(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
17 October 2024 (17.10.2024)

WIPO | PCT

(10) International Publication Number
WO 2024/215541 A1

(51) International Patent Classification:
G06T 19/00 (2011.01) *G09F 19/16* (2006.01)
G06V 10/74 (2022.01)

(21) International Application Number:
PCT/US2024/022836

(22) International Filing Date:
03 April 2024 (03.04.2024)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
63/495,599 12 April 2023 (12.04.2023) US
18/477,248 28 September 2023 (28.09.2023) US

(63) Related by continuation (CON) or continuation-in-part (CIP) to earlier application:
US 18/477,248 (CON)
Filed on 28 September 2023 (28.09.2023)

(71) Applicant: **SNAP INC.** [US/US]; 3000 31st Street, Santa Monica, California 90405 (US).

(72) Inventors: **BRADFORD, Adrian**; 3000 31st Street, Santa Monica, California 90405 (US). **CAVINS, Christopher**; 3000 31st Street, Santa Monica, California 90405 (US). **VONK, James**; 3000 31st Street, Santa Monica, California 90405 (US).

(74) Agent: **PERDOK, Monique M.** et al.; Schwegman Lundberg & Woessner, P.A., P.O. Box 2938, 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, CV, CZ, DE, DJ, DK, DM,

(54) Title: STATIONARY EXTENDED REALITY DEVICE

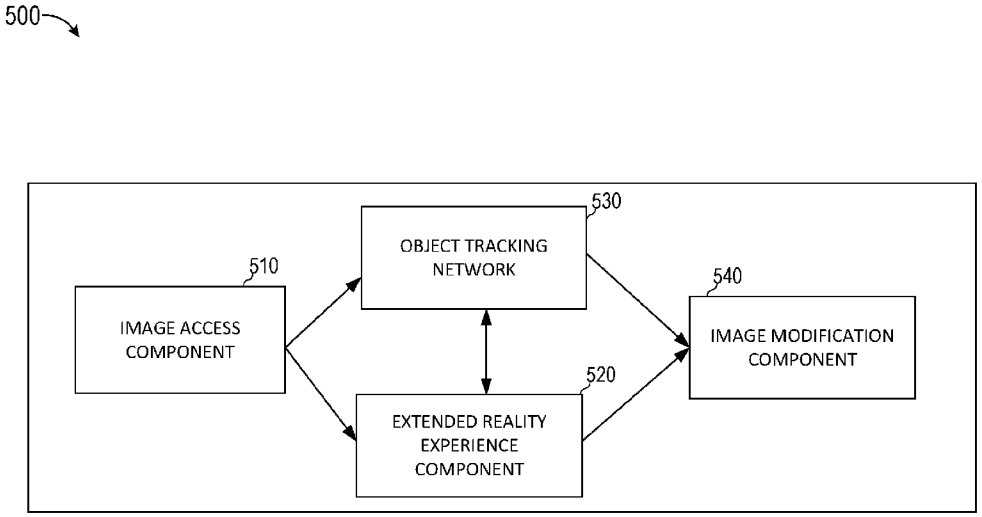


FIG. 5

(57) Abstract: Methods and systems are disclosed for generating an extended reality (XR) experience using a statically positioned device. The system receives, from a camera of a stationary device, an image depicting a real -world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment. The system analyzes the image using a machine learning model to predict tracking information for the real -world object, the machine learning model trained based on a plurality of training images depicting real -world objects in the specified field of view of the real-world environment and corresponding ground-truth tracking information for the real -world objects. The system selects an extended reality (XR) experience from a plurality of XR experiences and overlays one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

WO 2024/215541 A1

[Continued on next page]

DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

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

Published:

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

— *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))*

STATIONARY EXTENDED REALITY DEVICE

CLAIM OF PRIORITY

[0001] This application claims the benefit of priority to U.S. Provisional Application Serial No. 63/495,599, filed April 12, 2023, and U.S. Patent Application Serial No. 18/477,248, filed September 28, 2023, each of which is incorporated herein by reference in its entirety.

TECHNICAL FIELD

[0002] The present disclosure relates to extended reality (XR) devices, such as augmented reality (AR) and/or virtual reality (VR) devices.

BACKGROUND

[0003] AR is a modification of a virtual environment. For example, in VR, a user is completely immersed in a virtual world, whereas in AR, the user is immersed in a world where virtual objects are combined or superimposed on the real world. An AR system aims to generate and present virtual objects that interact realistically with a real-world environment and with each other. Examples of AR applications can include single or multiple player video games, instant messaging systems, and the like.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0004] In the drawings, which are not necessarily drawn to scale, like numerals may describe similar components in different views. To easily identify the discussion of any particular element or act, the most significant digit or digits in a reference number refer to the figure number in which that element is first introduced. Some nonlimiting examples are illustrated in the figures of the accompanying drawings in which:

[0005] FIG. 1 is a diagrammatic representation of a networked environment in which the present disclosure may be deployed, according to some examples.

[0006] FIG. 2 is a diagrammatic representation of a messaging system that has both client-side and server-side functionality, according to some examples.

[0007] FIG. 3 is a diagrammatic representation of a data structure as maintained in a database, according to some examples.

[0008] FIG. 4 is a diagrammatic representation of a message, according to some examples.

[0009] FIG. 5 is a diagrammatic representation of a stationary XR system, in accordance with some examples.

[0010] FIG. 6 is a diagrammatic representation of a stationary XR shoe mirror, in accordance with some examples.

[0011] FIG. 7 is a diagrammatic representation of a stationary XR vanity mirror, in accordance with some examples.

[0012] FIG. 8 is a diagrammatic representation of a stationary XR window, in accordance with some examples.

[0013] FIG. 9 is a diagrammatic representation of a general stationary XR system, in accordance with some examples.

[0014] FIG. 10 is a flowchart illustrating example operations of the stationary XR system, in accordance with some examples.

[0015] FIG. 11 is a diagrammatic representation of a machine in the form of a computer system within which a set of instructions may be executed to cause the machine to perform any one or more of the methodologies discussed herein, in accordance with some examples.

[0016] FIG. 12 is a block diagram showing a software architecture within which examples may be implemented.

[0017] FIG. 13 illustrates a system in which a head-wearable apparatus may be implemented, in accordance with some examples.

DETAILED DESCRIPTION

[0018] The description that follows includes systems, methods, techniques, instruction sequences, and computing machine program products that embody illustrative examples of the disclosure. In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide an understanding of various examples. It will be evident, however, to those skilled in the art, that examples may be practiced without these specific details. In general, well-known instruction instances, protocols, structures, and techniques are not necessarily shown in detail.

[0019] Typically, VR and AR systems display images representing a given user by capturing an image of the user and, in addition, obtaining a depth map using a depth sensor of the real-world human body depicted in the image. By processing the depth map and the image together, the VR and AR systems can detect positioning of the user in the image and can appropriately modify the user or background in the images. While such systems work well, the need for a depth sensor limits the scope of their applications. This is because adding depth sensors to user devices for the purpose of modifying images increases the

overall cost and complexity of the devices, making them less attractive and more difficult to implement in a mobile device setting.

[0020] Certain systems implement AR applications on mobile devices. These applications allow users to unlock various AR experiences on their mobile devices to interact with virtual objects. These AR applications usually rely on complex tracking operations using multiple machine learning models. These machine learning models are trained on a variety of data and are useful in the general sense to perform tracking of real-world objects in a variety of different environments. Because of this, the tracking in a specific environment of a given user may not be optimal and can result in poor performance and tracking. This can cause the virtual elements to behave erratically, which takes away from the illusion that the virtual elements are part of the real-world environment.

[0021] Also, in these mobile-implemented AR systems, a user needs to carefully position and hold a camera steady at a certain orientation and to ensure the camera parameters and lighting conditions are suitable for executing the AR experience. Any small deviation in the camera parameters or movement of the camera can cause tracking to be lost and the virtual elements to behave erratically. These AR systems also apply complex in-painting machine learning models to a captured image to occlude certain objects in the image to improve the realism in the display of the virtual elements. These complex in-painting processes also have limitations and are not tailored to the specific user's environment, which can also cause them to behave erratically. In this way, these systems can generally produce useful AR experiences but have many limitations and place a great deal of burden on users, detracting from the overall enjoyment and experience of using the systems. This can lead to lack of use and waste of resources.

[0022] The disclosed techniques seek to improve the way in which XR experiences are provided and used. Particularly, the disclosed techniques implement the XR experiences on a stationary device. In some cases, the stationary device includes a secure enclosure that houses various components including the camera, a screen, and processors used to execute machine learning models and operations of the XR experiences. This secure enclosure can be bolted to a surface, such as a ground, a table, or a wall. The disclosed XR experiences use one or more machine learning models that are specifically trained using specific training data to provide the corresponding XR experiences in the real-world environment in which the stationary device is placed. By tailoring the machine learning models to a specific real-world environment and by fixing the camera parameters and lighting conditions based on the real-world environment, the XR experiences provide more realistic displays of virtual

elements in a captured image. This improves the overall user experience and provides a true-to-life presentation of virtual content.

[0023] Also, by configuring the XR experiences to the real-world environment in which the device is physically anchored, statically positioned, or stationed, complex in-painting processes can be avoided. Namely, the XR experiences can be configured with information that identifies the field of view in which real-world objects are captured by the camera and that can be predetermined and may not change during the course of executing the XR experiences. In this way, the XR experiences can be configured to overlay virtual objects only within that field of view by using a predefined segmentation mask that defines the field of view, which avoids having to apply complex in-painting operations or executing in-painting machine learning models. This further improves the overall efficiencies of operating the device.

[0024] Specifically, the disclosed techniques receive, from a camera of a stationary device, an image depicting a real-world object. The camera can be directed in a stationary manner towards a specified field of view of a real-world environment. The disclosed techniques analyze the image using a machine learning model to predict tracking information for the real-world object. The machine learning model can be trained based on a plurality of training images depicting real-world objects in the specified field of view of the real-world environment and corresponding ground-truth tracking information for the real-world objects. The disclosed techniques select an XR experience from a plurality of XR experiences (in an automated manner, a preconfigured manner, or in response to user input) and overlay one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image. In some cases, the stationary device includes an XR shoe mirror, an XR vanity mirror, or an XR window.

[0025] The disclosed techniques simplify the process of adding AR graphics to an image or video, which significantly reduces design constraints and costs in generating such AR graphics and decreases the amount of processing complexities and power and memory requirements. This also improves the illusion of the AR graphics being part of a real-world environment depicted in an image or video that depicts real-world objects. This enables seamless and efficient addition of AR graphics to an underlying image or video in real time.

[0026] The disclosed techniques improve the overall experience of the user in using the electronic device. Also, by providing such AR experiences without using a depth sensor, the overall amount of system resources needed to accomplish a task is reduced. As used herein, “article of clothing,” “fashion item,” and “garment” are used interchangeably and should be

understood to have the same meaning. Article of clothing, garment, or fashion item can include a shirt, skirt, dress, shoes, purse, furniture item, household item, eyewear, eyeglasses, AR logo, AR emblem, pants, shorts, jacket, t-shirt, blouse, glasses, jewelry, earrings, bunny ears, a hat, earmuffs, makeup, and/or any other suitable item or object.

NETWORKED COMPUTING ENVIRONMENT

[0027] **FIG. 1** is a block diagram showing an example interaction system 100 for facilitating interactions (e.g., exchanging text messages, conducting text audio and video calls, or playing games) over a network. The interaction system 100 includes multiple user systems 102, each of which hosts multiple applications, including an interaction client 104 and other applications 106. Each interaction client 104 is communicatively coupled, via one or more communication networks including a network 108 (e.g., the Internet), to other instances of the interaction client 104 (e.g., hosted on respective other user systems 102), an interaction server system 110 and third-party servers 112). An interaction client 104 can also communicate with locally hosted applications 106 using Applications Program Interfaces (APIs).

[0028] Each user system 102 may include multiple user devices, such as a mobile device 114, head-wearable apparatus 116, and a computer client device 118 that are communicatively connected to exchange data and messages.

[0029] An interaction client 104 interacts with other interaction clients 104 and with the interaction server system 110 via the network 108. The data exchanged between the interaction clients 104 (e.g., interactions 120) and between the interaction clients 104 and the interaction server system 110 includes functions (e.g., commands to invoke functions) and payload data (e.g., text, audio, video, or other multimedia data).

[0030] The interaction server system 110 provides server-side functionality via the network 108 to the interaction clients 104. While certain functions of the interaction system 100 are described herein as being performed by either an interaction client 104 or by the interaction server system 110, the location of certain functionality either within the interaction client 104 or the interaction server system 110 may be a design choice. For example, it may be technically preferable to initially deploy particular technology and functionality within the interaction server system 110 but to later migrate this technology and functionality to the interaction client 104 where a user system 102 has sufficient processing capacity.

[0031] The interaction server system 110 supports various services and operations that are provided to the interaction clients 104. Such operations include transmitting data to, receiving data from, and processing data generated by the interaction clients 104. This data may include message content, client device information, geolocation information, media augmentation and overlays, message content persistence conditions, entity relationship information, and live event information. Data exchanges within the interaction system 100 are invoked and controlled through functions available via user interfaces of the interaction clients 104.

[0032] Turning now specifically to the interaction server system 110, an API server 122 is coupled to and provides programmatic interfaces to interaction servers 124, making the functions of the interaction servers 124 accessible to interaction clients 104, other applications 106, and third-party server 112. The interaction servers 124 are communicatively coupled to a database server 126, facilitating access to a database 128 that stores data associated with interactions processed by the interaction servers 124. Similarly, a web server 130 is coupled to the interaction servers 124 and provides web-based interfaces to the interaction servers 124. To this end, the web server 130 processes incoming network requests over Hypertext Transfer Protocol (HTTP) and several other related protocols.

[0033] The API server 122 receives and transmits interaction data (e.g., commands and message payloads) between the interaction servers 124 and the user systems 102 (and, for example, interaction clients 104 and other application 106) and the third-party server 112. Specifically, the API server 122 provides a set of interfaces (e.g., routines and protocols) that can be called or queried by the interaction client 104 and other applications 106 to invoke functionality of the interaction servers 124. The API server 122 exposes various functions supported by the interaction servers 124, including account registration; login functionality; the sending of interaction data, via the interaction servers 124, from a particular interaction client 104 to another interaction client 104; the communication of media files (e.g., images or video) from an interaction client 104 to the interaction servers 124; the settings of a collection of media data (e.g., a story); the retrieval of a list of friends of a user of a user system 102; the retrieval of messages and content; the addition and deletion of entities (e.g., friends) to an entity relationship graph (e.g., the entity graph 310); the location of friends within an entity relationship graph; and opening an application event (e.g., relating to the interaction client 104).

[0034] The interaction servers 124 host multiple systems and subsystems, described below with reference to FIG. 2.

LINKED APPLICATIONS

[0035] Returning to the interaction client 104, features and functions of an external resource (e.g., a linked application 106 or applet) are made available to a user via an interface of the interaction client 104. In this context, "external" refers to the fact that the application 106 or applet is external to the interaction client 104. The external resource is often provided by a third party but may also be provided by the creator or provider of the interaction client 104. The interaction client 104 receives a user selection of an option to launch or access features of such an external resource. The external resource may be the application 106 installed on the user system 102 (e.g., a "native app"), or a small-scale version of the application (e.g., an "applet") that is hosted on the user system 102 or remote of the user system 102 (e.g., on third-party servers 112). The small-scale version of the application includes a subset of features and functions of the application (e.g., the full-scale, native version of the application) and is implemented using a markup-language document. In some examples, the small-scale version of the application (e.g., an "applet") is a web-based, markup-language version of the application and is embedded in the interaction client 104. In addition to using markup-language documents (e.g., a .*ml file), an applet may incorporate a scripting language (e.g., a .*js file or a .json file) and a style sheet (e.g., a .*ss file).

[0036] In response to receiving a user selection of the option to launch or access features of the external resource, the interaction client 104 determines whether the selected external resource is a web-based external resource or a locally-installed application 106. In some cases, applications 106 that are locally installed on the user system 102 can be launched independently of and separately from the interaction client 104, such as by selecting an icon corresponding to the application 106 on a home screen of the user system 102. Small-scale versions of such applications can be launched or accessed via the interaction client 104 and, in some examples, no or limited portions of the small-scale application can be accessed outside of the interaction client 104. The small-scale application can be launched by the interaction client 104 receiving, from a third-party server 112 for example, a markup-language document associated with the small-scale application and processing such a document.

[0037] In response to determining that the external resource is a locally-installed application 106, the interaction client 104 instructs the user system 102 to launch the external resource by executing locally-stored code corresponding to the external resource. In

response to determining that the external resource is a web-based resource, the interaction client 104 communicates with the third-party servers 112 (for example) to obtain a markup-language document corresponding to the selected external resource. The interaction client 104 then processes the obtained markup-language document to present the web-based external resource within a user interface of the interaction client 104.

[0038] The interaction client 104 can notify a user of the user system 102, or other users related to such a user (e.g., "friends"), of activity taking place in one or more external resources. For example, the interaction client 104 can provide participants in a conversation (e.g., a chat session) in the interaction client 104 with notifications relating to the current or recent use of an external resource by one or more members of a group of users. One or more users can be invited to join in an active external resource or to launch a recently used but currently inactive (in the group of friends) external resource. The external resource can provide participants in a conversation, each using respective interaction clients 104, with the ability to share an item, status, state, or location in an external resource in a chat session with one or more members of a group of users. The shared item may be an interactive chat card with which members of the chat can interact, for example, to launch the corresponding external resource, view specific information within the external resource, or take the member of the chat to a specific location or state within the external resource. Within a given external resource, response messages can be sent to users on the interaction client 104. The external resource can selectively include different media items in the responses, based on a current context of the external resource.

[0039] The interaction client 104 can present a list of the available external resources (e.g., applications 106 or applets) to a user to launch or access a given external resource. This list can be presented in a context-sensitive menu. For example, the icons representing different ones of the application 106 (or applets) can vary based on how the menu is launched by the user (e.g., from a conversation interface or from a non-conversation interface).

SYSTEM ARCHITECTURE

[0040] **FIG. 2** is a block diagram illustrating further details regarding the interaction system 100, according to some examples. Specifically, the interaction system 100 is shown to comprise the interaction client 104 and the interaction servers 124. The interaction system 100 embodies multiple subsystems, which are supported on the client-side by the interaction client 104 and on the server-side by the interaction servers 124. Example subsystems are discussed below and can include a stationary XR system 500 that provides

an XR experience to one or more users in a dedicated and stationary physical location of a real-world environment. An illustrative implementation of the stationary XR system 500 is shown and described in connection with FIG. 5 below.

[0041] In some examples, these subsystems are implemented as microservices. A microservice subsystem (e.g., a microservice application) may have components that enable it to operate independently and communicate with other services. Example components of a microservice subsystem may include:

- Function logic: The function logic implements the functionality of the microservice subsystem, representing a specific capability or function that the microservice provides.
- API interface: Microservices may communicate with other components through well-defined APIs or interfaces, using lightweight protocols such as REST or messaging. The API interface defines the inputs and outputs of the microservice subsystem and how it interacts with other microservice subsystems of the interaction system 100.
- Data storage: A microservice subsystem may be responsible for its own data storage, which may be in the form of a database, cache, or other storage mechanism (e.g., using the database server 126 and database 128). This enables a microservice subsystem to operate independently of other microservices of the interaction system 100.
- Service discovery: Microservice subsystems may find and communicate with other microservice subsystems of the interaction system 100. Service discovery mechanisms enable microservice subsystems to locate and communicate with other microservice subsystems in a scalable and efficient way.
- Monitoring and logging: Microservice subsystems may need to be monitored and logged in order to ensure availability and performance. Monitoring and logging mechanisms enable the tracking of health and performance of a microservice subsystem.

[0042] In some examples, the interaction system 100 may employ a monolithic architecture, a service-oriented architecture (SOA), a function-as-a-service (FaaS) architecture, or a modular architecture:

[0043] An image processing system 202 provides various functions that enable a user to capture and augment (e.g., annotate or otherwise modify or edit) media content associated with a message.

[0044] A camera system 204 includes control software (e.g., in a camera application) that interacts with and controls camera hardware (e.g., directly or via operating system controls) of the user system 102 to modify and augment real-time images captured and displayed via the interaction client 104.

[0045] An augmentation system 206 provides functions related to the generation and publishing of augmentations (e.g., media overlays) for images captured in real-time by cameras of the user system 102 or retrieved from memory of the user system 102. For example, the augmentation system 206 operatively selects, presents, and displays media overlays (e.g., an image filter or an image lens) to the interaction client 104 for the augmentation of real-time images received via the camera system 204 or stored images retrieved from memory 1302 (shown in FIG. 13) of a user system 102. These augmentations are selected by the augmentation system 206 and presented to a user of an interaction client 104, based on a number of inputs and data, such as, for example:

- Geolocation of the user system 102; and
- Entity relationship information of the user of the user system 102.

[0046] An augmentation may include audio and visual content and visual effects. Examples of audio and visual content include pictures, texts, logos, animations, and sound effects. An example of a visual effect includes color overlaying. The audio and visual content or the visual effects can be applied to a media content item (e.g., a photo or video) at user system 102 for communication in a message, or applied to video content, such as a video content stream or feed transmitted from an interaction client 104. As such, the image processing system 202 may interact with, and support, the various subsystems of a communication system 208, such as a messaging system 210 and a video communication system 212.

[0047] A media overlay may include text or image data that can be overlaid on top of a photograph taken by the user system 102 or a video stream produced by the user system 102. In some examples, the media overlay may be a location overlay (e.g., Venice beach), a name of a live event, or a name of a merchant overlay (e.g., Beach Coffee House). In further examples, the image processing system 202 uses the geolocation of the user system 102 to identify a media overlay that includes the name of a merchant at the geolocation of the user system 102. The media overlay may include other indicia associated with the merchant. The media overlays may be stored in the databases 128 and accessed through the database server 126.

[0048] The image processing system 202 provides a user-based publication platform that enables users to select a geolocation on a map and upload content associated with the selected geolocation. The user may also specify circumstances under which a particular media overlay should be offered to other users. The image processing system 202 generates

a media overlay that includes the uploaded content and associates the uploaded content with the selected geolocation.

[0049] An augmentation creation system 214 supports AR developer platforms and includes an application for content creators (e.g., artists and developers) to create and publish augmentations (e.g., AR experiences) of the interaction client 104. The augmentation creation system 214 provides a library of built-in features and tools to content creators including, for example, custom shaders, tracking technology, and templates.

[0050] In some examples, the augmentation creation system 214 provides a merchant-based publication platform that enables merchants to select a particular augmentation associated with a geolocation via a bidding process. For example, the augmentation creation system 214 associates a media overlay of the highest bidding merchant with a corresponding geolocation for a predefined amount of time.

[0051] A communication system 208 is responsible for enabling and processing multiple forms of communication and interaction within the interaction system 100 and includes a messaging system 210, an audio communication system 216, and a video communication system 212. The messaging system 210 is responsible for enforcing the temporary or time-limited access to content by the interaction clients 104. The messaging system 210 incorporates multiple timers (e.g., within a user management system 218) that, based on duration and display parameters associated with a message or collection of messages (e.g., a story), selectively enable access (e.g., for presentation and display) to messages and associated content via the interaction client 104. The audio communication system 216 enables and supports audio communications (e.g., real-time audio chat) between multiple interaction clients 104. Similarly, the video communication system 212 enables and supports video communications (e.g., real-time video chat) between multiple interaction clients 104.

[0052] A user management system 218 is operationally responsible for the management of user data and profiles, and maintains entity information (e.g., stored in entity tables 308, entity graphs 310, and profile data 302 shown in FIG. 3) regarding users and relationships between users of the interaction system 100.

[0053] A collection management system 220 is operationally responsible for managing sets or collections of media (e.g., collections of text, image video, and audio data). A collection of content (e.g., messages, including images, video, text, and audio) may be organized into an “event gallery” or an “event story.” Such a collection may be made available for a specified time period, such as the duration of an event to which the content relates. For

example, content relating to a music concert may be made available as a "story" for the duration of that music concert. The collection management system 220 may also be responsible for publishing an icon that provides notification of a particular collection to the user interface of the interaction client 104. The collection management system 220 includes a curation function that allows a collection manager to manage and curate a particular collection of content. For example, the curation interface enables an event organizer to curate a collection of content relating to a specific event (e.g., to delete inappropriate content or redundant messages). Additionally, the collection management system 220 employs machine vision (or image recognition technology) and content rules to curate a content collection automatically. In certain examples, compensation may be paid to a user to include user-generated content into a collection. In such cases, the collection management system 220 operates to automatically make payments to such users to use their content.

[0054] A map system 222 provides various geographic location (e.g., geolocation) functions and supports the presentation of map-based media content and messages by the interaction client 104. For example, the map system 222 enables the display of user icons or avatars (e.g., stored in profile data 302 of FIG. 3) on a map to indicate a current or past location of "friends" of a user, as well as media content (e.g., collections of messages including photographs and videos) generated by such friends, within the context of a map. For example, a message posted by a user to the interaction system 100 from a specific geographic location may be displayed within the context of a map at that particular location to "friends" of a specific user on a map interface of the interaction client 104. A user can furthermore share his or her location and status information (e.g., using an appropriate status avatar) with other users of the interaction system 100 via the interaction client 104, with this location and status information being similarly displayed within the context of a map interface of the interaction client 104 to selected users.

[0055] A game system 224 provides various gaming functions within the context of the interaction client 104. The interaction client 104 provides a game interface providing a list of available games that can be launched by a user within the context of the interaction client 104 and played with other users of the interaction system 100. The interaction system 100 further enables a particular user to invite other users to participate in the play of a specific game by issuing invitations to such other users from the interaction client 104. The interaction client 104 also supports audio, video, and text messaging (e.g., chats) within the context of gameplay, provides a leaderboard for the games, and also supports the provision of in-game rewards (e.g., coins and items).

[0056] An external resource system 226 provides an interface for the interaction client 104 to communicate with remote servers (e.g., third-party servers 112) to launch or access external resources, i.e., applications or applets. Each third-party server 112 hosts, for example, a markup language (e.g., HTML5) based application or a small-scale version of an application (e.g., game, utility, payment, or ride-sharing application). The interaction client 104 may launch a web-based resource (e.g., application) by accessing the HTML5 file from the third-party servers 112 associated with the web-based resource. Applications hosted by third-party servers 112 are programmed in JavaScript leveraging a Software Development Kit (SDK) provided by the interaction servers 124. The SDK includes APIs with functions that can be called or invoked by the web-based application. The interaction servers 124 host a JavaScript library that provides a given external resource access to specific user data of the interaction client 104. HTML5 is an example of technology for programming games, but applications and resources programmed based on other technologies can be used.

[0057] To integrate the functions of the SDK into the web-based resource, the SDK is downloaded by the third-party server 112 from the interaction servers 124 or is otherwise received by the third-party server 112. Once downloaded or received, the SDK is included as part of the application code of a web-based external resource. The code of the web-based resource can then call or invoke certain functions of the SDK to integrate features of the interaction client 104 into the web-based resource.

[0058] The SDK stored on the interaction server system 110 effectively provides the bridge between an external resource (e.g., applications 106 or applets) and the interaction client 104. This gives the user a seamless experience of communicating with other users on the interaction client 104 while also preserving the look and feel of the interaction client 104. To bridge communications between an external resource and an interaction client 104, the SDK facilitates communication between third-party servers 112 and the interaction client 104. A bridge script running on a user system 102 establishes two one-way communication channels between an external resource and the interaction client 104. Messages are sent between the external resource and the interaction client 104 via these communication channels asynchronously. Each SDK function invocation is sent as a message and callback. Each SDK function is implemented by constructing a unique callback identifier and sending a message with that callback identifier.

[0059] By using the SDK, not all information from the interaction client 104 is shared with third-party servers 112. The SDK limits which information is shared based on the needs of the external resource. Each third-party server 112 provides an HTML5 file corresponding to the web-based external resource to interaction servers 124. The interaction servers 124

can add a visual representation (such as a box art or other graphic) of the web-based external resource in the interaction client 104. Once the user selects the visual representation or instructs the interaction client 104 through a graphical user interface (GUI) of the interaction client 104 to access features of the web-based external resource, the interaction client 104 obtains the HTML5 file and instantiates the resources to access the features of the web-based external resource.

[0060] The interaction client 104 presents a GUI (e.g., a landing page or title screen) for an external resource. During, before, or after presenting the landing page or title screen, the interaction client 104 determines whether the launched external resource has been previously authorized to access user data of the interaction client 104. In response to determining that the launched external resource has been previously authorized to access user data of the interaction client 104, the interaction client 104 presents another GUI of the external resource that includes functions and features of the external resource. In response to determining that the launched external resource has not been previously authorized to access user data of the interaction client 104, after a threshold period of time (e.g., 3 seconds) of displaying the landing page or title screen of the external resource, the interaction client 104 slides up (e.g., animates a menu as surfacing from a bottom of the screen to a middle or other portion of the screen) a menu for authorizing the external resource to access the user data. The menu identifies the type of user data that the external resource will be authorized to use. In response to receiving a user selection of an accept option, the interaction client 104 adds the external resource to a list of authorized external resources and allows the external resource to access user data from the interaction client 104. The external resource is authorized by the interaction client 104 to access the user data under an OAuth 2 framework.

[0061] The interaction client 104 controls the type of user data that is shared with external resources based on the type of external resource being authorized. For example, external resources that include full-scale applications (e.g., an application 106) are provided with access to a first type of user data (e.g., 2D avatars of users with or without different avatar characteristics). As another example, external resources that include small-scale versions of applications (e.g., web-based versions of applications) are provided with access to a second type of user data (e.g., payment information, 2D avatars of users, 3D avatars of users, and avatars with various avatar characteristics). Avatar characteristics include different ways to customize a look and feel of an avatar, such as different poses, facial features, clothing, and so forth.

[0062] An advertisement system 228 operationally enables the purchasing of advertisements by third parties for presentation to end-users via the interaction clients 104 and also handles the delivery and presentation of these advertisements.

[0063] An artificial intelligence and machine learning system 230 provides a variety of services to different subsystems within the interaction system 100. For example, the artificial intelligence and machine learning system 230 operates with the image processing system 202 and the camera system 204 to analyze images and extract information such as objects, text, or faces. This information can then be used by the image processing system 202 to enhance, filter, or manipulate images. The artificial intelligence and machine learning system 230 may be used by the augmentation system 206 to generate augmented content, XR experiences, and AR experiences, such as adding virtual objects or animations to real-world images. The communication system 208 and messaging system 210 may use the artificial intelligence and machine learning system 230 to analyze communication patterns and provide insights into how users interact with each other and provide intelligent message classification and tagging, such as categorizing messages based on sentiment or topic. The artificial intelligence and machine learning system 230 may also provide chatbot functionality to message interactions 120 between user systems 102 and between a user system 102 and the interaction server system 110. The artificial intelligence and machine learning system 230 may also work with the audio communication system 216 to provide speech recognition and natural language processing capabilities, allowing users to interact with the interaction system 100 using voice commands. In some cases, the artificial intelligence and machine learning system 230 can implement one or more machine learning models that are used by the stationary XR system 500 to provide an XR experience to a user at a specific real-world location/environment.

DATA ARCHITECTURE

[0064] **FIG. 3** is a schematic diagram illustrating data structures 300, which may be stored in a database 304 of the interaction server system 110, according to certain examples. While the content of the database 304 is shown to comprise multiple tables, it will be appreciated that the data could be stored in other types of data structures (e.g., as an object-oriented database).

[0065] The database 304 includes message data stored within a message table 306. This message data includes, for any particular message, at least message sender data, message recipient (or receiver) data, and a payload. Further details regarding information that may be

included in a message, and included within the message data stored in the message table 306, are described below with reference to **FIG. 4**.

[0066] An entity table 308 stores entity data, and is linked (e.g., referentially) to an entity graph 310 and profile data 302. Entities for which records are maintained within the entity table 308 may include individuals, corporate entities, organizations, objects, places, events, and so forth. Regardless of entity type, any entity regarding which the interaction server system 110 stores data may be a recognized entity. Each entity is provided with a unique identifier, as well as an entity type identifier (not shown).

[0067] The entity graph 310 stores information regarding relationships and associations between entities. Such relationships may be social, professional (e.g., work at a common corporation or organization), interest-based, or activity-based, merely for example. Certain relationships between entities may be unidirectional, such as a subscription by an individual user to digital content of a commercial or publishing user (e.g., a newspaper or other digital media outlet, or a brand). Other relationships may be bidirectional, such as a "friend" relationship between individual users of the interaction system 100.

[0068] Certain permissions and relationships may be attached to each relationship, and also to each direction of a relationship. For example, a bidirectional relationship (e.g., a friend relationship between individual users) may include authorization for the publication of digital content items between the individual users, but may impose certain restrictions or filters on the publication of such digital content items (e.g., based on content characteristics, location data or time of day data). Similarly, a subscription relationship between an individual user and a commercial user may impose different degrees of restrictions on the publication of digital content from the commercial user to the individual user, and may significantly restrict or block the publication of digital content from the individual user to the commercial user. A particular user, as an example of an entity, may record certain restrictions (e.g., by way of privacy settings) in a record for that entity within the entity table 308. Such privacy settings may be applied to all types of relationships within the context of the interaction system 100 or may selectively be applied to certain types of relationships.

[0069] The profile data 302 stores multiple types of profile data about a particular entity. The profile data 302 may be selectively used and presented to other users of the interaction system 100 based on privacy settings specified by a particular entity. Where the entity is an individual, the profile data 302 includes, for example, a username, telephone number, address, settings (e.g., notification and privacy settings), and a user-selected avatar

representation (or collection of such avatar representations). A particular user may then selectively include one or more of these avatar representations within the content of messages communicated via the interaction system 100 and on map interfaces displayed by interaction clients 104 to other users. The collection of avatar representations may include "status avatars," which present a graphical representation of a status or activity that the user may select to communicate at a particular time.

[0070] Where the entity is a group, the profile data 302 for the group may similarly include one or more avatar representations associated with the group, in addition to the group name, members, and various settings (e.g., notifications) for the relevant group.

[0071] The database 304 also stores augmentation data, such as overlays or filters, in an augmentation table 312. The augmentation data is associated with and applied to videos (for which data is stored in a video table 314) and images (for which data is stored in an image table 316).

[0072] Filters, in some examples, are overlays that are displayed as overlaid on an image or video during presentation to a recipient user. Filters may be of various types, including user-selected filters from a set of filters presented to a sending user by the interaction client 104 when the sending user is composing a message. Other types of filters include geolocation filters (also known as geo-filters), which may be presented to a sending user based on geographic location. For example, geolocation filters specific to a neighborhood or special location may be presented within a user interface by the interaction client 104, based on geolocation information determined by a Global Positioning System (GPS) unit of the user system 102.

[0073] Another type of filter is a data filter, which may be selectively presented to a sending user by the interaction client 104 based on other inputs or information gathered by the user system 102 during the message creation process. Examples of data filters include current temperature at a specific location, a current speed at which a sending user is traveling, battery life for a user system 102, or the current time.

[0074] Other augmentation data that may be stored within the image table 316 includes AR content items (e.g., corresponding to applying "lenses" or AR experiences). An AR content item may be a real-time special effect and sound that may be added to an image or a video.

[0075] A collections table 318 stores data regarding collections of messages and associated image, video, or audio data, which are compiled into a collection (e.g., a story or a gallery). The creation of a particular collection may be initiated by a particular user (e.g., each user for which a record is maintained in the entity table 308). A user may create a "personal

story" in the form of a collection of content that has been created and sent/broadcast by that user. To this end, the user interface of the interaction client 104 may include an icon that is user-selectable to enable a sending user to add specific content to his or her personal story.

[0076] A collection may also constitute a "live story," which is a collection of content from multiple users that is created manually, automatically, or using a combination of manual and automatic techniques. For example, a "live story" may constitute a curated stream of user-submitted content from various locations and events. Users whose client devices have location services enabled and are at a common location event at a particular time may, for example, be presented with an option, via a user interface of the interaction client 104, to contribute content to a particular live story. The live story may be identified to the user by the interaction client 104, based on his or her location. The end result is a "live story" told from a community perspective.

[0077] A further type of content collection is known as a "location story," which enables a user whose user system 102 is located within a specific geographic location (e.g., on a college or university campus) to contribute to a particular collection. In some examples, a contribution to a location story may employ a second degree of authentication to verify that the end-user belongs to a specific organization or other entity (e.g., is a student on the university campus).

[0078] As mentioned above, the video table 314 stores video data that, in some examples, is associated with messages for which records are maintained within the message table 306. Similarly, the image table 316 stores image data associated with messages for which message data is stored in the entity table 308. The entity table 308 may associate various augmentations from the augmentation table 312 with various images and videos stored in the image table 316 and the video table 314.

[0079] The databases 304 also include trained machine learning techniques 307 that stores parameters of one or more machine learning models that have been trained during training of the stationary XR system 500. For example, trained machine learning techniques 307 stores the trained parameters of one or more artificial neural network machine learning models or techniques.

DATA COMMUNICATIONS ARCHITECTURE

[0080] **FIG. 4** is a schematic diagram illustrating a structure of a message 400, according to some examples, generated by an interaction client 104 for communication to a further interaction client 104 via the interaction servers 124. The content of a particular message

400 is used to populate the message table 306 stored within the database 304, accessible by the interaction servers 124. Similarly, the content of a message 400 is stored in memory as "in-transit" or "in-flight" data of the user system 102 or the interaction servers 124. A message 400 is shown to include the following example components:

- Message identifier 402: a unique identifier that identifies the message 400.
- Message text payload 404: text, to be generated by a user via a user interface of the user system 102, and that is included in the message 400.
- Message image payload 406: image data, captured by a camera component of a user system 102 or retrieved from a memory component of a user system 102, and that is included in the message 400. Image data for a sent or received message 400 may be stored in the image table 316.
- Message video payload 408: video data, captured by a camera component or retrieved from a memory component of the user system 102, and that is included in the message 400. Video data for a sent or received message 400 may be stored in the image table 316.
- Message audio payload 410: audio data, captured by a microphone or retrieved from a memory component of the user system 102, and that is included in the message 400.
- Message augmentation data 412: augmentation data (e.g., filters, stickers, or other annotations or enhancements) that represents augmentations to be applied to message image payload 406, message video payload 408, or message audio payload 410 of the message 400. Augmentation data for a sent or received message 400 may be stored in the augmentation table 312.
- Message duration parameter 414: parameter value indicating, in seconds, the amount of time for which content of the message (e.g., the message image payload 406, message video payload 408, message audio payload 410) is to be presented or made accessible to a user via the interaction client 104.
- Message geolocation parameter 416: geolocation data (e.g., latitudinal and longitudinal coordinates) associated with the content payload of the message. Multiple message geolocation parameter 416 values may be included in the payload, each of these parameter values being associated with respect to content items included in the content (e.g., a specific image within the message image payload 406, or a specific video in the message video payload 408).
- Message story identifier 418: identifier values identifying one or more content collections (e.g., "stories" identified in the collections table 318) with which a particular content item in the message image payload 406 of the message 400 is associated. For

example, multiple images within the message image payload 406 may each be associated with multiple content collections using identifier values.

- Message tag 420: each message 400 may be tagged with multiple tags, each of which is indicative of the subject matter of content included in the message payload. For example, where a particular image included in the message image payload 406 depicts an animal (e.g., a lion), a tag value may be included within the message tag 420 that is indicative of the relevant animal. Tag values may be generated manually, based on user input, or may be automatically generated using, for example, image recognition.
- Message sender identifier 422: an identifier (e.g., a messaging system identifier, email address, or device identifier) indicative of a user of the user system 102 on which the message 400 was generated and from which the message 400 was sent.
- Message receiver identifier 424: an identifier (e.g., a messaging system identifier, email address, or device identifier) indicative of a user of the user system 102 to which the message 400 is addressed.

[0081] The contents (e.g., values) of the various components of message 400 may be pointers to locations in tables within which content data values are stored. For example, an image value in the message image payload 406 may be a pointer to (or address of) a location within an image table 316. Similarly, values within the message video payload 408 may point to data stored within an image table 316, values stored within the message augmentation data 412 may point to data stored in an augmentation table 312, values stored within the message story identifier 418 may point to data stored in a collections table 318, and values stored within the message sender identifier 422 and the message receiver identifier 424 may point to user records stored within an entity table 308.

STATIONARY XR SYSTEM

[0080] **FIG. 5** is a block diagram showing an example stationary XR system 500, according to some examples. The stationary XR system 500 can include an image access component 510, an XR experience component 520, an object tracking network 530, and/or an image modification component 540. Together these components of the stationary XR system 500 can receive, from a camera of a stationary device (e.g., user system 102), an image depicting a real-world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment. The stationary XR system 500 can analyze the image using a machine learning model to predict tracking information for the real-world object, the machine learning model trained based on a plurality of training images depicting real-world objects in the specified field

of view of the real-world environment and corresponding ground-truth tracking information for the real-world objects. The stationary XR system 500 selects an XR experience from a plurality of XR experiences and overlays one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

[0081] The stationary XR system 500 can be implemented by one or more devices (e.g., a display screen, a camera, and/or processors) that are housed or partially enclosed in a secure housing or enclosure. The housing of the stationary XR system 500 can be physically secured (e.g., bolted) to a physical surface, such as the ground, wall, and/or ceiling. The camera of the stationary XR system 500 can be oriented and configured with camera parameters that are best suited for providing a specific XR experience in the real-world environment in which the housing of the stationary XR system 500 is secured. By securing the stationary XR system 500 in this manner, the stationary XR system 500 cannot be tampered with or modified by an end user after the stationary XR system 500 is activated for user access and cannot be moved around. This ensures that the camera continues capturing images of a real-world environment in a specific field of view that is best suited for providing a particular XR experience which the stationary XR system 500 was trained to provide.

[0082] For example, the image access component 510 can implement one or more cameras and can access a real-time or previously stored image or video from the user system 102. The image can depict a real-world object in a specific field of view of the one or more cameras. The specific field of view can represent a specific physical portion of the real-world environment in which the stationary XR system 500 is physically secured. In some cases, the image access component 510 can select a frame or collection of frames from the accessed image or video for use in providing an XR experience. During training of the stationary XR system 500, the image access component 510 accesses a database of training data to retrieve images depicting real-world objects in the same real-world environment (or simulated real-world environment) as that in which the stationary XR system 500 is stationed (e.g., physically attached). The training data images also include corresponding ground truth tracking information for the depicted real-world objects to train one or more machine learning models of the stationary XR system 500.

[0083] During operation (e.g., after being trained), the image access component 510 provides the image depicting the real-world object to the object tracking network 530. The object tracking network 530 includes a machine learning model that is trained to process a received image and generate tracking information for the real-world object depicted in the received image.

[0084] To train the object tracking network 530, the image access component 510 retrieves a first training image that depicts a real-world object (e.g., a foot, face, landscape, sky, or any other

real-world object that is expected to be received or viewed in the real-world environment in which the stationary XR system 500 is stationed). The object tracking network 530 also receives the corresponding ground-truth tracking information of the real-world object. The object tracking network 530 analyzes or processes the first training image using a machine learning model to estimate tracking information for the real-world object depicted in the first training image. The object tracking network 530 computes a deviation between the estimated tracking information and the ground-truth tracking information. This deviation is used to generate a loss, which is used to update one or more parameters of the machine learning model. In this way, the machine learning model of the object tracking network 530 is trained to provide or estimate tracking information for real-world objects in a specific real-world environment and using a camera with certain camera parameters configured for the real-world environment in which the stationary XR system 500 is stationed or physically attached. This provides improved tracking information over other tracking systems that are trained using various training images depicting objects in various real-world environments as such systems generate more general tracking information. The object tracking network 530 iterates over a number of additional training images in this manner and updating the parameters of the machine learning model until a stopping criterion is reached. At that point, the machine learning model parameters are stored and can be used to generate tracking information for a new image depicting a new real-world object in the same or similar real-world environment as that depicted in the training images.

[0085] For example, the object tracking network 530 can provide tracking information for a foot that is placed in a field of view of the camera pointing in a particular preconfigured and stationary direction. In this case, the object tracking network 530 can be placed in a housing that is situated in a particular location in a shoe store or other clothing store. The object tracking network 530 can receive training images that depict various feet in the same particular location or in a simulated environment corresponding to the particular location. In this way, all of the training images can include the same real-world background as the background that will appear when a new foot is depicted in an image captured by the camera. This provides true-to-life tracking information for a foot depicted in an image and provides a more realistic presentation of virtual elements in the image.

[0082] For example, the object tracking network 530 can provide the tracking information to the XR experience component 520. The XR experience component 520 can receive input that selects a particular XR experience. Namely, the input can be received from a user that selects a particular virtual element, such as a virtual shoe, from a plurality of virtual elements. In some cases, the XR experience can be selected ahead of time and cannot be altered by any user without physically tampering with internal components of the stationary

XR system 500. Namely, a store operator can launch a particular XR experience that is associated with a particular virtual element or various virtual shoes. Once selected, the store operator can lock the stationary XR system 500 to prevent anyone else from changing the XR experience. In some cases, the XR experience can be configured to change over time, such as according to a schedule, to present different virtual elements or virtual shoes as objects in an image that depicts a foot captured by the camera.

[0083] Now, when a user inserts a foot into the dedicated field of view of the camera of the image access component 510, the XR experience component 520 selects and positions virtual elements corresponding to the selected XR experience in view of an image that depicts the user's foot. The XR experience component 520 provides this information to the image modification component 540. The image modification component 540 overlays the virtual element over the foot based on the positioning and tracking information and continues to dynamically adjust the position of the virtual element as the foot moves around the field of view.

[0084] In some cases, the XR experience component 520 receives a predetermined and pre-generated mask for the real-world environment or field of view of the camera of the stationary XR system 500. The mask can be generated automatically once and stored in association with the XR experience component 520 and/or object tracking network 530. The mask can define pixels of the image captured by the image access component 510 that are outside of the target portion of the image in which the real-world object is depicted. Namely, the mask can define regions that are depicted in the image that correspond to a background or portions that are outside of a target physical space in which the real-world object is expected to be placed. The image modification component 540 can obscure or replace any pixels that fall outside of the target physical space based on the mask with specified pixel values and/or a different background. Because the mask is pre-generated, there is no need to perform any real-time in-painting processes, which saves a great deal of time and resources. In some cases, the mask can be adjusted by a system operator or store operator and refined to designate the specific region that is the target physical space.

[0085] For example, the object tracking network 530 can provide tracking information for a face that is placed in a field of view of the camera pointing in a particular preconfigured and stationary direction. In this case, the object tracking network 530 can be placed in a housing that is situated in a particular location in a makeup store, earring store, and/or other fashion item store. The object tracking network 530 can receive training images that depict various faces in the same particular location or in a simulated environment corresponding to the particular location. In this way, all of the training images can include the same real-

world background as the background that will appear when a new face is depicted in an image captured by the camera. This provides true-to-life tracking information for a face depicted in an image and provides a more realistic presentation of virtual elements (e.g., virtual makeup and/or virtual fashion items) in the image.

[0086] For example, the object tracking network 530 can provide the tracking information to the XR experience component 520. The XR experience component 520 can receive input that selects a particular XR experience. Namely, the input can be received from a user that selects a particular virtual element, such as virtual makeup, from a plurality of virtual elements. In some cases, the XR experience can be selected ahead of time and cannot be altered by any user without physically tampering with internal components of the stationary XR system 500. Namely, a store operator can launch a particular XR experience that is associated with a particular virtual element or various virtual makeup. Once selected, the store operator can lock the stationary XR system 500 to prevent anyone else from changing the XR experience. In some cases, the XR experience can be configured to change over time, such as according to a schedule, to present different virtual elements or virtual makeup or virtual fashion items as objects in an image that depicts a face captured by the camera.

[0087] Now, when a user places a face into the dedicated field of view of the camera of the image access component 510, the XR experience component 520 selects and positions virtual elements corresponding to the selected XR experience in view of an image that depicts the user's face. The XR experience component 520 provides this information to the image modification component 540. The image modification component 540 overlays the virtual element over the face based on the positioning and tracking information and continues to dynamically adjust the position of the virtual element as the face moves around the field of view. In some cases, the image modification component 540 can activate a secondary camera, such as a rear-facing camera of the stationary XR system 500. The image modification component 540 can present images from the secondary camera in dedicated portions of the image that has been modified to present the virtual elements.

[0086] For example, the object tracking network 530 can provide tracking information for a real-world landscape, sky, city landscape, and/or buildings that is placed in a field of view of the camera pointing in a particular preconfigured and stationary direction. In this case, the object tracking network 530 can be placed in a housing that is situated in a particular location in a building, store, home, or physical space to appear as a virtual window. The object tracking network 530 can receive training images that depict various real-world landscape, sky, city landscape, and/or buildings in the same particular location or in a simulated environment corresponding to the particular location. In this way, all of the training images can include the same real-world background as the background that will appear when a new real-world landscape, sky, city landscape, and/or buildings is depicted in an image captured by the camera. This provides true-to-life tracking information for a real-world landscape, sky, city landscape, and/or buildings depicted in an image and provides a more realistic presentation of virtual elements (e.g., virtual overlays, virtual labels, and/or virtual information) in the image.

[0088] For example, the object tracking network 530 can provide the tracking information to the XR experience component 520. The XR experience component 520 can receive input that selects a particular XR experience. Namely, the input can be received from a user that selects a particular virtual element, such as virtual information, from a plurality of virtual elements. In some cases, the XR experience can be selected ahead of time and cannot be altered by any user without physically tampering with internal components of the stationary XR system 500.

[0089] Now, when the stationary XR system 500 detects the real-world landscape, sky, city landscape, and/or buildings in the dedicated field of view of the camera of the image access component 510, the XR experience component 520 selects and positions virtual elements corresponding to the selected XR experience in view of the image. The XR experience component 520 provides this information to the image modification component 540. The image modification component 540 overlays the virtual element over the real-world landscape, sky, city landscape, and/or buildings based on the positioning and tracking information and continues to dynamically adjust the position of the virtual element. In some cases, the image modification component 540 can receive accelerometer or movement information from a mobile device of a user that is within a threshold proximity of the stationary XR system 500. The image modification component 540 can adjust the positioning of the virtual elements in the images based on the accelerometer or movement information. This allows the stationary XR system 500 to act as an XR window that a user can interact with to obtain further information about various real-world objects that are depicted in the image. In some cases, the stationary XR system 500 is physically hung on a

solid wall and is in communication with a camera that is placed on the outside of the wall facing the real-world environment exterior to the wall. This provides the illusion that the images presented by the stationary XR system 500 are real and gives users the impression that the stationary XR system 500 is a window through which the real-world environment exterior to the wall can be seen.

[0090] **FIG. 6** is a diagrammatic representation of a stationary XR shoe mirror 600 and XR shoe mirror assembly 601, in accordance with some examples. The stationary XR shoe mirror 600 can include a housing 610, a user system 620 (e.g., user system 102) that includes a display device 622, a camera 630, and a base 640. The housing 610 can be arranged to be physically placed and attached in a particular physical location or a real-world environment, such as a physical shoe store. The housing 610 can be tamper proof to prevent access to any components inside of the housing 610 without a key. In this way, an individual user can only touch the display device 622 and place a real-world object, such as a real-world foot, in a field of view of the camera 630.

[0091] In some cases, the camera 630 can be configured with camera parameters that are optimized to generate an XR experience in the real-world environment in which the housing 610 is placed. For example, the focal length, frame rate, and other lens attributes and camera attributes can be fixed by a store operator before locking the housing 610, such as during training of machine learning models that estimate tracking information. After locking the housing 610, the camera parameters can be fixed and cannot be adjusted by any other user without a key to the housing 610. The camera 630 can be angled directly or substantially directly down towards the base 640. The base 640 defines the physical field of view of the camera 630 in which the real-world object is tracked by the object tracking network 530. Any pixels captured by the camera 630 outside of the base 640 can be masked by a specified mask that is generated by the stationary XR system 500.

[0092] The user system 620 can be placed in a portion of the housing 610 that is angled relative to the base 640 and a vertical segment of the housing 610. The base 640 can be attached or be part of the housing 610 and can be secured to a countertop or floor of a physical environment, such as using one or more bolts. The base 640 can be attached to a bottom portion of the vertical segment of the housing 610. A top of the vertical segment of the housing 610 can be attached to a screen-holding portion of the housing 610. The screen-holding portion can be angled (e.g., placed at a 45 degree angle) from the point at which the screen-holding portion is attached to the housing 610 and relative to the vertical segment of the housing 610. In this way, the display device 622 of the user system 620 can be inserted into the screen-holding portion and secured inside of the screen-holding portion to prevent

tampering with any user interface portions of the user system 620. In some cases, the user system 620 can be secured to the housing 610 via one or more brackets 612. The user system 620 can be bolted to the one or more brackets 612 to prevent movement of the user system 620 and/or the camera 630.

[0093] In some examples, a user can insert their foot and/or rest their foot on the base 640. In response to detecting the foot in an image captured by the camera 630 and/or in response to detecting pressure on the base 640, the preconfigured XR experience can be launched. Once the XR experience is launched, one or more virtual shoe elements can be overlaid on the foot that is depicted in the image captured by the camera 630 based on tracking information predicted by the machine learning models implemented by the user system 620 and/or processors housed within the base 640 coupled to the user system 620. In some examples, one or more wires can be routed via the vertical portion of the housing 610 from the one or more processors 650 housed inside of the base 640.

[0094] **FIG. 7** is a diagrammatic representation of a stationary XR vanity mirror 700, in accordance with some examples. The stationary XR vanity mirror 700 can include a housing 710, a user system 720 (e.g., user system 102) that includes a display device 722, a light element 730, a camera, a swivel 740, and a base 750. The housing 710 can be arranged to be physically placed and attached in a particular physical location or a real-world environment, such as a physical makeup or fashion item store. The housing 710 can be tamper proof to prevent access to any components inside of the housing 710 without a key. In this way, an individual user can only touch the display device 722 and place a real-world object, such as a real-world face, in a field of view of the camera of the user system 720.

[0095] In some cases, the camera of the user system 720 can be configured with camera parameters that are optimized to generate an XR experience in the real-world environment in which the housing 710 is placed. For example, the focal length, frame rate, and other lens attributes and camera attributes can be fixed by a store operator before locking the housing 710, such as during training of machine learning models that estimate tracking information of a face. After locking the housing 710, the camera parameters can be fixed and cannot be adjusted by any other user without a key to the housing 710. The camera of the user system 720 can be oriented directly or substantially outwards to point along a surface normal of the user system 720 and perpendicular to the base 750. The housing 710 can be rotated in a fixed or adjustable manner to point the camera in a particular direction about an axis of rotation of the swivel 740. The swivel 740 can be locked in a particular direction to define the physical field of view of the camera of the user system 720 in which the real-world object is tracked by the object tracking network 530.

[0096] The user system 720 can be placed in a portion of the housing 710. The base 750 can be attached or be part of the housing 710. The base 750 can be attached to a bottom portion of the housing 710 and can be physically secured to a countertop or floor of a physical environment, such as using one or more bolts. The display device 722 of the user system 720 can be inserted into a screen-holding portion of the housing 710 and secured inside of the screen-holding portion to prevent tampering with any user interface portions of the user system 720.

[0097] The light element 730 can border the user system 720. The light element 730 can include one or more light emitting diodes or other physical light generation device, such as a light bulb. The light element 730 can communicate with the user system 720 via one or more wires that can run through a portion of the housing 710. The user system 720 can receive information about lighting conditions of the real-world environment. The user system 720 can apply one or more machine learning models associated with a selected XR experience to predict lighting adjustments to make to optimize the XR experience. In some cases, the output of the one or more machine learning models can be used to alter the light temperature, color, or intensity of the light element 730. Different XR experiences can be associated with different light temperature, color, or intensity which can be dynamically controlled by the user system 720.

[0098] In some examples, a user can place their face within view of the camera of the user system 720. For example, the user can rotate the user system 720 using the swivel 740. In response to detecting the face in an image captured by the camera, the preconfigured XR experience can be launched. Once the XR experience is launched, one or more virtual makeup and/or fashion item elements can be overlaid on the face that is depicted in the image captured by the camera based on tracking information predicted by the machine learning models implemented by the user system 720 and/or processors housed within the base 750 coupled to the user system 720. In some examples, one or more wires can be routed via the housing 710 from one or more processors housed inside of the base 750 to provide the tracking information. Namely, the user system 720 can provide to the processors in the base 750 features of the image depicting the face and the processors in the base 750 can generate tracking information and overlaid virtual elements for presentation on the display device 722. The processors can communicate the tracking information and/or the overlaid images via the wires in the housing 710 to the display device 722.

[0099] **FIG. 8** is a diagrammatic representation of a stationary XR window 800, in accordance with some examples. The stationary XR window 800 can include a housing 810, a user system 812 (e.g., user system 102), and a camera (not shown) that are mounted to a

surface, such as a wall of a building, a floor, a ceiling, and so forth. The housing 810 can be arranged to be physically placed and attached in a particular physical location or a real-world environment, such as a wall of a building. The housing 810 can be tamper proof to prevent access to any components inside of the housing 810 without a key. In this way, an individual user can only touch the user system 812 to view and interact with virtual objects displayed over a real-world object presented by the user system 812. In some cases, the real-world object can be a landscape 820 (which can include buildings, mountains, and/or a sky) that is exterior to the building in which the housing 810 is placed. The camera of the user system 812 can be directed towards the landscape 820. In some cases, the housing 810 is placed on an interior wall of the building that does not abut an exterior of the building. In such cases, the camera of the user system 812 can be placed outside of the building and pointing towards the landscape 820. The images captured by the camera can be routed to the user system 812 for display to give the illusion that the content being viewed through a screen of the user system 812 is a window into the landscape 820.

[0100] In some cases, the camera of the user system 812 can be configured with camera parameters that are optimized to generate an XR experience in the real-world environment in which the housing 810 is placed. For example, the focal length, frame rate, and other lens attributes and camera attributes can be fixed by an operator before locking the housing 810, such as during training of machine learning models that estimate tracking information of a landscape visible by the camera of the user system 812. After locking the housing 810, the camera parameters can be fixed and cannot be adjusted by any other user without a key to the housing 810.

[0101] The user system 812 can be placed in a portion of the housing 810. The housing 810 can be attached to or physically secured to a surface of the building, such as using one or more bolts or framing elements. Once the XR experience is launched by the user system 812, one or more virtual elements (e.g., virtual labels or virtual weather or virtual enhancements) can be overlaid on the landscape 820 that is depicted in the image captured by the camera based on tracking information predicted by the machine learning models implemented by the user system 812 and/or processors housed within a base of the housing 810 coupled to the user system 812. The XR experience can be continuously activated and may not be turned OFF by anyone except the operator. In this way, any user that walks by the user system 812 can see the landscape 820 overlaid with the virtual elements.

[0102] In some cases, a user 830 can hold a mobile device 840. The mobile device 840 can communicate with the user system 812, such as via a short-range communication protocol (e.g., Bluetooth). The mobile device 840 can provide movement and/or accelerometer data

to the user system 812. Based on such information, the user system 812 provides various adjustments to the virtual elements displayed over the landscape 820. This enables the user 830 to interact with one or more of the displayed virtual elements.

[0103] **FIG. 9** is a diagrammatic representation of a general stationary XR system 900, in accordance with some examples. The stationary XR system 900 includes a display 920, a housing 910, and a camera 930 oriented to capture images from a preconfigured field of view 940. As mentioned above, the camera 930 can provide images from the preconfigured field of view 940 to a processor of the display 920. The processor can implement one or more machine learning models to generate or predict tracking information for one or more real-world elements depicted in the received images. The machine learning models may be trained based on training data collected using the same or simulated preconfigured field of view 940. Based on such predicted tracking information, the display 920 provides virtual elements overlaid on the received images.

[0104] **FIG. 10** is a flowchart of a process 1000 performed by the stationary XR system 500, in accordance with some examples. Although the flowchart can describe the operations as a sequential process, many of the operations can be performed in parallel or concurrently. In addition, the order of the operations may be re-arranged. A process is terminated when its operations are completed. A process may correspond to a method, a procedure, and the like. The steps of methods may be performed in whole or in part, may be performed in conjunction with some or all of the steps in other methods, and may be performed by any number of different systems or any portion thereof, such as a processor included in any of the systems.

[0105] At operation 1001, the stationary XR system 500 (e.g., a user system 102 or a server) receives, from a camera of a stationary device, an image depicting a real-world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment.

[0106] At operation 1002, the stationary XR system 500 analyzes the image using a machine learning model to predict tracking information for the real-world object, the machine learning model trained based on a plurality of training images depicting real-world objects in the specified field of view of the real-world environment and corresponding ground truth tracking information for the real-world objects.

[0107] At operation 1003, the stationary XR system 500 selects an extended reality (XR) experience from a plurality of XR experiences.

[0108] At operation 1004, the stationary XR system 500 overlays one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

[0109] In some examples, the stationary XR system 500 includes an XR shoe mirror. In some cases, the XR shoe mirror includes a display screen that is placed at a specified distance above the specified field of view, wherein the XR shoe mirror includes a physical area attached to the display screen that defines the specified field of view, and wherein the XR shoe mirror includes the camera pointed towards the physical area. In some cases, the display screen is mounted on a physical surface that is tilted at an angle relative to the physical area, and wherein the camera is integrated into the physical surface and is directed such that a surface normal of the camera is perpendicular to the physical area.

[0110] In some examples, the camera points down towards the physical area. In some cases, the stationary XR system 500 sets a focal length and one or more camera parameters to a specified value based on a distance between the camera and the physical area and one or more external conditions of the XR shoe mirror. The stationary XR system 500 identifies an area outside of the physical area of the XR shoe mirror and generates a mask based on the area outside of the physical area of the XR shoe mirror. The stationary XR system 500 occludes pixels of the image that fall within the identified area outside of the physical area of the XR shoe mirror while overlaying the one or more XR elements on the image. In some cases, the mask is generated when the XR shoe mirror is manufactured or placed within the real-world environment.

[0111] In some examples, the real-world object includes a foot, and the one or more XR elements include one or more virtual shoes.

[0112] In some examples, the stationary XR system 500 includes an XR vanity mirror. In some cases, the real-world object includes a face, and the one or more XR elements include one or more virtual fashion items or virtual makeup.

[0113] In some cases, the stationary XR system 500 accesses lighting conditions of the real-world environment and analyzes the lighting conditions with the machine learning model to predict a modification to one or more physical light parameters of the device and one or more light parameters of the one or more XR elements associated with the XR experience. The stationary XR system 500 automatically adjusts physical light that surrounds a screen on which the modified image is displayed based on the predicted one or more physical light parameters. In some cases, the one or more physical light parameters include a light temperature, color, or intensity.

[0114] In some examples, the stationary XR system 500 includes an XR window placed on an opening of a building or on an interior wall of the building. In such cases, the real-world object includes a sky or city landscape, and the one or more XR elements include one or more labels or virtual sky elements. In some examples, the stationary XR system 500 sets a focal length and one or more camera parameters to a specified value based on a distance between the camera and the sky or city landscape.

[0115] In some cases, the stationary XR system 500 receives, by the XR window, physical movement information of a mobile device that is external to the device and adjusts one or more visual properties of the one or more XR elements displayed by the XR window based on the physical movement information of the mobile device.

[0116] In some examples, the stationary XR system 500 trains the machine learning model by performing training operations including: selecting a first training image from the plurality of training images, the first training image depicting an individual real-world object having an associated individual ground-truth tracking information, the first training image being captured by the stationary device and depicting the individual real-world object in the field of view of the real-world environment; analyzing the first training image using the machine learning model to predict estimated tracking information for the individual real-world object; and generating loss based on a deviation between the estimated tracking information for the individual real-world object and the individual ground-truth tracking information of the individual real-world object; and updating one or more parameters of the machine learning model based on the loss.

EXAMPLES

[0117] Example 1. A method comprising: receiving, from a camera of a stationary device, an image depicting a real-world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment; analyzing the image using a machine learning model to predict tracking information for the real-world object, the machine learning model trained based on a plurality of training images depicting real-world objects in the specified field of view of the real-world environment and corresponding ground-truth tracking information for the real-world objects; selecting an extended reality (XR) experience from a plurality of XR experiences; and overlaying one or more XR

elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

[0118] Example 2. The method of Example 1, wherein the stationary device comprises an XR shoe mirror.

[0119] Example 3. The method of Example 2, wherein the XR shoe mirror includes a display screen that is placed at a specified distance above the specified field of view, wherein the XR shoe mirror includes a physical area attached to the display screen that defines the specified field of view, and wherein the XR shoe mirror includes the camera pointed towards the physical area.

[0120] Example 4. The method of Example 3, wherein the display screen is mounted on a physical surface that is tilted at an angle relative to the physical area, and wherein the camera is integrated into the physical surface and is directed such that a surface normal of the camera is perpendicular to the physical area.

[0121] Example 5. The method of any one of Examples 3-4, wherein the camera points down towards the physical area.

[0122] Example 6. The method of any one of Examples 3-5, further comprising setting a focal length and one or more camera parameters to a specified value based on a distance between the camera and the physical area and one or more external conditions of the XR shoe mirror.

[0123] Example 7. The method of any one of Examples 3-6, further comprising: identifying an area outside of the physical area of the XR shoe mirror; and generating a mask based on the area outside of the physical area of the XR shoe mirror; and occluding pixels of the image that fall within the identified area outside of the physical area of the XR shoe mirror while overlaying the one or more XR elements on the image.

[0124] Example 8. The method of Example 7, wherein the mask is generated when the XR shoe mirror is manufactured or placed within the real-world environment.

[0125] Example 9. The method of any one of Examples 2-8, wherein the real-world object comprises a foot, and wherein the one or more XR elements comprise one or more virtual shoes.

[0126] Example 10. The method of any one of Examples 1-9, wherein the stationary device comprises an XR vanity mirror.

[0127] Example 11. The method of Example 10, wherein the real-world object comprises a face, and wherein the one or more XR elements comprise one or more virtual fashion items or virtual makeup.

[0128] Example 12. The method of any one of Examples 10-11, further comprising: accessing lighting conditions of the real-world environment; analyzing the lighting conditions with the machine learning model to predict a modification to one or more physical light parameters of the stationary device and one or more light parameters of the one or more XR elements associated with the XR experience; and automatically adjusting physical light that surrounds a screen on which the modified image is displayed based on the predicted one or more physical light parameters.

[0129] Example 13. The method of Example 12, wherein the one or more physical light parameters comprise a light temperature, color, or intensity.

[0130] Example 14. The method of any one of Examples 1-13, wherein the stationary device comprises an XR window placed on an opening of a building or on an interior wall of the building.

[0131] Example 15. The method of Example 14, wherein the real-world object comprises a sky or city landscape, and wherein the one or more XR elements comprise one or more labels or virtual sky elements.

[0132] Example 16. The method of Example 15, further comprising setting a focal length and one or more camera parameters to a specified value based on a distance between the camera and the sky or city landscape.

[0133] Example 17. The method of any one of Examples 14-16, further comprising: receiving, by the XR window, physical movement information of a mobile device that is external to the stationary device; and adjusting one or more visual properties of the one or more XR elements displayed by the XR window based on the physical movement information of the mobile device.

[0134] Example 18. The method of any one of Examples 1-17, further comprising training the machine learning model by performing training operations comprising: selecting a first training image from the plurality of training images, the first training image depicting an individual real-world object having an associated individual ground-truth tracking information, the first training image being captured by the stationary device and depicting the individual real-world object in the field of view of the real-world environment; analyzing the first training image using the machine learning model to predict estimated tracking information for the individual real-world object; and generating loss based on a deviation between the estimated tracking information for the individual real-world object and the individual ground-truth tracking information of the individual real-world object; and updating one or more parameters of the machine learning model based on the loss.

[0135] Example 19. A system comprising: at least one processor; and at least one memory component having instructions stored thereon that, when executed by the at least one processor, cause the at least one processor to perform operations comprising: receiving, from a camera of a stationary device, an image depicting a real-world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment; analyzing the image using a machine learning model to predict tracking information for the real-world object, the machine learning model trained based on a plurality of training images depicting real-world objects in the specified field of view of the real-world environment and corresponding ground-truth tracking information for the real-world objects; selecting an extended reality (XR) experience from a plurality of XR experiences; and overlaying one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

[0136] Example 20. A non-transitory computer-readable storage medium having stored thereon instructions that, when executed by at least one processor, cause the at least one processor to perform operations comprising: receiving, from a camera of a stationary device, an image depicting a real-world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment; analyzing the image using a machine learning model to predict tracking information for the real-world object, the machine learning model trained based on a plurality of training images depicting real-world objects in the specified field of view of the real-world environment and corresponding ground-truth tracking information for the real-world objects; selecting an extended reality (XR) experience from a plurality of XR experiences; and overlaying one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

MACHINE ARCHITECTURE

[0137] **FIG. 11** is a diagrammatic representation of a machine 1100 within which instructions 1102 (e.g., software, a program, an application, an applet, an app, or other executable code) for causing the machine 1100 to perform any one or more of the methodologies discussed herein may be executed. For example, the instructions 1102 may cause the machine 1100 to execute any one or more of the methods described herein. The instructions 1102 transform the general, non-programmed machine 1100 into a particular machine 1100 programmed to carry out the described and illustrated functions in the manner described. The machine 1100 may operate as a standalone device or may be coupled (e.g.,

networked) to other machines. In a networked deployment, the machine 1100 may operate in the capacity of a server machine or a client machine in a server-client network environment, or as a peer machine in a peer-to-peer (or distributed) network environment. The machine 1100 may comprise, but not be limited to, a server computer, a client computer, a personal computer (PC), a tablet computer, a laptop computer, a netbook, a set-top box (STB), a personal digital assistant (PDA), an entertainment media system, a cellular telephone, a smartphone, a mobile device, a wearable device (e.g., a smartwatch), a smart home device (e.g., a smart appliance), other smart devices, a web appliance, a network router, a network switch, a network bridge, or any machine capable of executing the instructions 1102, sequentially or otherwise, that specify actions to be taken by the machine 1100. Further, while a single machine 1100 is illustrated, the term "machine" shall also be taken to include a collection of machines that individually or jointly execute the instructions 1102 to perform any one or more of the methodologies discussed herein. The machine 1100, for example, may comprise the user system 102 or any one of multiple server devices forming part of the interaction server system 110. In some examples, the machine 1100 may also comprise both client and server systems, with certain operations of a particular method or algorithm being performed on the server-side and with certain operations of the particular method or algorithm being performed on the client-side.

[0138] The machine 1100 may include processors 1104, memory 1106, and input/output (I/O) components 1108, which may be configured to communicate with each other via a bus 1110. In an example, the processors 1104 (e.g., a Central Processing Unit (CPU), a Reduced Instruction Set Computing (RISC) Processor, a Complex Instruction Set Computing (CISC) Processor, a Graphics Processing Unit (GPU), a Digital Signal Processor (DSP), an Application Specific Integrated Circuit (ASIC), a Radio-Frequency Integrated Circuit (RFIC), another processor, or any suitable combination thereof) may include, for example, a processor 1112 and a processor 1114 that execute the instructions 1102. The term "processor" is intended to include multi-core processors that may comprise two or more independent processors (sometimes referred to as "cores") that may execute instructions contemporaneously. Although FIG. 11 shows multiple processors 1104, the machine 1100 may include a single processor with a single-core, a single processor with multiple cores (e.g., a multi-core processor), multiple processors with a single core, multiple processors with multiples cores, or any combination thereof.

[0139] The memory 1106 includes a main memory 1116, a static memory 1118, and a storage unit 1120, all accessible to the processors 1104 via the bus 1110. The main memory 1106, the static memory 1118, and storage unit 1120 store the instructions 1102 embodying

any one or more of the methodologies or functions described herein. The instructions 1102 may also reside, completely or partially, within the main memory 1116, within the static memory 1118, within machine-readable medium 1122 within the storage unit 1120, within at least one of the processors 1104 (e.g., within the processor's cache memory), or any suitable combination thereof, during execution thereof by the machine 1100.

[0140] The I/O components 1108 may include a wide variety of components to receive input, provide output, produce output, transmit information, exchange information, capture measurements, and so on. The specific I/O components 1108 that are included in a particular machine will depend on the type of machine. For example, portable machines such as mobile phones may include a touch input device or other such input mechanisms, while a headless server machine will likely not include such a touch input device. It will be appreciated that the I/O components 1108 may include many other components that are not shown in FIG. 11. In various examples, the I/O components 1108 may include user output components 1124 and user input components 1126. The user output components 1124 may include visual components (e.g., a display such as a plasma display panel (PDP), a light-emitting diode (LED) display, a liquid crystal display (LCD), a projector, or a cathode ray tube (CRT)), acoustic components (e.g., speakers), haptic components (e.g., a vibratory motor, resistance mechanisms), other signal generators, and so forth. The user input components 1126 may include alphanumeric input components (e.g., a keyboard, a touch screen configured to receive alphanumeric input, a photo-optical keyboard, or other alphanumeric input components), point-based input components (e.g., a mouse, a touchpad, a trackball, a joystick, a motion sensor, or another pointing instrument), tactile input components (e.g., a physical button, a touch screen that provides location and force of touches or touch gestures, or other tactile input components), audio input components (e.g., a microphone), and the like. Any biometric collected by the biometric components is captured and stored with user approval and deleted on user request.

[0141] Further, such biometric data may be used for very limited purposes, such as identification verification. To ensure limited and authorized use of biometric information and other personally identifiable information (PII), access to this data is restricted to authorized personnel only, if allowed at all. Any use of biometric data may strictly be limited to identification verification purposes, and the data is not shared or sold to any third party without the explicit consent of the user. In addition, appropriate technical and organizational measures are implemented to ensure the security and confidentiality of this sensitive information.

[0142] In further examples, the I/O components 1108 may include biometric components 1128, motion components 1130, environmental components 1132, or position components 1134, among a wide array of other components. For example, the biometric components 1128 include components to detect expressions (e.g., hand expressions, facial expressions, vocal expressions, body gestures, or eye-tracking), measure biosignals (e.g., blood pressure, heart rate, body temperature, perspiration, or brain waves), identify a person (e.g., voice identification, retinal identification, facial identification, fingerprint identification, or electroencephalogram-based identification), and the like. The biometric components may include a brain-machine interface (BMI) system that allows communication between the brain and an external device or machine. This may be achieved by recording brain activity data, translating this data into a format that can be understood by a computer, and then using the resulting signals to control the device or machine.

[0143] Example types of BMI technologies include:

- Electroencephalography (EEG) based BMIs, which record electrical activity in the brain using electrodes placed on the scalp.
- Invasive BMIs, which use electrodes that are surgically implanted into the brain.
- Optogenetics BMIs, which use light to control the activity of specific nerve cells in the brain.

[0144] The motion components 1130 include acceleration sensor components (e.g., accelerometer), gravitation sensor components, rotation sensor components (e.g., gyroscope).

[0145] The environmental components 1132 include, for example, one or cameras (with still image/photograph and video capabilities), illumination sensor components (e.g., photometer), temperature sensor components (e.g., one or more thermometers that detect ambient temperature), humidity sensor components, pressure sensor components (e.g., barometer), acoustic sensor components (e.g., one or more microphones that detect background noise), proximity sensor components (e.g., infrared sensors that detect nearby objects), gas sensors (e.g., gas detection sensors to detection concentrations of hazardous gases for safety or to measure pollutants in the atmosphere), or other components that may provide indications, measurements, or signals corresponding to a surrounding physical environment.

[0146] With respect to cameras, the user system 102 may have a camera system comprising, for example, front cameras on a front surface of the user system 102 and rear cameras on a rear surface of the user system 102. The front cameras may, for example, be

used to capture still images and video of a user of the user system 102 (e.g., "selfies"), which may then be augmented with augmentation data (e.g., filters) described above. The rear cameras may, for example, be used to capture still images and videos in a more traditional camera mode, with these images similarly being augmented with augmentation data. In addition to front and rear cameras, the user system 102 may also include a 360° camera for capturing 360° photographs and videos.

[0147] Further, the camera system of the user system 102 may include dual rear cameras (e.g., a primary camera as well as a depth-sensing camera), or even triple, quad, or penta rear camera configurations on the front and rear sides of the user system 102. These multiple cameras systems may include a wide camera, an ultra-wide camera, a telephoto camera, a macro camera, and a depth sensor, for example.

[0148] The position components 1134 include location sensor components (e.g., a GPS receiver component), altitude sensor components (e.g., altimeters or barometers that detect air pressure from which altitude may be derived), orientation sensor components (e.g., magnetometers), and the like.

[0149] Communication may be implemented using a wide variety of technologies. The I/O components 1108 further include communication components 1136 operable to couple the machine 1100 to a network 1138 or devices 1140 via respective coupling or connections. For example, the communication components 1136 may include a network interface component or another suitable device to interface with the network 1138. In further examples, the communication components 1136 may include wired communication components, wireless communication components, cellular communication components, Near Field Communication (NFC) components, Bluetooth® components (e.g., Bluetooth® Low Energy), Wi-Fi® components, and other communication components to provide communication via other modalities. The devices 1140 may be another machine or any of a wide variety of peripheral devices (e.g., a peripheral device coupled via a universal serial bus (USB)).

[0150] Moreover, the communication components 1136 may detect identifiers or include components operable to detect identifiers. For example, the communication components 1136 may include Radio Frequency Identification (RFID) tag reader components, NFC smart tag detection components, optical reader components (e.g., an optical sensor to detect one-dimensional bar codes such as Universal Product Code (UPC) bar code, multi-dimensional bar codes such as Quick Response (QR) code, Aztec code, Data Matrix, Dataglyph™, MaxiCode, PDF417, Ultra Code, UCC RSS-2D bar code, and other optical

codes), or acoustic detection components (e.g., microphones to identify tagged audio signals). In addition, a variety of information may be derived via the communication components 1136, such as location via Internet Protocol (IP) geolocation, location via Wi-Fi® signal triangulation, location via detecting an NFC beacon signal that may indicate a particular location, and so forth.

[0151] The various memories (e.g., main memory 1116, static memory 1118, and memory of the processors 1104) and storage unit 1120 may store one or more sets of instructions and data structures (e.g., software) embodying or used by any one or more of the methodologies or functions described herein. These instructions (e.g., the instructions 1102), when executed by processors 1104, cause various operations to implement the disclosed examples.

[0152] The instructions 1102 may be transmitted or received over the network 1138, using a transmission medium, via a network interface device (e.g., a network interface component included in the communication components 1136) and using any one of several well-known transfer protocols (e.g., HTTP). Similarly, the instructions 1102 may be transmitted or received using a transmission medium via a coupling (e.g., a peer-to-peer coupling) to the devices 1140.

SOFTWARE ARCHITECTURE

[0153] **FIG. 12** is a block diagram 1200 illustrating a software architecture 1202, which can be installed on any one or more of the devices described herein. The software architecture 1202 is supported by hardware such as a machine 1204 that includes processors 1206, memory 1208, and I/O components 1210. In this example, the software architecture 1202 can be conceptualized as a stack of layers, where each layer provides a particular functionality. The software architecture 1202 includes layers such as an operating system 1212, libraries 1214, frameworks 1216, and applications 1218. Operationally, the applications 1218 invoke API calls 1220 through the software stack and receive messages 1222 in response to the API calls 1220.

[0154] The operating system 1212 manages hardware resources and provides common services. The operating system 1212 includes, for example, a kernel 1224, services 1226, and drivers 1228. The kernel 1224 acts as an abstraction layer between the hardware and the other software layers. For example, the kernel 1224 provides memory management, processor management (e.g., scheduling), component management, networking, and security settings, among other functionalities. The services 1226 can provide other common services

for the other software layers. The drivers 1228 are responsible for controlling or interfacing with the underlying hardware. For instance, the drivers 1228 can include display drivers, camera drivers, BLUETOOTH® or BLUETOOTH® Low Energy drivers, flash memory drivers, serial communication drivers (e.g., USB drivers), WI-FI® drivers, audio drivers, power management drivers, and so forth.

[0155] The libraries 1214 provide a common low-level infrastructure used by the applications 1218. The libraries 1214 can include system libraries 1230 (e.g., C standard library) that provide functions such as memory allocation functions, string manipulation functions, mathematic functions, and the like. In addition, the libraries 1214 can include API libraries 1232 such as media libraries (e.g., libraries to support presentation and manipulation of various media formats such as Moving Picture Experts Group-4 (MPEG4), Advanced Video Coding (H.264 or AVC), Moving Picture Experts Group Layer-3 (MP3), Advanced Audio Coding (AAC), Adaptive Multi-Rate (AMR) audio codec, Joint Photographic Experts Group (JPEG or JPG), or Portable Network Graphics (PNG)), graphics libraries (e.g., an OpenGL framework used to render in 2D and 3D in a graphic content on a display), database libraries (e.g., SQLite to provide various relational database functions), web libraries (e.g., WebKit to provide web browsing functionality), and the like. The libraries 1214 can also include a wide variety of other libraries 1234 to provide many other APIs to the applications 1218.

[0156] The frameworks 1216 provide a common high-level infrastructure that is used by the applications 1218. For example, the frameworks 1216 provide various GUI functions, high-level resource management, and high-level location services. The frameworks 1216 can provide a broad spectrum of other APIs that can be used by the applications 1218, some of which may be specific to a particular operating system or platform.

[0157] In an example, the applications 1218 may include a home application 1236, a contacts application 1238, a browser application 1240, a book reader application 1242, a location application 1244, a media application 1246, a messaging application 1248, a game application 1250, and a broad assortment of other applications such as a third-party application 1252. The applications 1218 are programs that execute functions defined in the programs. Various programming languages can be employed to create one or more of the applications 1218, structured in a variety of manners, such as object-oriented programming languages (e.g., Objective-C, Java, or C++) or procedural programming languages (e.g., C or assembly language). In a specific example, the third-party application 1252 (e.g., an application developed using the ANDROID™ or IOS™ SDK by an entity other than the vendor of the particular platform) may be mobile software running on a mobile operating

system such as IOS™, ANDROID™, WINDOWS® Phone, or another mobile operating system. In this example, the third-party application 1252 can invoke the API calls 1220 provided by the operating system 1212 to facilitate functionalities described herein.

SYSTEM WITH HEAD-WEARABLE APPARATUS

[0158] **FIG. 13** illustrates a system 1300 including a head-wearable apparatus 116 with a selector input device, according to some examples. FIG. 13 is a high-level functional block diagram of an example head-wearable apparatus 116 communicatively coupled to a mobile device 114 and various server systems 1304 (e.g., the interaction server system 110) via various networks 1316.

[0159] The head-wearable apparatus 116 includes one or more cameras, each of which may be, for example, a visible light camera 1306, an infrared emitter 1308, and an infrared camera 1310.

[0160] The mobile device 114 connects with head-wearable apparatus 116 using both a low-power wireless connection 1312 and a high-speed wireless connection 1314. The mobile device 114 is also connected to the server system 1304 and the network 1316.

[0161] The head-wearable apparatus 116 further includes two image displays of optical assembly 1318. The two image displays of optical assembly 1318 include one associated with the left lateral side and one associated with the right lateral side of the head-wearable apparatus 116. The head-wearable apparatus 116 also includes an image display driver 1320, an image processor 1322, low-power circuitry 1324, and high-speed circuitry 1326. The image display of optical assembly 1318 is for presenting images and videos, including an image that can include a GUI, to a user of the head-wearable apparatus 116.

[0162] The image display driver 1320 commands and controls the image display of optical assembly 1318. The image display driver 1320 may deliver image data directly to the image display of optical assembly 1318 for presentation or may convert the image data into a signal or data format suitable for delivery to the image display device. For example, the image data may be video data formatted according to compression formats, such as H.264 (MPEG-4 Part 10), HEVC, Theora, Dirac, RealVideo RV40, VP8, VP9, or the like, and still image data may be formatted according to compression formats such as PNG, JPEG, Tagged Image File Format (TIFF) or exchangeable image file format (EXIF) or the like.

[0163] The head-wearable apparatus 116 includes a frame and stems (or temples) extending from a lateral side of the frame. The head-wearable apparatus 116 further includes a user input device 1328 (e.g., touch sensor or push button), including an input

surface on the head-wearable apparatus 116. The user input device 1328 (e.g., touch sensor or push button) is to receive from the user an input selection to manipulate the GUI of the presented image.

[0164] The components shown in FIG. 13 for the head-wearable apparatus 116 are located on one or more circuit boards, for example a PCB or flexible PCB, in the rims or temples. Alternatively, or additionally, the depicted components can be located in the chunks, frames, hinges, or bridge of the head-wearable apparatus 116. Left and right visible light cameras 1306 can include digital camera elements such as a complementary metal oxide–semiconductor (CMOS) image sensor, charge-coupled device, camera lenses, or any other respective visible or light-capturing elements that may be used to capture data, including images of scenes with unknown objects.

[0165] The head-wearable apparatus 116 includes a memory 1302, which stores instructions to perform a subset or all of the functions described herein. The memory 1302 can also include a storage device.

[0166] As shown in FIG. 13, the high-speed circuitry 1326 includes a high-speed processor 1330, a memory 1302, and high-speed wireless circuitry 1332. In some examples, the image display driver 1320 is coupled to the high-speed circuitry 1326 and operated by the high-speed processor 1330 in order to drive the left and right image displays of the image display of optical assembly 1318. The high-speed processor 1330 may be any processor capable of managing high-speed communications and operation of any general computing system needed for the head-wearable apparatus 136. The high-speed processor 1330 includes processing resources needed for managing high-speed data transfers on a high-speed wireless connection 1314 to a wireless local area network (WLAN) using the high-speed wireless circuitry 1332. In certain examples, the high-speed processor 1330 executes an operating system such as a LINUX operating system or other such operating system of the head-wearable apparatus 116, and the operating system is stored in the memory 1302 for execution. In addition to any other responsibilities, the high-speed processor 1330 executing a software architecture for the head-wearable apparatus 116 is used to manage data transfers with high-speed wireless circuitry 1332. In certain examples, the high-speed wireless circuitry 1332 is configured to implement Institute of Electrical and Electronic Engineers (IEEE) 802.11 communication standards, also referred to herein as WiFi. In some examples, other high-speed communications standards may be implemented by the high-speed wireless circuitry 1332.

[0167] Low-power wireless circuitry 1334 and the high-speed wireless circuitry 1332 of the head-wearable apparatus 136 can include short-range transceivers (Bluetooth™) and wireless wide, local, or wide area network transceivers (e.g., cellular or WiFi). Mobile device 114, including the transceivers communicating via the low-power wireless connection 1312 and the high-speed wireless connection 1314, may be implemented using details of the architecture of the head-wearable apparatus 116, as can other elements of the network 1316.

[0168] The memory 1302 includes any storage device capable of storing various data and applications, including, among other things, camera data generated by the left and right visible light cameras 1306, the infrared camera 1310, and the image processor 1322, as well as images generated for display by the image display driver 1320 on the image displays of the image display of optical assembly 1318. While the memory 1302 is shown as integrated with high-speed circuitry 1326, in some examples, the memory 1302 may be an independent standalone element of the head-wearable apparatus 116. In certain such examples, electrical routing lines may provide a connection through a chip that includes the high-speed processor 1330 from the image processor 1322 or low-power processor 1336 to the memory 1302. In some examples, the high-speed processor 1330 may manage addressing of the memory 1302 such that the low-power processor 1336 will boot the high-speed processor 1330 any time that a read or write operation involving memory 1302 is needed.

[0169] As shown in FIG. 13, the low-power processor 1336 or high-speed processor 1330 of the head-wearable apparatus 136 can be coupled to the camera (visible light camera 1306, infrared emitter 1308, or infrared camera 1310), the image display driver 1320, the user input device 1328 (e.g., touch sensor or push button), and the memory 1302.

[0170] The head-wearable apparatus 116 is connected to a host computer. For example, the head-wearable apparatus 116 is paired with the mobile device 114 via the high-speed wireless connection 1314 or connected to the server system 1304 via the network 1316. The server system 1304 may be one or more computing devices as part of a service or network computing system, for example, that includes a processor, a memory, and network communication interface to communicate over the network 1316 with the mobile device 114 and the head-wearable apparatus 116.

[0171] The mobile device 114 includes a processor and a network communication interface coupled to the processor. The network communication interface allows for communication over the network 1316, low-power wireless connection 1312, or high-speed wireless connection 1314. Mobile device 114 can further store at least portions of the instructions for

generating binaural audio content in the memory of mobile device 114 to implement the functionality described herein.

[0172] Output components of the head-wearable apparatus 116 include visual components, such as a display such as a LCD, a PDP, a LED display, a projector, or a waveguide. The image displays of the optical assembly are driven by the image display driver 1320. The output components of the head-wearable apparatus 116 further include acoustic components (e.g., speakers), haptic components (e.g., a vibratory motor), other signal generators, and so forth. The input components of the head-wearable apparatus 116, the mobile device 114, and server system 1304, such as the user input device 1328, may include alphanumeric input components (e.g., a keyboard, a touch screen configured to receive alphanumeric input, a photo-optical keyboard, or other alphanumeric input components), point-based input components (e.g., a mouse, a touchpad, a trackball, a joystick, a motion sensor, or other pointing instruments), tactile input components (e.g., a physical button, a touch screen that provides location and force of touches or touch gestures, or other tactile input components), audio input components (e.g., a microphone), and the like.

[0173] The head-wearable apparatus 116 may also include additional peripheral device elements. Such peripheral device elements may include biometric sensors, additional sensors, or display elements integrated with the head-wearable apparatus 116. For example, peripheral device elements may include any I/O components including output components, motion components, position components, or any other such elements described herein.

[0174] For example, the biometric components include components to detect expressions (e.g., hand expressions, facial expressions, vocal expressions, body gestures, or eye-tracking), measure biosignals (e.g., blood pressure, heart rate, body temperature, perspiration, or brain waves), identify a person (e.g., voice identification, retinal identification, facial identification, fingerprint identification, or electroencephalogram-based identification), and the like. The biometric components may include a BMI system that allows communication between the brain and an external device or machine. This may be achieved by recording brain activity data, translating this data into a format that can be understood by a computer, and then using the resulting signals to control the device or machine.

[0175] The motion components include acceleration sensor components (e.g., accelerometer), gravitation sensor components, rotation sensor components (e.g., gyroscope), and so forth. The position components include location sensor components to

generate location coordinates (e.g., a GPS receiver component), Wi-Fi or Bluetooth™ transceivers to generate positioning system coordinates, altitude sensor components (e.g., altimeters or barometers that detect air pressure from which altitude may be derived), orientation sensor components (e.g., magnetometers), and the like. Such positioning system coordinates can also be received over low-power wireless connections 1312 and high-speed wireless connection 1314 from the mobile device 114 via the low-power wireless circuitry 1334 or high-speed wireless circuitry 1332.

GLOSSARY

[0176] "Carrier signal" refers, for example, to any intangible medium that is capable of storing, encoding, or carrying instructions for execution by the machine and includes digital or analog communications signals or other intangible media to facilitate communication of such instructions. Instructions may be transmitted or received over a network using a transmission medium via a network interface device.

[0177] "Client device" refers, for example, to any machine that interfaces to a communications network to obtain resources from one or more server systems or other client devices. A client device may be, but is not limited to, a mobile phone, desktop computer, laptop, portable digital assistant (PDA), smartphone, tablet, ultrabook, netbook, laptop, multi-processor system, microprocessor-based or programmable consumer electronics, game console, STB, or any other communication device that a user may use to access a network.

[0178] "Communication network" refers, for example, to one or more portions of a network that may be an ad hoc network, an intranet, an extranet, a virtual private network (VPN), a local area network (LAN), a WLAN, a wide area network (WAN), a wireless WAN (WWAN), a metropolitan area network (MAN), the Internet, a portion of the Internet, a portion of the Public Switched Telephone Network (PSTN), a plain old telephone service (POTS) network, a cellular telephone network, a wireless network, a Wi-Fi® network, another type of network, or a combination of two or more such networks. For example, a network or a portion of a network may include a wireless or cellular network, and the coupling may be a Code Division Multiple Access (CDMA) connection, a Global System for Mobile communications (GSM) connection, or other types of cellular or wireless coupling. In this example, the coupling may implement any of a variety of types of data transfer technology, such as Single Carrier Radio Transmission Technology (1xRTT), Evolution-Data Optimized (EVDO) technology, General Packet Radio Service (GPRS)

technology, Enhanced Data rates for GSM Evolution (EDGE) technology, third Generation Partnership Project (3GPP) including 3G, fourth-generation wireless (4G) networks, Universal Mobile Telecommunications System (UMTS), High Speed Packet Access (HSPA), Worldwide Interoperability for Microwave Access (WiMAX), Long Term Evolution (LTE) standard, others defined by various standard-setting organizations, other long-range protocols, or other data transfer technology.

[0179] "Component" refers, for example, to a device, physical entity, or logic having boundaries defined by function or subroutine calls, branch points, APIs, or other technologies that provide for the partitioning or modularization of particular processing or control functions. Components may be combined via their interfaces with other components to carry out a machine process. A component may be a packaged functional hardware unit designed for use with other components and a part of a program that usually performs a particular function of related functions. Components may constitute either software components (e.g., code embodied on a machine-readable medium) or hardware components.

[0180] A "hardware component" is a tangible unit capable of performing certain operations and may be configured or arranged in a certain physical manner. In various examples, one or more computer systems (e.g., a standalone computer system, a client computer system, or a server computer system) or one or more hardware components of a computer system (e.g., a processor or a group of processors) may be configured by software (e.g., an application or application portion) as a hardware component that operates to perform certain operations as described herein.

[0181] A hardware component may also be implemented mechanically, electronically, or any suitable combination thereof. For example, a hardware component may include dedicated circuitry or logic that is permanently configured to perform certain operations. A hardware component may be a special-purpose processor, such as a field-programmable gate array (FPGA) or an ASIC. A hardware component may also include programmable logic or circuitry that is temporarily configured by software to perform certain operations. For example, a hardware component may include software executed by a general-purpose processor or other programmable processors. Once configured by such software, hardware components become specific machines (or specific components of a machine) uniquely tailored to perform the configured functions and are no longer general-purpose processors. It will be appreciated that the decision to implement a hardware component mechanically, in dedicated and permanently configured circuitry, or in temporarily configured circuitry (e.g., configured by software), may be driven by cost and time considerations. Accordingly, the phrase "hardware component"(or "hardware-implemented component") should be

understood to encompass a tangible entity, be that an entity that is physically constructed, permanently configured (e.g., hardwired), or temporarily configured (e.g., programmed) to operate in a certain manner or to perform certain operations described herein.

[0182] Considering examples in which hardware components are temporarily configured (e.g., programmed), each of the hardware components need not be configured or instantiated at any one instance in time. For example, where a hardware component comprises a general-purpose processor configured by software to become a special-purpose processor, the general-purpose processor may be configured as respectively different special-purpose processors (e.g., comprising different hardware components) at different times. Software accordingly configures a particular processor or processors, for example, to constitute a particular hardware component at one instance of time and to constitute a different hardware component at a different instance of time. Hardware components can provide information to, and receive information from, other hardware components. Accordingly, the described hardware components may be regarded as being communicatively coupled. Where multiple hardware components exist contemporaneously, communications may be achieved through signal transmission (e.g., over appropriate circuits and buses) between or among two or more of the hardware components. In examples in which multiple hardware components are configured or instantiated at different times, communications between such hardware components may be achieved, for example, through the storage and retrieval of information in memory structures to which the multiple hardware components have access. For example, one hardware component may perform an operation and store the output of that operation in a memory device to which it is communicatively coupled. A further hardware component may then, at a later time, access the memory device to retrieve and process the stored output. Hardware components may also initiate communications with input or output devices, and can operate on a resource (e.g., a collection of information). The various operations of example methods described herein may be performed, at least partially, by one or more processors that are temporarily configured (e.g., by software) or permanently configured to perform the relevant operations. Whether temporarily or permanently configured, such processors may constitute processor-implemented components that operate to perform one or more operations or functions described herein.

[0183] As used herein, "processor-implemented component" refers to a hardware component implemented using one or more processors. Similarly, the methods described herein may be at least partially processor-implemented, with a particular processor or processors being an example of hardware. For example, at least some of the operations of a method may be performed by one or more processors or processor-implemented

components. Moreover, the one or more processors may also operate to support performance of the relevant operations in a "cloud computing" environment or as a "software as a service" (SaaS). For example, at least some of the operations may be performed by a group of computers (as examples of machines including processors), with these operations being accessible via a network (e.g., the Internet) and via one or more appropriate interfaces (e.g., an API). The performance of certain of the operations may be distributed among the processors, not only residing within a single machine, but deployed across a number of machines. In some examples, the processors or processor-implemented components may be located in a single geographic location (e.g., within a home environment, an office environment, or a server farm). In other examples, the processors or processor-implemented components may be distributed across a number of geographic locations.

[0184] "Computer-readable storage medium" refers, for example, to both machine-storage media and transmission media. Thus, the terms include both storage devices/media and carrier waves/modulated data signals. The terms “machine-readable medium,” “computer-readable medium,” and “device-readable medium” mean the same thing and may be used interchangeably in this disclosure. "Ephemeral message" refers, for example, to a message that is accessible for a time-limited duration. An ephemeral message may be a text, an image, a video and the like. The access time for the ephemeral message may be set by the message sender. Alternatively, the access time may be a default setting or a setting specified by the recipient. Regardless of the setting technique, the message is transitory.

[0185] "Machine storage medium" refers, for example, to a single or multiple storage devices and media (e.g., a centralized or distributed database, and associated caches and servers) that store executable instructions, routines and data. The term shall accordingly be taken to include, but not be limited to, solid-state memories, and optical and magnetic media, including memory internal or external to processors. Specific examples of machine-storage media, computer-storage media and device-storage media include non-volatile memory, including by way of example semiconductor memory devices, e.g., erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), FPGA, and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and CD-ROM and DVD-ROM disks The terms "machine-storage medium," "device-storage medium," "computer-storage medium" mean the same thing and may be used interchangeably in this disclosure.

[0186] The terms "machine-storage media," "computer-storage media," and "device-storage media" specifically exclude carrier waves, modulated data signals, and other such

media, at least some of which are covered under the term "signal medium." "Non-transitory computer-readable storage medium" refers, for example, to a tangible medium that is capable of storing, encoding, or carrying the instructions for execution by a machine. "Signal medium" refers, for example, to any intangible medium that is capable of storing, encoding, or carrying the instructions for execution by a machine and includes digital or analog communications signals or other intangible media to facilitate communication of software or data. The term "signal medium" shall be taken to include any form of a modulated data signal, carrier wave, and so forth. The term "modulated data signal" means a signal that has one or more of its characteristics set or changed in such a matter as to encode information in the signal. The terms "transmission medium" and "signal medium" mean the same thing and may be used interchangeably in this disclosure.

[0187] "User device" refers, for example, to a device accessed, controlled or owned by a user and with which the user interacts perform an action, or interaction on the user device, including interaction with other users or computer systems. "Carrier signal" refers to any intangible medium that is capable of storing, encoding, or carrying instructions for execution by the machine and includes digital or analog communications signals or other intangible media to facilitate communication of such instructions. Instructions may be transmitted or received over a network using a transmission medium via a network interface device. "Client device" refers to any machine that interfaces to a communications network to obtain resources from one or more server systems or other client devices. A client device may be, but is not limited to, a mobile phone, desktop computer, laptop, PDA, smartphone, tablet, ultrabook, netbook, laptop, multi-processor system, microprocessor-based or programmable consumer electronics, game console, STB, or any other communication device that a user may use to access a network.

[0188] "Communication network" refers to one or more portions of a network that may be an ad hoc network, an intranet, an extranet, a VPN, a LAN, a WLAN, a WAN, a WWAN, a MAN, the Internet, a portion of the Internet, a portion of the PSTN, a POTS network, a cellular telephone network, a wireless network, a Wi-Fi® network, another type of network, or a combination of two or more such networks. For example, a network or a portion of a network may include a wireless or cellular network, and the coupling may be a CDMA connection, a GSM connection, or other types of cellular or wireless coupling. In this example, the coupling may implement any of a variety of types of data transfer technology, such as 1xRTT, EVDO technology, GPRS technology, EDGE technology, 3GPP including 3G, 4G networks, UMTS, HSPA, WiMAX, LTE standard, others defined by various standard-setting organizations, other long-range protocols, or other data transfer technology.

[0189] Components may constitute either software components (e.g., code embodied on a machine-readable medium) or hardware components. A "hardware component" is a tangible unit capable of performing certain operations and may be configured or arranged in a certain physical manner. In various examples, one or more computer systems (e.g., a standalone computer system, a client computer system, or a server computer system) or one or more hardware components of a computer system (e.g., a processor or a group of processors) may be configured by software (e.g., an application or application portion) as a hardware component that operates to perform certain operations as described herein.

[0190] A hardware component may also be implemented mechanically, electronically, or any suitable combination thereof. For example, a hardware component may include dedicated circuitry or logic that is permanently configured to perform certain operations. A hardware component may be a special-purpose processor, such as a FPGA or an ASIC. A hardware component may also include programmable logic or circuitry that is temporarily configured by software to perform certain operations. For example, a hardware component may include software executed by a general-purpose processor or other programmable processor. Once configured by such software, hardware components become specific machines (or specific components of a machine) uniquely tailored to perform the configured functions and are no longer general-purpose processors. It will be appreciated that the decision to implement a hardware component mechanically, in dedicated and permanently configured circuitry, or in temporarily configured circuitry (e.g., configured by software), may be driven by cost and time considerations. Accordingly, the phrase "hardware component"(or "hardware-implemented component") should be understood to encompass a tangible entity, be that an entity that is physically constructed, permanently configured (e.g., hardwired), or temporarily configured (e.g., programmed) to operate in a certain manner or to perform certain operations described herein.

[0191] The various operations of example methods described herein may be performed, at least partially, by one or more processors that are temporarily configured (e.g., by software) or permanently configured to perform the relevant operations. Whether temporarily or permanently configured, such processors may constitute processor-implemented components that operate to perform one or more operations or functions described herein.

[0192] Changes and modifications may be made to the disclosed examples without departing from the scope of the present disclosure. These and other changes or modifications are intended to be included within the scope of the present disclosure, as expressed in the following claims.

What is claimed is:

1. A method comprising:

receiving, from a camera of a stationary device, an image depicting a real-world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment;

analyzing the image using a machine learning model to generate predicted tracking information for the real-world object, the machine learning model trained on a plurality of images depicting real-world objects in the specified field of view of the real-world environment and corresponding ground-truth tracking information for the real-world objects;

selecting an extended reality (XR) experience from a plurality of XR experiences; and

overlaying one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

2. The method of claim 1, wherein the stationary device comprises an XR shoe mirror.

3. The method of any one of claims 1-2, wherein the XR shoe mirror includes a display screen that is placed at a specified distance above the specified field of view, wherein the XR shoe mirror includes a physical area attached to the display screen that defines the specified field of view, and wherein the XR shoe mirror includes the camera pointed towards the physical area.

4. The method of claim 3, wherein the display screen is mounted on a physical surface that is tilted at an angle relative to the physical area, and wherein the camera is integrated into the physical surface and is directed such that a surface normal of the camera is perpendicular to the physical area.

5. The method of any one of claims 1-4, wherein the camera points down towards the physical area.

6. The method of any one of claims 1-5, further comprising setting a focal length and one or more camera parameters to a specified value based on a distance between the camera and the physical area and one or more external conditions of the XR shoe mirror.

7. The method of any one of claims 1-6, further comprising:

identifying an area outside of the physical area of the XR shoe mirror;

generating a mask based on the area outside of the physical area of the XR shoe mirror; and

occluding pixels of the image that fall within the identified area outside of the physical area of the XR shoe mirror while overlaying the one or more XR elements on the image.

8. The method of claim 7, wherein the mask is generated when the XR shoe mirror is manufactured or placed within the real-world environment.

9. The method of any one of claims 1-8, wherein the real-world object comprises a foot, and wherein the one or more XR elements comprise one or more virtual shoes.

10. The method of any one of claims 1-9, wherein the stationary device comprises an XR vanity mirror.

11. The method of claim 10, wherein the real-world object comprises a face, and wherein the one or more XR elements comprise one or more virtual fashion items or virtual makeup.

12. The method of any one of claims 1-11, further comprising:

accessing lighting conditions of the real-world environment;

analyzing the lighting conditions with the machine learning model to predict a modification to one or more physical light parameters of the stationary device and to predict one or more light parameters of the one or more XR elements associated with the XR experience; and

automatically adjusting physical light that surrounds a screen on which the modified image is displayed based on the predicted one or more physical light parameters.

13. The method of claim 12, wherein the one or more physical light parameters comprise a light temperature, color, or intensity.

14. The method of any one of claims 1-13, wherein the stationary device comprises an XR window placed on an opening of a building or on an interior wall of the building.

15. The method of claim 14, wherein the real-world object comprises a sky or city landscape, and wherein the one or more XR elements comprise one or more labels or virtual sky elements.

16. The method of any one of claims 1-15, further comprising setting a focal length and one or more camera parameters to a specified value based on a distance between the camera and the sky or city landscape.

17. The method of any one of claims 1-16, further comprising:

receiving, by the XR window, physical movement information of a mobile device that is external to the stationary device; and

adjusting one or more visual properties of the one or more XR elements displayed by the XR window based on the physical movement information of the mobile device.

18. The method of any one of claims 1-17, further comprising training the machine learning model by performing training operations comprising:

selecting a first training image from the plurality of training images, the first training image depicting an individual real-world object having associated individual ground-truth tracking information, the first training image being captured by the stationary device and depicting the individual real-world object in the specified field of view of the real-world environment;

analyzing the first training image using the machine learning model to predict estimated tracking information for the individual real-world object;

generating loss based on a deviation between the estimated tracking information for the individual real-world object and the associated individual ground-truth tracking information of the individual real-world object; and

updating one or more parameters of the machine learning model based on the loss.

19. A system comprising:

at least one processor; and

at least one memory component having instructions stored thereon that, when executed by the at least one processor, cause the at least one processor to perform operations comprising:

receiving, from a camera of a stationary device, an image depicting a real-world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment;

analyzing the image using a machine learning model to generate predicted tracking information for the real-world object, the machine learning model trained based on a plurality of images depicting real-world objects in the specified field of view of the

real-world environment and corresponding ground-truth tracking information for the real-world objects;

selecting an extended reality (XR) experience from a plurality of XR experiences; and

overlaying one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

20. A non-transitory computer-readable storage medium having stored thereon instructions that, when executed by at least one processor, cause the at least one processor to perform operations comprising:

receiving, from a camera of a stationary device, an image depicting a real-world object, the camera being directed in a stationary manner towards a specified field of view of a real-world environment;

analyzing the image using a machine learning model to generate predicted tracking information for the real-world object, the machine learning model trained based on a plurality of images depicting real-world objects in the specified field of view of the real-world environment and corresponding ground-truth tracking information for the real-world objects;

selecting an extended reality (XR) experience from a plurality of XR experiences; and

overlaying one or more XR elements associated with the XR experience on the image based on the predicted tracking information to generate a modified image.

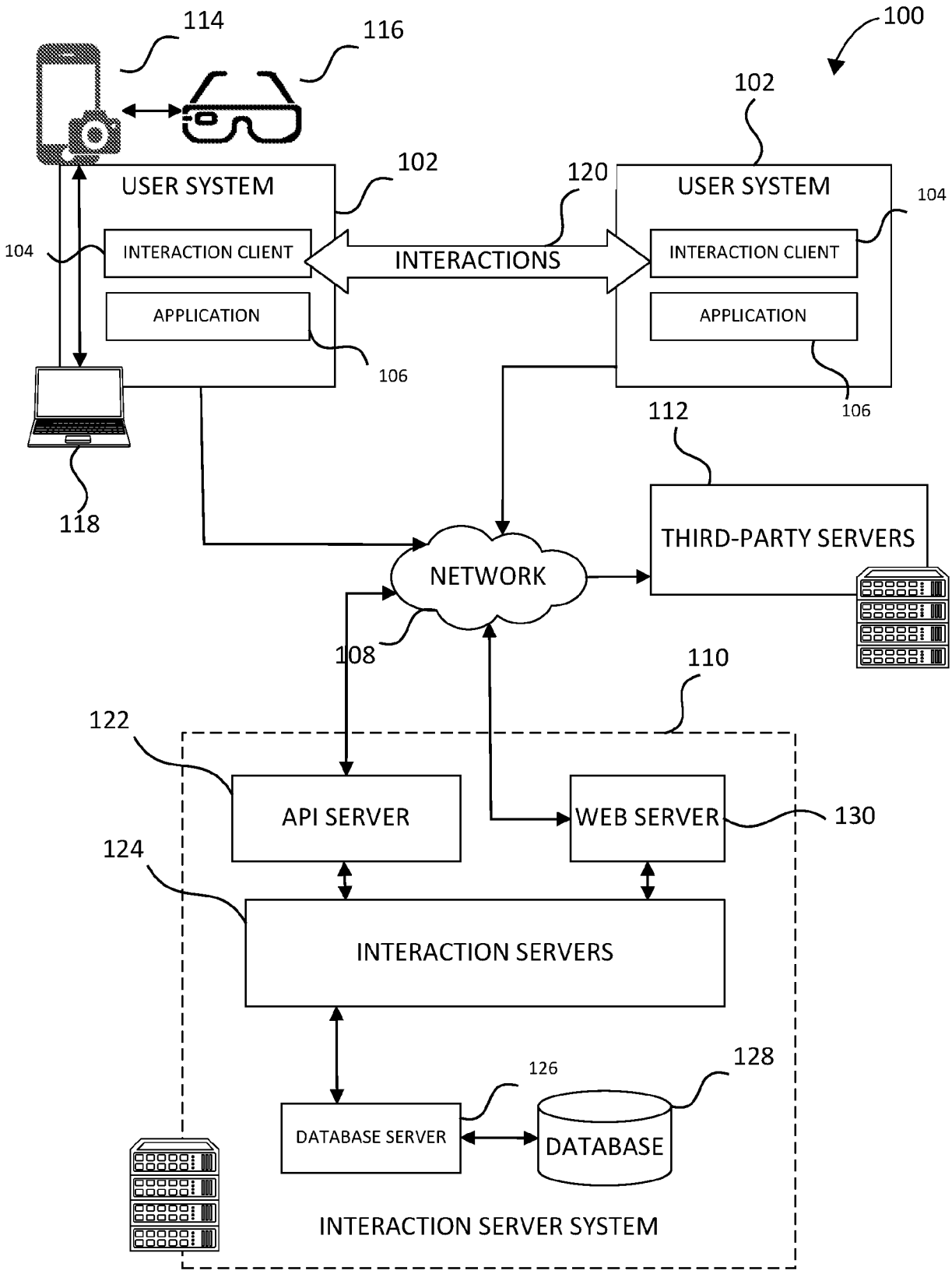

114
116
100
102
USER SYSTEM
102
120
USER SYSTEM
104
104
INTERACTION CLIENT
INTERACTIONS
INTERACTION CLIENT
APPLICATION
APPLICATION
106
106
112
118
THIRD-PARTY SERVERS
NETWORK
108
110
122
API SERVER
WEB SERVER
130
124
INTERACTION SERVERS
126
128
DATABASE SERVER
DATABASE
INTERACTION SERVER SYSTEM

FIG. 1

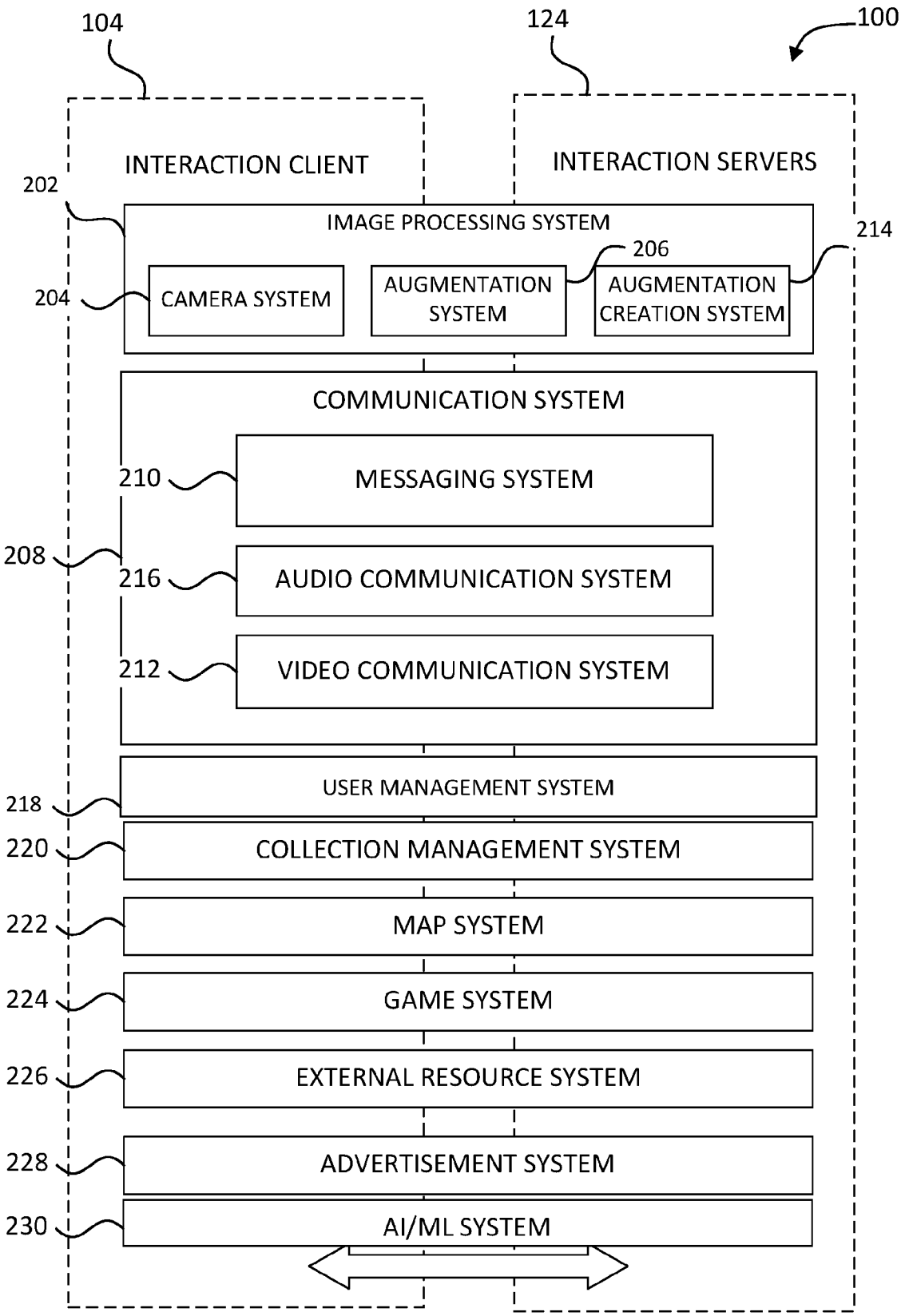

104
124
100
INTERACTION CLIENT
INTERACTION SERVERS
202
IMAGE PROCESSING SYSTEM
214
206
204
CAMERA SYSTEM
AUGMENTATION SYSTEM
AUGMENTATION CREATION SYSTEM
COMMUNICATION SYSTEM
210
MESSAGING SYSTEM
208
216
AUDIO COMMUNICATION SYSTEM
212
VIDEO COMMUNICATION SYSTEM
USER MANAGEMENT SYSTEM
218
220
COLLECTION MANAGEMENT SYSTEM
222
MAP SYSTEM
224
GAME SYSTEM
226
EXTERNAL RESOURCE SYSTEM
228
ADVERTISEMENT SYSTEM
230
AI/ML SYSTEM

FIG. 2

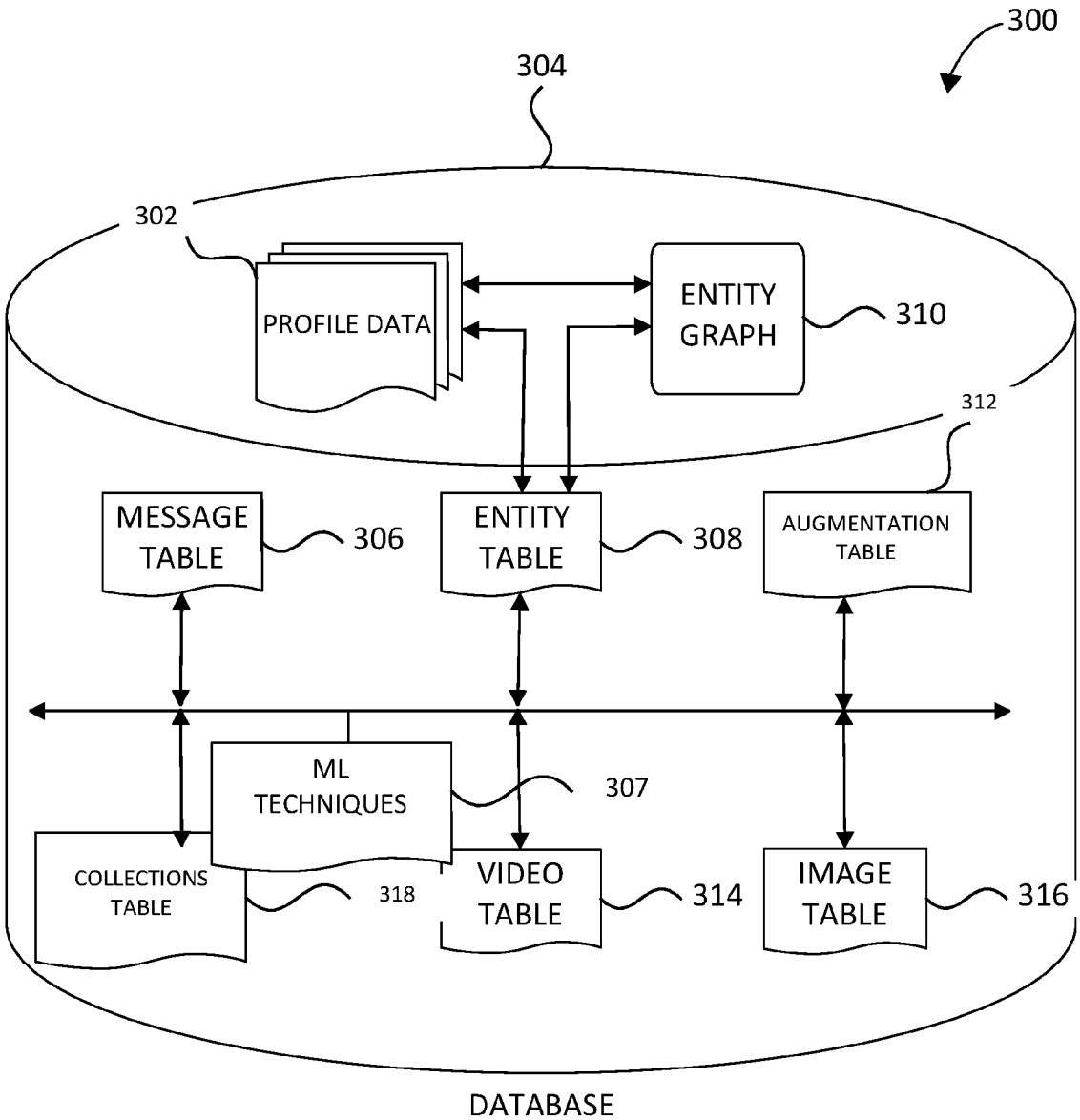

300
304
302
PROFILE DATA
ENTITY GRAPH
310
312
MESSAGE TABLE
306
ENTITY TABLE
308
AUGMENTATION TABLE
ML TECHNIQUES
307
COLLECTIONS TABLE
318
VIDEO TABLE
314
IMAGE TABLE
316
DATABASE

FIG. 3

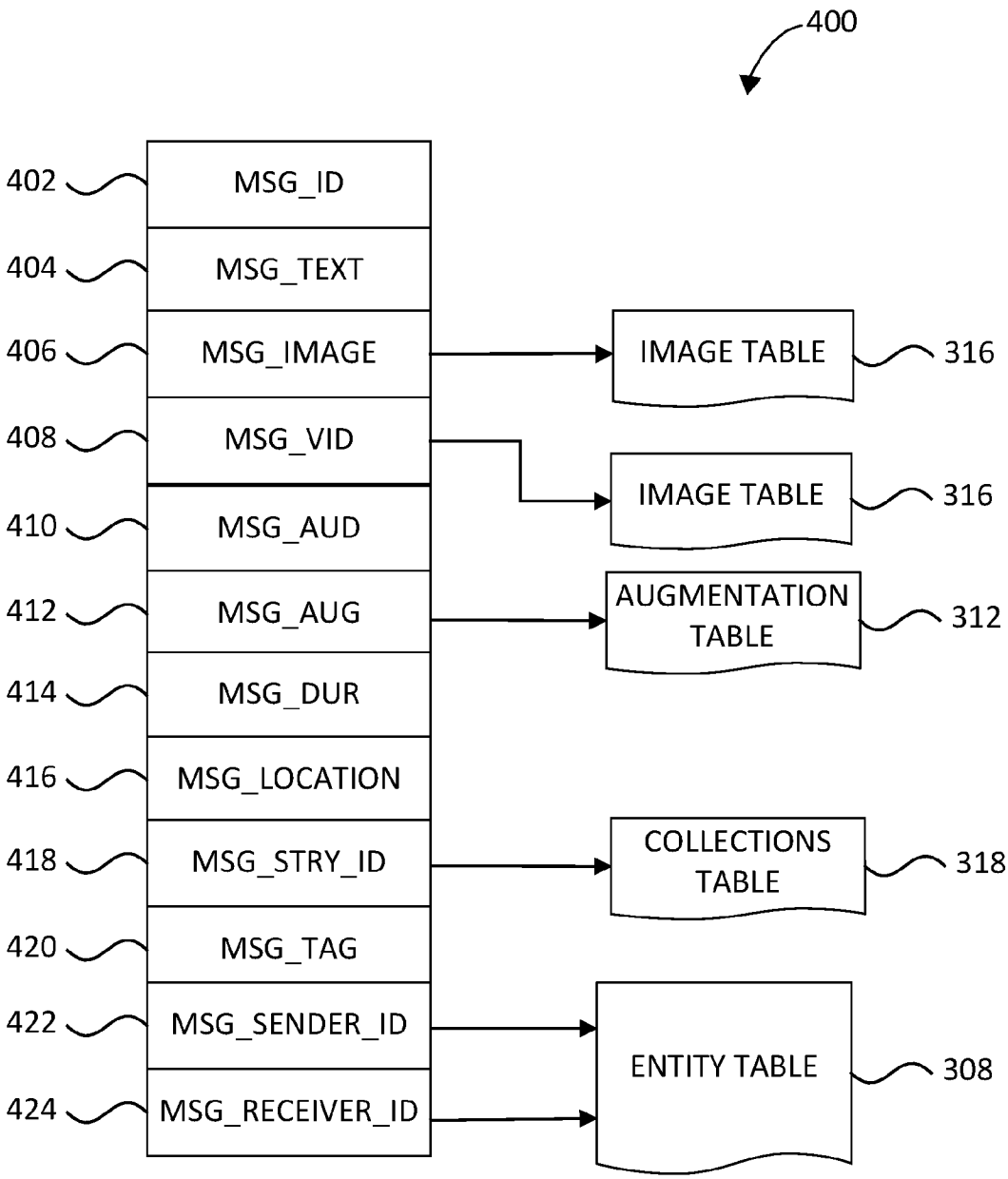

400
402
MSG_ID
404
MSG_TEXT
406
MSG_IMAGE
408
MSG_VID
410
MSG_AUD
412
MSG_AUG
414
MSG_DUR
416
MSG_LOCATION
418
MSG_STRY_ID
420
MSG_TAG
422
MSG_SENDER_ID
424
MSG_RECEIVER_ID
IMAGE TABLE
316
IMAGE TABLE
316
AUGMENTATION TABLE
312
COLLECTIONS TABLE
318
ENTITY TABLE
308

FIG. 4

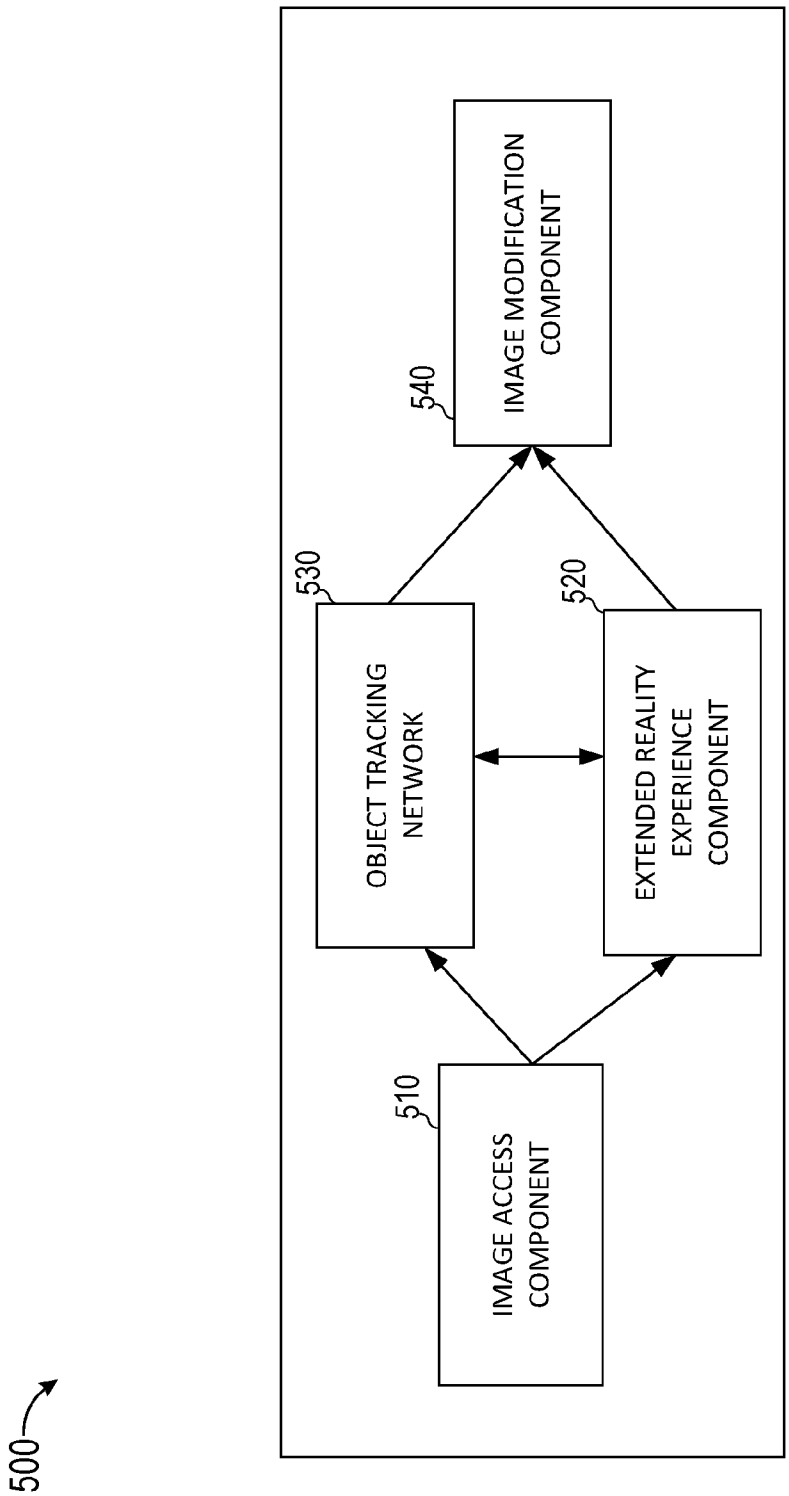

500
IMAGE ACCESS COMPONENT
510
OBJECT TRACKING NETWORK
530
EXTENDED REALITY EXPERIENCE COMPONENT
520
IMAGE MODIFICATION COMPONENT
540

FIG. 5

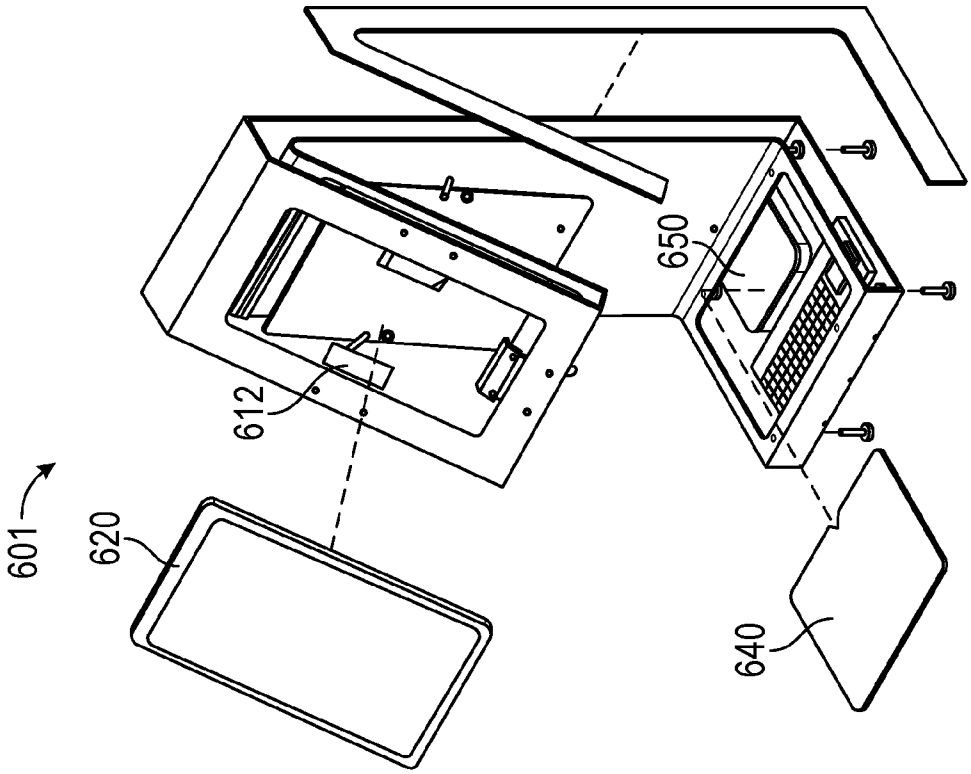
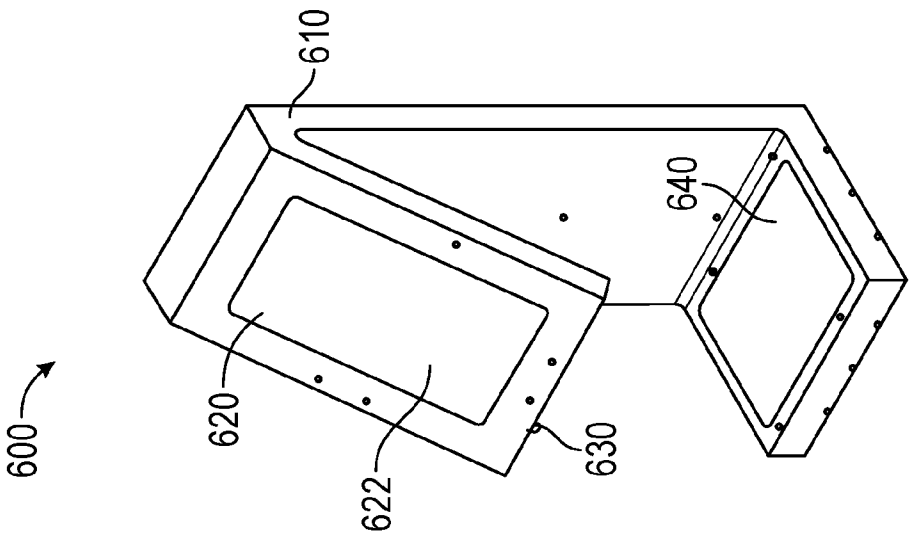


FIG. 6



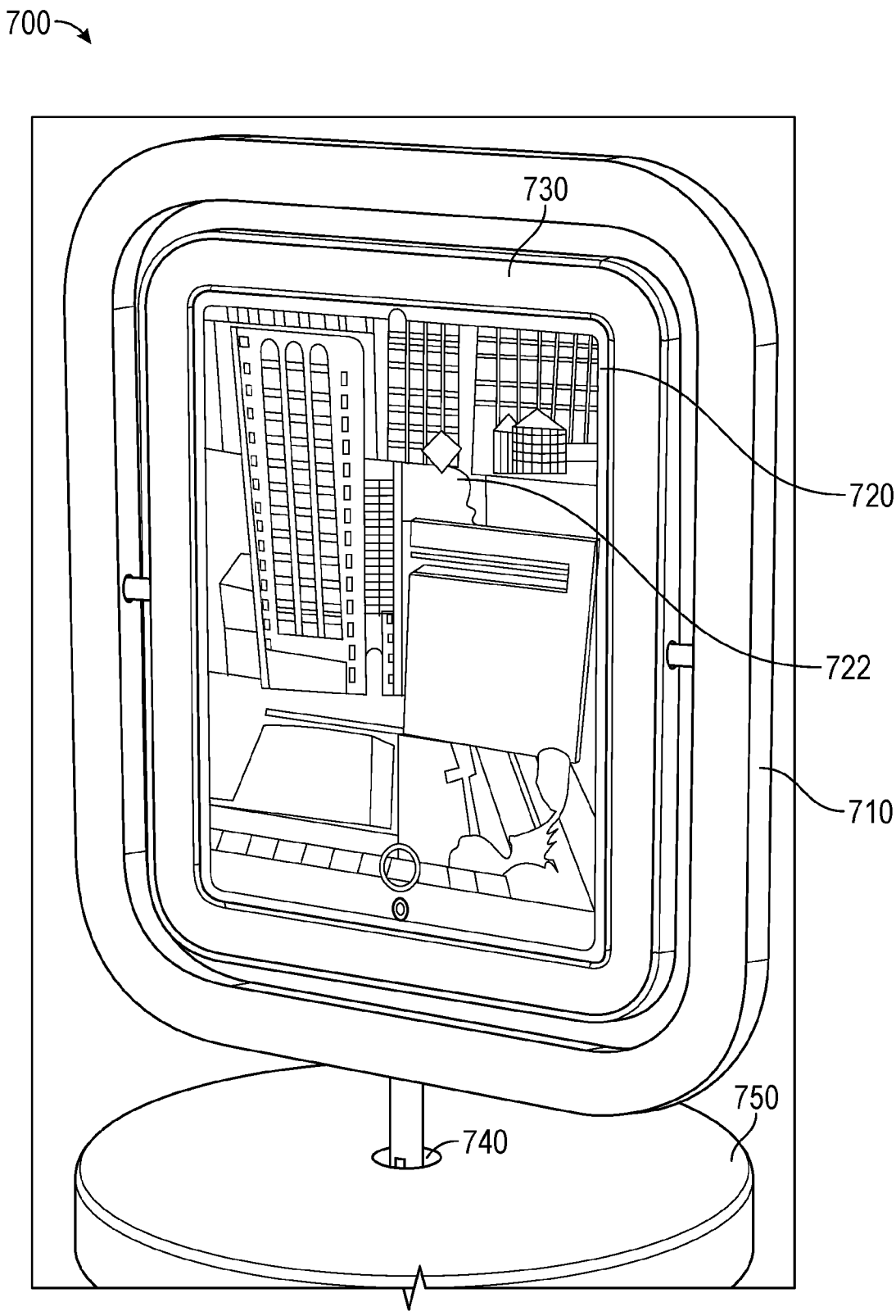


FIG. 7

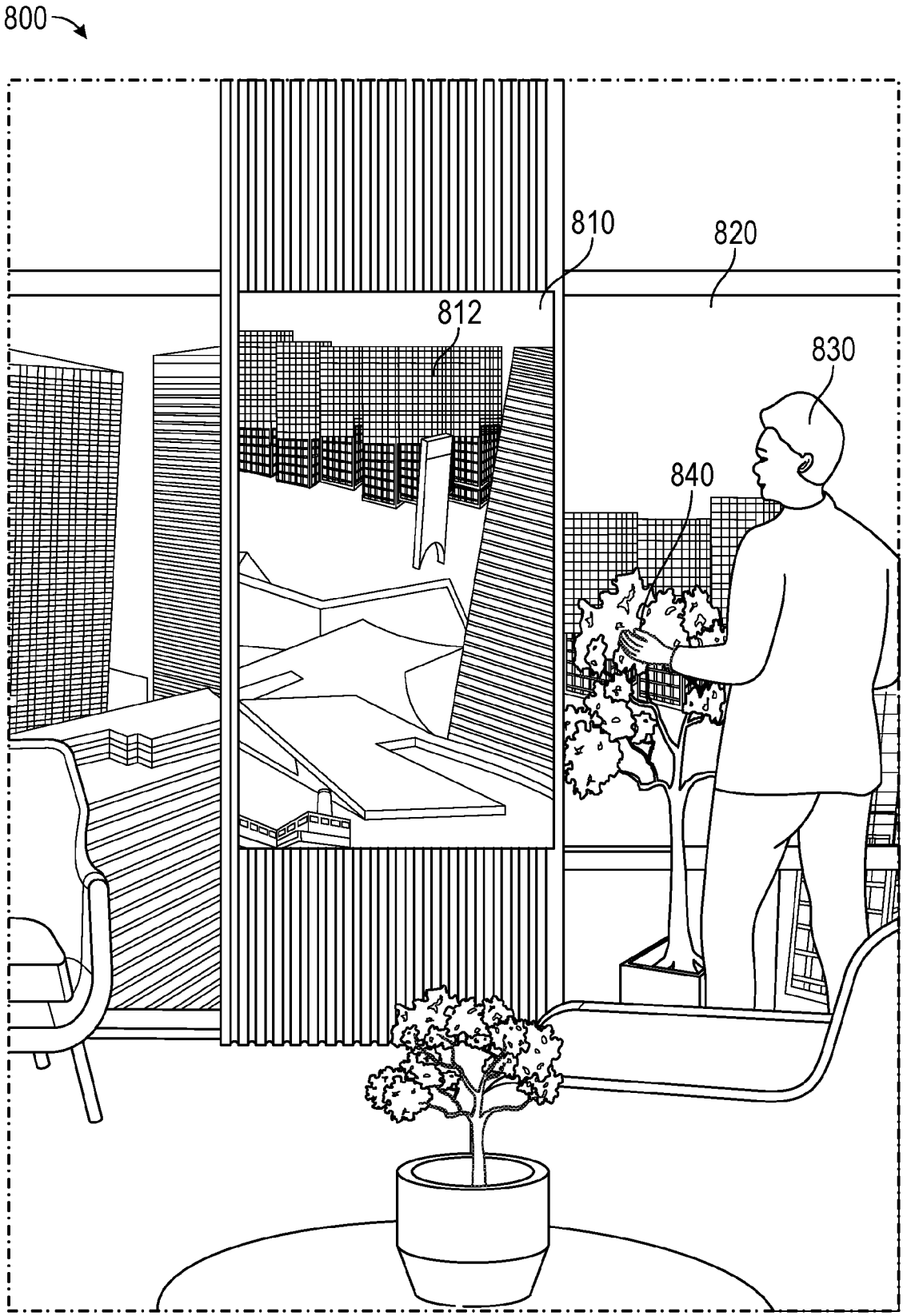


FIG. 8

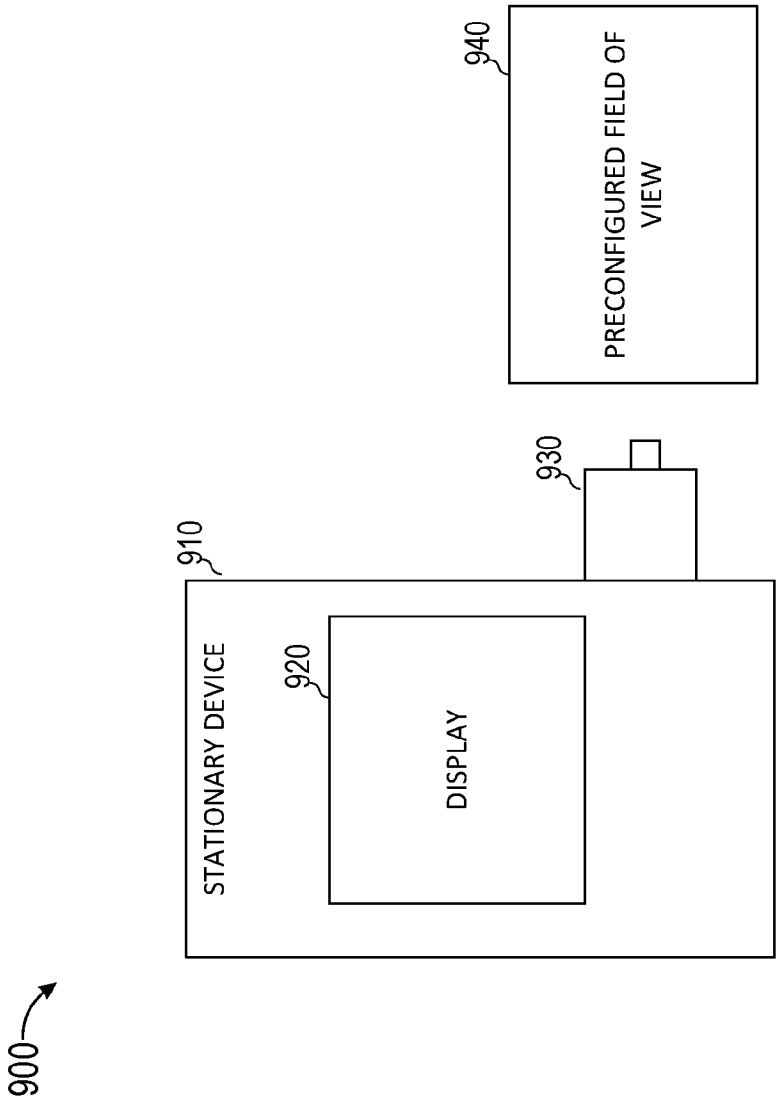

900
910
STATIONARY DEVICE
920
DISPLAY
930
940
PRECONFIGURED FIELD OF VIEW

FIG. 9

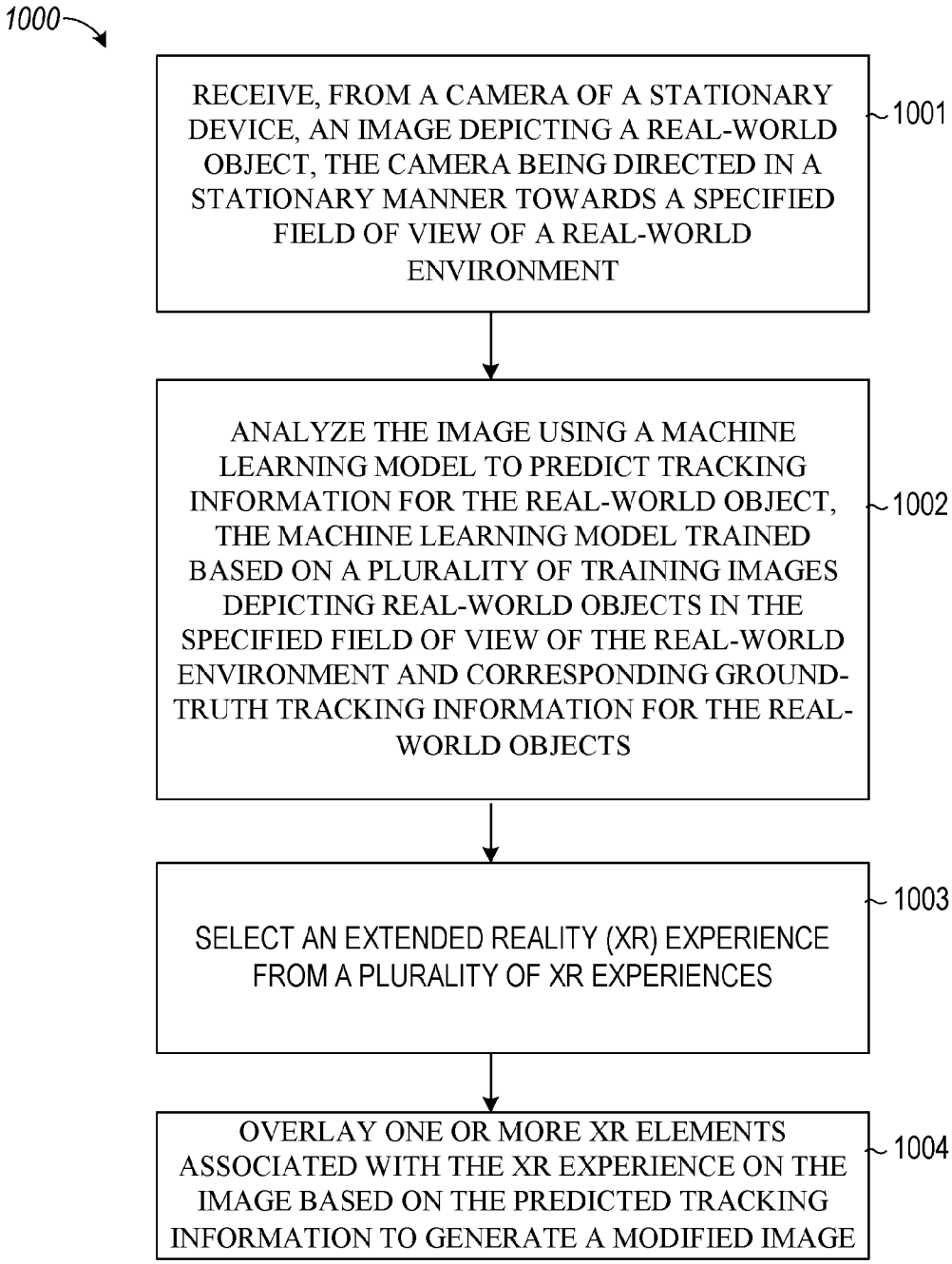

1000
RECEIVE, FROM A CAMERA OF A STATIONARY DEVICE, AN IMAGE DEPICTING A REAL-WORLD OBJECT, THE CAMERA BEING DIRECTED IN A STATIONARY MANNER TOWARDS A SPECIFIED FIELD OF VIEW OF A REAL-WORLD ENVIRONMENT
1001
ANALYZE THE IMAGE USING A MACHINE LEARNING MODEL TO PREDICT TRACKING INFORMATION FOR THE REAL-WORLD OBJECT, THE MACHINE LEARNING MODEL TRAINED BASED ON A PLURALITY OF TRAINING IMAGES DEPICTING REAL-WORLD OBJECTS IN THE SPECIFIED FIELD OF VIEW OF THE REAL-WORLD ENVIRONMENT AND CORRESPONDING GROUND-TRUTH TRACKING INFORMATION FOR THE REAL-WORLD OBJECTS
1002
SELECT AN EXTENDED REALITY (XR) EXPERIENCE FROM A PLURALITY OF XR EXPERIENCES
1003
OVERLAY ONE OR MORE XR ELEMENTS ASSOCIATED WITH THE XR EXPERIENCE ON THE IMAGE BASED ON THE PREDICTED TRACKING INFORMATION TO GENERATE A MODIFIED IMAGE
1004

FIG. 10

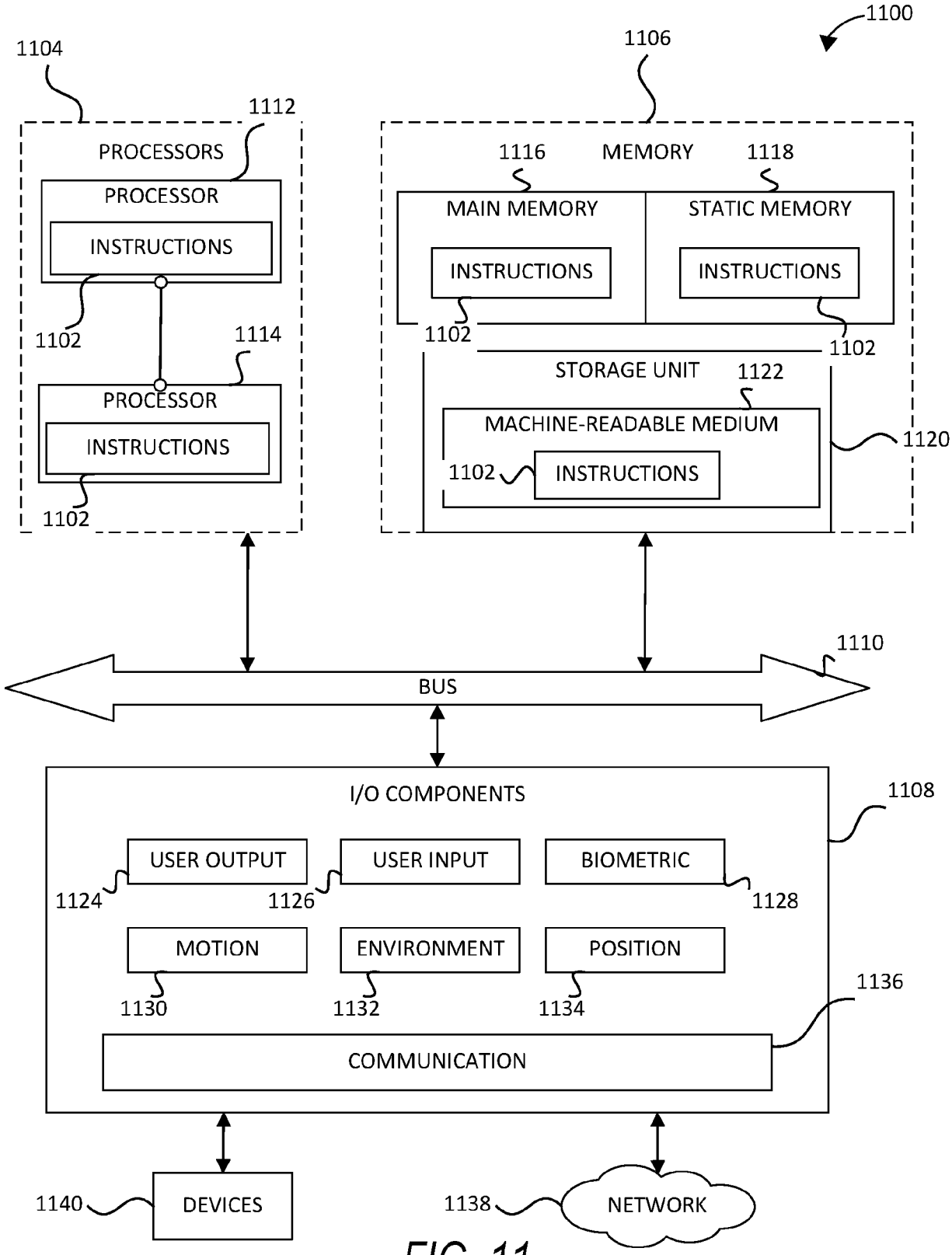

1100
1104
1106
1112
PROCESSORS
PROCESSOR
INSTRUCTIONS
1102
1114
PROCESSOR
INSTRUCTIONS
1102
1116
MEMORY
1118
MAIN MEMORY
STATIC MEMORY
INSTRUCTIONS
INSTRUCTIONS
1102
1102
STORAGE UNIT
1122
MACHINE-READABLE MEDIUM
1102
INSTRUCTIONS
1120
1110
BUS
I/O COMPONENTS
1108
USER OUTPUT
USER INPUT
BIOMETRIC
1124
1126
1128
MOTION
ENVIRONMENT
POSITION
1130
1132
1134
1136
COMMUNICATION
1140
DEVICES
1138
NETWORK

FIG. 11

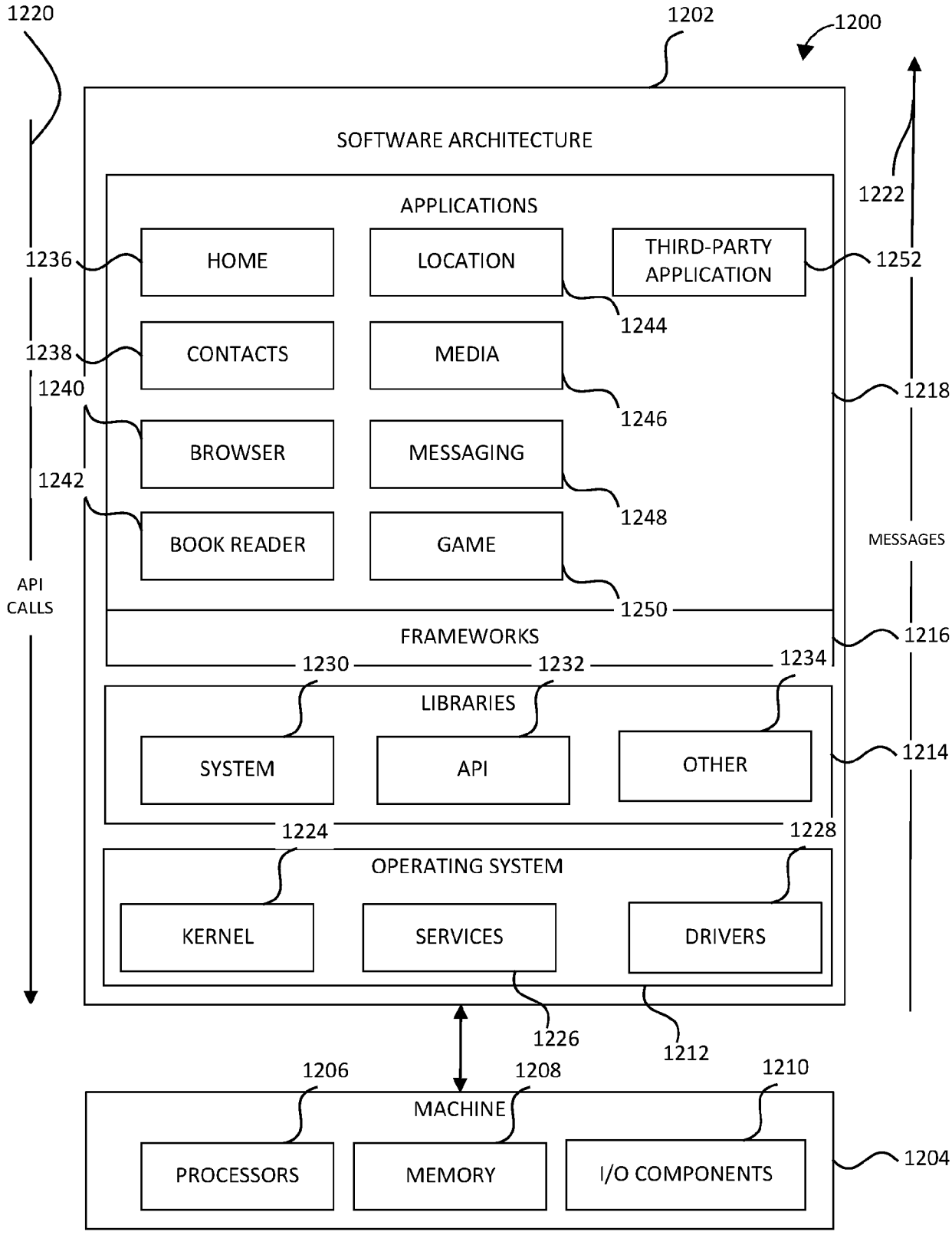

1220
1202
1200
SOFTWARE ARCHITECTURE
APPLICATIONS
1222
1236
HOME
LOCATION
THIRD-PARTY APPLICATION
1252
1244
1238
CONTACTS
MEDIA
1240
1218
1246
BROWSER
MESSAGING
1242
1248
BOOK READER
GAME
MESSAGES
API CALLS
1250
FRAMEWORKS
1216
1230
1232
1234
LIBRARIES
SYSTEM
API
OTHER
1214
1224
1228
OPERATING SYSTEM
KERNEL
SERVICES
DRIVERS
1226
1212
1206
1208
1210
MACHINE
PROCESSORS
MEMORY
I/O COMPONENTS
1204

FIG. 12

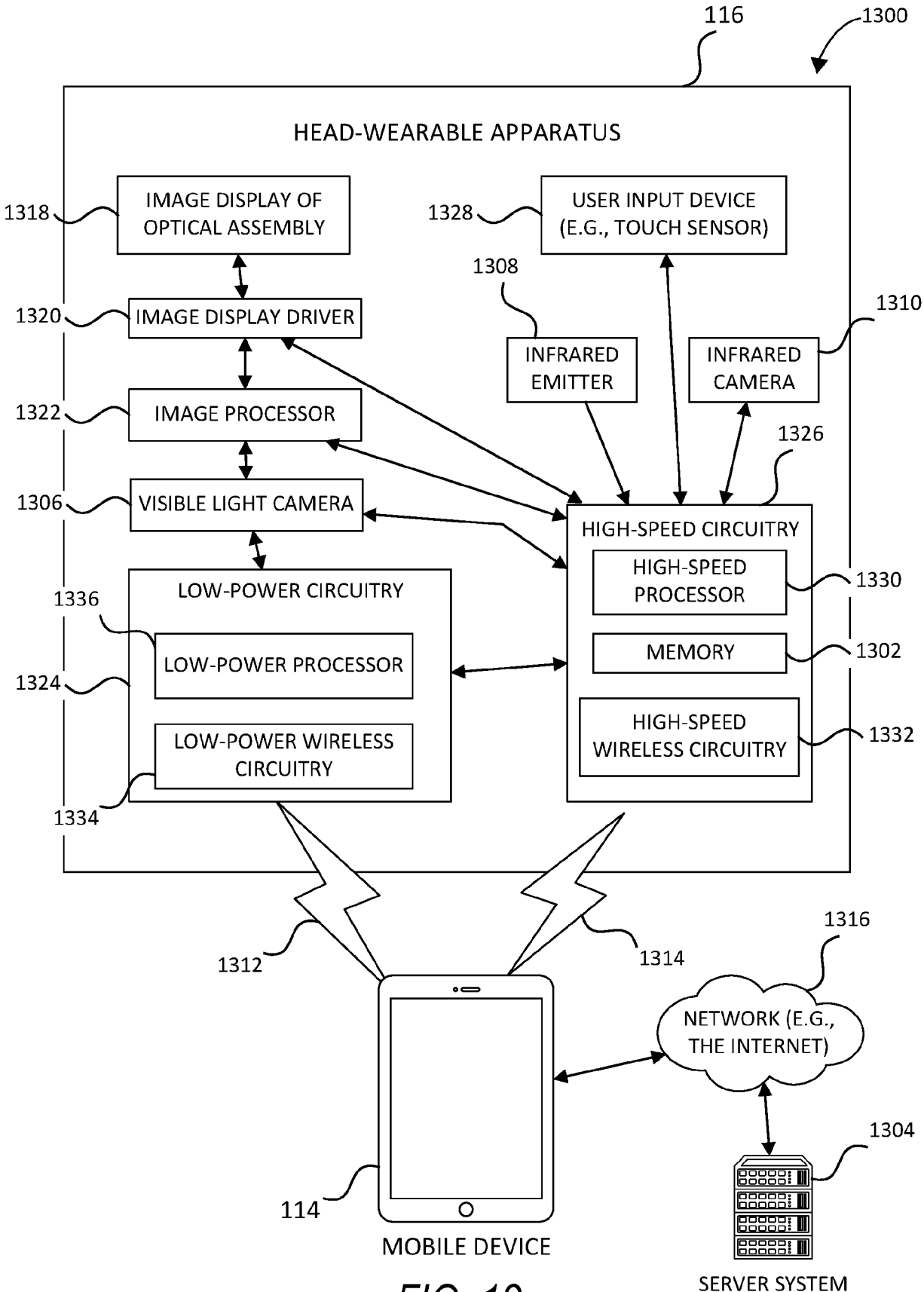

116
1300
HEAD-WEARABLE APPARATUS
1318
IMAGE DISPLAY OF OPTICAL ASSEMBLY
1328
USER INPUT DEVICE (E.G., TOUCH SENSOR)
1308
1320
IMAGE DISPLAY DRIVER
1310
INFRARED EMITTER
INFRARED CAMERA
1322
IMAGE PROCESSOR
1326
1306
VISIBLE LIGHT CAMERA
HIGH-SPEED CIRCUITRY
HIGH-SPEED PROCESSOR
1330
1336
LOW-POWER CIRCUITRY
MEMORY
1302
LOW-POWER PROCESSOR
1324
HIGH-SPEED WIRELESS CIRCUITRY
1332
LOW-POWER WIRELESS CIRCUITRY
1334
1312
1314
1316
NETWORK (E.G., THE INTERNET)
1304
114
MOBILE DEVICE
SERVER SYSTEM

FIG. 13

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2024/022836

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06T19/00 G06V10/74 G09F19/16
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)

G06T G06V G09F

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

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

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	KR 2021 0090900 A (KIM SANG AH [KR]; LEE MIN KYU [KR] ET AL.) 21 July 2021 (2021-07-21)	1,18-20
A	paragraphs [0022], [0034], [0046], [0049] - [0051], [0053], [0055], [0057] - [0059], [0063] - [0068], [0080] figure 3	2-17

Y	US 2022/327709 A1 (DUDOVITCH GAL [IL] ET AL) 13 October 2022 (2022-10-13)	1,18-20
A	paragraphs [0053] - [0058], [0103], [0106] - [0113]	2-17

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance;; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance;; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
12 September 2024	24/09/2024

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Gkeli, Maria

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2024/022836

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
KR 20210090900 A	21-07-2021	NONE	
US 2022327709 A1	13-10-2022	CN 117157667 A	01-12-2023
		EP 4323962 A2	21-02-2024
		KR 20230170722 A	19-12-2023
		US 2022327709 A1	13-10-2022
		WO 2022221243 A2	20-10-2022