

## (51) International Patent Classification:

G06F 9/455 (2006.01) G06T 17/00 (2006.01)

## (21) International Application Number:

PCT/US2015/033441

## (22) International Filing Date:

31 May 2015 (31.05.2015)

## (25) Filing Language:

English

## (26) Publication Language:

English

(71) Applicant: HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).

(72) Inventors: KOGAN, Olga; Shabazi 26, 56100 Yehud (IL). SAYERS, Yaniv; Shabazi 26, 56100 Yehud (IL).

(74) Agents: ALLEN, Matthew R. et al.; Hewlett Packard Enterprise, 3404 E. Harmony Road, Mail Stop 79, Fort Collins, CO 80528 (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, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, 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, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

## Declarations under Rule 4.17:

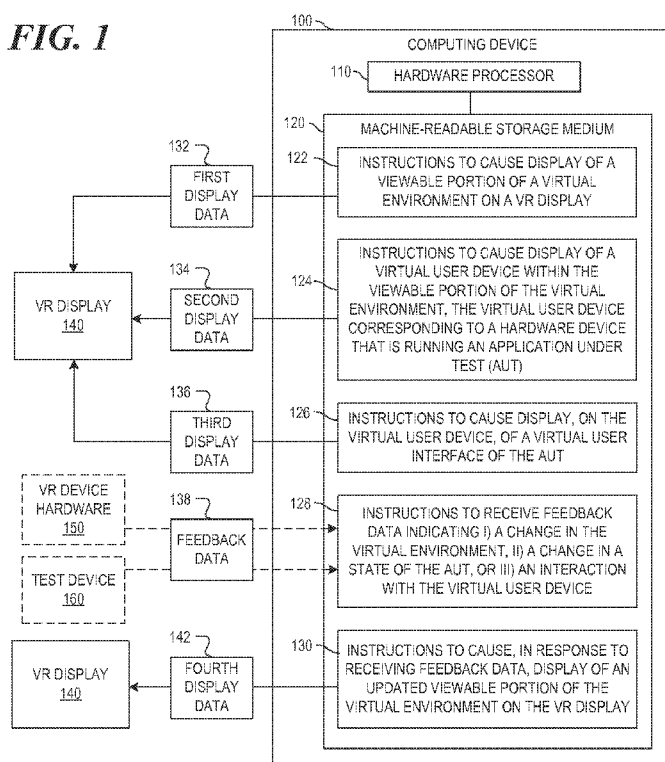
— as to the identity of the inventor (Rule 4.17(ii))

## Published:

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

## (54) Title: TESTING APPLICATIONS USING VIRTUAL REALITY

FIG. 1



(57) Abstract: Examples relate to testing applications using virtual reality. In one example, a computing device may: cause display of a viewable portion of a virtual environment on a VR display of the VR device; cause display of a virtual user device within the viewable portion of the virtual environment, the virtual user device corresponding to a hardware device that is running an application under test (AUT); cause display, on the virtual user device, of a virtual user interface of the AUT; receive feedback data indicating i) a change in the virtual environment, ii) a change in a state of the AUT, or iii) an interaction with the virtual user device; and in response to receiving feedback data, cause display of an updated viewable portion of the virtual environment on the VR display.

## TESTING APPLICATIONS USING VIRTUAL REALITY

### BACKGROUND

[0001] Applications are designed for use on many different types of computing devices, such as server computers, laptop computers, tablet computers, mobile phones, wearable computing devices, and embedded computing devices, such as those included in many consumer appliances and vehicles, to name a few. Applications are often tested during and after development, e.g., for the purposes of identifying errors and potential improvements.

### BRIEF DESCRIPTION OF THE DRAWINGS

[0002] The following detailed description references the drawings, wherein:

[0003] FIG. 1 is a block diagram of an example computing device for testing applications using virtual reality.

[0004] FIG. 2 is an example data flow for testing applications using virtual reality.

[0005] FIG. 3 is an illustration of an example virtual environment depicting the use of virtual reality to test an application.

[0006] FIG. 4 is a flowchart of an example method for testing applications using virtual reality.

### DETAILED DESCRIPTION

[0007] Applications for many different computing devices are often used to interact with the physical world. For example, a mobile phone application may be used to navigate through a city, a theme park, or a retail store; an automobile display application may be used to track and display a car's location, speed, fuel level, etc.; and an application running on a wearable computing device may make use of near-field communications (NFC) to interact with other nearby NFC

devices. To test user experience and obtain user feedback on these and other types of applications, virtual reality (VR) may be used to simulate, for a user, a physical world experience, without the need for real-world, or on location, testing.

[0008] Testing applications using virtual reality may have many advantages. For example, many different situations may be simulated, enabling the testing of user experience and feedback in a variety of situations, including situations that may only occur rarely in the physical world. Testing applications using VR may be safer, e.g., as in the case of testing an automobile heads-up display (HUD) application. VR testing may also make testing available to a wider audience, e.g., rather than needing to interact with the physical world, or a particular location within the physical world, testing may be performed in any location.

[0009] By way of example, a mobile phone application for navigating within a store, e.g., to find various products for purchase, may be tested using a VR system. A VR headset, worn by a user, may display a virtual store and a virtual phone to the user. The user interface of the application being tested, i.e., the application under test (AUT), may be displayed on the virtual phone. The user may test the application by interacting with the environment and/or the virtual phone. For example, in a situation where the AUT provides a map of the store and navigates the user to a particular product, the user may move around within the virtual store, observing the behavior of the AUT on the virtual phone. In this example, many aspects of the AUT may be tested, such as the accuracy of positional tracking, the accuracy of the destination with respect to the particular product, the usability of the AUT, interactions between the user and the virtual phone and/or the AUT, and the overall user experience.

[0010] The system for testing applications may be configured in a variety of ways, with functionality being spread across multiple devices or included in a single device. Further details regarding potential configurations, and for the testing applications using virtual reality, are described in further detail in the paragraphs that follow.

[0011] Referring now to the drawings, FIG. 1 is a block diagram of an example computing device 100 for testing applications using virtual reality. Computing

device 100 may be, for example, a server computer, a personal computer, a mobile computing device, a virtual reality device, or any other electronic device suitable for processing data. In the embodiment of FIG. 1, computing device 100 includes hardware processor 110 and machine-readable storage medium 120.

[0012] Hardware processor 110 may be one or more central processing units (CPUs), semiconductor-based microprocessors, and/or other hardware devices suitable for retrieval and execution of instructions stored in machine-readable storage medium 120. Hardware processor 110 may fetch, decode, and execute instructions, such as 122-130, to control the process for testing an application using virtual reality. As an alternative or in addition to retrieving and executing instructions, hardware processor 110 may include one or more electronic circuits that include electronic components for performing the functionality of one or more of instructions.

[0013] A machine-readable storage medium, such as 120, may be any electronic, magnetic, optical, or other physical storage device that contains or stores executable instructions. Thus, machine-readable storage medium 120 may be, for example, Random Access Memory (RAM), an Electrically Erasable Programmable Read-Only Memory (EEPROM), a storage device, an optical disc, and the like. In some implementations, storage medium 120 may be a non-transitory storage medium, where the term “non-transitory” does not encompass transitory propagating signals. As described in detail below, machine-readable storage medium 120 may be encoded with a series of executable instructions: 122-130, for testing applications using virtual reality.

[0014] A VR display 140 is in communication with the computing device 100, and is operable to display data for a user, such as a virtual environment, a virtual computing device, and a virtual representation of an application being tested. In implementations where the computing device 100 is included in a VR device, the VR display 140 may be the screen.

[0015] In some implementations, VR device hardware 150 may be in communication with the computing device 100, and is operable to provide feedback to the computing device 100. For example, the VR device hardware 150

may be a controller for controlling movement of the user within a virtual environment or sensors for tracking head movements and orientation. While VR device hardware 150 is represented in FIG. 1 by a single box, multiple and varying types of VR device hardware 150 may be used for providing feedback to the computing device 100.

[0016] In some implementations, test device 160 may be in communication with the computing device 100, and is operable to provide feedback to the computing device 100. For example, the test device 160 may be a computing device on which an AUT is running, and the feedback may data that comes from the AUT or from other applications running on the test device 160.

[0017] As shown in FIG. 1, the computing device 100 executes instructions (122) to cause display of a viewable portion of a virtual environment on a VR display 140, e.g., of a VR device. Using the example situation of an application being tested within a virtual store, the virtual environment may be a representation of the store using three dimensional graphics. The viewable portion may be the portion of the virtual environment within the user's field of view, e.g., the portion of the virtual store than the user wearing a VR device would see. The computing device 100 provides the VR display 140 with first display data 132 that causes the display of the viewable portion of the virtual environment.

[0018] The source of the first display data 132 may vary. For example, the computing device 100 may store pre-configured first display data 132 on an attached or remote machine-readable storage medium, such as storage medium 120. In some implementations, a separate virtual environment simulation module, running on the computing device 100 or a separate device, may provide the first display data 132 and/or data including a configuration of the virtual environment. For example, the developer of the AUT may provide the computing device with details of a virtual store environment, such as the placement and attributes of objects within the environment, and that information may be used by the computing device to produce the particular virtual environment.

[0019] The virtual environment is designed to simulate a real world environment, and may include a variety of objects and details to make the

simulation more realistic. For example, a virtual store environment may include various products on shelves, customers moving throughout the store, shopping carts, checkout lanes, store employees, and, in some implementations, may incorporate sounds, smells, and other sensory aspects capable of being simulated by technology in communication with the VR device. The virtual environment may also include virtual representations of objects that interact with the AUT. For example, wireless beacons, such as Wi-Fi and/or Bluetooth devices placed throughout the virtual store, may be included in the virtual environment for tracking the position of a user of the AUT. Other examples devices may include other devices running the AUT or other applications, point-of-sale devices and applications, smart tags capable of being scanned by a device running the AUT, and other types of objects capable of interacting with the AUT.

[0020] The virtual environment may include additional aspects that affect the AUT and/or the virtual device on which the AUT runs. For example, wireless beacons included in the virtual environment may have simulated signal patterns and simulated strength. Other objects within the environment, such as store shelving units, may have simulated interference attributes to simulate signal interference, e.g., between wireless beacons and the virtual device running the AUT. The additional aspects may vary greatly, and may be designed to make the virtual environment as realistic as possible or to simulate specific situations to be tested.

[0021] The computing device 100 executes instructions (124) to cause display of a virtual user device within the viewable portion of the virtual environment. The virtual user device corresponds to a hardware device that is running an application under test (AUT). For example, when testing a mobile phone application, the instructions 124 may send second display data 134 to the VR display, the second display data 124 including data operable to cause the VR display to display, within the viewable portion of the virtual environment, a virtual phone. The virtual user device to be displayed may vary, e.g., depending on the type of device on which the AUT is to be tested. In some implementations, the second display data 134, or data used by the computing device 100 to generate the second display data, may be provided by the hardware device that is running the AUT.

[0022] The computing device 100 executes instructions (126) to cause display, on the virtual user device, of a virtual user interface of the AUT. In the example situation where the AUT is a mobile phone application for navigating through a retail store, the third display data 136 may cause display of a map of the store, including a marker for the position of the user within the store and a destination where a particular product is located. The user interface is displayed on the virtual user device, e.g., on the virtual mobile phone.

[0023] In some implementations, the third display data 136 is provided to the computing device 100 by the hardware device running the AUT. For example, a mobile phone running the AUT may “cast” its screen to the computing device 100, which may in turn cause the VR display 140 to depict the screen cast, e.g., enabling the virtual user interface shown on the VR display to match the user interface displayed on the actual mobile phone.

[0024] The computing device 100 executes instructions (128) to receive feedback data 138 indicating i) a change in the virtual environment, ii) a change in a state of the AUT, or iii) an interaction with the virtual user device. As shown in FIG. 1, feedback data 138 may be provided by a VR hardware device 150 and/or a test device 160. In some implementations, the VR hardware device 150 may be, for example, a controller used to move the user within the virtual environment or a combination of sensors used to determine the orientation of the user’s head. In some implementations, the test device 160 may be a computing device on which the AUT is running, such as a mobile phone or a test computing running the AUT.

[0025] Feedback that indicates a change in the virtual environment may be, for example, the addition, removal, or change of an object within the viewable or non-viewable portion of the virtual environment, including any change in the virtual representation of the user. In the retail store example, this may include the addition or removal of shelving units, other virtual shoppers and/or employees, adding or removing wireless beacons or changing their signal strength, and adding or removing point-of-sale devices with which the AUT may interact, to name a few. Another example of feedback that indicates a change in the virtual environment

includes a change in a position, within the virtual environment, of a virtual user of the VR device or a change in a view orientation of a virtual user of the VR device. These changes may be tracked, for example, by the VR device itself and used, for example, to determine location based accuracy of the AUT and for testing movement-based aspects of an AUT.

[0026] Feedback that indicates a change in the state of the AUT may be, for example, changes that occur in the AUT without a change in the environment or user interaction, such as timed releases of information or changes to AUT settings or preferences. Using the retail store example, this may include periodically pushing a coupon or advertisement to the AUT for viewing by the user. Feedback that indicates an interaction with the virtual user device may be, for example, data sent to the virtual user device by another virtual object or an interaction with the virtual user device, or real user device running the AUT, by user input. Using the retail store example, this may include pushing a coupon or advertisement to the AUT for display when the user is within a certain range – e.g., determined by signal strength – of a beacon, or a button press or gesture or spoken instructions provided by the user of the virtual user device, e.g., detected by user input provided to the actual hardware device that corresponds to the virtual user device. Many other types of feedback data 138 may be provided to the computing device 100 instead of or in addition to the examples described above.

[0027] The computing device 100 executes instructions (130) to cause, in response to receiving the feedback data 138, display of an updated viewable portion of the virtual environment on the VR display 140. The fourth display data 142 provided to the VR display 140 may be, for example, data that causes a change to the user interface of the virtual user device depicted in the example environment. In the retail store example, in a situation where feedback data 138 indicates the position of the virtual user is within a certain range of a particular beacon, the fourth display data 142 may cause a coupon to be displayed on the virtual user device, e.g., a coupon for a product located near the user's virtual location within the virtual environment.

[0028] The computing device 100, using the foregoing instructions, is designed to produce a virtual experience that closely simulates a real-world experience for a user of the VR device, which enables testing of the AUT in conditions that resemble those that may be encountered by a user in the real world. In the virtual store example, a tester may determine how various things affect the user experience. For example, by tracking the user's gaze, testers may be able to determine if pushing a coupon to the virtual user device causes the user to a) look at the virtual user device, and/or b) find the product associated with the coupon. A distance threshold from a wireless beacon may be adjusted to help testers identify a threshold designed to maximize the chance that a user will find the product associated with the coupon. The speed with which a user moves throughout the virtual environment – measured, for example, by the VR device – may also have observable value to a tester, e.g., in the retail store example, a user may be moving too quickly for beacons to provide timely data, which may result in AUT developers implementing a speed threshold for determining when a coupon is eligible to be served to a nearby user. In addition, testers may be able to determine how signal interference and/or degradation affects the user experience. For example, if there are portions of the virtual store where signal is weak, testers may be able to determine if users are able to find what they are looking for or follow navigation instructions accurately. Testers may add, remove, or change the position of wireless beacons used for navigation throughout the virtual environment and determine how various configurations affect the user experience. In addition, interactions with other virtual devices, such as virtual point-of-sale devices, smart tags on store shelving units or products, or other virtual user devices running the AUT, may all be tested in the simulated real-world environment.

[0029] While the computing device 100 implements one possible configuration of a device for using virtual reality to test applications, further examples and details regarding the use of virtual reality in application testing are provided in the paragraphs that follow.

[0030] FIG. 2 is an example data flow 200 for testing applications using virtual reality. The data flow 200 depicts a testing device 210, which may be

implemented in a computing device, such as the computing device 100 described above with respect to FIG. 1. In some implementations, the testing device 210 is the hardware device that corresponds to the virtual hardware device, e.g., the device on which the AUT 220 is to be tested. In some implementations, the testing device 210 emulates or simulates the hardware device that corresponds to the virtual hardware device, e.g., a computer may run the AUT 220, alone or in addition to other testing tools, on an emulator that emulates a hardware device. The AUT 220 may be running on the testing device 210 and/or on a separate device in communication with the testing device 210. For example, the testing device 210 may be a computing device running testing tools while the AUT 220 is running on a hardware device in communication with the testing device 210.

[0031] The VR device 230 is in communication with the testing device 210, and is the device responsible for displaying the virtual environment 232 to a user. For example, the VR device 230 may be a virtual reality headset, which may include, among other things, a display screen and/or speakers. The VR device 230 is optionally in communication with one or more control devices, such as control device 235, for providing input to the VR device 230. For example, a user may remain stationary in the real world and use a joystick controller to move the virtual representation of the user within the virtual environment 232. Other forms of controlling the VR device 230 may also be used and may include, for example, sensors for detecting movement and/or orientation of the user's head, buttons, a touchpad, and/or a microphone for receiving voice commands, to name a few.

[0032] The virtual environment simulation module 240 is in communication with the testing device 210 and a virtual environment data 242 storage device. The virtual environment simulation module 240 may be used to provide the test device 210 with data to be represented in the virtual environment 232. In the retail store application testing example, the data may include the layout of the virtual store, the placement of all objects – including shelves, wireless beacons, other people, etc. – within the store, and attributes of those objects. The data may also include the virtual assets, e.g., the graphic components and art required to produce the virtual environment on the VR device. Movements of virtual people within the store, sounds that occur in the store, and other sensations that can be simulated

may also be included in the virtual environment data 242 for use in simulating a real-world environment. The virtual environment simulation module 240 may be implemented, in whole or in part, in the testing device 210 and/or a separate computing device. In some implementations, the virtual environment simulation module 240 is included in or in communication with the VR device 230.

[0033] While the testing device 210, AUT 220, VR device 230, virtual environment simulation module 240, and virtual environment data 242 are all depicted separately, multiple configurations are possible. And indicated by box 250, each of the foregoing components may be implemented in single device.

[0034] In the example data flow 200, when the AUT is to be tested, the test device 210 provides virtual environment (VE) data 212 to the VR device 230. The virtual environment data 212 specifies the virtual environment 232 in which the AUT 220 is to be tested. As discussed above, in the virtual retail store example, this includes data specifying the details of the virtual store layout and objects within the virtual store.

[0035] In the example data flow 200, the testing device 210 also provides virtual computing device (VCD) data 214 to the VR device 230. The virtual computing device data 214 specifies the virtual computing device 234 on which the AUT 220 is to be tested. The virtual computing device 234 corresponds to the computing device, e.g., the actual mobile phone on which the AUT 220 is to be run and/or tested. For example, VCD data 214 for a mobile phone may include graphical data required to produce a virtual version of the mobile phone and, in some implementations, features of the mobile phone to be used during testing, e.g., an earpiece if one is being used, a current state of the mobile phone including other applications running on it, and settings – such as phone volume level and/or screen brightness level.

[0036] The test device 210 also provides virtual user interface (VUI) data 216 to the VR device 230. The virtual user interface data 216 specifies data to be displayed, by the VR device 230, on the virtual display 236 of the virtual computing device 234. For example, the VUI data 216 may include the user interface of the AUT 220 for reproduction on the virtual mobile phone display 236. As discussed

above, the VUI data may be provided by screen casting from the testing device 210 or, in implementations where the testing device is separate from the hardware device on which the AUT is being simulated, the separate hardware device, e.g., an actual mobile phone running the AUT.

[0037] The virtual environment 232 depicted in the example data flow 200 depicts a scene that places a user of the VR device 230 in a virtual store near shelving units. The virtual environment 232 includes a beacon device 238, which may be a Bluetooth enabled beacon that, in the example AUT 200, is designed to push a coupon to display when the virtual user device 234 is within range of the beacon device 238.

[0038] The VR device 230 provides feedback data 218 to the testing device 210. The feedback data 218 indicates i) a change in position, within the virtual environment 232, of the virtual user of the VR device 230, or ii) a change in a view orientation of a virtual user of the VR device 230. For example, if the user either causes the virtual user to move within the virtual environment 232 or looks around within the virtual environment 232, feedback data 218 indicating the change is sent to the testing device 210.

[0039] The testing device 210 provides the virtual environment simulation module 240 with sensory data 222 that is based on the feedback data 218. The sensory data indicates the position and/or orientation of the virtual user of the VR device 230. In some implementations, the virtual environment simulation module 240, alone or in conjunction with the testing device 210, may use the sensory data 222 to determine a new state of the virtual computing device 234. For example, when if the virtual user moved closer to the beacon 238, the virtual environment simulation module 240 may determine a simulated signal strength level for wireless signal received by the virtual user device 234.

[0040] The virtual environment simulation module 240 provides the testing device 210 with computing device state data 224 that indicates the change in the simulated state of the computing device being simulated. The testing device 210 may use the computing device state data 224 and the AUT 220 to obtain updated VUI data. For example, in a situation where the AUT 220 is configured to display

a coupon when the virtual computing device 234 is within a certain signal strength, when the computing device state change data 224 indicates that the virtual computing device 234 is within the certain signal strength level, the AUT 220 may provide updated VUI data that includes a graphical representation of a coupon for display on the virtual display 236.

[0041] The updated VUI data is provided to the VR device 230, which causes the virtual display 236 of the virtual user device 234 to be updated. For example, the coupon may be displayed. Other example use cases in the retail store example may include: prompting the user to interact with the beacon 238 using a NFC connection to obtain a coupon, using the beacon 238 and other beacons to determine the user's location within the virtual store and provide a map indicating the location for display, permitting the user to use the virtual computing device 234 to scan a smart tag, e.g., by taking a picture of a virtual representation of a smart tag that causes the AUT to display more information about a particular product, and facilitating a virtual interaction between the virtual computing device 234 and another virtual device, such as a point-of-sale device.

[0042] In some implementations, the testing device 210 may obtain data indicating an interaction with the virtual user device 234. For example, if a user interacts with the virtual user device 234 – e.g., within the virtual environment 232 or in the real world with a real computing device that is screen casting to the virtual computing device, the VR device 230 or computing device running the AUT – whichever was used to interact – may provide data indicating the interaction to the testing device 210. In this situation, the testing device 210 may again obtain updated AUT state data that is based on the interaction. For example, if a user interacts with the AUT by using voice commands to search for a product within the virtual store, the AUT may change state by displaying a search or navigational interface. The testing device 210 may then provide data to the VR device 230 that causes display of the updated user interface within the virtual environment 232.

[0043] In some implementations, an update to the state of the AUT and virtual environment display may be caused by a change in the virtual environment 232. For example, if a tester changes the configuration of the virtual environment or

objects within it, this may affect the state of the AUT, which may cause another update to the AUT display. In some implementations, the AUT may cause a change in the state of the AUT. For example, in situations where the AUT uses a time-based release of information, the AUT may determine to update the display based on its configuration. As with the examples above, in this situation the testing device 210 will also cause the VR device 230 to update the virtual display 236 of the virtual computing device 234.

[0044] As discussed above, many aspects of the user experience with the AUT may be tested when using virtual reality. For example, the range at which certain events are triggered may affect the user's ability to find a particular beacon or product; audible noise within the virtual environment may affect the user's ability to hear any audible notifications the AUT causes the virtual computing device 234 to produce; and inaccurate positional tracking may affect how a user interacts with the AUT or moved around the virtual environment 232. Many other aspects of an AUT may be tested in a wide variety of environments.

[0045] FIG. 3 is an illustration of an example virtual environment 310 depicting the use of virtual reality to test an application. The application being tested, e.g., the AUT, is a heads-up-display (HUD) for a vehicle. The AUT causes a semi-transparent HUD to display various information about the vehicle, such as the travelling speed, fuel level, and navigational information. Testing the AUT using virtual reality may, in this example, be safer for a user than testing the AUT would be in the real world, and many conditions – such as weather, obstructions, other vehicles, and distractions – may be included in the virtual environment 310 for testing situations that might not be encountered often in the real world.

[0046] The example virtual environment 310, depicts a cellular tower 330 and a satellite 340, which may be in communication with the virtual computing device causing display of the HUD, such as a computer included in the automobile. Many things may affect the signal between the cellular tower 330, satellite 340, other cellular towers and satellites that are not depicted, and the virtual computing device. As noted above, weather may affect signal, e.g., GPS signal sent from a GPS; obstructions, such as bridges and buildings, may also affect signal. Many

other aspects, aside from how signals affect vehicle navigation, may be tested, such as the user experience with respect to various notifications or other information that is displayed on the HUD. Eye tracking technology may be used to determine where user's direct their attention, allowing developers of the AUT to determine, for example, which notifications are most useful to users without being disruptive or distracting. Facial expression tracking and emotion measurement technology may be implemented to determine how a user reacts or feels in different situations.

[0047] Many other applications in many different settings and on many different devices may be tested using virtual reality. For example, a smart watch application may be used to navigate through a theme park and receive location-based content, testing of which may be performed using virtual reality. As another example, a medical application designed to run on a tablet computer and assist with medical procedures may be tested using virtual reality, allowing it to be tested on a virtual patient. In addition, the flexibility of virtual reality configurations may allow for a single VR testing system to be used to simulate and test a variety of different applications on a variety of different devices.

[0048] FIG. 4 is a flowchart of an example method 400 for testing applications using virtual reality. The method may be implemented by a computing device, such as computing device 100 described above with reference to FIG. 1.

[0049] Virtual environment data is provided to a virtual reality (VR) device, the virtual environment data specifying a virtual environment in which an application under test (AUT) is to be tested (402). In the vehicle HUD example, the virtual environment may specify the virtual environment as the inside of an automobile on a road or in a parking lot.

[0050] Virtual computing device data is provided to the VR device, the virtual computing device data specifying a virtual computing device on which the AUT is to be tested (404). For example, a test computing device may provide data specifying that an on-board computer of the vehicle is the computing device on which the HUD application is to be tested.

[0051] Virtual user interface data is provided to the VR device (406). The virtual user interface data is i) based on a current state of the AUT, and ii) specifies data to be displayed, by the VR device, on a virtual display of the virtual computing device. For example, virtual interface data for the vehicle HUD application may be based on a simulated location and status of the vehicle, as well as the status of the HUD application, e.g., actively navigating and/or tracking speed, and the data displayed on the HUD may be specified by the virtual user interface data.

[0052] Updated AUT state data is obtained from the AUT, indicating a change in the current state of the AUT (408). For example, the AUT running on the test computing device may change state, e.g., a tester may change the AUT while it is running, altering the data to be displayed on the vehicle HUD.

[0053] Updated virtual user interface data is provided to the VR device for display on the virtual display of the virtual computing device (410). In the example situation where the information to be displayed on the vehicle HUD is to be changed, the test computing device may provide, to the VR device, the information necessary to update the virtual HUD.

[0054] The foregoing disclosure describes a number of example implementations for testing applications using virtual reality. As detailed above, examples provide a mechanism for simulating the real world and applications within a virtual environment and potential applications of a system that is capable of providing a mechanism to test applications using virtual reality.

## CLAIMS

### We claim:

1. A non-transitory machine-readable storage medium encoded with instructions executable by a hardware processor of a virtual reality (VR) device for testing applications using virtual reality, the machine-readable storage medium comprising instructions to cause the hardware processor to:

cause display of a viewable portion of a virtual environment on a VR display of the VR device;

cause display of a virtual user device within the viewable portion of the virtual environment, the virtual user device corresponding to a hardware device that is running an application under test (AUT);

cause display, on the virtual user device, of a virtual user interface of the AUT;

receive feedback data indicating i) a change in the virtual environment, ii) a change in a state of the AUT, or iii) an interaction with the virtual user device; and

in response to receiving feedback data, cause display of an updated viewable portion of the virtual environment on the VR display.

2. The storage medium of claim 1, wherein the feedback data indicates a change in the virtual environment, the change in the virtual environment comprising at least one of:

a change in a position, within the virtual environment, of a virtual user of the VR device;

a change in a view orientation of a virtual user of the VR device;

an addition, removal, or change of an object within the viewable portion of the virtual environment; or

an addition, removal, or change of an object within a non-viewable portion of the virtual environment.

3. The storage medium of claim 2, wherein the updated viewable portion of the virtual environment includes an updated virtual user interface of the AUT.

4. The storage medium of claim 1, wherein:
  - the feedback data is received from a separate computing device;
  - the feedback data indicates a change in the state of the AUT, and
  - the updated viewable portion of the virtual environment includes an updated virtual user interface of the AUT.
5. The storage medium of claim 1, wherein:
  - the feedback data indicates an interaction with the virtual user device,
  - and
  - the updated viewable portion of the virtual environment includes an updated virtual user interface of the AUT.
6. The storage medium of claim 1, wherein the virtual environment includes a second virtual device that corresponds to a second AUT, and wherein the instructions further cause the hardware processor to:
  - receive second AUT data from a separate computing device; and
  - in response to receiving second AUT data, cause display, on the virtual user device, of an updated virtual user interface of the AUT.
7. The storage medium of claim 1, wherein:
  - the feedback data indicates a change in position, within the virtual environment, of a virtual user of the VR device, and
  - the updated viewable portion of the virtual environment is based on the change in position, and wherein the instructions further cause the hardware processor to:
    - send data indicating the change in position to the hardware device running the AUT;
    - receive, from the hardware device, AUT user interface data, the AUT user interface data being based on the change in position; and
    - cause display of an updated virtual user interface of the AUT, the updated virtual user interface being based on the AUT user interface data.

8. A computing device for testing applications using virtual reality, the computing device comprising:

a hardware processor; and

a data storage device storing instructions that, when executed by the hardware processor, cause the hardware processor to:

provide virtual environment data to a virtual reality (VR) device, the virtual environment data specifying a virtual environment in which an application under test (AUT) is to be tested;

provide virtual computing device data to the VR device, the virtual computing device data specifying a virtual computing device on which the AUT is to be tested, the virtual computing device corresponding to the computing device;

provide virtual user interface data to the VR device, the virtual user interface data specifying data to be displayed, by the VR device, on a virtual display of the virtual computing device;

receive, from the VR device, feedback data indicating i) a change in position, within the virtual environment, of a virtual user of the VR device, or ii) a change in a view orientation of a virtual user of the VR device;

provide a virtual environment simulation module with sensory data indicating at least one of a position or orientation of the virtual user of the VR device, the sensory data being based on the feedback data;

receive, from the virtual environment simulation module, computing device state data indicating a change in a simulated state of the computing device;

obtain, using the AUT, updated virtual user interface data that is based on the change in the simulated state of the computing device; and

provide, to the VR device, the updated virtual user interface data for display on the virtual display of the virtual computing device.

9. The computing device of claim 8, wherein the instructions further cause the hardware processor to:

obtain interaction test data indicating an interaction with the virtual user device;

obtain updated AUT state data using the interaction test data, the updated AUT state data indicating a change in a state of the AUT; and

provide, to the VR device, updated display data for display on a VR display of the VR device, the updated display data being based on the updated AUT state data.

10. The computing device of claim 8, wherein the instructions further cause the hardware processor to:

obtain environment test data indicating a change in the virtual environment;

obtain updated AUT state data using the environment test data, the updated AUT state data indicating a change in a state of the AUT; and

provide, to the VR device, updated display data for display on a VR display of the VR device, the updated display data being based on the updated AUT state data.

11. The computing device of claim 8, wherein the instructions further cause the hardware processor to:

receive, from the AUT, updated AUT state data indicating a change in a state of the AUT; and

provide, to the VR device, updated display data for display on a VR display of the VR device, the updated display data being based on the updated AUT state data.

12. A method for testing applications using virtual reality implemented by at least one data processor, the method comprising:

providing virtual environment data to a virtual reality (VR) device, the virtual environment data specifying a virtual environment in which an application under test (AUT) is to be tested;

providing virtual computing device data to the VR device, the virtual

computing device data specifying a virtual computing device on which the AUT is to be tested;

providing virtual user interface data to the VR device, the virtual user interface data i) being based on a current state of the AUT, and ii) specifying data to be displayed, by the VR device, on a virtual display of the virtual computing device;

obtain, from the AUT, updated AUT state data indicating a change in the current state of the AUT; and

provide, to the VR device, updated virtual interface data for display on the virtual display of the virtual computing device, the updated virtual interface data being based on the updated AUT state data.

13. The method of claim 12, further comprising:

receiving, from the VR device, feedback data indicating i) a change in position, within the virtual environment, of a virtual user of the VR device, or ii) a change in a view orientation of a virtual user of the VR device;

providing a virtual environment simulation module with sensory data indicating at least one of a position or orientation of the virtual user of the VR device, the sensory data being based on the feedback data;

receive, from the virtual environment simulation module, computing device state data indicating a change in a simulated state of the computing device;

obtain, using the AUT, second updated virtual user interface data that is based on the change in the simulated state of the computing device; and

provide, to the VR device, the second updated virtual user interface data for display on the virtual display of the virtual computing device.

14. The method of claim 12, further comprising:

obtaining interaction test data indicating an interaction with the virtual user device;

obtaining second updated AUT state data using the interaction test data, the second updated AUT state data indicating a second change in the current

state of the AUT; and

provide, to the VR device, updated display data for display on a VR display of the VR device, the updated display data being based on the second updated AUT state data.

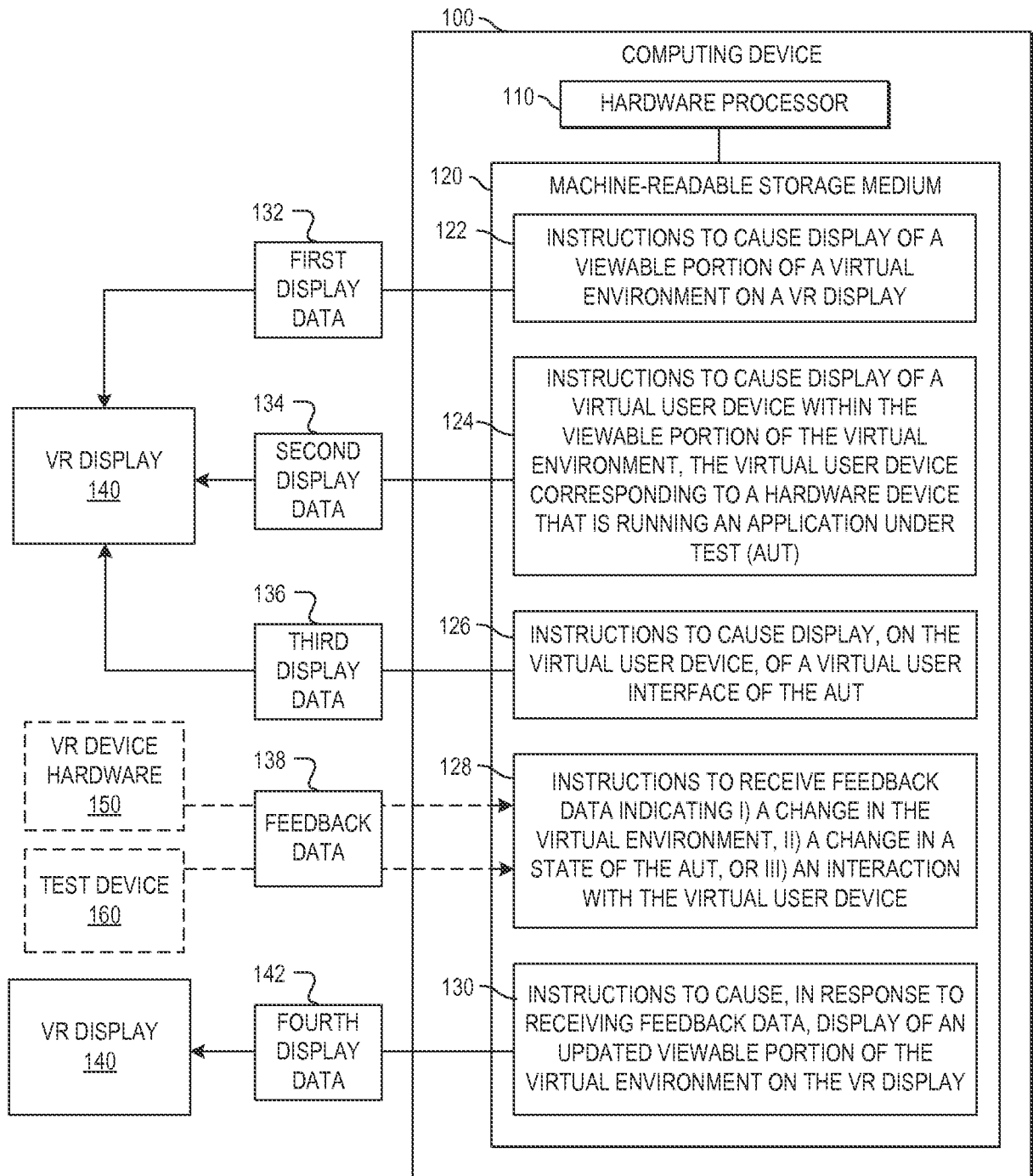
15. The method of claim 12, further comprising:

obtaining environment test data indicating a change in the virtual environment;

obtaining updated AUT state data using the environment test data, the updated AUT state data indicating a change in a state of the AUT; and

providing, to the VR device, updated display data for display on a VR display of the VR device, the updated display data being based on the updated AUT state data.

1/4

**FIG. 1**

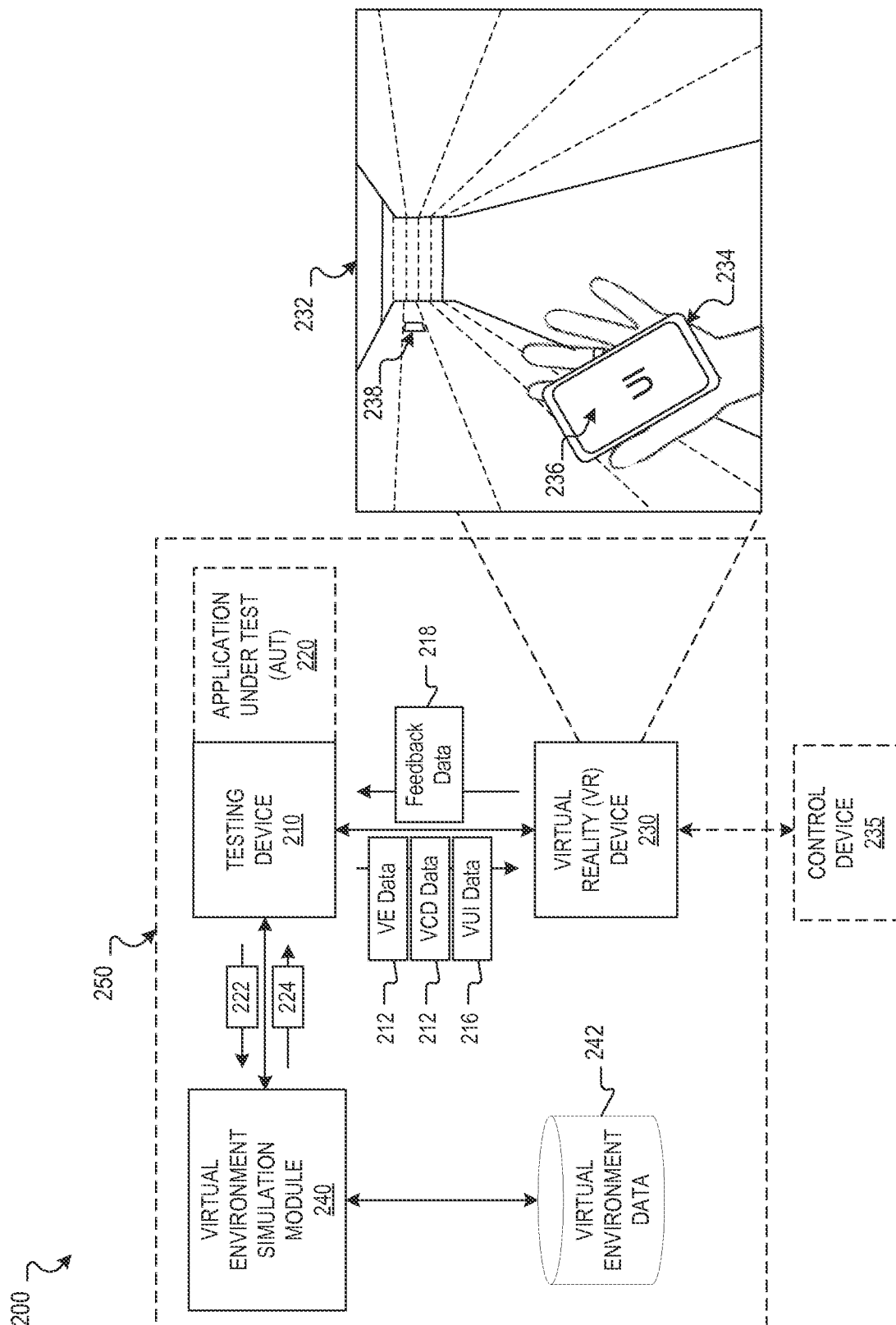


FIG. 2

300 ~~~~~>

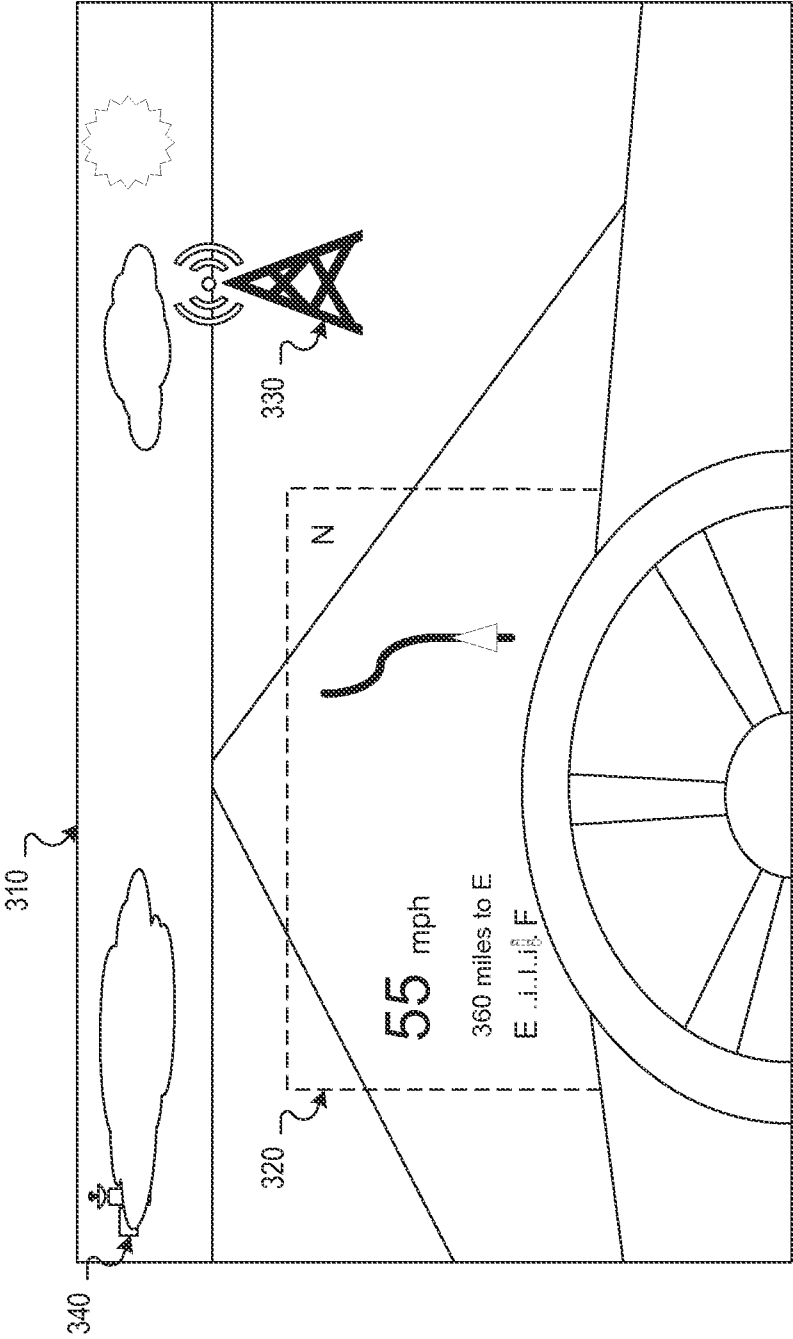
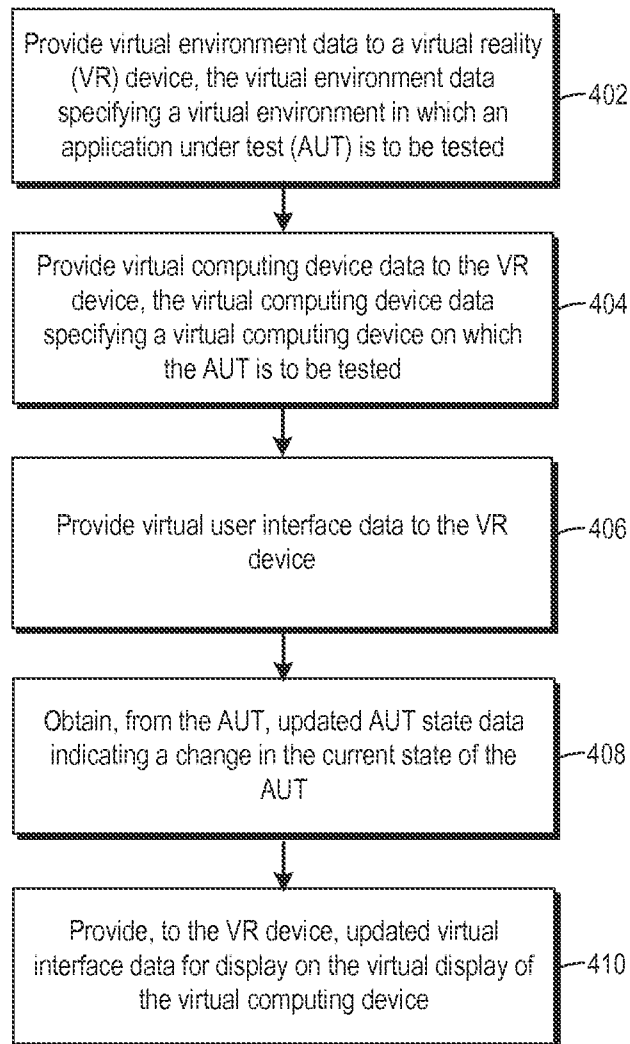


FIG. 3

4/4

400

**FIG. 4**

**A. CLASSIFICATION OF SUBJECT MATTER****G06F 9/455(2006.01)i, G06T 17/00(2006.01)i**

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 9/455; G06F 15/16; G06F 9/44; G06T 17/00

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS(KIPO internal) &amp; Keywords: application, test, virtual environment, simulation

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category* | Citation of document, with indication, where appropriate, of the relevant passages                              | Relevant to claim No. |
|-----------|-----------------------------------------------------------------------------------------------------------------|-----------------------|
| A         | US 2010-0146487 A1 (CHUN, I. et al.) 10 June 2010<br>See abstract; paragraphs [0067]-[0084]; and figs. 5 and 6. | 1-15                  |
| A         | US 2014-0180666 A1 (MUTTIK, I.) 26 June 2014<br>See abstract; paragraphs [0020]-[0028]; and fig. 5.             | 1-15                  |
| A         | US 2013-0275945 A1 (KOLLBERG, D. et al.) 17 October 2013<br>See abstract; paragraphs [0017]-[0033]; and fig. 3. | 1-15                  |
| A         | US 2006-0212540 A1 (CHON, K. et al.) 21 September 2006<br>See abstract; paragraphs [0068]-[0072]; and fig. 5.   | 1-15                  |
| A         | US 2011-0083122 A1 (CHEN, J. et al.) 07 April 2011<br>See abstract; paragraphs [0045]-[0050]; and fig. 2.       | 1-15                  |



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

"&amp;" document member of the same patent family

Date of the actual completion of the international search

11 February 2016 (11.02.2016)

Date of mailing of the international search report

**12 February 2016 (12.02.2016)**

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-472-7140

Authorized officer

YU, Jintae

Telephone No. +82-42-481-8530



**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2015/033441**

| Patent document<br>cited in search report | Publication<br>date | Patent family<br>member(s)                              | Publication<br>date                    |
|-------------------------------------------|---------------------|---------------------------------------------------------|----------------------------------------|
| US 2010-0146487 A1                        | 10/06/2010          | KR 10-2010-0065046 A<br>US 8549491 B2                   | 15/06/2010<br>01/10/2013               |
| US 2014-0180666 A1                        | 26/06/2014          | WO 2014-099623 A1                                       | 26/06/2014                             |
| US 2013-0275945 A1                        | 17/10/2013          | US 2014-372991 A1<br>US 8782615 B2                      | 18/12/2014<br>15/07/2014               |
| US 2006-0212540 A1                        | 21/09/2006          | US 7441236 B2<br>WO 2006-047741 A2<br>WO 2006-047741 A3 | 21/10/2008<br>04/05/2006<br>07/05/2009 |
| US 2011-0083122 A1                        | 07/04/2011          | None                                                    |                                        |