

**[0109]** Example 3: The method of example 2, wherein the interaction mode is a ray-based collision mode and the method further includes generating a virtual ray that extends in the pointing direction from the finger of the user.

**[0110]** Example 4: The method of any preceding example, wherein the monitoring the hand of the user to determine a hand property includes monitoring the hand to determine a position of a finger of the hand.

**[0111]** Example 5: The method of example 4, wherein the interaction mode is a point-based collision mode and the method further includes generating a collision point based on the position of the finger of the hand.

**[0112]** Example 6: The method of any preceding example, wherein the monitoring the hand of the user to determine a hand property includes monitoring the user's hand to determine a gesture performed by the hand.

**[0113]** Example 7: The method of any preceding example, wherein the monitoring the hand of the user to determine a hand property includes monitoring the user's hand to determine a movement of the hand from a first position to a second position.

**[0114]** Example 8: The method of any preceding example, wherein the contextual factor is a mode selection and the determining the contextual factor for the user includes: presenting a user interface that includes a plurality of user-selectable interface elements, each of the plurality of user-selecting interface elements being associated with a mode; receiving a selection from the user of one of the user-selectable interface elements; and determining the contextual factor for the user to be the mode associated with the selected user-selectable interface element.

**[0115]** Example 9: The method of any preceding example, wherein the contextual factor is a distance and the determining the contextual factor for the user includes: determining a position of the user's hand; mapping the position of the user's hand to a virtual hand position in the immersive environment; and determining a distance from the virtual hand position to a user-interface entity in the immersive environment.

**[0116]** Example 10: The method of example 9, wherein the selecting an interaction mode based on the contextual factor includes: comparing the distance to a threshold distance; responsive to determining that the distance is not greater than the threshold distance, selecting a ray-based collision mode as the interaction mode; and responsive to determining that the distance is greater than the threshold distance, selecting a point-based collision mode as the

interaction mode.

**[0117]** Example 11: The method of any preceding example, wherein the contextual factor is a focus position in the immersive environment and the determining the contextual factor for the user includes: tracking a gaze of the user; and determining the focus position based on the gaze.

**[0118]** Example 12: The method of example 11, wherein the selecting an interaction mode based on the contextual factor includes: identifying an entity in the immersive environment associated with the focus position; responsive to determining that the identified entity is a user-interface entity, selecting a ray-based collision mode as the interaction mode; and responsive to determining that the identified entity is not a user-interface entity, selecting a point-based collision mode as the interaction mode.

**[0119]** Example 13: The method of any preceding example, wherein determining a contextual factor for the user includes determining a pose or an arrangement of the user's hand.

**[0120]** Example 14: The method of any preceding example, wherein the immersive environment is an augmented reality environment.

**[0121]** Example 15: The method of any preceding example, wherein the immersive environment is a virtual reality environment.

**[0122]** Example 16: A system for context-sensitive hand interaction with an immersive environment comprising: at least one processor; and memory storing instructions that, when executed by the at least one processor, cause the system to: determine a contextual factor for a user; select an interaction mode based on the contextual factor; monitor the hand of the user hand to determine a hand property; and determine an interaction with an immersive environment based on the interaction mode and the hand property.

**[0123]** Example 17: The system of example 16, further including a hand tracker, wherein the instructions, when executed by the at least one processor, further cause the system to: determine a position of the hand using the hand tracker; map the position of the hand to a virtual hand position in the immersive environment; and determine a distance from the virtual hand position to a user-interface entity in the immersive environment.

**[0124]** Example 18: The system of example 17, wherein the instructions that cause the system to select an interaction mode based on the contextual factor include instructions that, when executed by the at least one processor, cause the system to: compare the distance to a threshold distance; responsive to determining that the distance is not greater than the threshold distance, select a ray-based collision mode as the interaction mode; and responsive

to determining that the distance is greater than the threshold distance, select a point-based collision mode as the interaction mode.

**[0125]** Example 19: The method of example 1, comprising: generating an immersive environment that includes a three-dimensional virtual canvas; wherein determining a contextual factor for a user comprises determining an orientation of a handheld electronic device held in a first hand of a user; wherein selecting an interaction mode based on the contextual factor comprises selecting based on the orientation of the handheld electronic device indicating the first hand is in a first state, a positional interaction mode; wherein monitoring a hand of the user to determine a hand property comprises using a hand tracker to monitor a position of a second hand of the user; wherein determining an interaction with an immersive environment based on the interaction mode and the hand property comprises mapping the position of the second hand to a virtual hand position in the immersive environment; the method further comprising: generating a visual entity in the three-dimensional virtual canvas based on the virtual hand position; monitoring for a change to the orientation of the handheld electronic device; and responsive to the orientation of the handheld device changing to a changed orientation: determining that the changed orientation indicates the first hand is in a second state; and responsive to determining that the first hand is in the second state, selecting a ray-based collision mode as the interaction mode and generating a virtual ray that extends in a pointing direction from the virtual hand position.

**[0126]** Example 20: The method of example 19, wherein the first state of the first hand is down by a side of the user.

**[0127]** Example 21: The method of example 19 or example 20, wherein the second state of the first hand is in front of the user.

**[0128]** Example 22: The method of any one of examples 19 to 21, further comprising determining an orientation of a finger of the user; and determining the pointing direction based on the orientation of the finger.

**[0129]** Example 23: A method comprising: presenting a user interface that includes a plurality of user-selectable interface elements, each of the plurality of user-selectable interface elements being associated with an interaction mode; receiving a selection from the user of one of the user-selectable interface elements; monitoring a user's hand to determine a hand property; and determining an interaction with an immersive environment based on the hand property and the interaction mode associated with the selected user-selectable interface element.

**[0130]** Example 24: A method comprising: generating an immersive environment;

using a hand tracker to determine a position of a user's hand; mapping the position of the user's hand to a virtual hand position in the immersive environment; determining a distance from the virtual hand position to a user-interface entity in the immersive environment comparing the distance to a threshold distance; responsive to determining that the distance is not greater than the threshold distance: determining a pointing direction of a finger of the user's hand; generating a virtual ray in the immersive environment that extends in the pointing direction from the virtual hand position; and actuating a user-interface element of the user-interface entity based on an intersection of the virtual ray with the user-interface element; responsive to determining that the distance is greater than the threshold distance: determining a position of the finger of the user's hand; and interacting with the immersive environment based on the position of the user's finger relative to the virtual hand position.

**[0131]** Example 25: A method comprising: generating an immersive environment; using an eye tracker to track a user's gaze; determining a focus position based on the user's gaze; identifying an entity in the immersive environment associated with the focus position; responsive to determining that the identified entity is a user-interface entity, selecting a ray-based collision mode and generating a virtual ray that extends in a pointing direction based on an orientation of a finger of the user's hand; and responsive to determining that the identified entity is not a user-interface entity, selecting a point-based collision mode and generating a collision point based on a position of the finger of the user's hand.

**[0132]** Example 26: A method comprising: generating an immersive environment that includes a three-dimensional virtual canvas; determining an orientation of a handheld electronic device held in a first hand of a user; based on the orientation of the handheld electronic device indicating the first hand is in a first state, selecting a positional interaction mode; using a hand tracker to determine a position of a second hand of the user; mapping the position of the second hand to a virtual hand position in the immersive environment; generating a visual entity in the three-dimensional virtual canvas based on the virtual hand position; monitoring for a change to the orientation of the handheld electronic device; and responsive to the orientation of the handheld device changing to a changed orientation: determining that the changed orientation indicates the first hand is in a second state; and responsive to determining that the first hand is in the second state, selecting a ray-based collision mode as the interaction mode and generating a virtual ray that extends in a pointing direction from the virtual hand position.

**[0133]** Example 27: The method of example 26, wherein the first state of the first hand is down by a side of the user.

**[0134]** Example 28: The method of example 26 or example 27, wherein the second state of the first hand is in front of the user.

**[0135]** Example 29: The method of any one of examples 26 to 28, further comprising determining an orientation of a finger of the user; and determining the pointing direction based on the orientation of the finger.

**WHAT IS CLAIMED IS:**

1. A method for context-sensitive hand interaction with an immersive environment comprising:
  - determining a contextual factor for a user;
  - selecting an interaction mode based on the contextual factor;
  - monitoring a hand of the user to determine a hand property; and
  - determining an interaction with an immersive environment based on the interaction mode and the hand property.
2. The method of claim 1, wherein the monitoring the hand of the user to determine a hand property includes monitoring the hand to determine a pointing direction of a finger of the user's hand.
3. The method of claim 2, wherein the interaction mode is a ray-based collision mode and the method further includes generating a virtual ray that extends in the pointing direction from the finger of the user.
4. The method of any preceding claim, wherein the monitoring the hand of the user to determine a hand property includes monitoring the hand to determine a position of a finger of the hand.
5. The method of claim 4, wherein the interaction mode is a point-based collision mode and the method further includes generating a collision point based on the position of the finger of the hand.
6. The method of any preceding claim, wherein the monitoring the hand of the user to determine a hand property includes monitoring the user's hand to determine a gesture performed by the hand.
7. The method of any preceding claim, wherein the monitoring the hand of the user to determine a hand property includes monitoring the user's hand to determine a movement of the hand from a first position to a second position.

8. The method of any preceding claim, wherein the contextual factor is a mode selection and the determining the contextual factor for the user includes:

presenting a user interface that includes a plurality of user-selectable interface elements, each of the plurality of user-selecting interface elements being associated with a mode;

receiving a selection from the user of one of the user-selectable interface elements;  
and

determining the contextual factor for the user to be the mode associated with the selected user-selectable interface element.

9. The method of any preceding claim, wherein the contextual factor is a distance and the determining the contextual factor for the user includes:

determining a position of the user's hand;

mapping the position of the user's hand to a virtual hand position in the immersive environment; and

determining a distance from the virtual hand position to a user-interface entity in the immersive environment.

10. The method of claim 9, wherein the selecting an interaction mode based on the contextual factor includes:

comparing the distance to a threshold distance;

responsive to determining that the distance is not greater than the threshold distance, selecting a ray-based collision mode as the interaction mode; and

responsive to determining that the distance is greater than the threshold distance, selecting a point-based collision mode as the interaction mode.

11. The method of any preceding claim, wherein the contextual factor is a focus position in the immersive environment and the determining the contextual factor for the user includes:

tracking a gaze of the user; and

determining the focus position based on the gaze.

12. The method of claim 11, wherein the selecting an interaction mode based on the contextual factor includes:

identifying an entity in the immersive environment associated with the focus position;  
responsive to determining that the identified entity is a user-interface entity, selecting  
a ray-based collision mode as the interaction mode; and

responsive to determining that the identified entity is not a user-interface entity,  
selecting a point-based collision mode as the interaction mode.

13. The method of any preceding claim, wherein determining a contextual factor for the user includes determining a pose of the user's hand.

14. The method of any preceding claim, wherein the immersive environment is an augmented reality environment.

15. The method of any preceding claim, wherein the immersive environment is a virtual reality environment.

16. A system for context-sensitive hand interaction with an immersive environment comprising:

at least one processor; and

memory storing instructions that, when executed by the at least one processor,

cause the system to:

determine a contextual factor for a user;

select an interaction mode based on the contextual factor;

monitor the hand of the user hand to determine a hand property; and

determine an interaction with an immersive environment based on the interaction mode and the hand property.

17. The system of claim 16, further including a hand tracker, wherein the instructions, when executed by the at least one processor, further cause the system to:

determine a position of the hand using the hand tracker;

map the position of the hand to a virtual hand position in the immersive environment;

and

determine a distance from the virtual hand position to a user-interface entity in the immersive environment.

18. The system of claim 17, wherein the instructions that cause the system to select an interaction mode based on the contextual factor include instructions that, when executed by the at least one processor, cause the system to:

- compare the distance to a threshold distance;
- responsive to determining that the distance is not greater than the threshold distance, select a ray-based collision mode as the interaction mode; and
- responsive to determining that the distance is greater than the threshold distance, select a point-based collision mode as the interaction mode.

19. A method according to claim 1 comprising:

- generating an immersive environment that includes a three-dimensional virtual canvas;

- wherein determining a contextual factor for a user comprises determining an orientation of a handheld electronic device held in a first hand of a user;

- wherein selecting an interaction mode based on the contextual factor comprises selecting based on the orientation of the handheld electronic device indicating the first hand is in a first state, a positional interaction mode;

- wherein monitoring a hand of the user to determine a hand property comprises using a hand tracker to monitor a position of a second hand of the user;

- wherein determining an interaction with an immersive environment based on the interaction mode and the hand property comprises mapping the position of the second hand to a virtual hand position in the immersive environment;

- the method further comprising:

- generating a visual entity in the three-dimensional virtual canvas based on the virtual hand position;

- monitoring for a change to the orientation of the handheld electronic device; and

- responsive to the orientation of the handheld device changing to a changed orientation:

- determining that the changed orientation indicates the first hand is in a second state; and

- responsive to determining that the first hand is in the second state, selecting a ray-based collision mode as the interaction mode and generating a virtual ray that extends in a pointing direction from the virtual hand position.

20. The method of claim 19, wherein the first state of the first hand is down by a side of the user.

21. The method of claim 19 or claim 20, wherein the second state of the first hand is in front of the user.

22. The method of any one of claims 19 to 21, further comprising determining an orientation of a finger of the user; and determining the pointing direction based on the orientation of the finger.

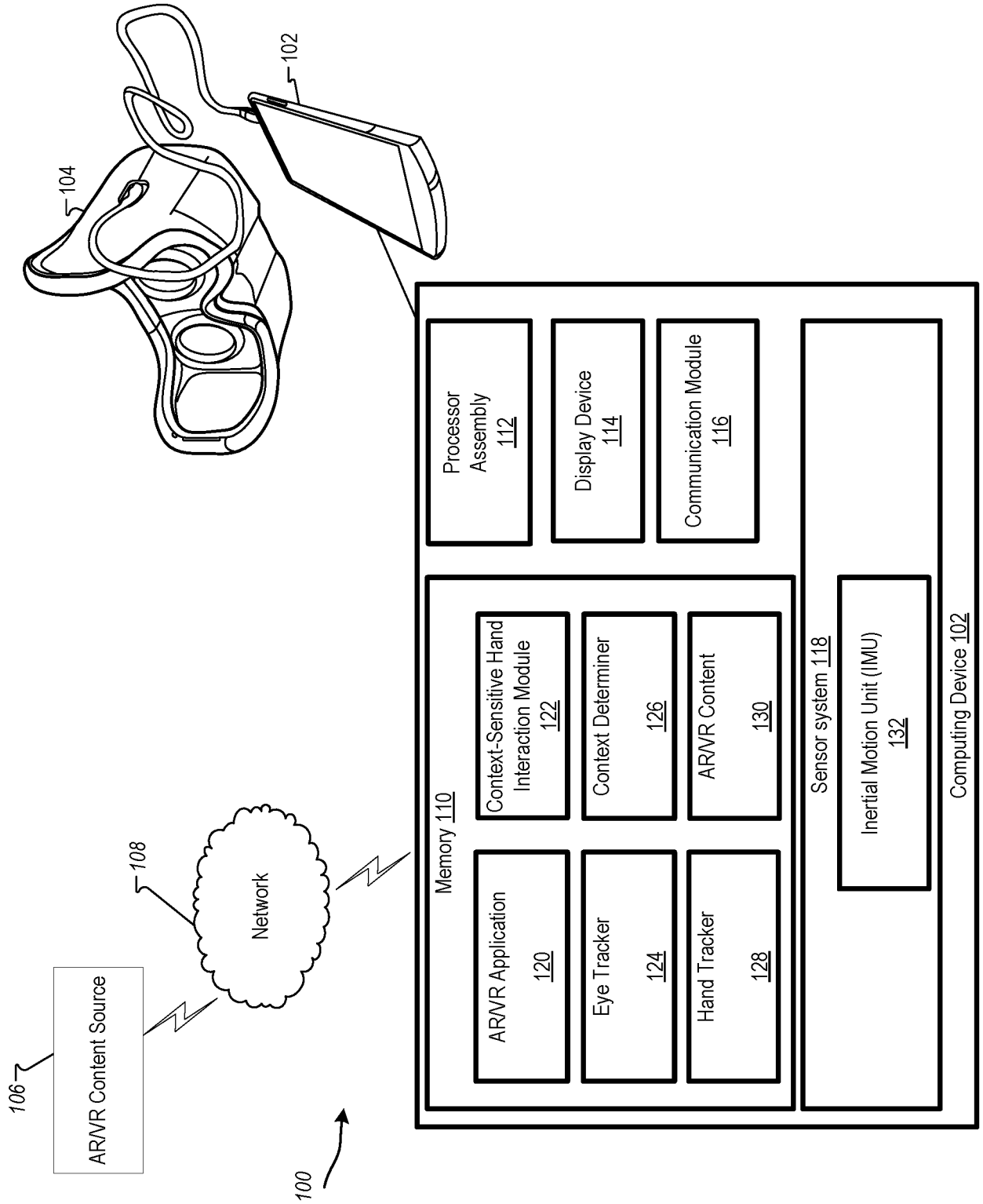


FIG. 1

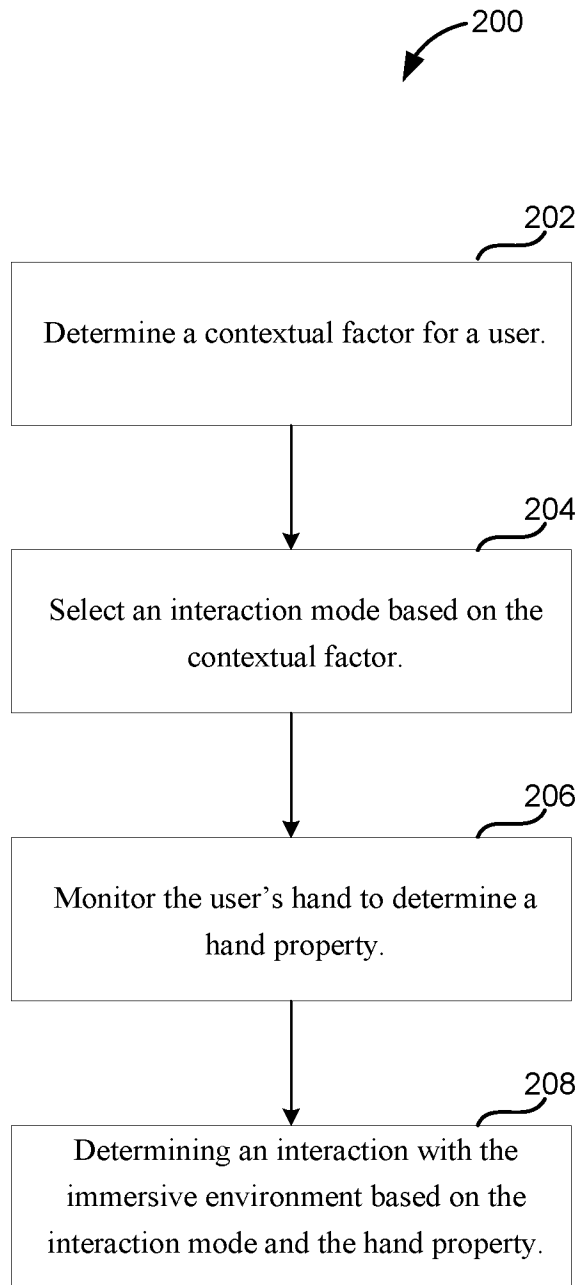


FIG. 2

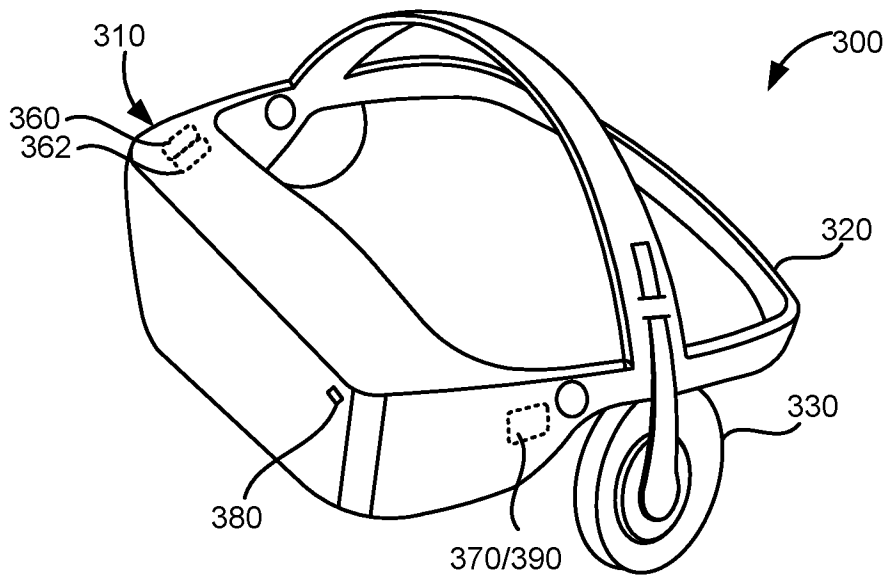


FIG. 3A

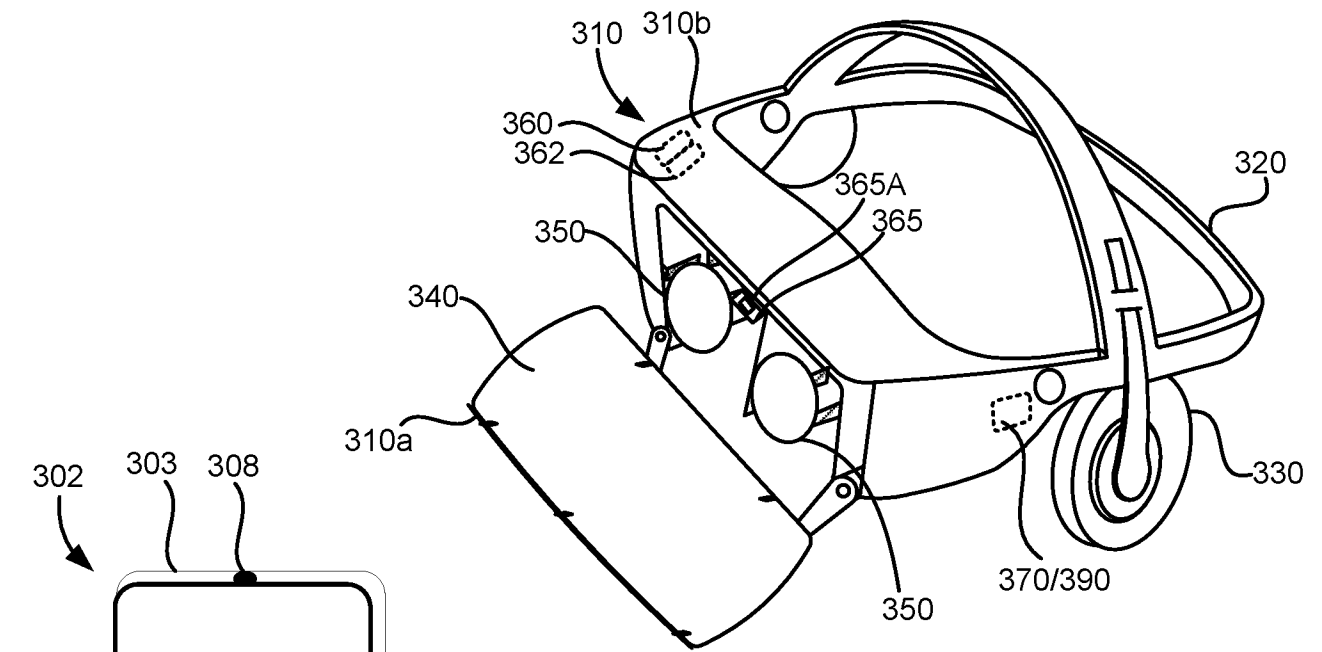


FIG. 3B

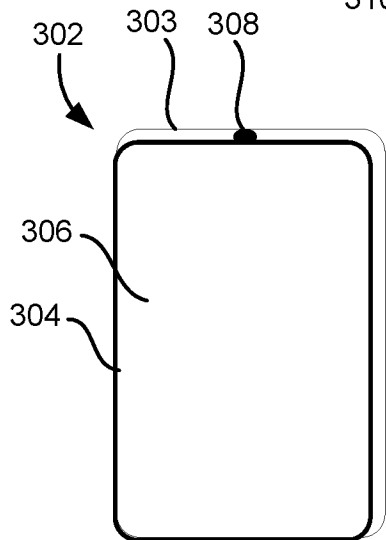


FIG. 3C

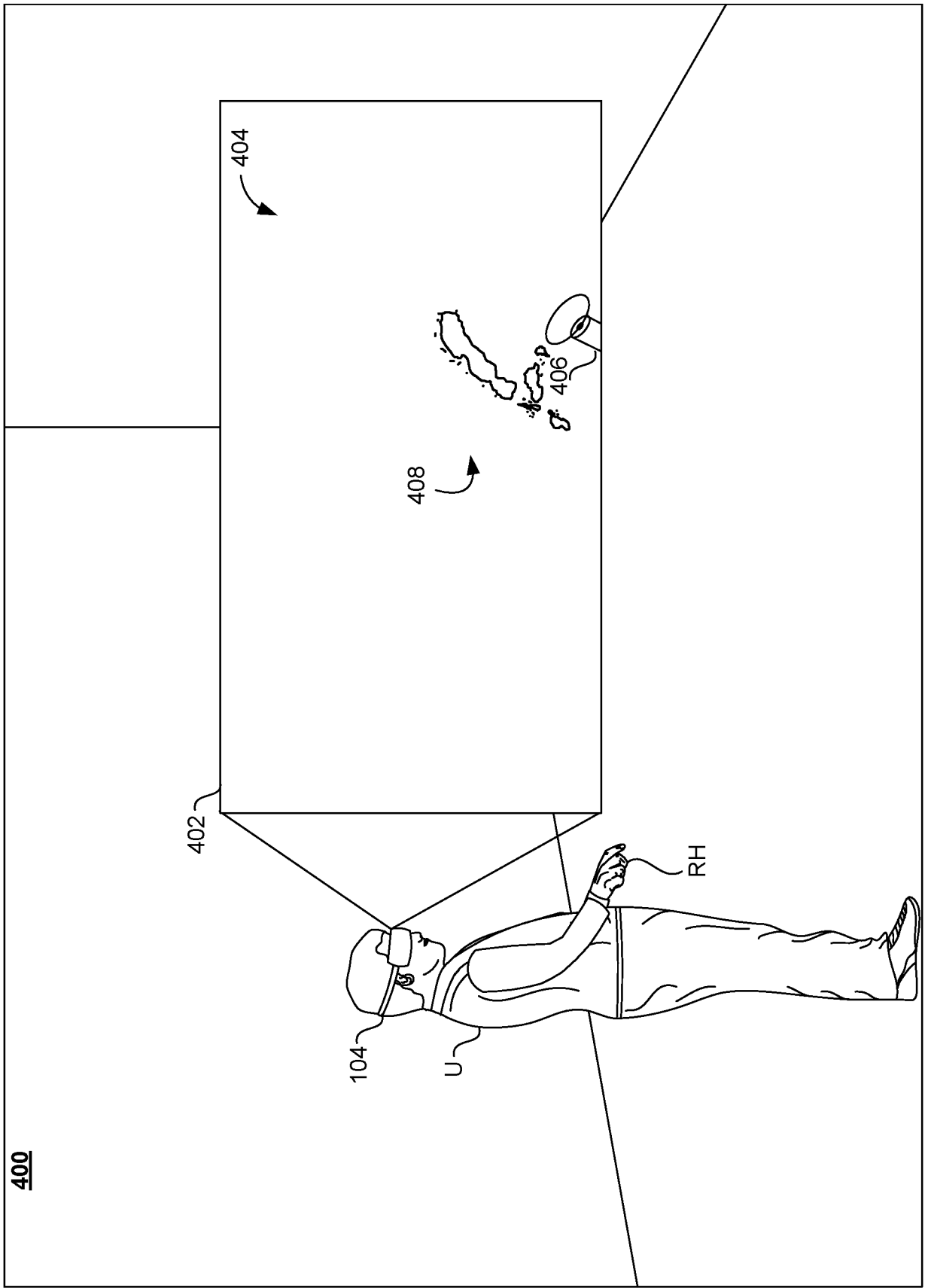


FIG. 4

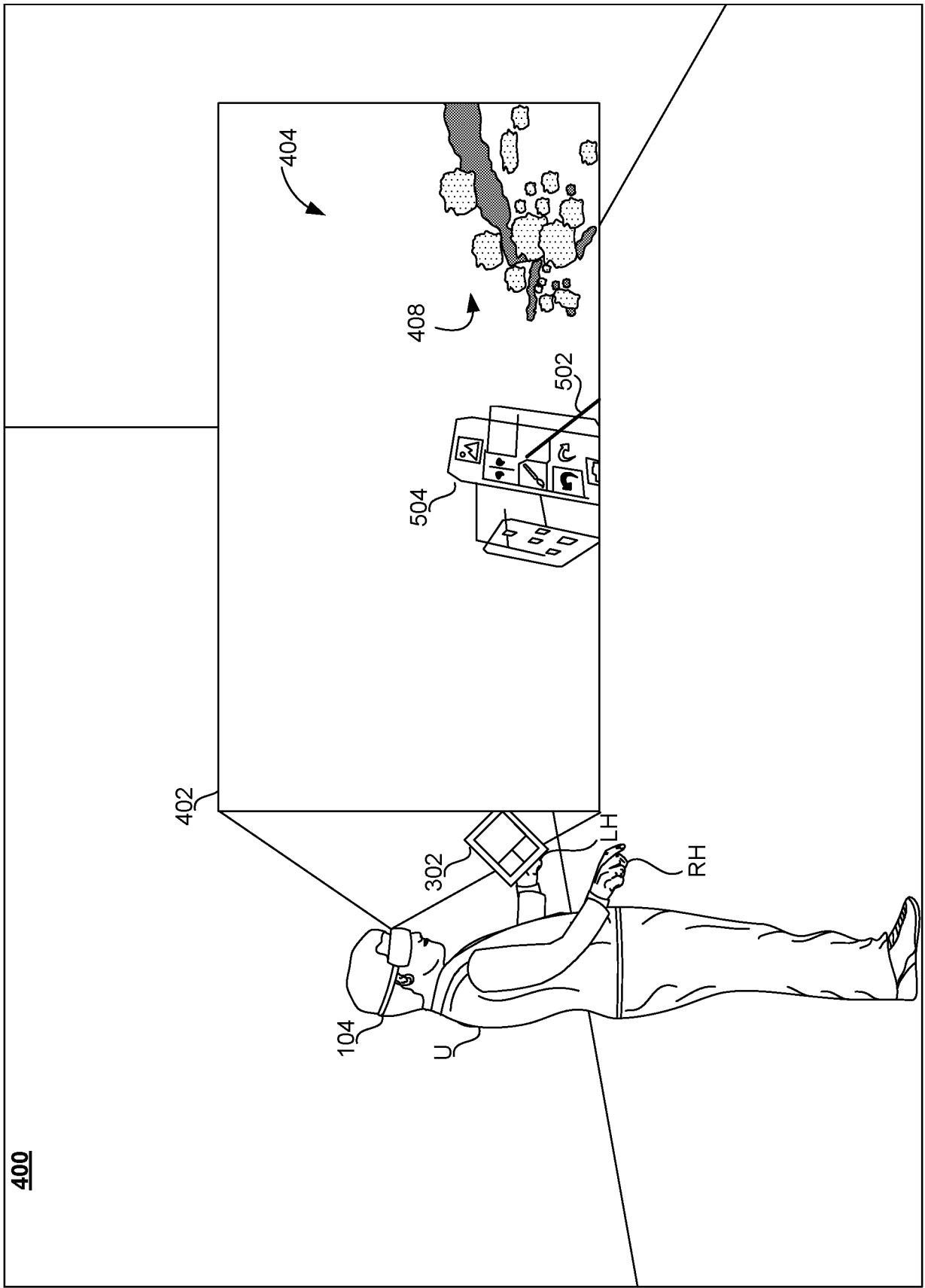


FIG. 5

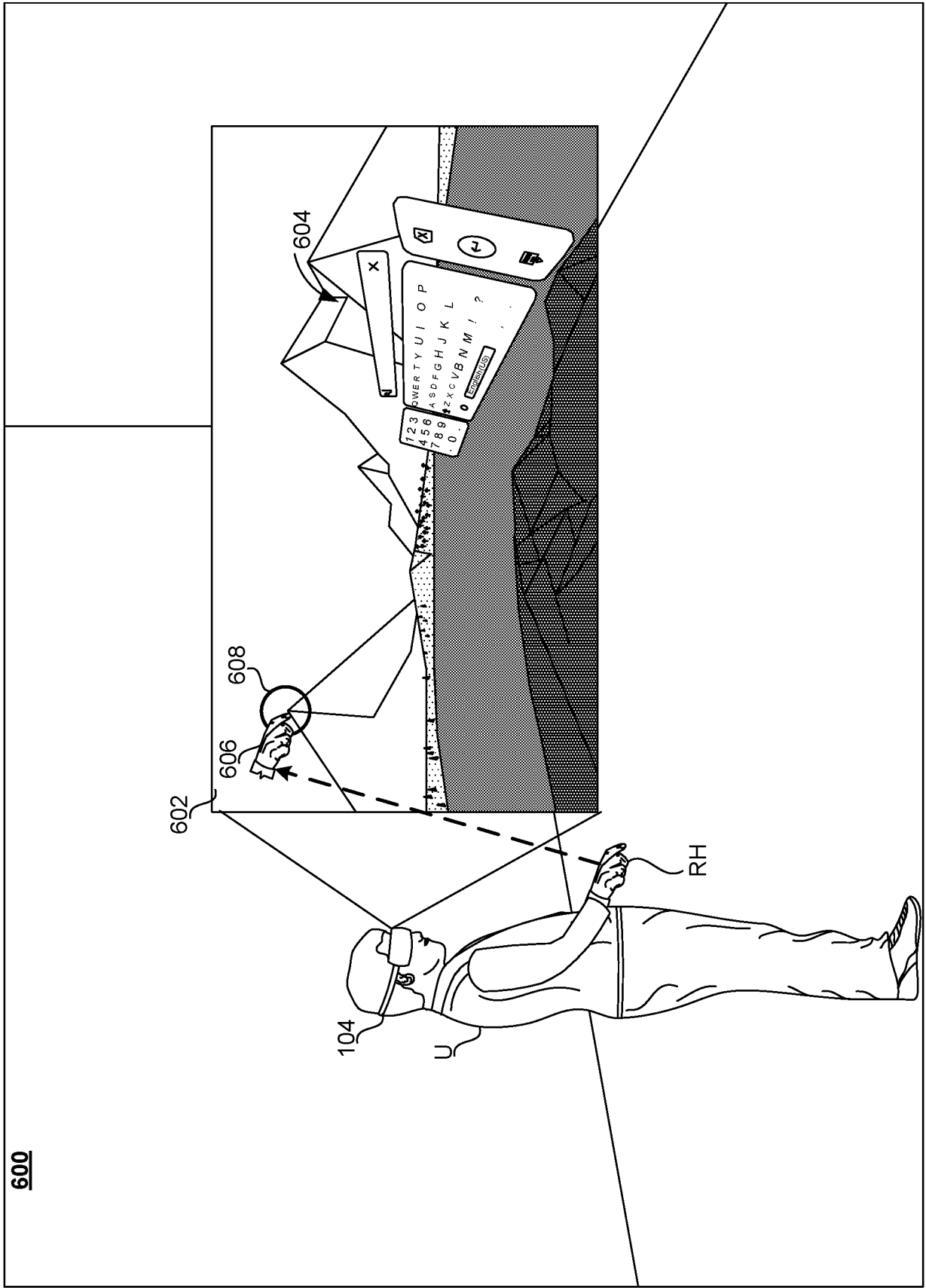
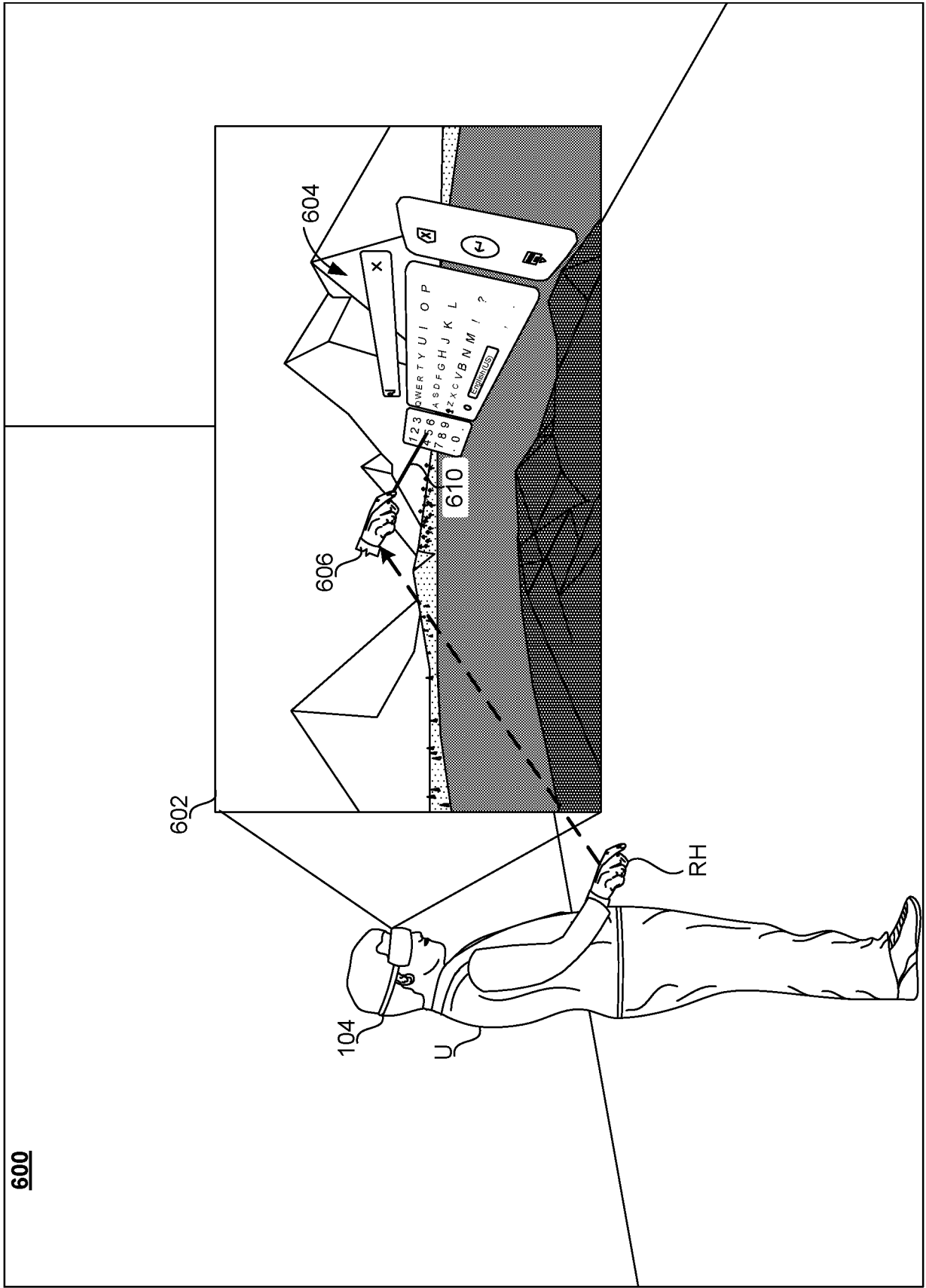


FIG. 6



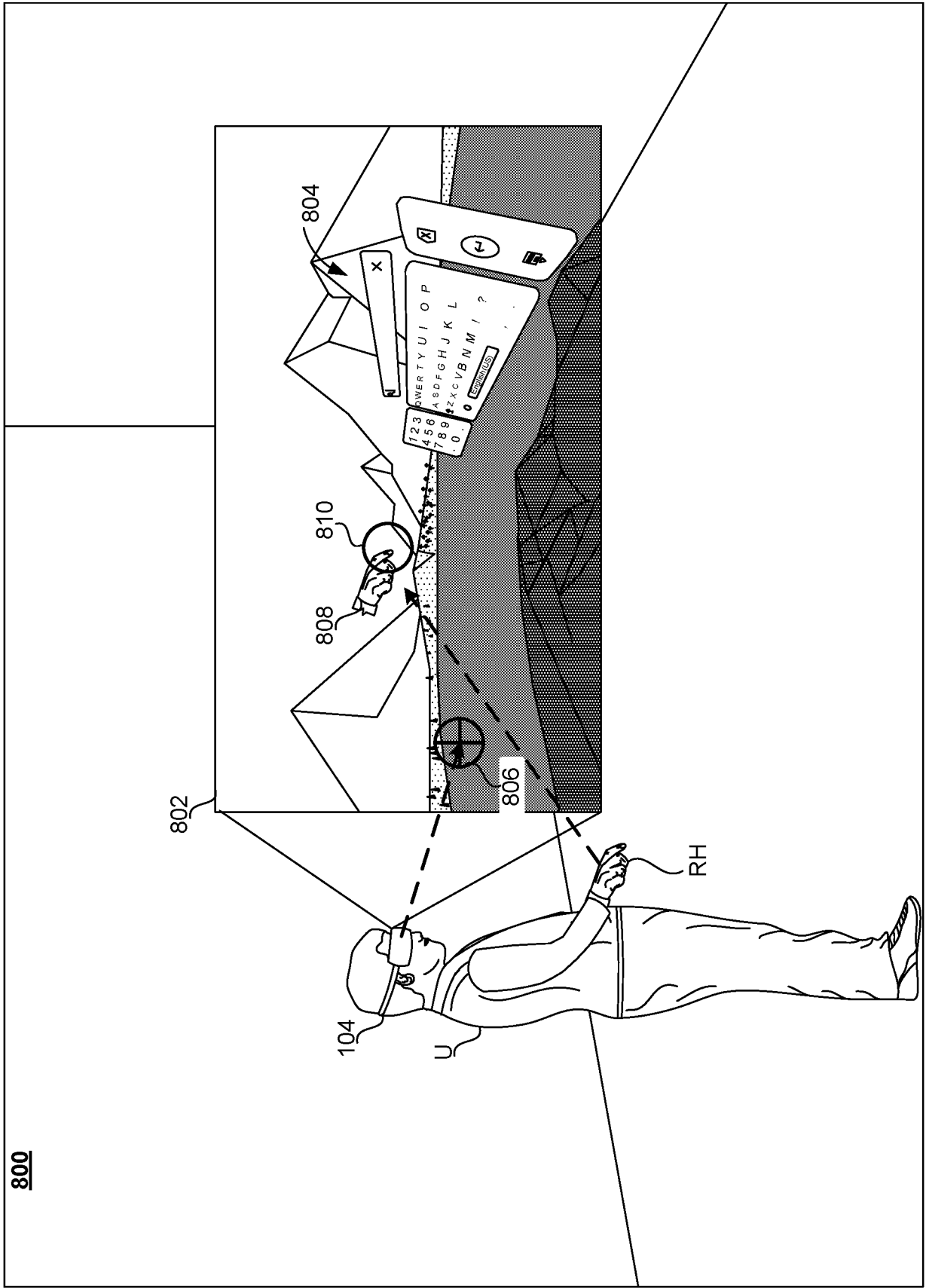
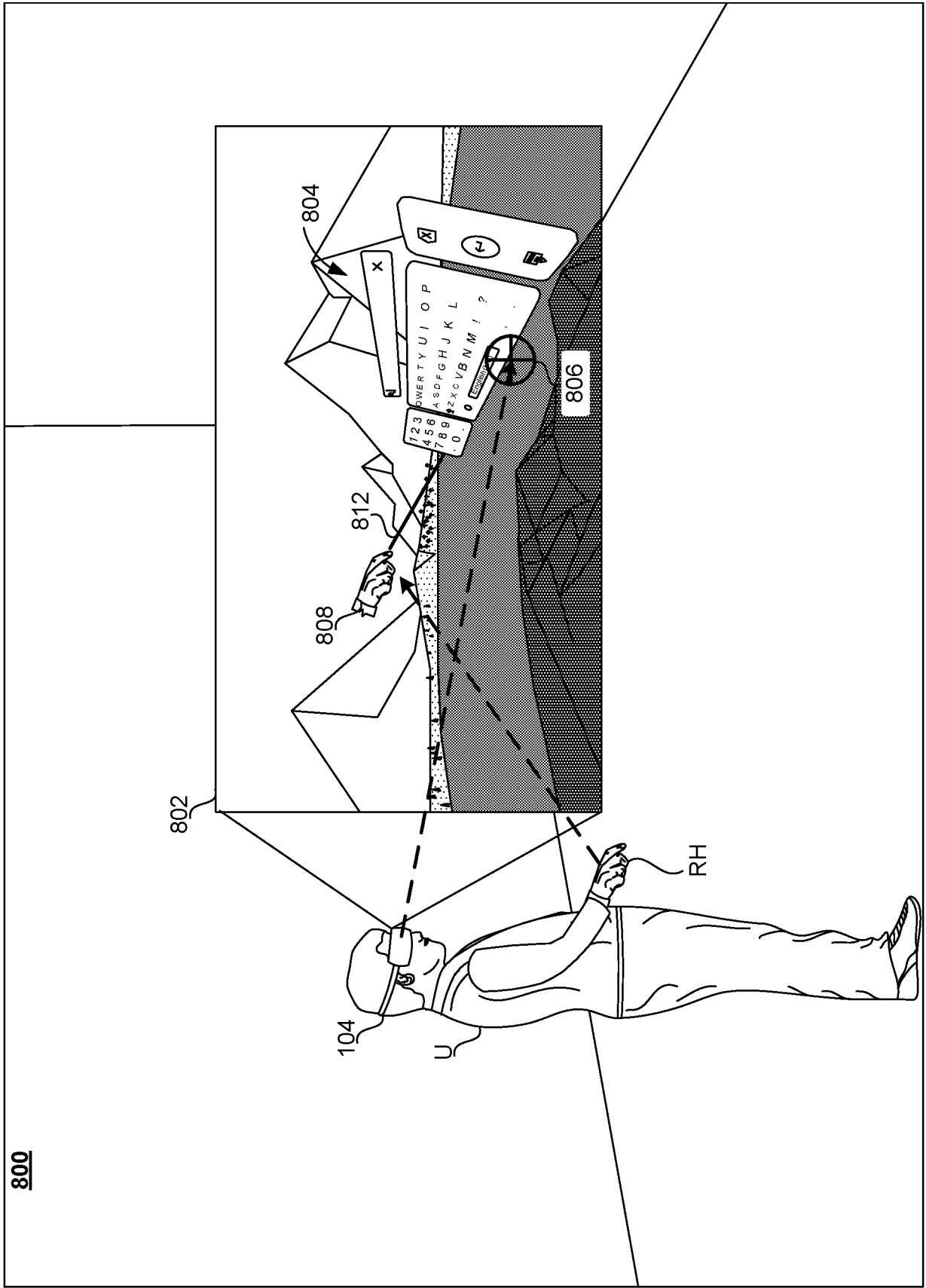


FIG. 8



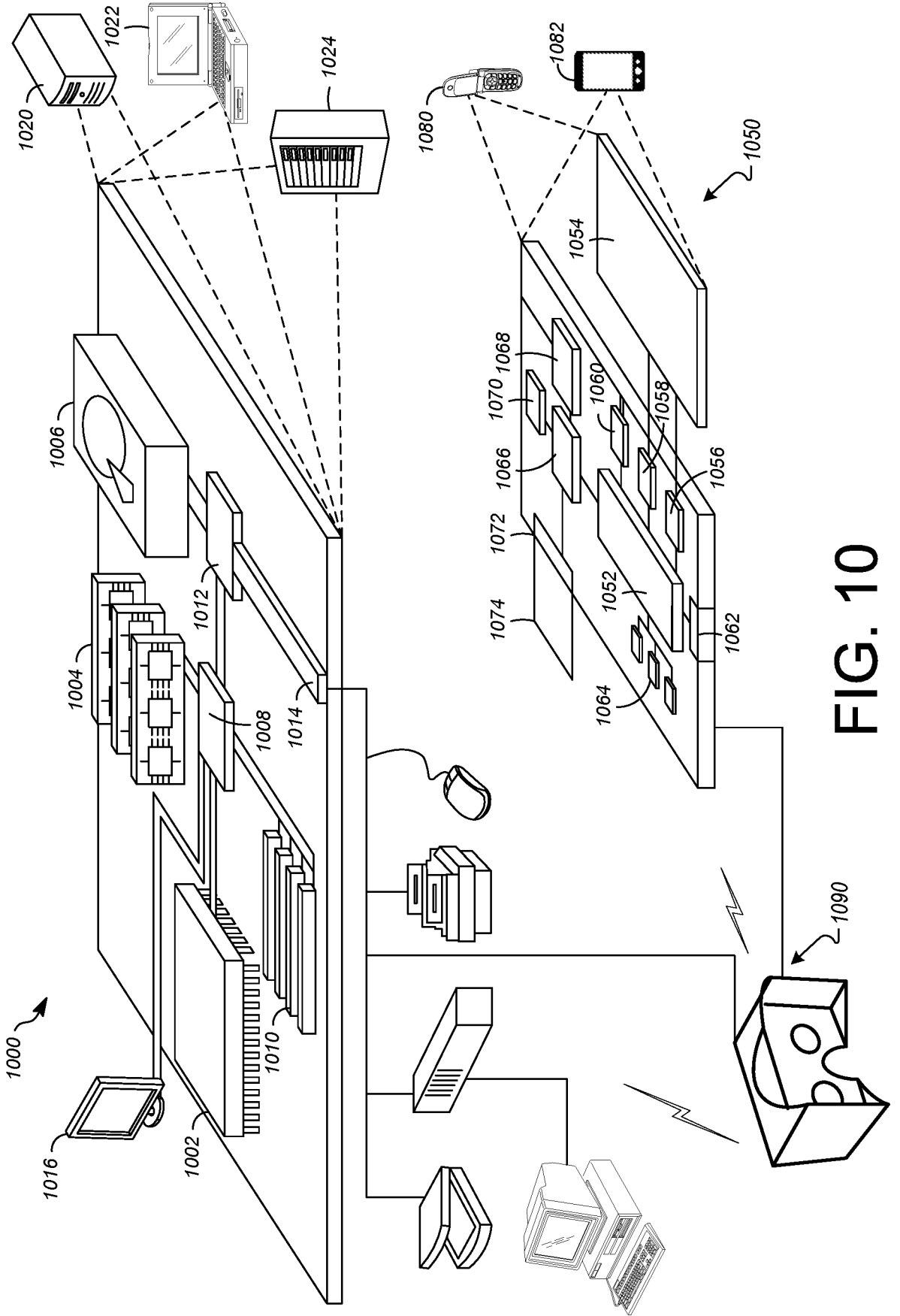


FIG. 10

# INTERNATIONAL SEARCH REPORT

International application No.  
PCT/US2018/046223

## Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1.  Claims Nos.:  
because they relate to subject matter not required to be searched by this Authority, namely:
  
2.  Claims Nos.:  
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
  
3.  Claims Nos.:  
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

## Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

1.  As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
  
2.  As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of additional fees.
  
3.  As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
  
4.  No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:  
  
1-7, 9, 13-18

### Remark on Protest

- The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- No protest accompanied the payment of additional search fees.

**INTERNATIONAL SEARCH REPORT**

International application No  
PCT/US2018/046223

**A. CLASSIFICATION OF SUBJECT MATTER**  
INV. G06F3/01  
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, WPI Data

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2017/024142 A1 (GOOGLE INC [US]) 9 February 2017 (2017-02-09) paragraph [0044] paragraph [0043] - paragraph [0047] paragraph [0047] -----	1,2,4-7, 9,13-18

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  
  
17 October 2018

Date of mailing of the international search report  
  
15/01/2019

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  
  
Fulcheri, Alessandro

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2018/046223

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2017024142 A1	09-02-2017	CN 107533373 A	02-01-2018
		EP 3332314 A1	13-06-2018
		JP 2018522310 A	09-08-2018
		KR 20170130586 A	28-11-2017
		US 2017038830 A1	09-02-2017
		WO 2017024142 A1	09-02-2017
-----			

**FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210**

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

## 1. claims: 1-7, 9, 13-18

A method for context-sensitive hand interaction with an immersive environment.

The interaction mode is a ray-based collision mode and the method further includes generating a virtual ray that extends in the pointing direction from the finger of the user.

---

## 2. claims: 8, 11, 12

A method for context-sensitive hand interaction with an immersive environment, comprising presenting a user interface that includes a plurality of user-selectable interface elements, each of the plurality of user-selectable interface elements being associated with a mode; receiving a selection from the user of one of the user-selectable interface elements; and determining the contextual factor for the user to be the mode associated with the selected user-selectable interface element.

---

## 3. claim: 10

A method for context-sensitive hand interaction with an immersive environment, wherein the selecting an interaction mode based on the contextual factor includes: comparing the distance to a threshold distance; responsive to determining that the distance is not greater than the threshold distance, selecting a ray-based collision mode as the interaction mode; and responsive to determining that the distance is greater than the threshold distance, selecting a point-based collision mode as the interaction mode.

---

## 4. claims: 19-22

A method for context sensitive hand interaction with an immersive environment, comprising generating an immersive environment that includes a three-dimensional virtual canvas; wherein determining a contextual factor for a user comprises determining an orientation of a handheld electronic device held in a first hand of a user; wherein selecting an interaction mode based on the contextual factor comprises selecting based on the orientation of the handheld electronic device indicating the first hand is in a first state, a positional interaction mode; wherein monitoring a hand of the user to determine a hand property comprises using a hand tracker to monitor a position of a second hand of the user; wherein determining an interaction with an

**FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210**

immersive environment based on the interaction mode and the hand property comprises mapping the position of the second hand to a virtual hand position in the immersive environment; the method further comprising: generating a visual entity in the three-dimensional virtual canvas based on the virtual hand position; monitoring for a change to the orientation of the handheld electronic device; and responsive to the orientation of the handheld device changing to a changed orientation: determining that the changed orientation indicates the first hand is in a second state; and responsive to determining that the first hand is in the second state, selecting a ray-based collision mode as the interaction mode and generating a virtual ray that extends in a pointing direction from the virtual hand position.

---