



(51) International Patent Classification:

G06V 20/17 (2022.01) G06V 20/56 (2022.01)
G06V 20/52 (2022.01)

(21) International Application Number:

PCT/US2024/024618

(22) International Filing Date:

15 April 2024 (15.04.2024)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

18/136,166 18 April 2023 (18.04.2023) US

(63) Related by continuation (CON) or continuation-in-part (CIP) to earlier application:

US 18/136,166 (CON)

Filed on 18 April 2023 (18.04.2023)

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

(72) Inventors: GURGUL, Piotr; 3000 31st Street, Santa Monica, California 90405 (US). MOLL, Sharon; 3000 31st Street, Santa Monica, California 90405 (US).

(74) Agent: PERDOK, Monique M. et al.; 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, 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,

(54) Title: IMAGE CAPTURING WHILE CIRCUMNAVIGATING OBJECTS USING MOBILE DEVICES

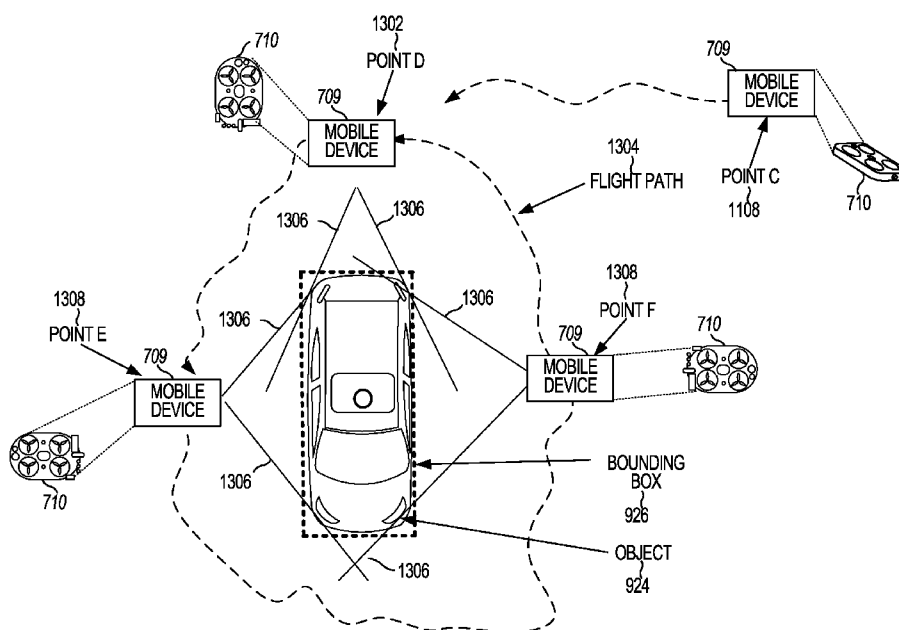


FIG. 14

(57) Abstract: Apparatuses, computer readable medium, and methods for image capturing while circumnavigating objects using mobile devices are disclosed. Example methods include capturing an image, processing the image to identify an object within the image, determining a path around the object and a number of images to capture of the object, dividing the path by the number of images to determine a number of waypoints, and navigating the mobile device to the waypoints and capturing an image of the object at each waypoint of the waypoints. Examples include a person pointing at an object and the mobile device identifying the object based on the person pointing at the object. The mobile device determines a bounding box and a geometric center of the bounding box to determine the path to circumnavigate the object. The mobile device determines a height above a ground to assist in navigation.

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))*

IMAGE CAPTURING WHILE CIRCUMNAVIGATING OBJECTS USING MOBILE DEVICES

PRIORITY CLAIM

[0001] This application claims the benefit of U.S. Patent Application Serial No. 18/136,166, filed on April 18, 2023, which is hereby incorporated by reference in its entirety.

TECHNICAL FIELD

[0002] Examples of the present disclosure relate generally to image capturing while circumnavigating objects using mobile device. More particularly, but not by way of limitation, the present disclosure addresses systems, methods, and apparatuses for determining a path around an object and navigating around the object while capturing images at fixed distances along the path. The captured images may be used for a 360-degree view of the object.

BACKGROUND

[0003] Mobile devices that provide photographic services to users are becoming more and more popular. But mobile device designs are limited by size and power constraints. And users of mobile devices continue to demand more and more services from the mobile devices.

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 non-limiting 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, in accordance with some examples.

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

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

[0008] FIG. 4 is a diagrammatic representation of a message, in accordance with some examples.

[0009] FIG. 5 is a flowchart for an access-limiting process, in accordance with some examples.

[0010] FIG. 6 illustrates examples of components for a mobile device, in accordance with some examples.

[0011] FIG. 7 is a schematic diagram illustrating a mobile device system, in accordance with some examples.

[0012] FIG. 8 illustrates a mobile device, in accordance with some examples.

[0013] FIG. 9 illustrates a system 900 for image capturing while circumnavigating objects using mobile devices, in accordance with some examples.

[0014] FIG. 10 illustrates unlocking the mobile device, in accordance with some embodiments.

[0015] FIG. 11 illustrates a mobile device flying, in accordance with some embodiments.

[0016] FIG. 12 illustrates a person selecting an object with a mobile device, in accordance with some embodiments.

[0017] FIG. 13 illustrates a system for object selection, in accordance with some examples.

[0018] FIG. 14 illustrates the mobile device determining a bounding box for the object, in accordance with some embodiments.

[0019] FIG. 15 illustrates an image capture user interface (UI), in accordance with some embodiments.

[0020] FIG. 16 illustrates a system for adjusting parameters for image capturing while circumnavigating objects using mobile devices, in accordance with some examples.

[0021] FIG. 17 illustrates a system for image capturing while circumnavigating objects using mobile devices, in accordance with some examples.

[0022] FIG. 18 illustrates a system for image capturing while circumnavigating objects using mobile devices, in accordance with some examples.

[0023] FIG. 19 illustrates a system for image capturing while circumnavigating objects using mobile devices, in accordance with some examples.

[0024] FIG. 20 is a perspective view of a head-wearable apparatus in the form of glasses, in accordance with some examples.

[0025] FIG. 21 illustrates a method for image capturing while circumnavigating objects using mobile devices, in accordance with some examples.

[0026] FIG. 22 is a diagrammatic representation of a machine in the form of a computer system within which a set of instructions may be executed for causing the machine to perform any one or more of the methodologies discussed herein, in accordance with some examples.

[0027] FIG. 23 is a block diagram showing a software architecture within which examples may be implemented.

[0028] FIG. 24 is a diagrammatic representation of a processing environment, in accordance with some examples.

DETAILED DESCRIPTION

[0029] 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 of the inventive subject matter. It will be evident, however, to those skilled in the art, that examples of the inventive subject matter may be practiced without these specific details. In general, well-known instruction instances, protocols, structures, and techniques are not necessarily shown in detail.

[0030] Often images of products are captured for marketing purposes. The products may be placed in a machine that captures images of the product by spinning the product or a camera around the product and capturing images. The

images provide a 360-degree view of the product. However, the machines that provide the images for a 360-degree view may be expensive and may only accommodate products that fit within the machine.

[0031] Referring to FIGS. 6, 7, 9, and 19, a challenge is how to provide a 360-degree view of an object 924 with images 914. The challenge is addressed by a mobile device 708, 709 that identifies an object 924, determines a bounding box 926, and a path 934 around the object 924 and captures images at image capture 1902 locations. The mobile device 708, 709 may be a smartphone, a wearable device 719, an augmented reality (AR), mixed reality (MR), or virtual reality (VR) wearable device 719, a self-propelled device 710, an autonomous device, and so forth. A navigation system 916, which may be resident within the mobile device 708, 709, navigates the mobile device 708, 709 by either controlling a propulsion system 684 to move the mobile device 708, 709 or instructing a person to move the mobile device 708, 709, in accordance with some examples.

[0032] The navigation system 916 navigates the mobile device 708, 709 to capture images of the object 924 and determines a bounding box 926 around the object 924 based on the images. The bounding box 926 enables the navigation system 916 to determine a geometric center 925 of the bounding box 926. The navigation system 916 determines a path 934 around the object 924 based on the geometric center 925 and a size of the object 924. The navigation system 916 then navigates the mobile device 708, 709 around the path 934 capturing images 914 at intervals around the path 934. The navigation system 916 directly controls the mobile device 708, 709 or instructs a person 718 to move the mobile device 708, 709. The captured images 914 can be shown in rapid succession to provide a 360-degree view of the object 924, which may be a product. The path 934 may be limited to a ground path or a water path.

[0033] Additionally, the autonomous device 710 provides the person 718 with the option of adjusting the parameters for taking the images 914 such as the number of images 914 to take while navigating along the path 934. The navigation system 916 may present the object selection UI 1302 of FIG. 13 to the person 718 to enable the person 718 to select the object 924 from captured images. The mobile device 708, 709 in the example of a self-propelled device 710 may confirm with the person 718 that the proper object 924 was identified

by navigating towards the object 924 and then turning back to the person 718 for a hand signal that indicates the correct object 924 was selected.

NETWORKED COMPUTING ENVIRONMENT

[0034] FIG. 1 is a block diagram showing an example messaging system 100 for exchanging data (e.g., messages and associated content) over a network. The messaging system 100 includes multiple instances of a client device 102, each of which hosts a number of applications, including a messaging client 104 and other applications 106. Each messaging client 104 is communicatively coupled to other instances of the messaging client 104 (e.g., hosted on respective other client devices 102), a messaging server system 108 and third-party servers 110 via a network 112 (e.g., the Internet). A messaging client 104 can also communicate with locally-hosted applications 106 using Applications Program Interfaces (APIs).

[0035] A messaging client 104 is able to communicate and exchange data with other messaging clients 104 and with the messaging server system 108 via the network 112. The data exchanged between messaging clients 104, and between a messaging client 104 and the messaging server system 108, includes functions (e.g., commands to invoke functions) as well as payload data (e.g., text, audio, video or other multimedia data).

[0036] The messaging server system 108 provides server-side functionality via the network 112 to a particular messaging client 104. While certain functions of the messaging system 100 are described herein as being performed by either a messaging client 104 or by the messaging server system 108, the location of certain functionality either within the messaging client 104 or the messaging server system 108 may be a design choice. For example, it may be technically preferable to initially deploy certain technology and functionality within the messaging server system 108 but to later migrate this technology and functionality to the messaging client 104 where a client device 102 has sufficient processing capacity.

[0037] The messaging server system 108 supports various services and operations that are provided to the messaging client 104. Such operations include transmitting data to, receiving data from, and processing data generated by the messaging client 104. This data may include message content, client

device information, geolocation information, media augmentation and overlays, message content persistence conditions, social network information, and live event information, as examples. Data exchanges within the messaging system 100 are invoked and controlled through functions available via user interfaces (UIs) of the messaging client 104.

[0038] Turning now specifically to the messaging server system 108, an Application Program Interface (API) server 116 is coupled to, and provides a programmatic interface to, application servers 114. The application servers 114 are communicatively coupled to a database server 120, which facilitates access to a database 126 that stores data associated with messages processed by the application servers 114. Similarly, a web server 128 is coupled to the application servers 114, and provides web-based interfaces to the application servers 114. To this end, the web server 128 processes incoming network requests over the Hypertext Transfer Protocol (HTTP) and several other related protocols.

[0039] The Application Program Interface (API) server 116 receives and transmits message data (e.g., commands and message payloads) between the client device 102 and the application servers 114. Specifically, the Application Program Interface (API) server 116 provides a set of interfaces (e.g., routines and protocols) that can be called or queried by the messaging client 104 in order to invoke functionality of the application servers 114. The Application Program Interface (API) server 116 exposes various functions supported by the application servers 114, including account registration, login functionality, the sending of messages, via the application servers 114, from a particular messaging client 104 to another messaging client 104, the sending of media files (e.g., images or video) from a messaging client 104 to a messaging server 118, and for possible access by another messaging client 104, the settings of a collection of media data (e.g., story), the retrieval of a list of friends of a user of a client device 102, the retrieval of such collections, the retrieval of messages and content, the addition and deletion of entities (e.g., friends) to an entity graph (e.g., a social graph), the location of friends within a social graph, and opening an application event (e.g., relating to the messaging client 104).

[0040] The application servers 114 host a number of server applications and subsystems, including for example a messaging server 118, an image processing server 122, and a social network server 124. The messaging server 118 implements a number of message processing technologies and functions, particularly related to the aggregation and other processing of content (e.g., textual and multimedia content) included in messages received from multiple instances of the messaging client 104. As will be described in further detail, the text and media content from multiple sources may be aggregated into collections of content (e.g., called stories or galleries). These collections are then made available to the messaging client 104. Other processor and memory intensive processing of data may also be performed server-side by the messaging server 118, in view of the hardware requirements for such processing.

[0041] The application servers 114 also include an image processing server 122 that is dedicated to performing various image processing operations, typically with respect to images or video within the payload of a message sent from or received at the messaging server 118.

[0042] The social network server 124 supports various social networking functions and services and makes these functions and services available to the messaging server 118. To this end, the social network server 124 maintains and accesses an entity graph 308 (as shown in FIG. 3) within the database 126. Examples of functions and services supported by the social network server 124 include the identification of other users of the messaging system 100 with which a particular user has relationships or is “following,” and also the identification of other entities and interests of a particular user.

[0043] Returning to the messaging client 104, features and functions of an external resource (e.g., an application 106 or applet) are made available to a user via an interface of the messaging client 104. In this context, “external” refers to the fact that the application 106 or applet is external to the messaging client 104. The external resource is often provided by a third party but may also be provided by the creator or provider of the messaging client 104. The messaging 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 client device 102 (e.g., a “native app”), or a

small-scale version of the application (e.g., an “applet”) that is hosted on the client device 102 or remote of the client device 102 (e.g., on third-party servers 110). 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 one example, 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 messaging 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).

[0044] In response to receiving a user selection of the option to launch or access features of the external resource, the messaging 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 client device 102 can be launched independently of and separately from the messaging client 104, such as by selecting an icon, corresponding to the application 106, on a home screen of the client device 102. Small-scale versions of such applications can be launched or accessed via the messaging client 104 and, in some examples, no or limited portions of the small-scale application can be accessed outside of the messaging client 104. The small-scale application can be launched by the messaging client 104 receiving, from a third-party server 110 for example, a markup-language document associated with the small-scale application and processing such a document.

[0045] In response to determining that the external resource is a locally-installed application 106, the messaging client 104 instructs the client device 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 messaging client 104 communicates with the third-party servers 110 (for example) to obtain a markup-language document corresponding to the selected external resource. The messaging client 104 then processes the obtained markup-language document to present the web-based external resource within a user interface of the messaging client 104.

[0046] The messaging client 104 can notify a user of the client device 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 messaging client 104 can provide participants in a conversation (e.g., a chat session) in the messaging 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 messaging clients 104, with the ability to share an item, status, state, or location in an external resource with one or more members of a group of users into a chat session. 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 messaging client 104. The external resource can selectively include different media items in the responses, based on a current context of the external resource.

[0047] The messaging 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).

[0048] SYSTEM ARCHITECTURE

[0049] FIG. 2 is a block diagram illustrating further details regarding the messaging system 100, according to some examples. Specifically, the messaging system 100 is shown to comprise the messaging client 104 and the application servers 114. The messaging system 100 embodies a number of subsystems, which are supported on the client-side by the messaging client 104 and on the server-side by the application servers 114. These subsystems include, for example, an ephemeral timer system 202, a collection management system 204,

an augmentation system 208, a map system 210, a game system 212, an external resource system 214, and a mobile device management system 216.

[0050] The ephemeral timer system 202 is responsible for enforcing the temporary or time-limited access to content by the messaging client 104 and the messaging server 118. The ephemeral timer system 202 incorporates a number of timers 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 messaging client 104. Further details regarding the operation of the ephemeral timer system 202 are provided below.

[0051] The collection management system 204 is 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 204 may also be responsible for publishing an icon that provides notification of the existence of a particular collection to the user interface of the messaging client 104.

[0052] The collection management system 204 furthermore includes a curation interface 206 that allows a collection manager to manage and curate a particular collection of content. For example, the curation interface 206 enables an event organizer to curate a collection of content relating to a specific event (e.g., delete inappropriate content or redundant messages). Additionally, the collection management system 204 employs machine vision (or image recognition technology) and content rules to automatically curate a content collection. In certain examples, compensation may be paid to a user for the inclusion of user-generated content into a collection. In such cases, the collection management system 204 operates to automatically make payments to such users for the use of their content.

[0053] The augmentation system 208 provides various functions that enable a user to augment (e.g., annotate or otherwise modify or edit) media content

associated with a message. For example, the augmentation system 208 provides functions related to the generation and publishing of media overlays for messages processed by the messaging system 100. The augmentation system 208 operatively supplies a media overlay or augmentation (e.g., an image filter) to the messaging client 104 based on a geolocation of the client device 102. In another example, the augmentation system 208 operatively supplies a media overlay to the messaging client 104 based on other information, such as social network information of the user of the client device 102. A media overlay 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, a digital object,) at the client device 102. For example, the media overlay may include text or image that can be overlaid on top of a photograph taken by the client device 102. In another example, the media overlay includes an identification of 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 another example, the augmentation system 208 uses the geolocation of the client device 102 to identify a media overlay that includes the name of a merchant at the geolocation of the client device 102. The media overlay may include other indicia associated with the merchant. The media overlays may be stored in the database 126 and accessed through the database server 120.

[0054] In some examples, the augmentation system 208 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 augmentation system 208 generates a media overlay that includes the uploaded content and associates the uploaded content with the selected geolocation.

[0055] In other examples, the augmentation system 208 provides a merchant-based publication platform that enables merchants to select a particular media overlay associated with a geolocation via a bidding process. For example, the augmentation system 208 associates the media overlay of the highest bidding merchant with a corresponding geolocation for a predefined amount of time.

[0056] The map system 210 provides various geographic location functions and supports the presentation of map-based media content and messages by the messaging client 104. For example, the map system 210 enables the display of user icons or avatars (e.g., stored in profile data 316) 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 messaging 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 messaging 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 messaging system 100 via the messaging client 104, with this location and status information being similarly displayed within the context of a map interface of the messaging client 104 to selected users.

[0057] The game system 212 provides various gaming functions within the context of the messaging client 104. The messaging client 104 provides a game interface providing a list of available games that can be launched by a user within the context of the messaging client 104 and played with other users of the messaging system 100. The messaging 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 messaging client 104. The messaging client 104 also supports both the voice 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).

[0058] The external resource system 214 provides an interface for the messaging client 104 to communicate with remote servers (e.g., third-party servers 110) to launch or access external resources, i.e., applications or applets. Each third-party server 110 hosts, for example, a markup language (e.g., HTML5) based application or small-scale version of an application (e.g., game, utility, payment, or ride-sharing application). The messaging client 104 may launch a web-based resource (e.g., application) by accessing the HTML5 file from the third-party servers 110 associated with the web-based resource. In certain examples, applications hosted by third-party servers 110 are programmed in JavaScript leveraging a Software Development Kit (SDK)

provided by the messaging server 118. The SDK includes Application Programming Interfaces (APIs) with functions that can be called or invoked by the web-based application. In certain examples, the messaging server 118 includes a JavaScript library that provides a given external resource access to certain user data of the messaging client 104. HTML5 is used as an example technology for programming games, but applications and resources programmed based on other technologies can be used.

[0059] In order to integrate the functions of the SDK into the web-based resource, the SDK is downloaded by a third-party server 110 from the messaging server 118 or is otherwise received by the third-party server 110. 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 messaging client 104 into the web-based resource.

[0060] The SDK stored on the messaging server 118 effectively provides the bridge between an external resource (e.g., applications 106 or applets and the messaging client 104. This provides the user with a seamless experience of communicating with other users on the messaging client 104, while also preserving the look and feel of the messaging client 104. To bridge communications between an external resource and a messaging client 104, in certain examples, the SDK facilitates communication between third-party servers 110 and the messaging client 104. In certain examples, a WebViewJavaScriptBridge running on a client device 102 establishes two one-way communication channels between an external resource and the messaging client 104. Messages are sent between the external resource and the messaging 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.

[0061] By using the SDK, not all information from the messaging client 104 is shared with third-party servers 110. The SDK limits which information is shared based on the needs of the external resource. In certain examples, each third-party server 110 provides an HTML5 file corresponding to the web-based

external resource to the messaging server 118. The messaging server 118 can add a visual representation (such as a box art or other graphic) of the web-based external resource in the messaging client 104. Once the user selects the visual representation or instructs the messaging client 104 through a GUI of the messaging client 104 to access features of the web-based external resource, the messaging client 104 obtains the HTML5 file and instantiates the resources necessary to access the features of the web-based external resource.

[0062] The messaging client 104 presents a graphical user interface (e.g., a landing page or title screen) for an external resource. During, before, or after presenting the landing page or title screen, the messaging client 104 determines whether the launched external resource has been previously authorized to access user data of the messaging client 104. In response to determining that the launched external resource has been previously authorized to access user data of the messaging client 104, the messaging client 104 presents another graphical user interface 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 messaging 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 messaging client 104 slides up (e.g., animates a menu as surfacing from a bottom of the screen to a middle of 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 messaging client 104 adds the external resource to a list of authorized external resources and allows the external resource to access user data from the messaging client 104. In some examples, the external resource is authorized by the messaging client 104 to access the user data in accordance with an OAuth 2 framework.

[0063] The messaging 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., only two-dimensional 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, two-dimensional avatars of users, three-dimensional 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.

[0064] The mobile device management system 216 supports, referring to FIG. 7, the off-site client device 704, the server 706, the mobile device 708, 709, wearable device 719, and the self-propelled device 710, in accordance with some examples. The mobile device management system 216 receives requests from the off-site client device 704, the server 706, the mobile device 708, 709, wearable device 719, and/or the self-propelled device 710 and responds to the requests. In some examples, an associated application 717 is resident in a host device such as the client devices 102, off-site client device 704, server 706, and/or mobile device 708, 709. For example, referring to FIGS. 6, 7, and 9, the associated application 717 enables the user to set thresholds, the path 934, height 932, object 924, center 925, bounding box 926, number of images 928, flight plans for the control knob 614, and other preferences. In some examples, the mobile device management system 216 provides functions and routines to assist the navigation system 916.

DATA ARCHITECTURE

[0065] FIG. 3 is a schematic diagram illustrating data structures 300, which may be stored in the database 126 of the messaging server system 108, according to certain examples. While the content of the database 126 is shown to comprise a number of tables, it will be appreciated that the data could be stored in other types of data structures (e.g., as an object-oriented database).

[0066] The database 126 includes message data stored within a message table 302. 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 302 is described below with reference to FIG. 4.

[0067] An entity table 306 stores entity data, and is linked (e.g., referentially) to an entity graph 308 and profile data 316. Entities for which records are maintained within the entity table 306 may include individuals, corporate entities, organizations, objects, places, events, and so forth. Regardless of entity type, any entity regarding which the messaging server system 108 stores data may be a recognized entity. Each entity is provided with a unique identifier, as well as an entity type identifier (not shown).

[0068] The entity graph 308 stores information regarding relationships and associations between entities. Such relationships may be social, professional (e.g., work at a common corporation or organization) interested-based or activity-based, merely for example.

[0069] The profile data 316 stores multiple types of profile data about a particular entity. The profile data 316 may be selectively used and presented to other users of the messaging system 100, based on privacy settings specified by a particular entity. Where the entity is an individual, the profile data 316 includes, for example, a username, telephone number, address, settings (e.g., notification and privacy settings), as well as 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 messaging system 100, and on map interfaces displayed by messaging 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 316 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 126 also stores augmentation data, such as overlays or filters, in an augmentation table 310. The augmentation data is associated with and applied to videos (for which data is stored in a video table 304) and images (for which data is stored in an image table 312).

[0072] Filters, in one example, 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 messaging 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 messaging client 104, based on geolocation information determined by a Global Positioning System (GPS) unit of the client device 102.

[0073] Another type of filter is a data filter, which may be selectively presented to a sending user by the messaging client 104, based on other inputs or information gathered by the client device 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 client device 102, or the current time.

[0074] Other augmentation data that may be stored within the image table 312 includes augmented reality content items (e.g., corresponding to applying Lenses or augmented reality experiences). An augmented reality content item may be a real-time special effect and sound that may be added to an image or a video.

[0075] As described above, augmentation data includes augmented reality content items, overlays, image transformations, AR images, and similar terms refer to modifications that may be applied to image data (e.g., videos or images). This includes real-time modifications, which modify an image as it is captured using device sensors (e.g., one or multiple cameras) of a client device 102 and then displayed on a screen of the client device 102 with the modifications. This also includes modifications to stored content, such as video clips in a gallery that may be modified. For example, in a client device 102 with access to multiple augmented reality content items, a user can use a single video clip with multiple augmented reality content items to see how the different augmented reality content items will modify the stored clip. For example, multiple augmented reality content items that apply different pseudorandom

movement models can be applied to the same content by selecting different augmented reality content items for the content. Similarly, real-time video capture may be used with an illustrated modification to show how video images currently being captured by sensors of a client device 102 would modify the captured data. Such data may simply be displayed on the screen and not stored in memory, or the content captured by the device sensors may be recorded and stored in memory with or without the modifications (or both). In some systems, a preview feature can show how different augmented reality content items will look within different windows in a display at the same time. This can, for example, enable multiple windows with different pseudorandom animations to be viewed on a display at the same time.

[0076] Data and various systems using augmented reality content items or other such transform systems to modify content using this data can thus involve detection of objects (e.g., faces, hands, bodies, cats, dogs, surfaces, objects, etc.), tracking of such objects as they leave, enter, and move around the field of view in video frames, and the modification or transformation of such objects as they are tracked. In various examples, different methods for achieving such transformations may be used. Some examples may involve generating a three-dimensional mesh model of the object or objects and using transformations and animated textures of the model within the video to achieve the transformation. In other examples, tracking of points on an object may be used to place an image or texture (which may be two dimensional or three dimensional) at the tracked position. In still further examples, neural network analysis of video frames may be used to place images, models, or textures in content (e.g., images or frames of video). Augmented reality content items thus refer both to the images, models, and textures used to create transformations in content, as well as to additional modeling and analysis information needed to achieve such transformations with object detection, tracking, and placement.

[0077] Real-time video processing can be performed with any kind of video data (e.g., video streams, video files, etc.) saved in a memory of a computerized system of any kind. For example, a user can load video files and save them in a memory of a device or can generate a video stream using sensors of the device. Additionally, any objects can be processed using a computer animation

model, such as a human's face and parts of a human body, animals, or non-living things such as chairs, cars, or other objects.

[0078] In some examples, when a particular modification is selected along with content to be transformed, elements to be transformed are identified by the computing device, and then detected and tracked if they are present in the frames of the video. The elements of the object are modified according to the request for modification, thus transforming the frames of the video stream. Transformation of frames of a video stream can be performed by different methods for different kinds of transformation. For example, for transformations of frames mostly referring to changing forms of object's elements characteristic points for each element of an object are calculated (e.g., using an Active Shape Model (ASM) or other known methods). Then, a mesh based on the characteristic points is generated for each of the at least one element of the object. This mesh is used in the following stage of tracking the elements of the object in the video stream. In the process of tracking, the mentioned mesh for each element is aligned with a position of each element. Then, additional points are generated on the mesh. A first set of first points is generated for each element based on a request for modification, and a set of second points is generated for each element based on the set of first points and the request for modification. Then, the frames of the video stream can be transformed by modifying the elements of the object on the basis of the sets of first and second points and the mesh. In such methods, a background of the modified object can be changed or distorted as well by tracking and modifying the background.

[0079] In some examples, transformations changing some areas of an object using its elements can be performed by calculating characteristic points for each element of an object and generating a mesh based on the calculated characteristic points. Points are generated on the mesh, and then various areas based on the points are generated. The elements of the object are then tracked by aligning the area for each element with a position for each of the at least one element, and properties of the areas can be modified based on the request for modification, thus transforming the frames of the video stream. Depending on the specific request for modification properties of the mentioned areas can be transformed in different ways. Such modifications may involve changing color

of areas; removing at least some part of areas from the frames of the video stream; including one or more new objects into areas that are based on a request for modification; and modifying or distorting the elements of an area or object. In various examples, any combination of such modifications or other similar modifications may be used. For certain models to be animated, some characteristic points can be selected as control points to be used in determining the entire state-space of options for the model animation.

[0080] In some examples of a computer animation model to transform image data using face detection, the face is detected on an image with the use of a specific face detection algorithm (e.g., Viola-Jones). Then, an Active Shape Model (ASM) algorithm is applied to the face region of an image to detect facial feature reference points.

[0081] Other methods and algorithms suitable for face detection can be used. For example, in some examples, features are located using a landmark, which represents a distinguishable point present in most of the images under consideration. For facial landmarks, for example, the location of the left eye pupil may be used. If an initial landmark is not identifiable (e.g., if a person has an eyepatch), secondary landmarks may be used. Such landmark identification procedures may be used for any such objects. In some examples, a set of landmarks forms a shape. Shapes can be represented as vectors using the coordinates of the points in the shape. One shape is aligned to another with a similarity transform (allowing translation, scaling, and rotation) that minimizes the average Euclidean distance between shape points. The mean shape is the mean of the aligned training shapes.

[0082] In some examples, a search for landmarks from the mean shape aligned to the position and size of the face determined by a global face detector is started. Such a search then repeats the steps of suggesting a tentative shape by adjusting the locations of shape points by template matching the image texture around each point and then conforming the tentative shape to a global shape model until convergence occurs. In some systems, individual template matches are unreliable, and the shape model pools the results of the weak template matches to form a stronger overall classifier. The entire search is repeated at each level in an image pyramid, from coarse to fine resolution.

[0083] A transformation system can capture an image or video stream on a client device (e.g., the client device 102) and perform complex image manipulations locally on the client device 102 while maintaining a suitable user experience, computation time, and power consumption. The complex image manipulations may include size and shape changes, emotion transfers (e.g., changing a face from a frown to a smile), state transfers (e.g., aging a subject, reducing apparent age, changing gender), style transfers, graphical element application, and any other suitable image or video manipulation implemented by a convolutional neural network that has been configured to execute efficiently on the client device 102.

[0084] In some examples, a computer animation model to transform image data can be used by a system where a user may capture an image or video stream of the user (e.g., a selfie) using a client device 102 having a neural network operating as part of a messaging client 104 operating on the client device 102. The transformation system operating within the messaging client 104 determines the presence of a face within the image or video stream and provides modification icons associated with a computer animation model to transform image data, or the computer animation model can be present as associated with an interface described herein. The modification icons include changes that may be the basis for modifying the user's face within the image or video stream as part of the modification operation. Once a modification icon is selected, the transform system initiates a process to convert the image of the user to reflect the selected modification icon (e.g., generate a smiling face on the user). A modified image or video stream may be presented in a graphical user interface displayed on the client device 102 as soon as the image or video stream is captured, and a specified modification is selected. The transformation system may implement a complex convolutional neural network on a portion of the image or video stream to generate and apply the selected modification. That is, the user may capture the image or video stream and be presented with a modified result in real-time or near real-time once a modification icon has been selected. Further, the modification may be persistent while the video stream is being captured, and the selected modification icon remains toggled. Machine-taught neural networks may be used to enable such modifications.

[0085] The graphical user interface, presenting the modification performed by the transform system, may supply the user with additional interaction options. Such options may be based on the interface used to initiate the content capture and selection of a particular computer animation model (e.g., initiation from a content creator user interface). In various examples, a modification may be persistent after an initial selection of a modification icon. The user may toggle the modification on or off by tapping or otherwise selecting the face being modified by the transformation system and store it for later viewing or browsing to other areas of the imaging application. Where multiple faces are modified by the transformation system, the user may toggle the modification on or off globally by tapping or selecting a single face modified and displayed within a graphical user interface. In some examples, individual faces, among a group of multiple faces, may be individually modified, or such modifications may be individually toggled by tapping or selecting the individual face or a series of individual faces displayed within the graphical user interface.

[0086] A story table 314 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 306). A user may create a “personal story” in the form of a collection of content that has been created and sent/broadcasted by that user. To this end, the user interface of the messaging 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.

[0087] 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 messaging client 104, to contribute content to a particular live story. The live story may be identified to the user by the messaging client 104, based on his or her location. The end result is a “live story” told from a community perspective.

[0088] A further type of content collection is known as a “location story,” which enables a user whose client device 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 require 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).

[0089] As mentioned above, the video table 304 stores video data that, in one example, is associated with messages for which records are maintained within the message table 302. Similarly, the image table 312 stores image data associated with messages for which message data is stored in the entity table 306. The entity table 306 may associate various augmentations from the augmentation table 310 with various images and videos stored in the image table 312 and the video table 304.

[0090] DATA COMMUNICATIONS ARCHITECTURE

[0091] FIG. 4 is a schematic diagram illustrating a structure of a message 400, according to some examples, generated by a messaging client 104 for communication to a further messaging client 104 or the messaging server 118. The content of a particular message 400 is used to populate the message table 302 stored within the database 126, accessible by the messaging server 118. Similarly, the content of a message 400 is stored in memory as “in-transit” or “in-flight” data of the client device 102 or the application servers 114. 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 client device 102, and that is included in the message 400.
- message image payload 406: image data, captured by a camera component of a client device 102 or retrieved from a memory component of a client device 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 312.
- message video payload 408: video data, captured by a camera component or retrieved from a memory component of the client device 102, and that is

included in the message 400. Video data for a sent or received message 400 may be stored in the video table 304.

- message audio payload 410: audio data, captured by a microphone or retrieved from a memory component of the client device 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 310.
- 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 messaging 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 into 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 story table 314) 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 Client device 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 client device 102 to which the message 400 is addressed.

[0092] 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 312. Similarly, values within the message video payload 408 may point to data stored within a video table 304, values stored within the message augmentations 412 may point to data stored in an augmentation table 310, values stored within the message story identifier 418 may point to data stored in a story table 314, 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 306.

[0093] TIME-BASED ACCESS LIMITATION ARCHITECTURE

[0094] FIG. 5 is a schematic diagram illustrating an access-limiting process 500, in terms of which access to content (e.g., an ephemeral message 502, and associated multimedia payload of data) or a content collection (e.g., an ephemeral message group 504) may be time-limited (e.g., made ephemeral).

[0095] An ephemeral message 502 is shown to be associated with a message duration parameter 506, the value of which determines an amount of time that the ephemeral message 502 will be displayed to a receiving user of the ephemeral message 502 by the messaging client 104. In one example, an ephemeral message 502 is viewable by a receiving user for up to a maximum of 10 seconds, depending on the amount of time that the sending user specifies using the message duration parameter 506.

[0096] The message duration parameter 506 and the message receiver identifier 424 are shown to be inputs to a message timer 510, which is responsible for determining the amount of time that the ephemeral message 502 is shown to a particular receiving user identified by the message receiver identifier 424. In

particular, the ephemeral message 502 will only be shown to the relevant receiving user for a time period determined by the value of the message duration parameter 506. The message timer 510 is shown to provide output to a more generalized ephemeral timer system 202, which is responsible for the overall timing of display of content (e.g., an ephemeral message 502) to a receiving user.

[0097] The ephemeral message 502 is shown in FIG. 5 to be included within an ephemeral message group 504 (e.g., a collection of messages in a personal story, or an event story). The ephemeral message group 504 has an associated group duration parameter 508, a value of which determines a time duration for which the ephemeral message group 504 is presented and accessible to users of the messaging system 100. The group duration parameter 508, for example, may be the duration of a music concert, where the ephemeral message group 504 is a collection of content pertaining to that concert. Alternatively, a user (either the owning user or a curator user) may specify the value for the group duration parameter 508 when performing the setup and creation of the ephemeral message group 504.

[0098] Additionally, each ephemeral message 502 within the ephemeral message group 504 has an associated group participation parameter 512, a value of which determines the duration of time for which the ephemeral message 502 will be accessible within the context of the ephemeral message group 504. Accordingly, a particular ephemeral message group 504 may “expire” and become inaccessible within the context of the ephemeral message group 504, prior to the ephemeral message group 504 itself expiring in terms of the group duration parameter 508. The group duration parameter 508, group participation parameter 512, and message receiver identifier 424 each provide input to a group timer 514, which operationally determines, firstly, whether a particular ephemeral message 502 of the ephemeral message group 504 will be displayed to a particular receiving user and, if so, for how long. Note that the ephemeral message group 504 is also aware of the identity of the particular receiving user as a result of the message receiver identifier 424.

[0099] Accordingly, the group timer 514 operationally controls the overall lifespan of an associated ephemeral message group 504, as well as an individual

ephemeral message 502 included in the ephemeral message group 504. In one example, each and every ephemeral message 502 within the ephemeral message group 504 remains viewable and accessible for a time period specified by the group duration parameter 508. In a further example, a certain ephemeral message 502 may expire, within the context of ephemeral message group 504, based on a group participation parameter 512. Note that a message duration parameter 506 may still determine the duration of time for which a particular ephemeral message 502 is displayed to a receiving user, even within the context of the ephemeral message group 504. Accordingly, the message duration parameter 506 determines the duration of time that a particular ephemeral message 502 is displayed to a receiving user, regardless of whether the receiving user is viewing that ephemeral message 502 inside or outside the context of an ephemeral message group 504.

[0100] The ephemeral timer system 202 may furthermore operationally remove a particular ephemeral message 502 from the ephemeral message group 504 based on a determination that it has exceeded an associated group participation parameter 512. For example, when a sending user has established a group participation parameter 512 of 24 hours from posting, the ephemeral timer system 202 will remove the relevant ephemeral message 502 from the ephemeral message group 504 after the specified 24 hours. The ephemeral timer system 202 also operates to remove an ephemeral message group 504 when either the group participation parameter 512 for each and every ephemeral message 502 within the ephemeral message group 504 has expired, or when the ephemeral message group 504 itself has expired in terms of the group duration parameter 508.

[0101] In certain use cases, a creator of a particular ephemeral message group 504 may specify an indefinite group duration parameter 508. In this case, the expiration of the group participation parameter 512 for the last remaining ephemeral message 502 within the ephemeral message group 504 will determine when the ephemeral message group 504 itself expires. In this case, a new ephemeral message 502, added to the ephemeral message group 504, with a new group participation parameter 512, effectively extends the life of an ephemeral message group 504 to equal the value of the group participation parameter 512.

[0102] Responsive to the ephemeral timer system 202 determining that an ephemeral message group 504 has expired (e.g., is no longer accessible), the ephemeral timer system 202 communicates with the messaging system 100 (and, for example, specifically the messaging client 104) to cause an indicium (e.g., an icon) associated with the relevant ephemeral message group 504 to no longer be displayed within a user interface of the messaging client 104. Similarly, when the ephemeral timer system 202 determines that the message duration parameter 506 for a particular ephemeral message 502 has expired, the ephemeral timer system 202 causes the messaging client 104 to no longer display an indicium (e.g., an icon or textual identification) associated with the ephemeral message 502.

IMAGE CAPTURING WHILE CIRCUMNAVIGATING OBJECTS USING DEVICE

[0103] FIG. 6 illustrates examples of components 600 for a mobile device, in accordance with some examples. The components 600 illustrated in FIG. 6 are part of a mobile device such as mobile device 708, 709, illustrated in FIGS. 7, 8, 10, 11, 14, 17, 18, and 19. The components 600 may be part of an autonomous device or controlled device such as the self-propelled device 710. The components 600 are organized into functional groups that include input/output devices 602, a processor 604, memory 606, a battery 608, a power chip 610, sensors 648, wireless connections 666, power and communications connections 646, and a propulsion system 684. One skilled in the art would recognize that the components 600 may be organized into different functional groups or may all individually be part of the self-propelled device.

[0104] The components 600 are connected via power and communications connections 646. The power and communication connections 646 include one or more communication buses, power buses, and/or point-to-point connections, in accordance with some examples. Additionally, one or more of the components 600 may be optional. And the components 600 may include additional components. Moreover, the number of the components as illustrated may be different. For example, there may be multiple processors 604. The terms electrical and electronic may be used to refer to either electrical components and/or electronic components.

[0105] The propulsion system 684 is included in some examples of the mobile device 708, 709. The propulsion system 684 includes electrical motors 686, where each electrical motor 686 includes a rotor 688 associated with a propeller 690. The propellers 690 provide aerodynamic lift to the self-propelled device 710, as well as to accelerate and rotate the self-propelled device 710, in accordance with some examples. The electrical motors 686 once actuated in response to signals including electrical motor operation instructions from, for example, the processor 604, spin the rotors 688, which spin the propellers 690. The electrical motors 686 and actuators 638 are powered by the battery 608 and/or power chip 610 and are controlled by signals from the processor 604. The electrical motors 686 are variable electrical motors in accordance with some examples. In some examples, the electrical motors 686 have a low setting, which the processor 604 may use to indicate to a user of the self-propelled device 710 that the self-propelled device 710 is preparing to takeoff. In some examples, in addition to or instead of the propellers 690, the propulsion system 684 is composed of one or more of: tires, gears, wheels, caterpillar tracks for a land device, water propellers, turbines or impellers for a water-based device, or another means of propulsion. Additionally, the self-propelled device 710 may be an airplane, boat, helicopter, and so forth, and the propulsion system 684 may include at least one engine for a water propeller 690, jet engine, or another type of propulsion system 684. Moreover, the self-propelled device 710 may be remote controlled by an associated application 717, which may be controlled by a person.

[0106] Having more than one propeller 690 enables the self-propelled device 710 to continue to fly when one or more of the electrical motors 686, rotors 688, or propellers 690 fail. For instance, if one of the electrical motors 686 fails, the self-propelled device 710 can still stay aloft with the remaining electrical motors 686 working in concert to compensate. In some examples, the propulsion system 684 sends a signal to the processor 604 that indicates an electrical motor 686 is not functioning properly. In some examples, the electrical motors 686 provide signals to the processor 604 that indicate whether the electrical motors 686 are operating properly. In addition, the greater the number electrical motors 686 are operating properly. In addition, the greater the number electrical motors 686 that are incorporated into the self-propelled device 710, the more lift the

self-propelled device 710 will generate, allowing the self-propelled device 710 to carry a heavier payload such as one or more sensors 648.

[0107] In some examples, the propulsion system 684 includes one or more actuators 638 that tilt the electrical motors 686 so that the electrical motors 686 may operate at an angle relative to a frame of reference of the self-propelled device 710. For example, each electrical motor 686 is rotationally mounted on the self-propelled device with a single axis of rotation where the actuator 638 controls the angle of the electrical motor 686. In some examples, each electrical motor 686 is rotationally mounted with two or more axes of rotation controlled by one or more actuators 638. In some examples, one or more actuators 638 control the angle of more than one electrical motor 686.

[0108] The functional groups include sensors 648 with components 600 including photography camera 650, navigation camera 651, inertial measurement unit (IMU) A 652, altimeter 654, gyroscope 656, IMU B 658, accelerometer 665, height detector 660, which includes light 662 and light detector 664, magnetic sensor 659, wind speed sensor 661, clock 663, motion sensor 667, microphone 669, orientation sensor 671, and so forth. In some examples, the sensors 648 are the sensors that are available on the mobile device 708, 709. For example, the mobile device 708, 709, such as a smartphone or glasses 1500

[0109] In some examples, the sensors 648 generate data that is processed by the processor 604 and stored as data in a memory 606 such as the memory 1 626 as data 630 or memory N 632 as data 636. For instance, the altimeter 654 is an instrument for determining attained altitude. So, when the self-propelled device is set to hover in place, the processor 604 uses the data from the altimeter 654 to determine a height and adjusts the propulsion system 684 to maintain that height.

[0110] Alternatively, or in addition, the height detector 660 is used to generate data that can be used to determine the height of the self-propelled device above the ground. For example, the light 662 of the height detector 660 is mounted on the bottom of the self-propelled device 710 to shine light down to the ground which bounces off and hits the light detector 664. The processor 604 uses the data generated from shining the light 662 and receiving the light at the light

detector 664 to determine a height of the self-propelled device above a ground based on a time-of-flight of the light and the speed of light. The light 662 is a suitable light source strong enough to produce a detectable reflection from the ground back to the light detector 664. In some examples, the light 662 emits electro-magnetic radiation at a specific wavelength that the light detector 664 is manufactured to detect. In some examples, the height is determined further based on a roll and pitch of the self-propelled device to account for the fact that the light 662 may not be shining light straight down. In some examples, the height detector 660 is based on sonar.

[0111] The IMU A 652 and IMU B 658 output measurements such as the self-propelled device's specific force, angular rate, and the orientation of the self-propelled device 710, using a combination of accelerometers, gyroscopes, and, optionally, magnetometers. The altimeter 654, gyroscope 656, accelerometer 665, and other sensors 648 may be replaced by the IMU A 652 and/or IMU B 658. Various combinations of sensors 648 may be used to generate the data needed to navigate the self-propelled device. In some examples, the sensors 648 include a lidar system, a radar system, a light sensor, or another form of sensor that may be used to assist in navigation and/or photography. In some examples, sensors 648 are included that enable the self-propelled device to determine a pitch, yaw, and roll of the self-propelled device. In some examples, the sensors 648 include a motion sensor 667 that does not require power but generates a signal based on the self-propelled device being moved. The motion sensor 667 may be used to generate an event that the processor 604 responds to.

[0112] The photography camera 650 as well as other sensors 648 generate data that may be captured for the purposes of displaying or saving the data for a user of the self-propelled device. The photography camera 650 comprises a digital sensor such as a charged-coupled device (CCD) or a complementary metal-oxide semiconductor (CMOS) that is divided into pixels. The sensor generates an electrical signal based on the light that strikes the sensor. In some examples, the photography camera 650 is mounted horizontally relative to an axis of propellers 690 of the self-propelled device 710 and the navigation camera 651 is mounted vertically relative to the axis of the propellers 690 and is directed downward.

[0113] In some examples, generated data is associated with, referring to FIG. 9, a position 952 of the self-propelled device 710 and a time 936. In some examples, IMU A 652 is coupled with the photography camera 650, which enables the processor 604 to digitally or physically stabilize the photography camera 650 for improved photography and determine if the photographs are not reliable because the photography camera 650 has not been stable. The photography camera 651 may be rotationally mounted on the body of the self-propelled device, being coupled to the body via one or more actuators or other mechanisms to control orientation of the photography cameras 650 relative to the body of the self-propelled device. The IMU B 658 is attached to the self-propelled device to provide data for navigation. The processor 604 uses the data generated by the IMUs for navigation and, in some examples, to stabilize the photography camera 650. In some examples, there is more than one photography camera 650.

[0114] The navigation camera 651 is mounted on a body of the self-propelled device 710 for providing data to assist in navigation of the self-propelled device. There may be more than one navigation camera 651. For example, the navigation may be a front camera that is mounted onto the body of the self-propelled device, where the navigation camera 651 is positioned to pick up images of the scene towards which the self-propelled device is directed. Additionally, or instead of forward mounting, the navigation camera 651 may be mounted vertically on the body of the self-propelled device, where the navigation camera 651 is positioned to pick up images of the terrain being overflown. The one or more navigation cameras 651 may be movably or fixedly mounted on the body of the self-propelled device, being coupled to the body via one or more actuators or other mechanisms to control orientation of the one or more navigation cameras 651 relative to the body of the self-propelled device. In some examples, the photography camera 650 is also used as a navigation camera 651. In some examples, the photography camera 650 provides a greater pixel resolution and requires more power to operate than the navigation camera 651.

[0115] Data such as video and digital images captured by the photography camera 650 and navigation camera 651 may be stored in memory 606 as data 630, 636. Further, data captured by the photography camera 650 and/or

navigation camera 651 may be streamed in near-real time wirelessly, using wireless connections, to an external device 682. Additionally, the self-propelled device may send or receive data, which includes instructions, to or from an external device 682.

[0116] The magnetic sensor 659 provides data regarding the orientation of the self-propelled device within a magnetic field. The wind speed sensor 661 indicates a wind speed, which may be an apparent wind speed and/or direction, which the processor 604 can use to estimate the true wind speed based on flight characteristics and settings of the self-propelled device such as the power to the electrical motors 686. The clock 663 generates data that indicates a time. In some embodiments, the clock 663 indicates a Greenwich Mean Time. In some embodiments, the clock 663 generates a time relative to an event such as the powering up of the self-propelled device. The microphone 669 turns sound waves into electrical signals that may be stored as data 630, 636 are processed by the processor 604. The orientation 671 sensor is a sensor that indicates the orientation of the self-propelled device 710. For example, the orientation 671 sensor indicates whether the self-propelled device is right side up or upside down.

[0117] The wireless connections 666 include one or more wireless protocols that may include radio waves and/or light waves. As illustrated, the wireless connections 666 include a GPS chip 668, a lower-power wireless chip 670 connected to a transceiver/antenna 672, and a higher-power wireless chip 676 connected to a transceiver/antenna 678. The GPS chip 668 is connected to an antenna/transceiver that may either be internal to the chip or external. The GPS chip 668 receives communications from satellites and uses the information from receiving signals from multiple satellites to determine a position of the self-propelled device. GPS chips 668 are higher-power chips 676. In some examples, the processor 604 receives set-up data from an external device 682 that is needed for the operation of the GPS chip 668 where the set-up data may include information about the satellites that the GPS chip 668 receives signals from where the data may include positional information about the satellites. Some examples do not include one or more of the components 600 such as the GPS chip 668.

[0118] The lower-power wireless chip 670 may include chips that perform one or more lower-power wireless protocols. For example, Bluetooth Low-Energy (BLE) may be used to communicate with nearer external devices 682. The higher-power wireless chip 676 includes chips that perform one or more higher-power wireless protocols. For example, 3GPP protocols and IEEE 802.11 protocols.

[0119] The input/output devices 602 provide input and output to the self-propelled device 710 that may be used by a user. The indicator lights 612 indicate a status of the self-propelled device and are visible to a user of the self-propelled device. For example, the indicator lights 612 may indicate on vs. off, a charge level, a charge state such as charging or not charging, a standby state, whether there are photographs or videos in the memory, whether the memory is full or not, whether the wireless connection 666 is on or off, and so forth. An electronic display such as an LCD display may be in addition to or replace the indicator lights 612. In some embodiments, one of the buttons 618 is a flight button that when pressed indicates to the self-propelled device 710 that it should take off. In some examples, the input/output devices 602 are the devices available on the mobile device 708.

[0120] The control knob 614, which may take other forms, is a knob mounted on the outside of the self-propelled device providing a user the ability to control the operation of the self-propelled device. The control knob 614 may be termed a control user interface device or another similar term, in accordance with some embodiments. The control knob 614 has a number of positions or states such as off, on, transfer for transferring data such as images out of the self-propelled device, various flight paths and behaviors, and so forth. The state 615 is an internal state that provides the processor 604 with information regarding the setting of the control knob 614. Other input and output user interface items may be used in addition to or instead of the control knob 614.

[0121] Connectors 616 are outside connectors that provide either power and/or data connections from external devices 682 to the self-propelled device. For example, there is a power connector 616 for charging the self-propelled device. The power connector 616 or another connector 616 may be used to transfer data to a host device or to receive power from another power source. In some

examples, the connector 616 is a wireless rechargeable connector 616 so that the self-propelled device is placed on or near a charging base. In some examples there is a connector 616 for a micro secure digital (SD) card or another external storage device.

[0122] In some examples, there are buttons 618 to perform one or more functions. For example, a button 618 that when pressed instructs the self-propelled device to perform whatever function is indicated by the control knob 614 such as takeoff and take a portrait photograph of the user of the self-propelled device as quickly as possible.

[0123] The processor 604 performs instructions 620 to process data 630, 636, and/or to control the operations of the self-propelled device. The instructions 620 are machine instructions specifying the operations that the processor 604 is to perform and may be stored in a cache memory that is part of the processor 604 chip. The power portion 1 622 through power portion N 624 indicate that the processor 604 is divided into different portions so that the power chip 610 may select which portions of the processor 604 to provide power to in accordance with different power states 641 of the self-propelled device. Throughout this discussion, the processor 604 is described as the actor in determining various functions but one skilled in the art will recognize that special purpose chips may be included in various components of the self-propelled device to perform specific functions. For example, the height detector 660 may include a processing circuitry that determines the height above the ground and outputs data indicating the height above the ground for consumption by the processor 604.

[0124] The memory 606 includes memory 1 626 through memory N 632. Memory 1 626 and memory N 632 includes instructions 628, 634 and data 630, 636, respectively. The memories are accessible to the processor 604 and one or more other components 600 via the power and communication connections 646. A memory of the memory 1 626 through memory N 632 is a main memory that is used for storing data generated by the sensors 648 and for other data such as communications to and from the wireless connections 666. The main memory is a dynamic memory such as a DRAM or RAM, in accordance with some examples. Another memory of memory 1 626 through memory N 632 is a static

memory such as a SRAM or ROM that does not need power to maintain a state. In some examples, another memory of memory 1 626 through memory N 632 is a storage unit that is a machine-readable medium. For example, the storage unit is a removable micro-SD card. The power chip 610 has connections to different memories so that the power chip 610 may provide power to one or more of the memories of memory 1 626 through memory N 632. The instructions 628, 634, and data 630, 636 reside, completely or partially, within the main memory, static memory, the storage unit, and/or within the processor 604 such as within a cache memory, or any suitable combination thereof. The main memory, static memory, the storage unit, and the memory of processor 604 are examples of machine-readable media.

[0125] The self-propelled device 710 may have preprogrammed paths 960, as illustrated in FIG. 9, that the navigation system 916 uses to control the path and operation of the self-propelled device 710. The preprogrammed paths 960 may indicate a path, a flight plan, or operations for the navigation system 916 to perform. The preprogrammed path 960 may be associated with positions or states 615 of the control knob 614 or another input/output device 602. The memory 606 stores the preprogrammed paths 960. In some examples, the self-propelled device 710 downloads new preprogrammed paths 960 from external devices 682. In some examples where the self-propelled device 710 does not fly, the preprogrammed paths 960 are used by the navigation system 916 to control the self-propelled device 710 as it traverses another medium such as the ground or water. In some examples, the navigation system 916 controls the self-propelled device 710, which may be a mobile device 708, by sending and receiving commands from a person interacting with the associated application 717.

[0126] Alternatively, or in addition to, movement of the self-propelled device 710 may be controlled by a remote controller such as an external device 682 that is a remote-control device that a pilot or user may use to launch, land, take photographs or video, and navigate the self-propelled device if the self-propelled device is not acting as a self-propelled device. In these embodiments, the self-propelled device is not acting as a self-propelled device but a remote-controlled device. Remote controllers can take many forms, from gamepad-like controllers to smartphones and tablets. Regardless of their design, remote

controllers require communication with the self-propelled device, and typically do that using radio waves. For example, devices are typically run by 2.4 gigahertz radio waves. To communicate with an self-propelled device, many device controllers use one of the communication protocols of IEEE 802.11, which may be termed Wi-Fi, which can be transmitted on the 2.4 gigahertz spectrum, and is used by smartphones and tablets for communication. In one example, referring to FIG. 7, device 710 communicates with mobile device 708, using Wi-Fi or BLE®. The external devices 682 include remote-control/host devices such as off-site client device 704, server 706, mobile device 708, or another device.

[0127] A power source is required to power the electrical motors 686 and power the other components 600. In some examples, the self-propelled device comprises one or more batteries 608 as sources of power for the components 600 such as the electrical motors 686. The batteries 608 are removable, in accordance with some examples. In some examples, the batteries 608 are rechargeable where one or more connectors 616 connect to the battery 608 either directly or via an electrical or electronic component. In some examples, a power chip 640 manages the batteries 608 by performing various functions such as determining a charge of the batteries 608, turning on or off the recharging, provisioning the output of the batteries 608 with capacitors, resistors, and/or inductors, and so forth. In some examples, the power chip 640 includes a power state 641, which indicates which of the components 600 of the self-propelled device are currently being powered. The power state 641 includes different power states 641 such as power state 1 642 through power state N 644. The different power states 641 provide power to different subsets of the components 600 and, thus, consume different amounts of power and provide different levels of functionality for the self-propelled device as is discussed herein. The different power states 641 are achieved by providing power to different sets of power and communications connections 646. In some examples, the components 600 are included as part of the machine 1900. In some examples, one or more of the components 600 are part of a motherboard with the processor 604 running a real-time operating system or an operating system such as Linux®.

[0128] FIG. 7 is a schematic diagram illustrating a mobile device system 700, in accordance with some examples. The mobile device may be mobile device 708, 709, self-propelled device 710, wearable device 719, and so forth. When the mobile device 708, 709 is not a self-propelled device 710, then mobile device 708, 709, may be moved by the person 718 or in another way. The mobile device 708, 709 may be autonomous or controlled by, for example, a person 718. The wearable device 719 may be the glasses 2000. The mobile device 708 may be the client device 102 such as a mobile phone. In some examples, the self-propelled device 710 is a self-propelled or a semi-self-propelled device. One skilled in the art would recognize that the mobile device 708, 709 may be another type of device.

[0129] The mobile device 708, 709, communicates by sending communications 712, 713 using wireless connections 666 of FIG. 6 to the remote-control/host device such as off-site client device 704, server 706, mobile phone or mobile device 708, or another device. The wireless network 702 is a cellular telephone network such as an LTE network, an IEEE 802.11 network, a Bluetooth® network, private network, or another wireless network using another wireless communication protocol. In some examples, a mobile device 709 such as the self-propelled device 710 communicates directly with another device such as mobile device 708, client device 704, and/or server 706 via communications 713 where communications 713 are sent using a communication protocol such as the communication protocols discussed for the wireless network 702.

[0130] In examples the mobile device 708, 709, such as self-propelled device 710 or wearable device 719 send communications 712, 713 that includes data and/or commands or requests to another device such as the mobile device 708. In some instances, communication between the remote-control/host device such as the mobile device 708 and the self-propelled device 710 may be via the wireless network 702. The wireless network 702 may include access to the internet and/or the mobile device 708, 709 may access the internet via another connected device. For example, wearable device 719 or self-propelled device 710 may connect to the internet via the mobile device 708 or mobile device 709, respectively.

[0131] In some examples, the server 706 provides a social networking service, to enable communication of content such as photos, videos, status updates, media content messages, and the like, directly to social-media sites such as Snapchat® from the mobile device 708, 709, which may be in motion such as in flight. In some examples, the server 706 is messaging server system 108 and the data captured by a photography camera of mobile device 708, 709 is broadcasted or otherwise communicated via a wireless network 702, which may be in near-real time, to a remote-control/host device such as mobile device 708, to servers 706, client devices 704, or another device. The self-propelled device 710 may be in contact with device management system 216 of FIG. 2 either directly or via another device.

[0132] One or more of the remote-control/host devices such as the mobile device 708, 709 may assist in processing of the data such as data 630 by receiving the data wirelessly, processing the data, and then sending back information wirelessly to the wearable device 718 or self-propelled device 710. For example, the mobile device 709 may receive an image from the self-propelled device 710 and determine that the image includes an object 924 of FIG. 9. The mobile device 708 may send back information that is used by the self-propelled device 710 to assist in a flight path 934 associated with the object 924. The mobile device 708, 709 may contact device management system 216 to perform functions, which may be for another mobile device 708, 709. The mobile device 708, 709 may contact the device management system 216 by sending commands to the device management system 216 such as store data for a user, request a user or purchaser of the self-propelled device, and so forth.

[0133] In some examples, an associated application 717 runs on the mobile device 708, 709 or the remote-control/host device. The associated application 717 may be used by a user or device to control a mobile device 708 such as a self-propelled device 710 or send instructions to the self-propelled device 710 such as return to user, take a particular set flight, move to the left, move to right, move up or down, tilt, take a set of photographs, turn off, and so forth. The associated application 717 may provide real-time or near real-time images of the videos that the mobile device 708, 709 is capturing. In some examples, the associated application 717 enables the person 718 to configure the mobile device 708, 709 such as the self-propelled device 710 by setting timeouts,

conditions, and/or thresholds. In some examples, the associated application 717 includes user interfaces and enables the user to select one or more parameters for the operation of the self-propelled device 710. For example, in some examples, referring to FIG. 16, the user can select a path 934, a height 932, a bounding box 926, a number of images 928, an object 924, a center 925, and so forth. In some examples, the person 718 can receive a real-time feed of images 914 captured by the mobile device 708, 709. In some examples, the associated application 717 provides instructions to the person to move the mobile device 708, which acts as the self-propelled device 710. For example, referring to FIG. 15, an image capture UI 1502 is used to guide the person 718 carrying the mobile device 708, 709 with an image capturing device.

[0134] Additionally, the user may select configurations regarding the wireless connections 666 to indicate which wireless protocols should be used in which states of the self-propelled device. In some examples, the remote-control/host device acts as a router or passes through messages or packets to other devices connected to the wireless network 702 directly or indirectly. For example, the mobile device 708, 709 receives an image via communications 713 from the self-propelled device 710. The mobile device 708, 709 takes the image and sends it to server 706 for posting on a social media site, which may be in near-real time. The server 706 may be hosting the device management system 216. A remote-control/host device such as the mobile device 708, 709 controls a state of the wearable device 719 or self-propelled device 710 by sending instructions to the wearable device 719 or self-propelled device 710 via communications 713, 712, in accordance with some examples. The associated application 717 be run by the mobile device 708, 709 such as the wearable device 719.

[0135] FIG. 8 illustrates an self-propelled device 710, in accordance with some examples. The self-propelled device 710 is illustrated from a bottom view of the self-propelled device 710. Referring to FIGS. 6 and 8, the self-propelled device 710 includes propeller 690, control knob 614, which may be on the top, height detector 660, batteries 608, photography camera 650, indicator lights 612, connector 616, center case 804, navigation camera 651, and flight button 806. In some examples, the navigation camera 651 is mounted horizontally the same or similar as the photography camera 650. In some examples, the navigation camera 651 and photography camera 650 are the same camera, which may have

different settings to reduce an image size and power consumption for navigation. The charging/transfer cable 802 is plugged into the connector 616. The center case 804 includes various components 600 such as the processor 604, memory 606, wireless connections 666, and so forth. In some examples, the self-propelled device 710 is plastic and approximately six inches in length and four inches in width. In some examples, the self-propelled device 710 is a quadrotor. The flight button 806 is a button 618 that when pressed indicates that the self-propelled device 710 should takeoff and perform a flight plan. The state 615 of the control knob 614 selects the preprogrammed path 960 of FIG. 9, in accordance with some embodiments.

[0136] FIG. 9 illustrates a system 900 for image capturing while circumnavigating objects using mobile devices, in accordance with some examples. The navigation system 916 guides a mobile device 708, 709 such as the wearable device 719 or self-propelled device 710 along a selected path 930 or to waypoints 910. The navigation system 916 controls the mobile device 708, 709, in cases where the mobile device 708, 709 is self-propelled such as for the self-propelled device 710. In cases where the mobile device 708, 709 is not self-propelled, then the navigation system 916 guides a person 718 to move the mobile device 708, 709, to the proper locations such via image capture UI 1502. The navigation system 916 is stored in a memory such as memory 606 and performed by one or more processors such as processor 604, in accordance with some examples. The navigation system 916 may send sensor data 912 such as images 914 to the off-site client device 704, the server 706, and/or the mobile device 708, 709 for processing.

[0137] The navigation system 916 generates the waypoints 910 based on the selected path 930. For example, a preprogrammed path 960 such as a flightpath for the self-propelled device 710 is to fly or move a couple of feet away from and above a head of a user. The navigation system 916 has to determine a waypoint 910 that conforms to the preprogrammed path 960. Examples of preprogrammed paths 960 include for the self-propelled device 710 to go to a waypoint 910 in front of person and hover to identify the object 924 and various preprogrammed path 960 to circumnavigate an object 924. The preprogrammed paths 960 include various information such as: a target distance from a person associated with the face used to unlock the self-propelled device 710, a target

height of the self-propelled device 710 above the ground, and so forth. The preprogrammed paths 960 may be modified by the person 718 using the associated application 717, in accordance with some examples. One skilled in the art would recognize that the preprogrammed paths 960 would be different for different types of self-propelled devices 710 such as cars, boats, helicopters and so forth. Additionally, the preprogrammed paths 960 indicate paths that the navigation system 916 guides the person 718 through when the mobile device 708, 709 is not self-propelled.

[0138] The sensor data 912 is data that is generated by the sensors 648 and/or other components 600 such as the wireless connections 666. The sensor data 912 also includes information regarding the input/output devices 602 such as the state 615 of the control knob 614. The propulsion system commands 915 includes determining a power for the electrical motors 686 and actuator 638 positions. The sensor data 912 includes user interface actions received from the associated application 717, in accordance with some examples. The sensor data 912 indicates data from mobile device 708, 709, in accordance with some embodiments. For example, the sensor data 912 may include data regarding the light conditions, a position of the mobile device 708, 709, a live video feed, and so forth.

[0139] In some examples, the navigation system 916 determines an self-propelled device speed 935, which is associated with a time 936, direction 950, and a position 952. A velocity of the self-propelled device 710 is the self-propelled device speed 935 and the direction 950. In some examples, the navigation system 916 provides for the self-propelled device 710 to be autonomous where the self-propelled device 710 takes off, flies, lands, and, optionally, captures images or video, without additional input from an electronic control device external to the self-propelled device 710.

[0140] In some examples, for the self-propelled device 710, the navigation system 916 determines the self-propelled device speed 935 or velocity based on images from the navigation camera 651, and, in some examples, based further on height estimates using the altimeter 654 or another sensor 648. In examples for non-self-propelled mobile device 708, 709, the navigation system 916 may determine the position 952 and/or a height estimate based on analyzing sensor

data 912 such as sensor data 912 from a camera such as photography camera 650. In some examples, the navigation system 916 determines the self-propelled device speed 935 or velocity using dead reckoning using the sensor data 912. One skilled in the art will recognize that the self-propelled device speed 935 may be determined in other ways using the sensor data 912.

[0141] The navigation system 916 for self-propelled devices 710, determines the wind speed either directly by a sensor such as wind speed sensor 661 or based on a difference in an expected velocity of the self-propelled device 710 compared with an actual velocity of the self-propelled device 710. In some examples, the navigation system 916 determines the wind speed by the self-propelled device 710 hovering in place and determining an electrical power applied to the electrical motors 686 to keep the self-propelled device 710 in one place hovering.

[0142] In some examples, for self-propelled devices 710, the navigation system 916 determines the velocity of the self-propelled device 710 based on differences in locations of the self-propelled device 710, which may be based on a GPS chip 668 or objects detected in images. In some examples, the navigation system 916 determines the location of the device 708, 709 based on a wireless protocol such as IEEE 802.11 where messages are sent between the device 708, 709 and one or more hosts to determine a location of the self-propelled device 710. In some examples, the navigation system 916 determines the location of the device 708, 709 based on a location of a host such as another mobile device 708, 709 and information about a distance the mobile device 708, 709 is from the other mobile device 708, 709 which may be coupled with a height above the ground to determine coordinates of the location of the mobile device 708, 709. For example, the position of the wearable device 719 or self-propelled device 710 may be determined based on a position of the mobile device 708 and mobile device 709, respectively, and a distance of the wearable device 719 and self-propelled device 710, from the mobile device 708 and mobile device 709, respectively.

[0143] The navigation system 916 determines the height 927 based on the sensor data 912 such as from the navigation camera 651, the height detector 660, or another component 600. For example, the height 927 may be determined

based on analyzing sensor data 912 of an image captured by a camera of the mobile device 708, 709. The height 927 is associated with a time 929. The navigation system 916 may determine the device speed 935 and height 927 in different ways. The time 929 associated with the height 927 and the time 936 associated with the self-propelled device speed 935 is generated by a clock 663. The navigation system 916 includes a state 972, which indicates a goal or purpose of the mobile device 708, 709 or navigation system 916. For example, states 972 include a “start-up” state 972, an “object detection” state 972, a “bounding box” state 972, a “circumnavigate state” state 972, and so forth. In examples, the mobile device 708, 709 is not self-propelled and the navigation system 916 does not determine a device speed 935.

[0144] For self-propelled devices 710, the navigation system 916 has two navigation types. The first type is used for a period of time when the self-propelled device 710 identifies an initial face 968, takes off and hovers. The first type may include other portions of the navigation such as identifying the object 924, determining the bounding box 926, and so forth. During the first type of navigation the orientation of the self-propelled device 710 is based on a position relative to the initial face 968 and/or position 952 estimates by means such as a GPS and so forth. During the first type of navigation, the self-propelled device 710 captures images and receives sensor data 912 that are used by the navigation module 962 to build a frame of reference of the environment where the self-propelled device 710 is flying. For example, the second type used by the navigation module 962 is based on visual odometry where the motion of the self-propelled device 710 is determined based on using information from a sequence of images. In other examples, Simultaneous Localization and Mapping (SLAM) is used as the second type of navigation. The type of navigation used by the navigation module 962 may be termed motion navigation based on using information from different positions of the self-propelled device 710 to build structure or structural information regarding the real world. The navigation module 962 may build a 3D coordinate system of the environment around the self-propelled device 710.

[0145] In some examples, the navigation module 962 continues to perform one or more of the following functions: move around to waypoints 910, capture images from the navigation camera 651, capture images from the photography

camera 650, and collect sensor data 912 such as from IMU A 652, until the navigation module 962 is capable of using the second type of navigation for navigating the self-propelled device 710. The navigation module 962 builds a model of the area around the self-propelled device 710 and a model of how to interpret the images to determine the movement of the self-propelled device 710 in the second type of navigation. The navigation module 962 transitions from the first type to the second type after the navigation module 962 has built a rudimentary model of the environment.

[0146] In some embodiments, information regarding the environment that helps the navigation module 962 to transition to type 2 navigation is received by the self-propelled device 710 from the off-site client device 704, the server 706, and/or the mobile device 708 for processing.

[0147] In some examples, the mobile device 708, 709 is not self-propelled. The navigation system 916 determines its position based on analyzing sensor data 912 such as GPS data and analyzing images. The navigation system 916 additionally may determine a position 952 based on analyzing an image of the object 924. For example, the navigation system 916 may guide a human 718 to waypoints 910 of a selected path 930 by capturing an image of the object 924 and determining where the mobile device 708, 709 is relative to the object 924.

[0148] In some examples, the face recognition module 970, identify object module 917, the identify hand point module 918, and so forth, may be a trained special purpose neural network. For example, the identify hand point module 918 may be a trained neural network to identify, referring to FIG. 11, a pointing hand 1202 and to return a position or direction that the hand is pointing. The identify hand point module 918 may provide additional services such as analyzing an image 914 and identifying a likely location of the object 924. The face recognition module 970 may be a neural network to identify a face and a closeness of the face to the self-propelled device 710. The identify object module 917 is configured to identify objects within an image 914 and use the information from the identify hand pointing module 918 to select a likely object 924. FIG. 11 illustrates the examples where the mobile device 708, 709 is a self-propelled device 710. In some examples, the mobile device 708, 709 is the wearable device 719 where the camera would be located on the person 718.

[0149] In some examples, the mobile device 708, 709 transmits an image 914 to the off-site client device 704, the server 706, and/or another mobile device 708, 709 for processing. For example, the identify object module 917 sends an image 914 from self-propelled device 710 to the mobile device 709 for processing and the mobile device 709 returns locations and names of objects 924 identified within the image 914. Similarly, the identify hand point module 918 may send an image 914 to the off-site client device 704, the server 706, and/or another mobile device 708, 709 for processing. For example, the identify hand point module 918 resident in the wearable device 719 sends an image to the mobile device 708 for processing and the mobile device 708 returns a location of the pointing hand 1102 and an indication of what the pointing hand 1102 is pointing at. For example, a location of the object 924 within the sent image 914, a location of the object 924 within a 3D coordinate location, and/or a location relative to a location of the wearable device 719.

[0150] The face recognition module 970 recognizes faces. The face recognition module 970 is based on neural networks, feature recognition and placement, or another method. In some examples, the face recognition module 970 after identifying the initial face 968, uploads the initial face 968 to a server such as self-propelled device management system 216, which trains a neural network to recognize the face in the initial face 968 and downloads weights to the self-propelled device 710 to make the process of identifying the initial face 968 in subsequent images easier.

[0151] The face recognition module 970 determines an initial face 968. The initial face 968 may include more than just the face of a person but also the body of the person. The unlock module 967 unlocks or turns the self-propelled device 710 on based on sensor data 912. In some examples, the unlock module 967 receives an indication of a selection of a flight button 806, and in response sends a command to the photography camera 650 to capture an image, which is processed by the face recognition module 970. If the face recognition module 970 determines there is a suitable face in the image, then the unlock module 967 unlocks the self-propelled device 710 for flight. The navigation module 962 is called to initiate the flight. In some examples, the unlock module 967 does not receive a command from an external electronic control indicating the self-propelled device 710 should be unlocked. The unlock module 967 unlocks the

self-propelled device 710 based on a user interacting with input/output devices 602 of the self-propelled device 710. In some examples, the unlock module 967 unlocks the self-propelled device 710 in a different way. For example, a button may be depressed on the self-propelled device 710 or an instruction may be received from the off-site client device 704, the server 706, and/or the mobile device 708. The control knob 614 may be set to a state 615 for image capturing while circumnavigating an object using self-propelled modes. The select flight plan module 922 selects a flight plan to identify an object 924 and perform image capturing while circumnavigating the identified object 924. In some examples, the mobile device 708, 709 are unlocked differently. For example, a smartphone may have a password and wearable device 719 may be unlocked by simply pressing a button.

[0152] The navigation module 962 performs an initial navigation for the self-propelled device 710. The navigation module 962 sends propulsion system commands 915 to lift the self-propelled device 710 into flight on an initial flight plan. The navigation module 962 determines the waypoint 910, which is based on the preprogrammed path 960. In some examples, a first waypoint 910 is to liftoff several inches to several feet, turn the photography camera 650 towards the initial face 968, and then hover. The navigation module 962 navigates the self-propelled device 710 along to the waypoints 910 based on receiving sensor data 912 and sending propulsion system commands 915.

[0153] The navigation module 962 hovers the self-propelled device 710 pointed towards the initial face 968 and captures images 914 for the identify hand pointing module 918. The identify hand pointing module 918 identifies, referring to FIG. 11, a pointing hand 1102. The identify object module 917 uses information from the identify hand pointing module 918 to identify an object 924 within the image 914 or a subsequent image 914. For example, the identify hand pointing module 918 may be a trained neural network that identifies the pointing hand 1102 and an object 924 that is pointed to. In another example, the identify hand pointing module 918 identifies the pointing hand 1102 and a direction or area of the image 914 where the object 924 is likely located. The identify object module 917 selects the object 924. The navigation module 962 starts to navigate to and point the self-propelled device 710 at the object 924. The person 1006 may be able to identify the object 924 the identify object

module 917 has selected based on the flight path and orientation of the self-propelled device 710.

[0154] In some examples, when the mobile device 708, 709 is not the self-propelled device 710, the navigation module 962 indicates to the person 718 where to move the mobile device 708, 709. For example, the navigation module 962 may use the image capture UI 1502.

[0155] The identify object module 917 provides the person 1006 with an option to select an object 924 within an image 914. For example, referring to FIG. 13, the associated application 717 includes a user interface for the person 718 to select the object 924 from an image 914 provided by the mobile device 708, 709. In some examples, a different mobile device 708, 709 takes images and processes the images to perform object identification. The person 718 then selects an object 924 and the different mobile device 708, 709 communicates a location of the object 924 to the mobile device 708, 709. For example, mobile device 708 such as a smartphone provides object selection UI 1302 of FIG. 13 and then sends an indication of the selection of the object 924 by the person 718 to the wearable device 719 or the self-propelled device 710. For example, the indication of the location of the object 924 may be a portion of an image 914, a location in a 3D coordinate system, a location relative to the self-propelled device 710, wearable device 719, and so forth.

[0156] The navigation module 962 then uses images 914 of the object 924 to determine a bounding box 926. The bound object module 920 may indicate to the navigation module 962 a flight path or waypoints 910 to go to for constructing the bounding box 926. The bound object module 920 uses object 924 information from the identify object module 917 to determine a path for the navigation module 962 to follow to provide the bound object module 920 enough information to determine the bounding box 926. In some examples, the bounding box 926 is a two- or three-dimensional geometric shape used to represent the object 924. In some embodiments, the bounding box 926 is a two-dimensional rectangle. In some embodiments, the bounding box 926 encloses portions of the object 924 but may not enclose projections of the object 924 such as side mirrors of an automobile. The navigation module 962 may determine the path 934 in a different way. Additionally, the navigation module 962 may

indicate to the person 718 such as through the image capture UI 1502 that the person 718 should walk around the object 924 to provide the bound object module 920 enough information to determine the bounding box 926.

[0157] The bound object module 920 may provide the person 718 with an option to select or modify the bounding box 926. For example, the associated application 717 may include a user interface for the person 1006 to select or modify the bounding box 926 with an interactive user interface such as adjustments UI 1610 of FIG. 16. The associated application 717 sends an indication of the bounding box 926 to the found object module 920.

[0158] The bound object module 920 then determines a center 925 of the object 924. The center 925 is based on a center of the bounding box 926, in accordance with some embodiments. If the bounding box 926 is two-dimensional, then the bound object module 920 estimates a height of the object 924 and selects the center to be at a half the height and at the center of the bounding box 926, in accordance with some examples. The bound object module 920 may determine the center 925 as a center of mass of the bounding box 926 assuming a uniform mass of the object 924. In some examples, the bound object module 920 determines a centroid based on determining the surfaces or approximating the surfaces of the object 924 with polygons.

[0159] The navigation module 962 then selects default values for the height 932, number of images 914, and path 934. For example, the height 932 may be half the height of the object 924, the number of images 914 may be 50 images, and the path 934 may be a circular path about the center 925 with a radius a threshold value from a nearest surface of the object 924 in traversing the path 934 (e.g. a path 934 that does not come closer than 1 to 10 feet from the object 924.) The height 932 may be selected based on the mobile device 708, 709. For example, the height 932 may be the height 932 of the wearable device 719 on the person 718 or how high a person 718 can hold a mobile device 708 that is a smartphone.

[0160] The navigation module 962 may provide the person 1006 with an option to select or modify the height 932, number of images 914, and path 934. For example, the associated application 717 may include a user interface such as adjustments UI 1610 of FIG. 16 for the person 718 to select or modify the

height 932, number of images 914, and path 934 with an interactive user interface. The associated application 717 sends an indication of the height 932, number of images 914, and path 934 to the navigation module 962.

[0161] The navigation module 962 divides path 934 by the number of images 914 and sets waypoints 910 at each position where an image is to be captured. In some examples, if the person 718 adjusts the position of where the images are to be captured, then navigation module 962 sets the waypoints 910 to the adjusted positions of where the images are to be captured. The navigation module 962 then navigates the device 708, 709 to each of the waypoints 910. In the case where the self-propelled device 710 is autonomous, the navigation module 962 accesses a device speed 935, time 936, direction 950, position 952, height 927, and time 929 to assist in navigating the self-propelled device 710. The navigation module 962 uses SLAM to navigate to each of the waypoints 910. In some examples, the navigation module 962 builds a representation of the object 924 and flies the self-propelled device 710 to each waypoint 910 based on a position 952 relative to the object 924. For example, the self-propelled device 710 can determine a distance from the center 925 based on the representation of the object 924 and a position 952 within the path 934. The height 927 can then be determined to match the selected height 932. The navigation module 962 has different modes based on whether the mobile device 708, 709 is self-propelled and autonomous. In the case when the mobile device 708, 709 is self-propelled device 710 and autonomous, then the navigation module 962 controls the movement of the mobile device 708, 709.

[0162] In other examples, the navigation module 962 controls the movement of the mobile device 708, 709 by providing instructions to a person 718 to move the mobile device 708, 709. For example, the navigation system 962 may use image capture UI 1502 of FIG. 15 to instruct a person 718 to move the mobile device 708, 709 comprising a camera or image capturing device. In some examples, the navigation system 916 provides navigation to a boat, a plane, a device restricted to operating on the ground such as a car, or another self-propelled device 710.

[0163] The navigation module 962 ensures that the self-propelled device 710 is still or motionless to within a threshold for each image 914 captured and may

adjust the settings of the photography camera 650 to take higher quality images and set the exposure time to adjust for low light. The navigation module 962 may adjust the exposure time to attempt to have all the images 914 have a same exposure.

[0164] In some examples, the object 924 is moving so the navigation module 962 sets the waypoints 910 relative to the position of the object 924 and matches the speed of the object 924 during image 914 capture by updating the location of the center of the bounding box based on captured images of the object 924. After the mobile device 708, 709 has completed taking the images 914 of the object 924, the self-propelled device 710 returns to the person 1006, in accordance with some examples.

[0165] FIG. 10 illustrates unlocking 1000 the self-propelled device 710, in accordance with some embodiments. The person 718, with the self-propelled device 710 in their open hand 1008, presses the flight button 806 and points the photography camera 650 towards their face 1002. The face recognition module 970 determines whether the face 1002 is suitable as an initial face 968, which is based on the distance 1004 of the face 1002 from the self-propelled device 710. The unlock module 967 unlocks the self-propelled device 710 and recognizes the combination of the face 1002 with the pressing of the flight button 806 as a “fly instruction”, which transitions the navigation system 916 to the navigation module 962. In some embodiments, the self-propelled device 710 unlocks in a different manner or may not need to be unlocked.

[0166] FIG. 11 illustrates a self-propelled device 710 flying, in accordance with some embodiments. The self-propelled device 710 has flown out of the open hand 1008 to point A 1108 and then to point B 1106, which is a hover location. The navigation module 962 initially navigated the self-propelled device 710 to fly above the open hand 1008 to point A 1108 and then to the point B 1106, which is a hover location. The navigation module 962 uses one or more of the images from the photography camera 650, images from the navigation camera 651, and data from the height sensor 1104 to hover. The initial portion of the flight from the open hand 1008 to point B 1106 may be a bit unpredictable as the person 1006 may move their hand or launch the self-propelled device 710. So, the hovering at point B 1106 may be performed at

different locations in accordance with how the self-propelled device 710 took off from the open hand 1008 or another location.

[0167] The navigation system 916 causes an image 914 or images 914 to be taken to identify the object 924 and to begin to be able to navigate using SLAM. As illustrated, the identify hand pointing module 918 indicates that a pointing hand 1102 is detected and information related to the pointing is passed to the identify object module 917. The identify object module 917 then makes an initial estimate where the object 924 that the person 718 would like as the object 924 is located. The navigation module 962 navigates the self-propelled device 710 to point C 1109 where the self-propelled device 710 captures another image 914 and the identify object module 917 identifies the object 924.

[0168] The navigation module 962 access information from the identify object module 917 to begin navigating the self-propelled device 710 towards the identified object 924. The person 718 may be able to discern the object 924 selected by the navigation system 916 based on the movement of the self-propelled device 710 towards the identified object 924. In some examples, the person 1006 may indicate an affirmation of the object 924 based on an additional hand signal. For example, the self-propelled device 710 may move toward the object 924 so that the person 1006 can identify which object 924 the self-propelled device 710 has determined is the selected object 924. The self-propelled device 710 may then turn towards the person 1006 to receive confirmation by another hand signal such as an “OK” hand signal that the identified object 924 is correct.

[0169] In some embodiments, the self-propelled drone 710 identifies the object 924 in a different manner such as the person 1006 indicating the object 924 in a user interface provided by the associated application 717 as illustrated in FIG. 13. In some embodiments, the person 718 has access to an associated application 717 that confirms the selection of the identified object 924, enables the person 1006 to select a different object 924, or enables the person 1006 to select the object 924 without hand signals. The navigation camera 651 is taking images during the navigation to begin navigation beyond the frame of reference or orientation of the person 718 or face 1002. There may be other waypoints 910 before the navigation system 916 switches to a different type of navigation

or the navigation system 916 may just use the frame of reference or orientation of the person 718, face 1002, or object 924 to perform a flight.

[0170] FIG. 12 illustrates a person selecting an object with a mobile device 708, in accordance with some embodiments. The person 718 points the mobile device 708 or wearable device 711 at the object 924 and captures an image with the photography camera 650. The navigation system 916 of FIG. 9 may then provide the object selection UI 1302 to the person 718 to confirm the object 924 or change to a different object. The person 718 may use this method of selecting an object 924 for the different mobile devices 708, 709 by pointing the photography camera 650 of the mobile device 708, 709 at the object 924.

[0171] FIG. 13 illustrates a system 1300 for object selection, in accordance with some examples. The object selection user interface (UI) 1302 is being displayed on the display 1306 by the associated application 717. The person 718 may use a mobile device 708, 709 with a display 1306. In some embodiments, the self-propelled device 710 includes the display 1206. The associated application 717 is displaying an image 914 captured by the mobile device 708, 709. The object 924 is highlighted as the object 924 identified by the identify object module 917. The associated application 717 enables the person 718 to select another object. For example, the display 1306 is a touch screen enabling the person 718 to select another object by touch. In some examples, the person 718 may indicate that they would like an image 914 with a different portion of the real world displayed. The associated application 717 causes the mobile device 708, 709 to send another image 914 to the associated application 717 for displaying on the display 1306. In some embodiments, the person 718 just captures another image 914, which is then used by the object selection UI 1302 after the identify object module 917 makes an estimated selection of the object 924. The associated application 717 indicates to the mobile device 708, 709 the object 924 selected 1204 by the person 718.

[0172] FIG. 14 illustrates the mobile device 709 determining a bounding box 926 for the object 924, in accordance with some embodiments. The mobile device 709 may be the self-propelled device 710, the mobile device 708, the wearable device 719, or another mobile device. The mobile device 709 may be autonomous and may be self-propelled. The navigation module 962 navigates

the mobile device 708, 709 around the object 924 from point C 1409, to point D 1402, to point E 1408, and then to point F 1408. In examples, where the mobile device 708, 709 is self-propelled and autonomous, the navigation module 962 controls the movement of the mobile device 708, 709. The navigation module 962 directs a person 718 to move the mobile device 708, 709 around the object 924 from point C 1409, to point D 1402, to point E 1408, and then to point F 1408. For example, the navigation module 962 may use image capture UI 1502.

[0173] The navigation module 962 determines the flight path 1404 based on images of the object 924. The bound object module 920 determines the bounding box 926 by analyzing images 914 and determining the boundaries of the object 924 from angles from edge rays 1406. Additionally, the bound object module 920 estimates a distance the mobile device 708, 709 is from the object 924 to determine a size of the object 924. The bounding box 926 may be two- or three-dimensional. The path 1404 may include the mobile device 708, 709 passing over the top of the object 924, the mobile device 708, 709 may travel around the object 924 more than once, and the mobile device 708, 709 may move closer or further from the object 924. The bounding box 926 may be determined based on using the techniques of SLAM to determine the size and location of the object 924. In some examples, the path 1404 may be limited to be on the ground or in the water and may be termed a flight path in the examples where the mobile device 708, 709 is a self-propelled mobile device 710. In some examples, the path 1404 is followed by a person 718 using a mobile device 708 and being guided by the navigation system 916. In some examples, the path 1404 is performed by a person 718 walking around the object 924 and pointing the camera at the object 924. The navigation module 962 may then analyze the video or images captured and determine if there is enough information for the navigation module 962 to determine the bounding box 926 or selected path 930.

[0174] FIG. 15 illustrates an image capture user interface (UI) 1502, in accordance with some embodiments. A live image is displayed on the display 1306 where the live image is from a image capturing device of the mobile device 708, 709. A person 718 views the display 1306. The navigation module 962 directs the person 718 to move the mobile device 708, 709 in the direction of the direction indicator 1504. The steady indicator 1506 indicates how stead

the mobile device 708, 709 is for purposes of capturing an image of the object 924. The navigation module 962 may use the image capture UI 1502 to direct the person 718 to capture images at specific locations where the mobile device 708, 709 is held steady. The number of images 928 may indicate which image of the total number of images is being captured. The person 718 may be controlling a mobile device 708, 709 that is self-propelled but not autonomous with a remote control.

[0175] FIG. 16 illustrates a system 1600 for adjusting parameters for image capturing while circumnavigating objects using mobile devices, in accordance with some examples. The adjustments UI 1610 enables the person 718 to adjust one or more of the parameters. The parameters that can be adjusted include a center 925, which is used as the center of the path 934, a number of images 914 with a number 1608, a height 1606, which indicates how high the self-propelled device 710 should be when the images 914 are captures, a radius 1604, and a bounding box 926.

[0176] The adjustments UI 1610 is being displayed on the display 1206 by the associated application 717 or the navigation module 962. The person 718 may use a mobile device 708, 709 with a display 1306. In some embodiments, the mobile device 708, 709 includes the display 1306. The associated application 717 is displaying an image 914 captured by the mobile device 708, 709. The object 924 is highlighted as the object 924 identified by the identify object module 917. The associated application 717 enables the person 718 to select another object. For example, the display 1306 is a touch screen enabling the person 718 to select another object by touch. In some examples, the person 718 may indicate that they would like to choose another object 924 where the object selection UI 1302 would be displayed.

[0177] The number of images 914, bounding box 926, height 1606, path 934, center 925 all have default values. The person 718 may select the image capture 1602 locations, or the navigation system 916 splits up the path 934 into equal portions based on the number of images 928. The path 934 defaults to a circle but any path 934 may be selected such as a fly over the object 924, curly lines, and so forth. In some examples, the person 718 may draw the path 934 with their finger. The associated application 717 sends the parameter values back to

the navigation system 916, in accordance with some examples. In some examples, the navigation system 916 controls the operation of the adjustments UI 1610. In some examples, the person 718 can set the angle at which the mobile device 708, 709 takes the images 914.

[0178] FIG. 17 illustrates a system 1700 for image capturing while circumnavigating objects using mobile devices, in accordance with some examples. The navigation system 916 controls the mobile device 708, 709 as described herein around the path 934 and captures a number of images 928 along the path 934 at a height 932 (not illustrated in FIG. 17). The center 925 is at a center of the object 924 so the mobile device 708, 709 follows a symmetric path 934 where the radius A 1502 and radius B 1504 from the center 925 remain the same for a circular path 934. The images 914 captured by the mobile device 708, 709 may be at different distance from a surface of the object 924, but the images 914 when played in sequence provide a 360-degree view of the object 924 in a symmetrical fashion with each image 914 captured at the same distance from a center of the object 924, where the center 925 in this case is determined as a center 925 of the 3D dimensional bounding box 926. The navigation system 916 navigates the mobile device 708, 709 either directly for autonomous mobile device 708, 709 or via navigating a person 718 if the mobile device 708, 709 is not autonomous. In some examples, the mobile device 708, 709 is a self-propelled mobile device 710 that is autonomous.

[0179] FIG. 18 illustrates a system 1800 for image capturing while circumnavigating objects using mobile devices, in accordance with some examples. The navigation system 916 controls the mobile device 708, which as pictured is the self-propelled device 710 as described herein around the path 1606 and captures a number of images 928 along the path 1606 at a height 932 (not illustrated in FIG. 18). In some embodiments, the center 925 is not a center, centroid, or center of mass of the object 924 so the mobile device 710 does not follow a symmetric path 934 where the radius A 1602 and radius B 1604 from the center 925 remain the same for a circular path 934 or elliptical path. The images 914 captured by the self-propelled device 710 may be at different distance from a surface of the object 924, because the center 925 is not at a center or center of the bounding box 926 of the object 924. The images 914 when played in sequence provide a 360-degree view of the object 924 in an

asymmetrical fashion with each image 914 captured a same distance from the center 925 but the distance from a surface of object 924 varies due to the center 925 not being a center 925 of the bounding box 926. In some examples, the path 934 may be on the ground or water.

[0180] FIG. 19 illustrates a system 1900 for image capturing while circumnavigating objects using mobile devices, in accordance with some examples. The navigation module 962 navigates the mobile device 708, which as illustrated is a self-propelled device 710, around the object 924 from one image capture 1902 position to the next at a height 1706. In some examples, the self-propelled device 710 may be limited to the ground or water. The self-propelled device 710 may adjust a camera angle based on the height 1906 or adjust a camera arm based on the height 1906 or instruct a person 718 to move a mobile device 708. The navigation module 962 sets the image capture 1702 positions as waypoints 910. The navigation module 962 determines the flight path to take from one image capture 1902 position to the next based on navigation techniques described herein including the image capture UI 1502 and autonomous navigation. The image capture 1902 positions are either set by the person 718 or by the navigation module 962, which divides up the path 934 by the number of images 914. The radius 1704 indicates a distance from the center 925 to the path 934. The lines 1708 indicate a field of view of the image 914 that will be captured. The navigation module 962 stops the mobile device 708 at each image capture 1902 location and captures an image 914 where the navigation module 962 may perform these operations autonomously or under the direction of a person 718. The navigation module 962 attempts to keep the self-propelled device 710 steady and at a constant angle relative to the path 934 or a plane of the path 934. For example, a horizontal angle relative to the path 934. In some examples, the mobile device 710 maintains an orientation parallel with a ground on which the object 924 rests. In some examples, the path 934 has an orientation that is parallel with a face of the bounding box 926 such as a top of the bounding box 926. The navigation module 962 or mobile device 708, 709 may set the exposure time to adjust for the lighting conditions so that all the images 914 have a similar exposure or a constant exposure. The exposure time may be set further based on how steady the navigation module 962 can keep the mobile device 710, which may be based on autonomous control of the mobile

device 708, 709 or a person 718. For example, in windy conditions or under the control of a person 718 with a shaky hand, a shorter or faster exposure time may be used to prevent blurry images. Example image A 1910, example image B 1912, and example image C 1914 are examples of the images 914 captured at the image capture 1902 positions or waypoints 910. The captured images 914 can be played to provide a 360-degree view of the object 924. The center 925 being selected in the geographic center of the bounding box 926 ensures that the captured images 914 when played for the 360-degree view will provide a centered view with respect to the object 924. In some examples, the path 934 may be on the ground or water.

[0181] FIG. 20 is a perspective view of a head-wearable apparatus in the form of glasses 2000, in accordance with some examples. The glasses 2000 are an article of eyewear including electronics, which operate within a network system for communicating image and video content. FIG. 20 illustrates an example of the head-wearable apparatus 116. In some examples, the wearable electronic device is termed AR glasses. The glasses 2000 can include a frame 2032 made from any suitable material such as plastic or metal, including any suitable shape memory alloy. The frame 2032 can have a front piece 2033 that can include a first or left lens, display, or optical element holder 2036 and a second or right lens, display, or optical element holder 2037 connected by a bridge 2038. The front piece 2033 additionally includes a left end portion 2041 and a right end portion 2042. A first or left optical element 2044 and a second or right optical element 2043 can be provided within respective left and right optical element holders 2036, 2037. Each of the optical elements 2043, 2044 can be a lens, a display, a display assembly, or a combination of the foregoing. In some examples, for example, the glasses 2000 are provided with an integrated near-eye display mechanism that enables, for example, display to the user of preview images for visual media captured by cameras 2069 of the glasses 2000.

[0182] The frame 2032 additionally includes a left arm or temple piece 2046 and a right arm or temple piece 2047 coupled to the respective left and right end portions 2041, 2042 of the front piece 2033 by any suitable means such as a hinge (not shown), so as to be coupled to the front piece 2033, or rigidly or fixedly secured to the front piece 2033 so as to be integral with the front piece 2033. Each of the temple pieces 2046 and 2047 can include a first portion 2051

that is coupled to the respective end portion 2041 or 2042 of the front piece 2033 and any suitable second portion 2052, such as a curved or arcuate piece, for coupling to the ear of the user. In one example, the front piece 2033 can be formed from a single piece of material, so as to have a unitary or integral construction. In one example, the entire frame 2032 can be formed from a single piece of material so as to have a unitary or integral construction.

[0183] The glasses 2000 include a computing device, such as a computer 2061, which can be of any suitable type so as to be carried by the frame 2032 and, in one example, of a suitable size and shape, so as to be at least partially disposed in one or more of the temple pieces 2046 and 2047. In one example, the computer 2061 has a size and shape similar to the size and shape of one of the temple pieces 2046, 2047 and is thus disposed almost entirely if not entirely within the structure and confines of such temple pieces 2046 and 2047.

[0184] In one example, the computer 2061 can be disposed in both of the temple pieces 2046, 2047. The computer 2061 can include one or more processors with memory, wireless communication circuitry, and a power source. The computer 2061 comprises low-power circuitry, high-speed circuitry, location circuitry, and a display processor. Various other examples may include these elements in different configurations or integrated together in different ways. Additional details of aspects of the computer 2061 may be implemented as described with reference to the description that follows.

[0185] The computer 2061 additionally includes a battery 2062 or other suitable portable power supply. In one example, the battery 2062 is disposed in one of the temple pieces 2046 or 2047. In the glasses 2000 shown in FIG. 20, the battery 2062 is shown as being disposed in the left temple piece 2046 and electrically coupled using a connection 2074 to the remainder of the computer 2061 disposed in the right temple piece 2047. One or more input and output devices can include a connector or port (not shown) suitable for charging a battery 2062 accessible from the outside of the frame 2032, a wireless receiver, transmitter, or transceiver (not shown), or a combination of such devices.

[0186] The glasses 2000 include digital cameras 2069. Although two cameras 2069 are depicted, other examples contemplate the use of a single or additional (i.e., more than two) cameras 2069. For ease of description, various features

relating to the cameras 2069 will be described further with reference to only a single camera 2069, but it will be appreciated that these features can apply, in suitable examples, to both cameras 2069.

[0187] In various examples, the glasses 2000 may include any number of input sensors or peripheral devices in addition to the cameras 2069. The front piece 2033 is provided with an outward-facing, forward-facing, front, or outer surface 2066 that faces forward or away from the user when the glasses 2000 are mounted on the face of the user, and an opposite inward-facing, rearward-facing, rear, or inner surface 2067 that faces the face of the user when the glasses 2000 are mounted on the face of the user. Such sensors can include inward-facing video sensors or digital imaging modules such as cameras 2069 that can be mounted on or provided within the inner surface 2067 of the front piece 2033 or elsewhere on the frame 2032 so as to be facing the user, and outward-facing video sensors or digital imaging modules such as the cameras 2069 that can be mounted on or provided with the outer surface 2066 of the front piece 2033 or elsewhere on the frame 2032 so as to be facing away from the user. Such sensors, peripheral devices, or peripherals can additionally include biometric sensors, location sensors, accelerometers, or any other such sensors. In some examples, projectors (not illustrated) are used to project images on the inner surface of the optical elements 2043, 2044 (or lenses) to provide a mixed reality or augmented reality experience for the user of the glasses 2000.

[0188] The glasses 2000 further include an example of a camera control mechanism or user input mechanism comprising a camera control button mounted on the frame 2032 for haptic or manual engagement by the user. The camera control button provides a bi-modal or single-action mechanism in that it is disposable by the user between only two conditions, namely an engaged condition and a disengaged condition. In this example, the camera control button is a push button that is by default in the disengaged condition, being depressible by the user to dispose it to the engaged condition. Upon release of the depressed camera control button, it automatically returns to the disengaged condition.

[0189] In other examples, the single-action input mechanism can instead be provided by, for example, a touch-sensitive button comprising a capacitive sensor mounted on the frame 2032 adjacent to its surface for detecting the presence of a user's finger, to dispose the touch-sensitive button to the engaged condition when the user touches a finger to the corresponding spot on the outer surface 2066 of the frame 2032. It will be appreciated that the above-described camera control button and capacitive touch button are but two examples of a haptic input mechanism for single-action control of the camera 2069, and that other examples may employ different single-action haptic control arrangements.

[0190] The computer 2061 is configured to perform the methods described herein. In some examples, the computer 2061 is coupled to one or more antennas for reception of signals from a GNSS and circuitry for processing the signals where the antennas and circuitry are housed in the glasses 2000. In some examples, the computer 2061 is coupled to one or more wireless antennas and circuitry for transmitting and receiving wireless signals where the antennas and circuitry are housed in the glasses 2000. In some examples, there are multiple sets of antennas and circuitry housed in the glasses 2000. In some examples, the antennas and circuitry are configured to operate in accordance with a communication protocol such as Bluetooth™, Low-energy Bluetooth™, IEEE 802, IEEE 802.11az/be, and so forth. In some examples, PDR sensors housed in glasses 2000 and coupled to the computer 2061. In some examples, the glasses 2000 are VR headsets where optical elements 2043, 2044 are opaque screens for displaying images to a user of the VR headset. In some examples, the computer 2061 is coupled to user interface elements such as slide or touchpad 2076 and button 2078. A long press of button 2078 resets the glasses 2000. The slide or touchpad 2076 and button 2078 are used for a user to provide input to the computer 2061 and/or other electronic components of the glasses 2000. The glasses 2000 include one or more microphones 2082 that are coupled to the computer 2061. The glasses 2000 include one or more gyroscopes 2080.

[0191] FIG. 21 illustrates a method 2100 for image capturing while circumnavigating objects using mobile devices, in accordance with some examples. The method 2100 begins at operation 2102 with capturing an image using an image capturing device of the self-propelled device. For example, the mobile device 708, 709 captures an image 914 using the photography camera

650 in FIGS. 11 and 12. The method 2100 continues at operation 2104 with processing the image to identify an object within the image. For example, the mobile device 708, 709 at point B 1106 of FIG. 11 identifies the object 924. Similarly, mobile device 708 or wearable device 711 capture an image 914 and identify object 924 using, for example, identify object module 917.

[0192] The method 2100 continues at operation 2106 with accessing a number of images to capture of the object. For example, the mobile device 708, 709 of FIGS. 11 and 12 access a default value of a number of images to capture or receives a number of images to capture from the associated application 717. In some examples, the person 718 enters a number of images using adjustments UI 1610 of FIG. 16.

[0193] The method 2100 continues at operation 2106 with determining a path around the object and a number of images to capture of the object. For example, the mobile device 708, 709 of FIG. 14 determines the path 934. The method 2100 continues at operation 2108 by dividing the path by the number of images to determine a number of waypoints. For example, the mobile device 708, 709 of FIGS. 11 and 12 access a default value of a number of images to capture or receives a number of images to capture from the associated application 717. In some examples, the person 718 enters a number of images using adjustments UI 1610 of FIG. 16. The mobile device 708, 709 then determines the image capture 1602 positions of FIG. 16, which are also waypoints 910.

[0194] The method 2100 continues at operation 2110 with navigating the device to the waypoints and capturing an image of the object at each waypoint of the waypoints. For example, the mobile device 708, 709 of FIG. 19 navigates the path 934 and captures an image 914 at each of the image capture 1902 positions or waypoints 910. The navigation may be realized by a person 718 under the direction of the navigation system 916 using, in some examples, image capture UI 1502. In some examples where the mobile device 708, 709 is autonomous and self-propelled, the navigation system 916 controls the movement of the mobile device 708, 709.

[0195] The method 2100 may be performed by one or more devices or apparatuses of devices discussed herein either alone or in conjunction with one another. For example, the self-propelled device 710, messaging system 100,

mobile device 708, another device, or an apparatus of the device, may perform the method 2100 either alone or in conjunction with one another. One or more of the operations of method 2100 may be optional. Method 2100 may include one or more additional operations. One or more operations of method 2100 may be performed in a different order.

MACHINE ARCHITECTURE

[0196] FIG. 22 is a diagrammatic representation of the machine 2200 within which instructions 2210 (e.g., software, a program, an application, an applet, an app, or other executable code) for causing the machine 2200 to perform any one or more of the methodologies discussed herein may be executed. For example, the instructions 2210 may cause the machine 2200 to execute any one or more of the methods described herein. The instructions 2210 transform the general, non-programmed machine 2200 into a particular machine 2200 programmed to carry out the described and illustrated functions in the manner described. The machine 2200 may operate as a standalone device or may be coupled (e.g., networked) to other machines. In a networked deployment, the machine 2200 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 2200 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 2210, sequentially or otherwise, that specify actions to be taken by the machine 2200. Further, while only a single machine 2200 is illustrated, the term “machine” shall also be taken to include a collection of machines that individually or jointly execute the instructions 2210 to perform any one or more of the methodologies discussed herein. The machine 2200, for example, may comprise the client device 102 or any one of a number of server devices forming part of the messaging server system 108. In some examples, the machine 2200 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.

[0197] The machine 2200 may include processors 2204, memory 2206, and input/output I/O components 2202, which may be configured to communicate with each other via a bus 2240. In an example, the processors 2204 (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 2208 and a processor 2212 that execute the instructions 2210. 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. 22 shows multiple processors 2204, the machine 2200 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.

[0198] The memory 2206 includes a main memory 2214, a static memory 2216, and a storage unit 2218, both accessible to the processors 2204 via the bus 2240. The main memory 2206, the static memory 2216, and storage unit 2218 store the instructions 2210 embodying any one or more of the methodologies or functions described herein. The instructions 2210 may also reside, completely or partially, within the main memory 2214, within the static memory 2216, within machine-readable medium 2220 within the storage unit 2218, within at least one of the processors 2204 (e.g., within the Processor's cache memory), or any suitable combination thereof, during execution thereof by the machine 2200.

[0199] The I/O components 2202 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 2202 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 2202 may include many other components that are not shown in FIG. 22. In various examples, the I/O components 2202 may include user output components 2226 and user input components 2228. The user output components 2226 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 2228 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.

[0200] In further examples, the I/O components 2202 may include biometric components 2230, motion components 2232, environmental components 2234, or position components 2236, among a wide array of other components. For example, the biometric components 2230 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 motion components 2232 include acceleration sensor components (e.g., accelerometer), gravitation sensor components, rotation sensor components (e.g., gyroscope).

[0201] The environmental components 2234 include, for example, one or more 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.

[0202] With respect to cameras, the client device 102 may have a camera system comprising, for example, front cameras on a front surface of the client device 102 and rear cameras on a rear surface of the client device 102. The front cameras may, for example, be used to capture still images and video of a user of the client device 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 client device 102 may also include a 360° camera for capturing 360° photographs and videos.

[0203] Further, the camera system of a client device 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 client device 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.

[0204] The position components 2236 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.

[0205] Communication may be implemented using a wide variety of technologies. The I/O components 2202 further include communication components 2238 operable to couple the machine 2200 to a network 2222 or devices 2224 via respective coupling or connections. For example, the communication components 2238 may include a network interface Component or another suitable device to interface with the network 2222. In further examples, the communication components 2238 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 2224 may be another machine or any of a wide variety of peripheral devices (e.g., a peripheral device coupled via a USB).

[0206] Moreover, the communication components 2238 may detect identifiers or include components operable to detect identifiers. For example, the communication components 2238 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 2238, 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.

[0207] The various memories (e.g., main memory 2214, static memory 2216, and memory of the processors 2204) and storage unit 2218 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 2210), when executed by processors 2204, cause various operations to implement the disclosed examples.

[0208] The instructions 2210 may be transmitted or received over the network 2222, using a transmission medium, via a network interface device (e.g., a network interface component included in the communication components 2238) and using any one of several well-known transfer protocols (e.g., hypertext transfer protocol (HTTP)). Similarly, the instructions 2210 may be transmitted or received using a transmission medium via a coupling (e.g., a peer-to-peer coupling) to the devices 2224.

SOFTWARE ARCHITECTURE

[0209] FIG. 23 is a block diagram 2300 illustrating a software architecture 2304, which can be installed on any one or more of the devices described herein. The software architecture 2304 is supported by hardware such as a machine 2302 that includes processors 2320, memory 2326, and I/O components 2338. In this example, the software architecture 2304 can be conceptualized as a stack of layers, where each layer provides a particular functionality. The software architecture 2304 includes layers such as an operating system 2312, libraries 2310, frameworks 2308, and applications 2306. Operationally, the applications 2306 invoke API calls 2350 through the software stack and receive messages 2352 in response to the API calls 2350.

[0210] The operating system 2312 manages hardware resources and provides common services. The operating system 2312 includes, for example, a kernel 2314, services 2316, and drivers 2322. The kernel 2314 acts as an abstraction layer between the hardware and the other software layers. For example, the kernel 2314 provides memory management, processor management (e.g., scheduling), component management, networking, and security settings, among other functionalities. The services 2316 can provide other common services for the other software layers. The drivers 2322 are responsible for controlling or interfacing with the underlying hardware. For instance, the drivers 2322 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.

[0211] The libraries 2310 provide a common low-level infrastructure used by the applications 2306. The libraries 2310 can include system libraries 2318 (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 2310 can include API libraries 2324 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 two dimensions (2D)

and three dimensions (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 2310 can also include a wide variety of other libraries 2328 to provide many other APIs to the applications 2306.

[0212] The frameworks 2308 provide a common high-level infrastructure that is used by the applications 2306. For example, the frameworks 2308 provide various graphical user interface (GUI) functions, high-level resource management, and high-level location services. The frameworks 2308 can provide a broad spectrum of other APIs that can be used by the applications 2306, some of which may be specific to a particular operating system or platform.

[0213] In an example, the applications 2306 may include a home application 2336, a contacts application 2330, a browser application 2332, a book reader application 2334, a location application 2342, a media application 2344, a messaging application 2346, a game application 2348, and a broad assortment of other applications such as a third-party application 2340. The self-propelled device management 2341 system manages the self-propelled device as disclosed in conjunction with the self-propelled device management system 216 and herein. The applications 2306 are programs that execute functions defined in the programs. Various programming languages can be employed to create one or more of the applications 2306, 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 2340 (e.g., an application developed using the ANDROID™ or IOS™ software development kit (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 2340 can invoke the API calls 2350 provided by the operating system 2312 to facilitate functionality described herein.

PROCESSING COMPONENTS

[0214] Turning now to FIG. 24 there is shown a diagrammatic representation of a processing environment 2400, which includes a processor 2402, a processor 2406, and a processor 2408 (e.g., a GPU, CPU or combination thereof).

[0215] The processor 2402 is shown to be coupled to a power source 2404, and to include (either permanently configured or temporarily instantiated) modules, namely a identify component 2410, a navigation component 2412, and a bounding component 2414. The identify component 2410 identifies the object 924. For example, the identify component 2410 performs the functions described in conjunction with the identify object module 917, in accordance with some examples. The navigation component 2412 performs the functions related to navigating the self-propelled device in accordance with a navigation module 962. The bounding component 2414 manages the determination of the bounding box 926 and performs the functions disclosed in conjunction with the bound object module 920. The processor 2402 is a special purpose processor 2402 designed specifically for the self-propelled device 710, in accordance with some examples. In some examples, the processor 2402 is part of a motherboard with the processor 2402 running a real-time operating system or an operating system such as Linux®. The processor 2402 communicates with other processing circuitry that is included in the mobile device 708, 709 such as components 600.

GLOSSARY

[0216] "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.

[0217] "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, portable digital assistants (PDAs), smartphones, tablets, ultrabooks, netbooks, laptops, multi-processor systems, microprocessor-based or programmable consumer electronics, game consoles,

set-top boxes, or any other communication device that a user may use to access a network.

[0218] "Communication network" refers 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 wireless LAN (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.

[0219] "Component" refers 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. 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. 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 application specific integrated circuit (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. 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. 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.

[0220] "Computer-readable storage medium" refers 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.

[0221] "Ephemeral message" refers 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.

[0222] "Machine storage medium" refers 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. 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."

[0223] "Non-transitory computer-readable storage medium" refers to a tangible medium that is capable of storing, encoding, or carrying the instructions for execution by a machine.

[0224] "Signal medium" refers 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 manner 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.

CLAIMS

What is claimed is:

1. An apparatus of a mobile device comprising:
a processor; and
a memory storing instructions that, when executed by the processor, configure the processor to perform operations comprising:
capturing an image using an image capturing device of the mobile device;
processing the image to identify an object within the image;
determining a path around the object;
determine a number of waypoints by dividing the path by a number of images to capture of the object; and
navigating the mobile device to the waypoints and capturing an image of the object at each waypoint of the waypoints.
2. The apparatus of claim 1, wherein the mobile device is an autonomous device and wherein the capturing the image of the object further comprises:
hovering at the waypoint and capturing the image with a constant angle of the mobile device relative to a plane of the path.
3. The apparatus of claim 2 wherein the capturing further comprises:
adjusting an exposure of the image capturing device to achieve a constant exposure for each of the number of images.
4. The apparatus of claim 1 wherein the image is a first image and wherein processing the image to identify the object within the image further comprises:
capturing a second image of a person;
processing the second image of the person to identify a pointing hand;
capturing a third image; and
processing the third image to identify the object, wherein the object is identified based on the direction indicated by the pointing hand.
5. The apparatus of claim 4 wherein the operations further comprise:

navigating the mobile device towards the object;
navigating the mobile device to point at a person having the pointing hand;
capturing a second image; and
processing the second image to identify an indication, from the person, of a confirmation that a correct object was selected.

6. The apparatus of claim 5, wherein the operations further comprise:
navigating the mobile device towards a direction indicated by the pointing hand.

7. The apparatus of any of claims 1-5, wherein the wherein processing the image to identify the object within the image further comprises:
processing the image to identify a pointing hand, and wherein the processing the image to identify the object is based on a direction of the pointing hand.

8. The apparatus of any of claims 1-5, wherein determining the path around the object further comprises:
navigating the mobile device around the object and capturing images of the object;
determining a bounding box based on the images, the bounding box enclosing the object;
determining a geometric center of the bounding box; and
determining the path to have a center of the geometric center.

9. The apparatus of claim 8 wherein the path is further determined to have an orientation parallel with a ground on which the object rests or the path is further determined to have an orientation parallel with a face of the bounding box.

10. The apparatus of claim 8, wherein the path is a circular path having a radius based on a size of the object.

11. The apparatus of any of claims 1-5, wherein the mobile device is an autonomous drone, a plane, a helicopter, a boat, an augmented reality (AR) mobile device, a mixed reality (MR) mobile device, a virtual reality (VR) wearable device, a smartphone, or a smart camera.
12. The apparatus of any of claims 1-5, wherein the navigating is based on identifying the object in subsequent images captured by the image capturing device.
13. The apparatus of any of claims 1-5, wherein the operations further comprise:
 - determining a height above a ground based on sensor data, and wherein the navigating is further based on the height above the ground.
14. The apparatus of any of claims 1-5, wherein the navigating is based on orienting the mobile device in space relative to the object.
15. The apparatus of any of claims 1-5, wherein the image is a first image and wherein the operations further comprise:
 - capturing a second image; and
 - processing the second image to determine one or more additional objects, and wherein the navigating is further based on relative positions of the one or more additional objects.
16. The apparatus of any of claims 1-5, wherein the navigating is autonomous and based on controlling a propulsion system of the mobile device.
17. The apparatus of any of claims 1-5, wherein the navigation is based on instructing a person to move the mobile device.
18. A method performed on an apparatus of a mobile device, the method comprising:
 - capturing an image using an image capturing device of the mobile device;

processing the image to identify an object within the image;
accessing a number of images to capture of the object;
determining a path around the object;
dividing the path by the number of images to determine a number of waypoints; and
navigating the mobile device to the waypoints and capturing an image of the object at each waypoint of the waypoints.

19. The method of claim 18, wherein determining the path around the object further comprises:

navigating the mobile device around the object and capturing images of the object;
determining a bounding box based on the images, the bounding box enclosing the object;
determining a geometric center of the bounding box; and
determining the path to have a center of the geometric center.

20. A non-transitory computer-readable storage medium, the non-transitory computer-readable storage medium including instructions that, when executed by at least one processor of an apparatus of a mobile device, cause the at least one processor to perform operations comprising:

capturing an image using an image capturing device of the mobile device;
processing the image to identify an object within the image;
accessing a number of images of the object;
determining a path around the object based on the number of objects;
dividing the path by the number of images to determine a number of waypoints; and
navigating the mobile device to the waypoints and capturing an image of the object at each waypoint of the waypoints.

1 / 24

100

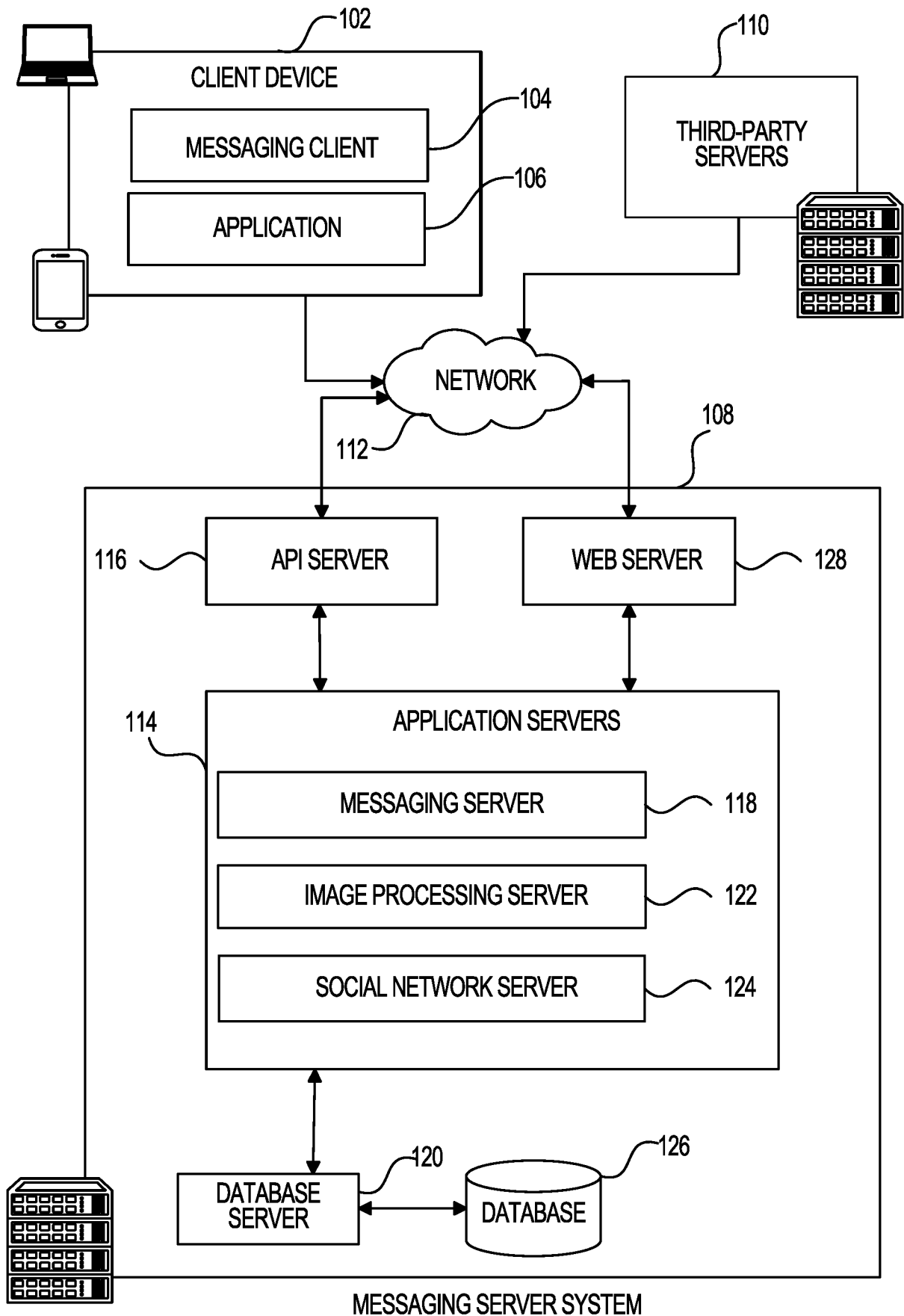


FIG. 1

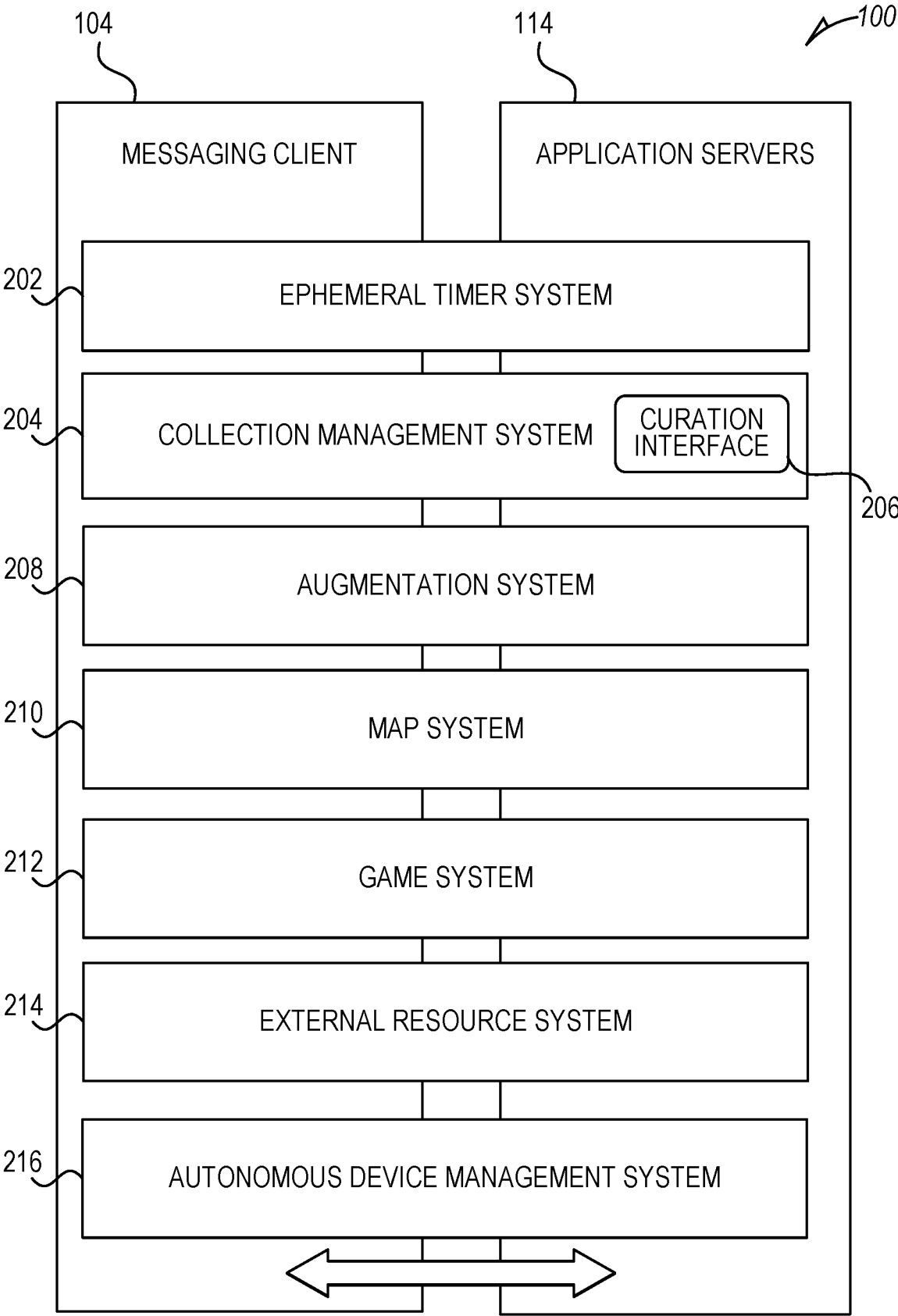


FIG. 2

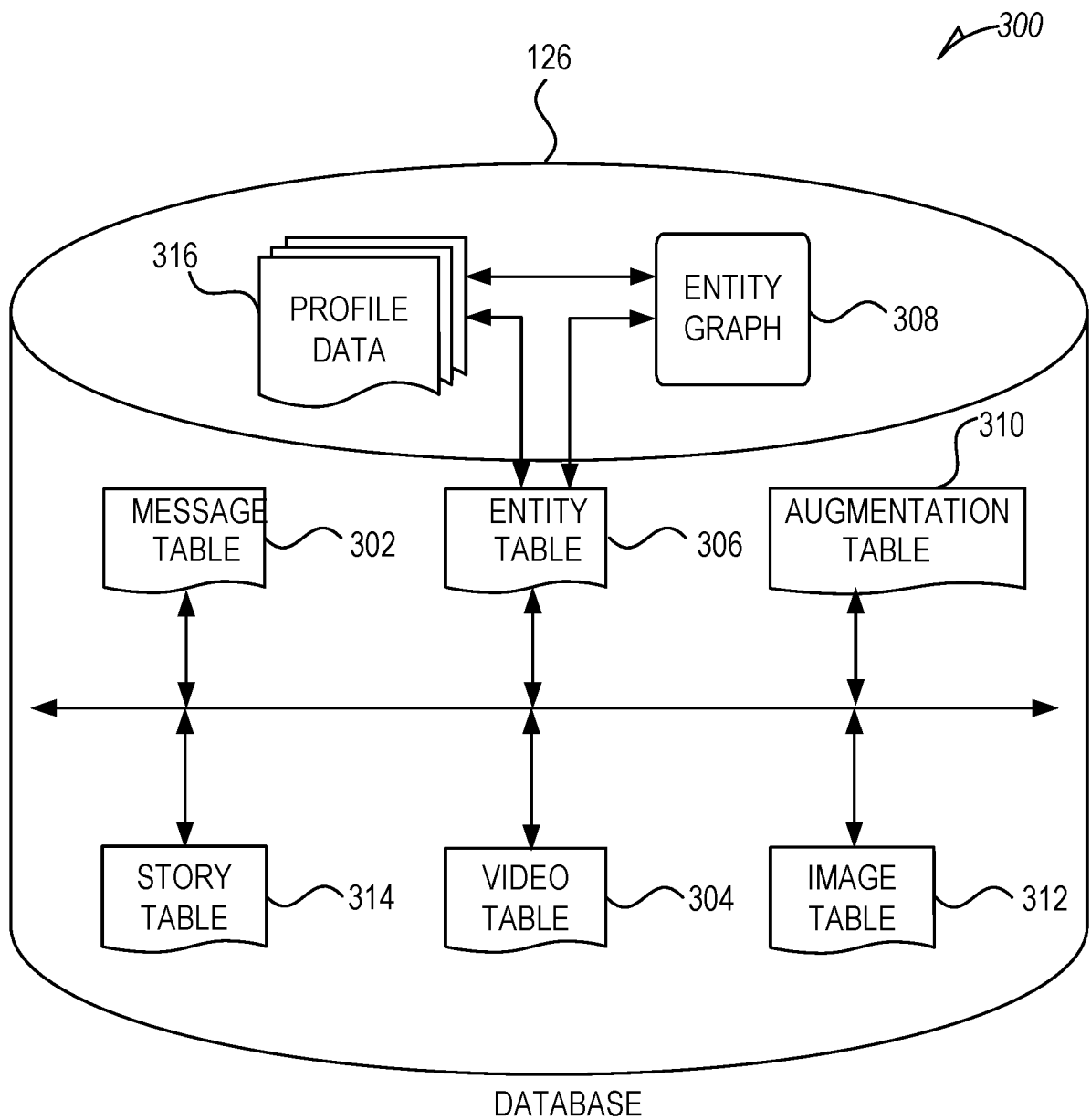


FIG. 3

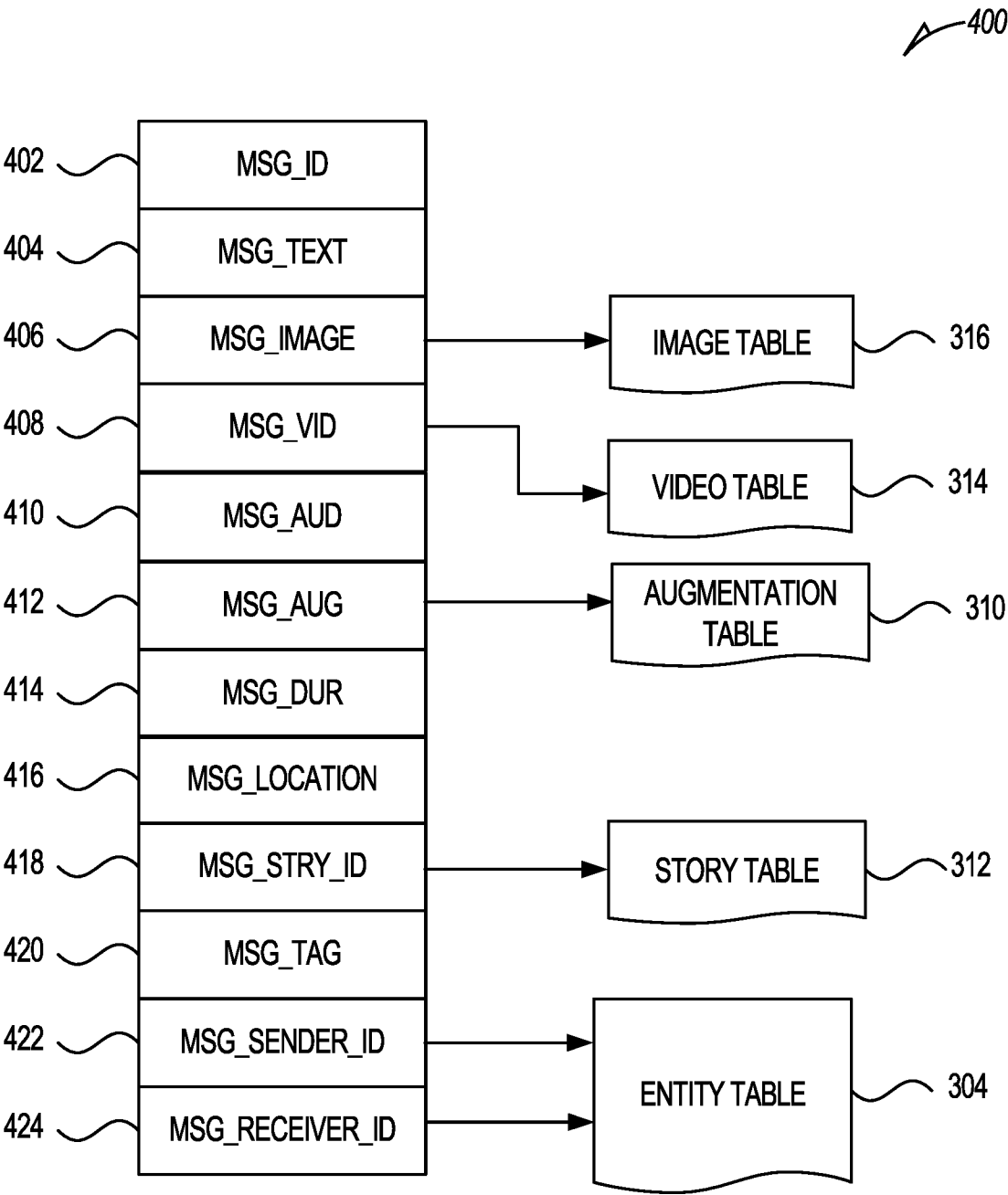


FIG. 4

5 / 24

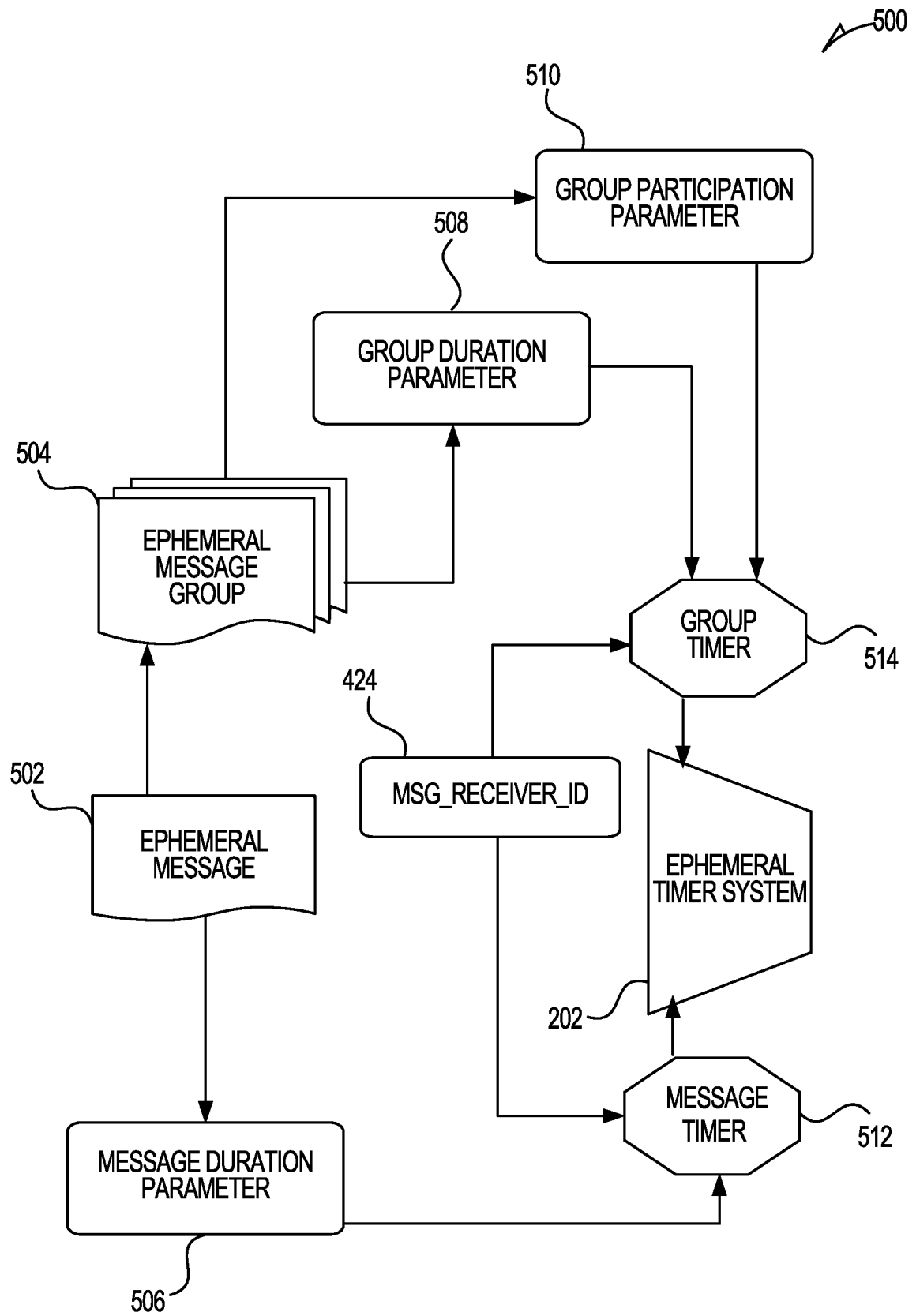


FIG. 5

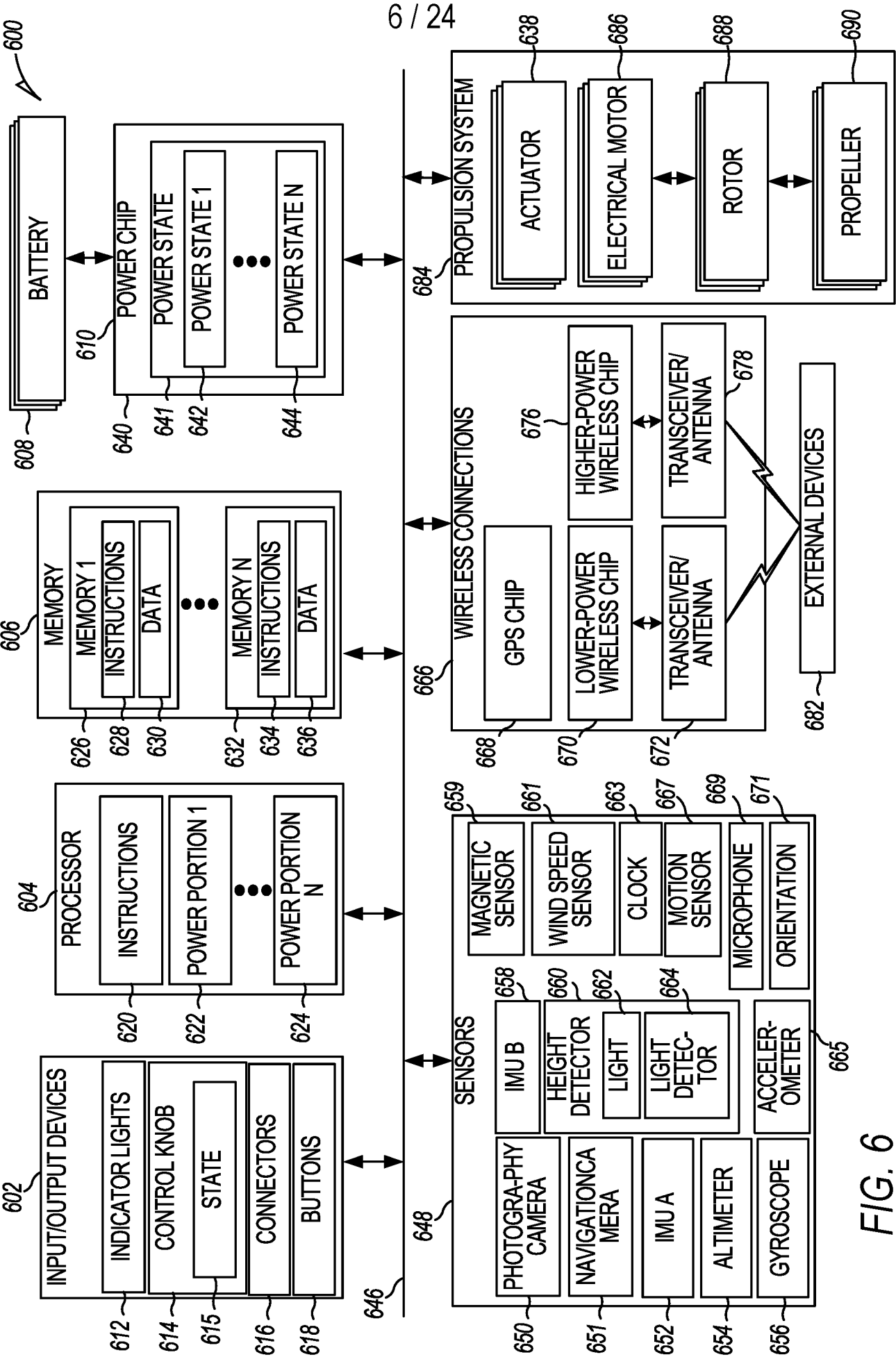


FIG. 6

700

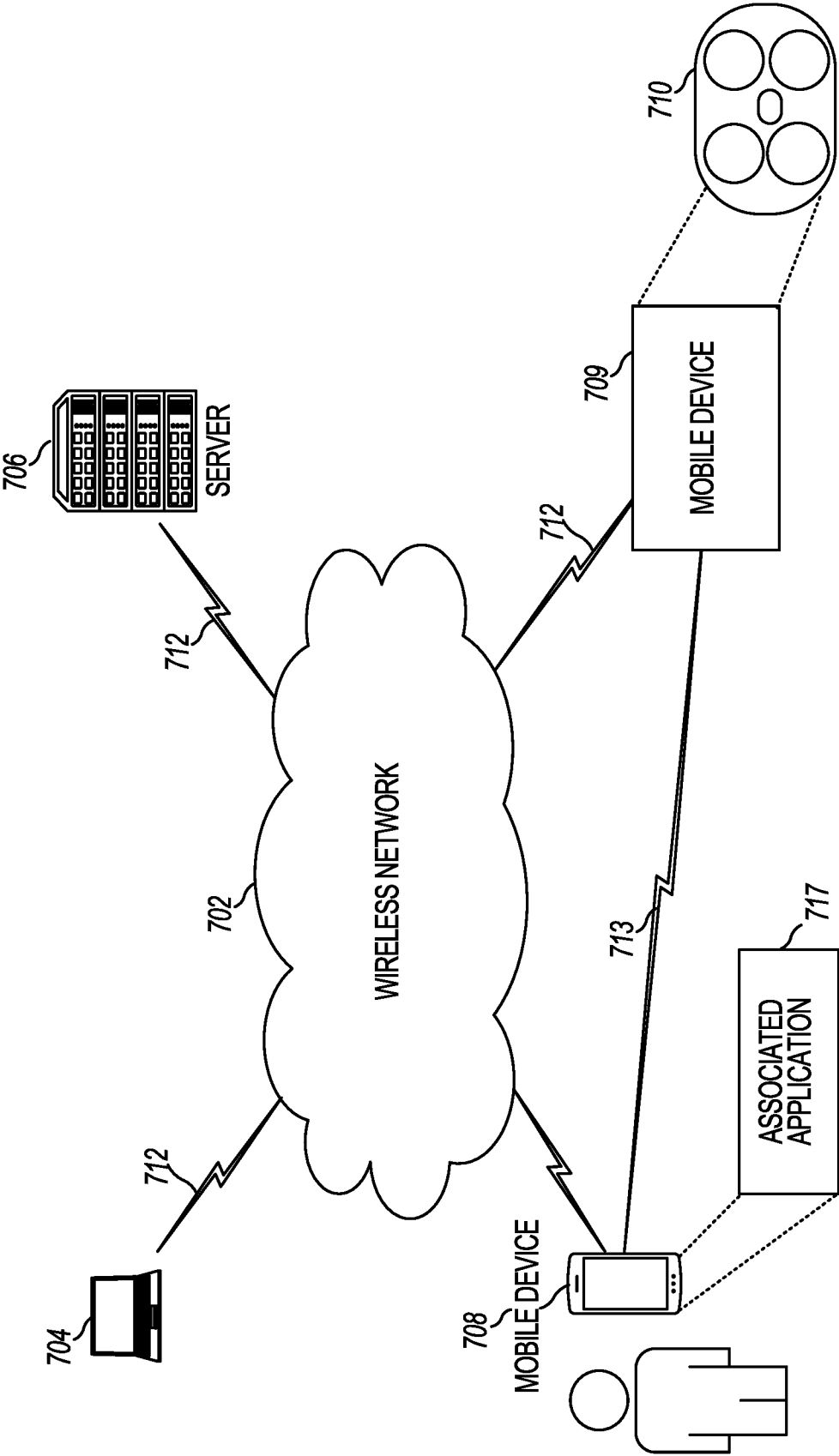


FIG. 7

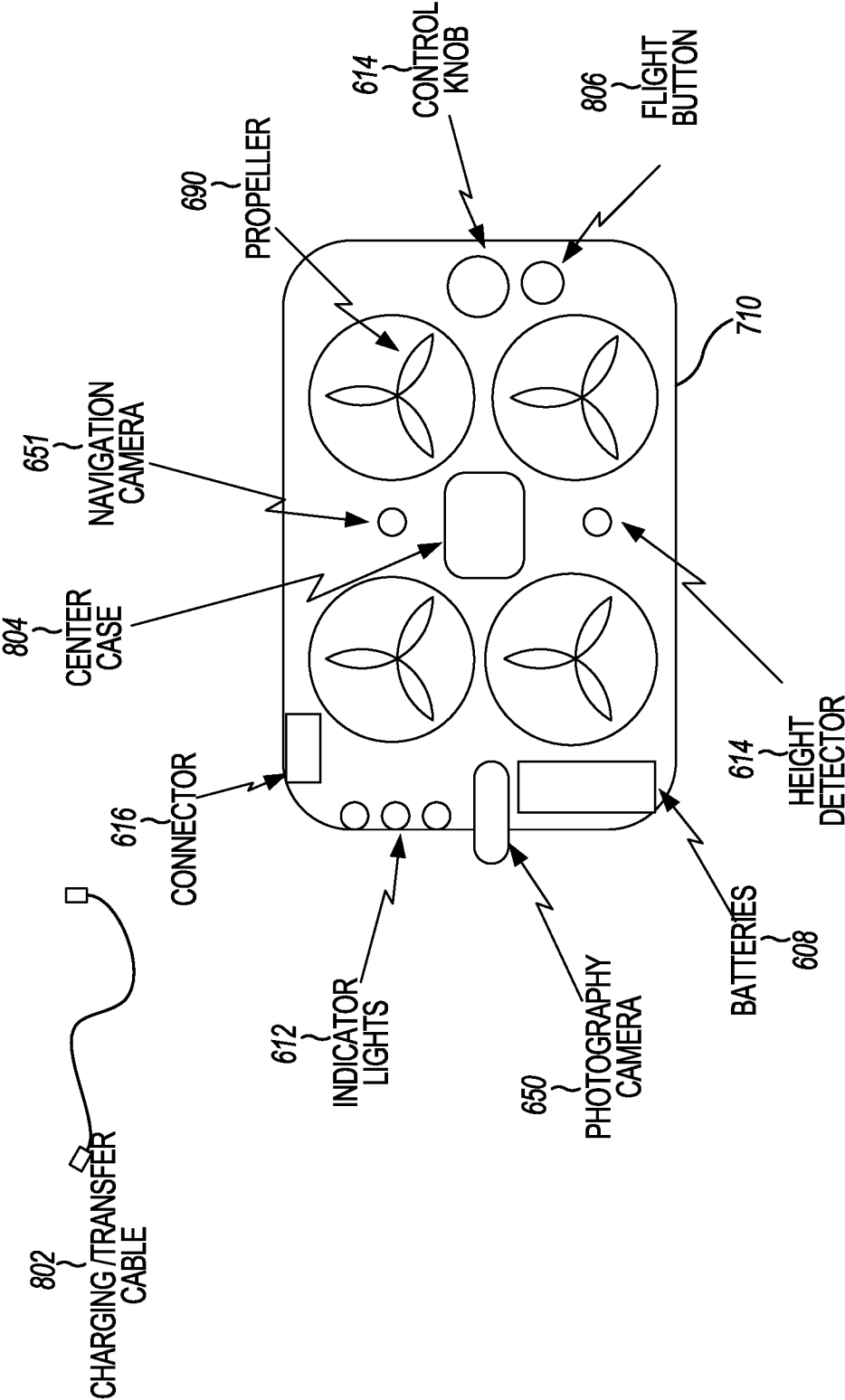


FIG. 8

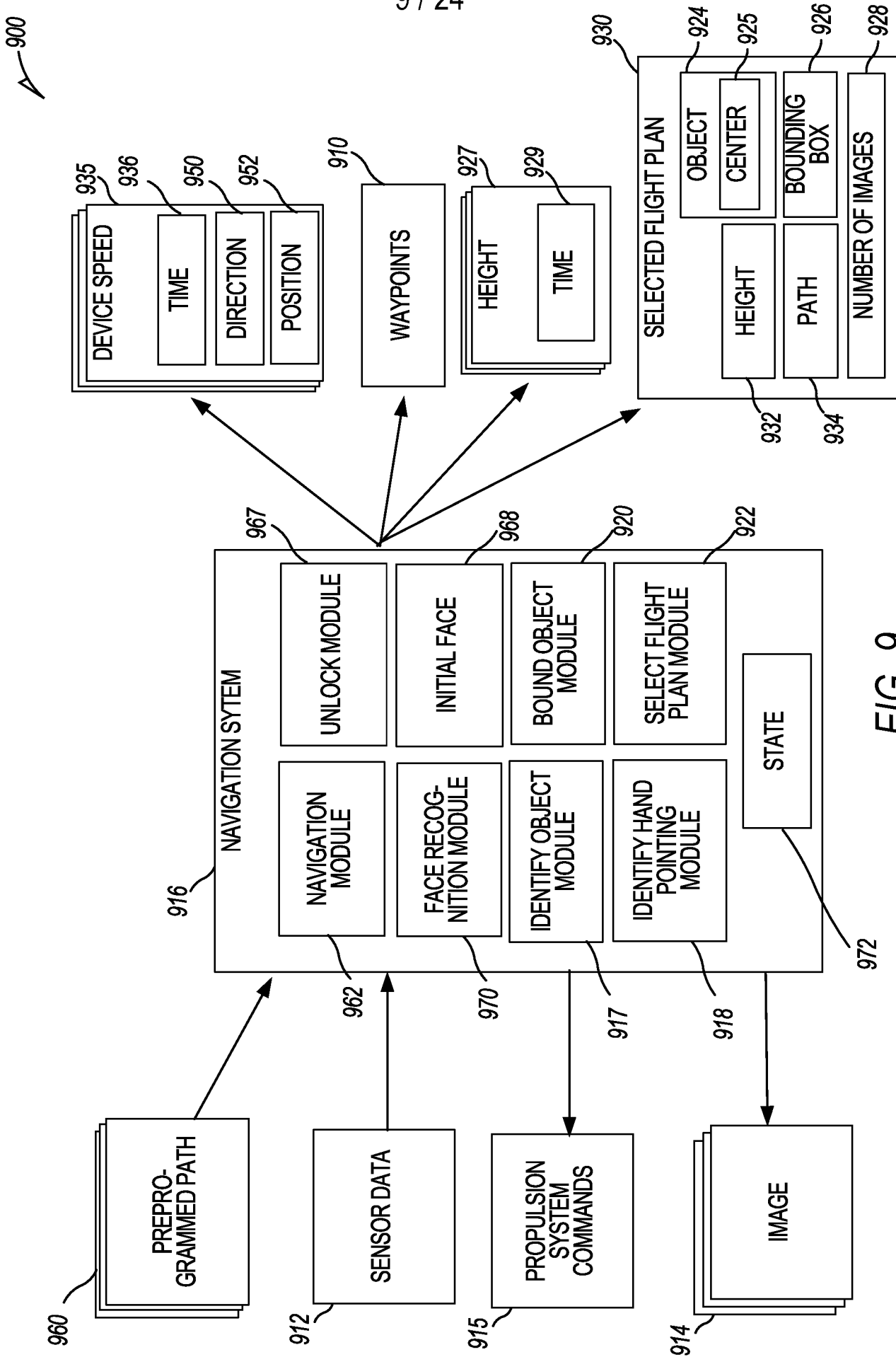


FIG. 9

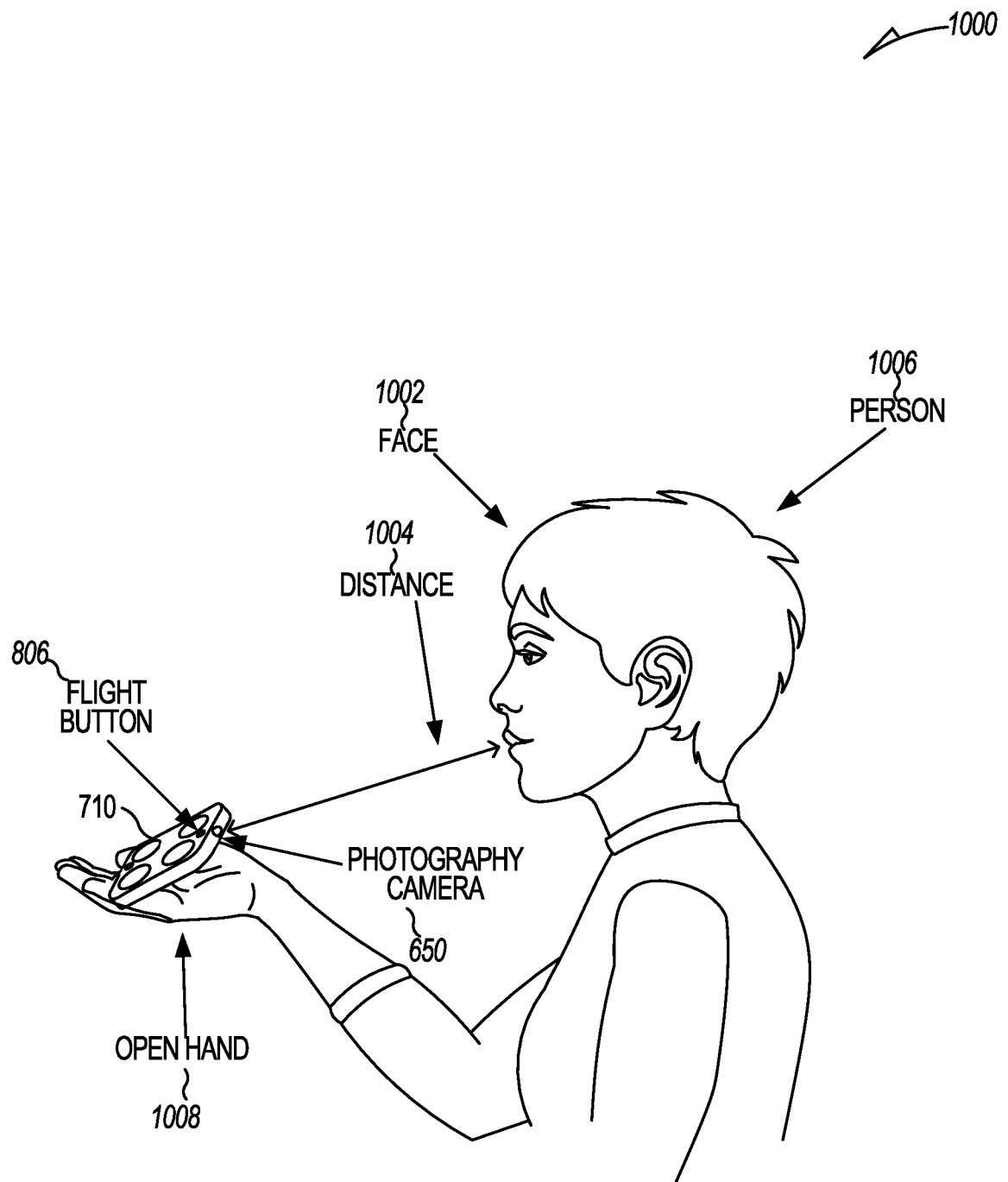


FIG. 10

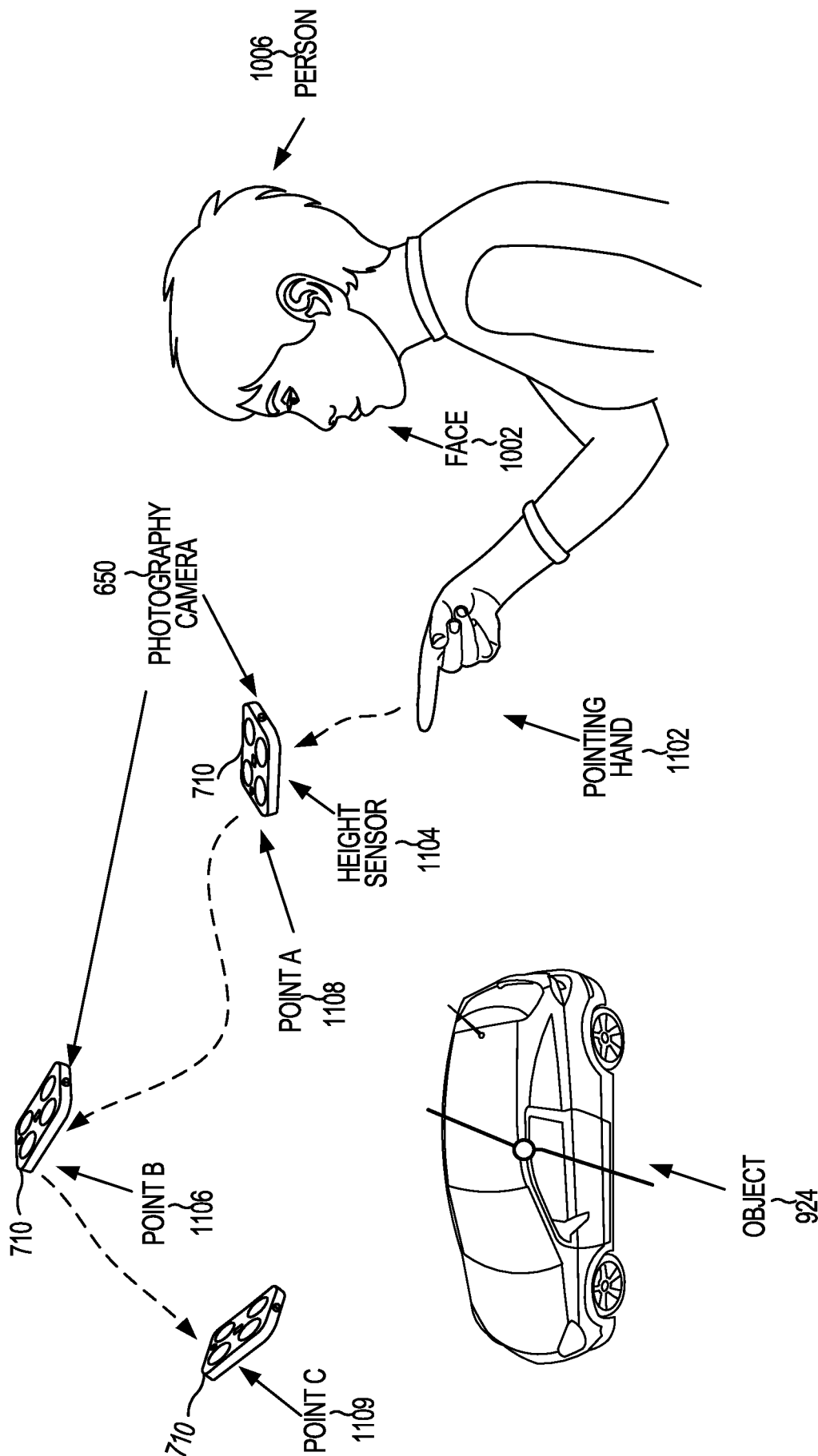


FIG. 11

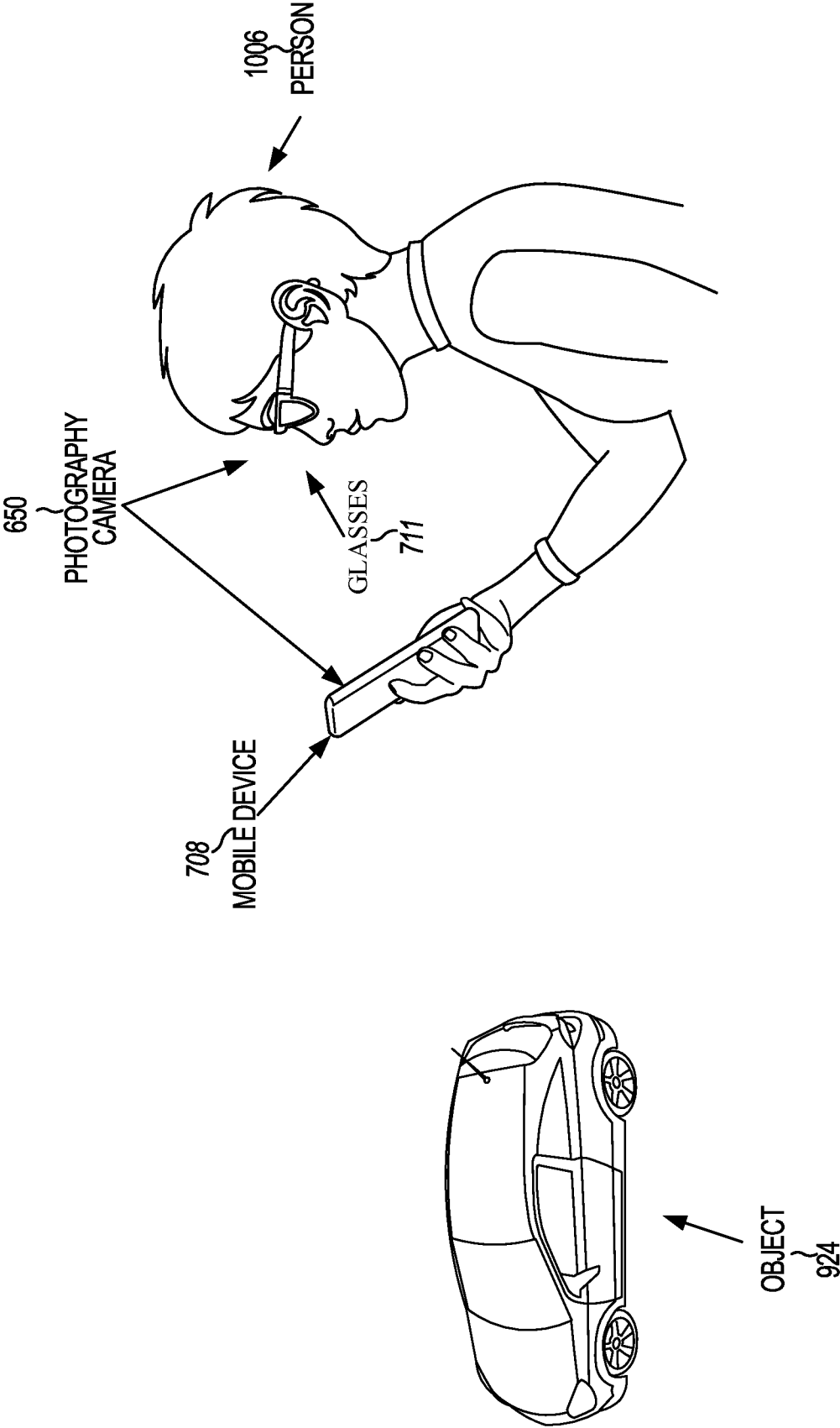


FIG. 12

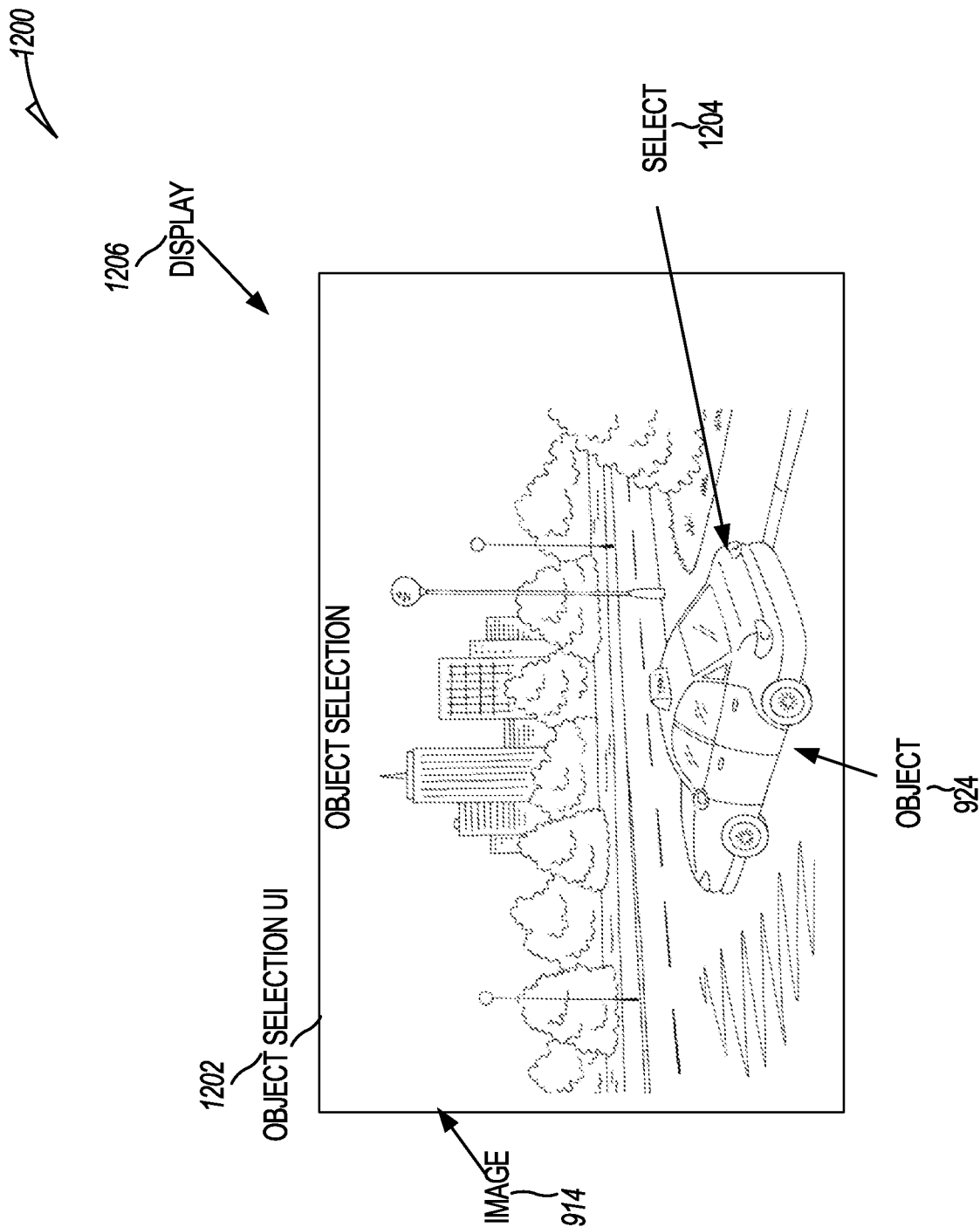


FIG. 13

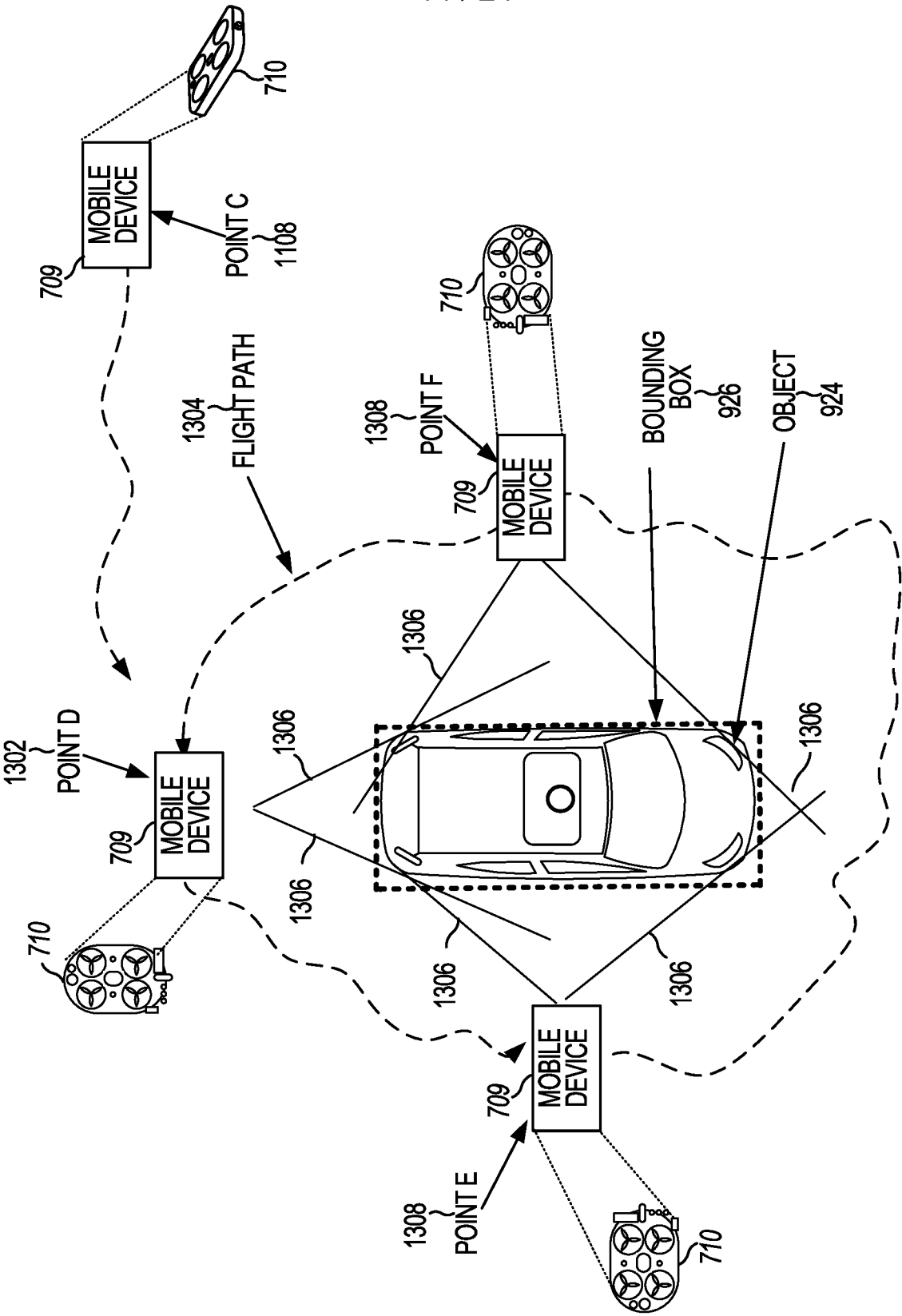


FIG. 14

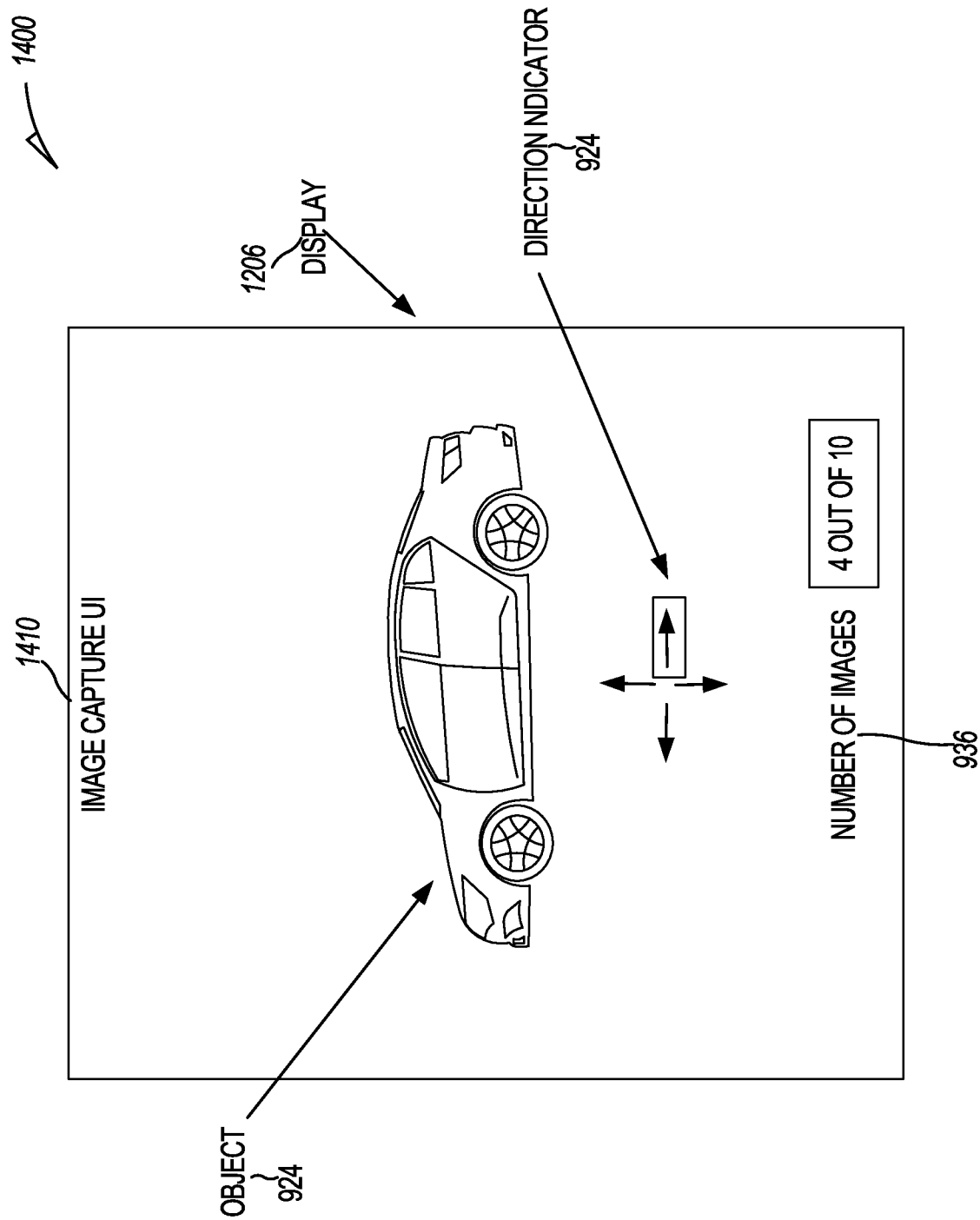


FIG. 15

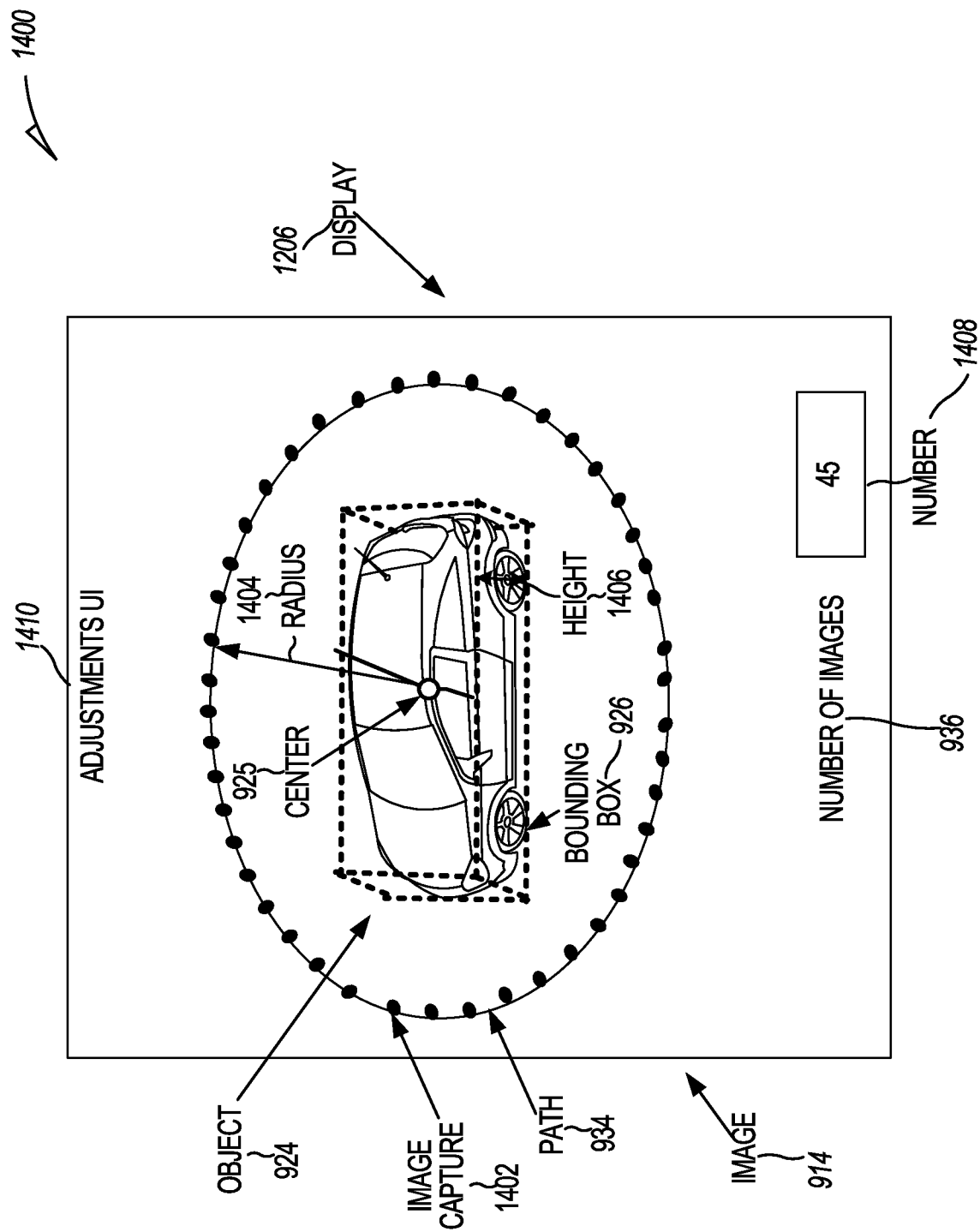


FIG. 16

1500

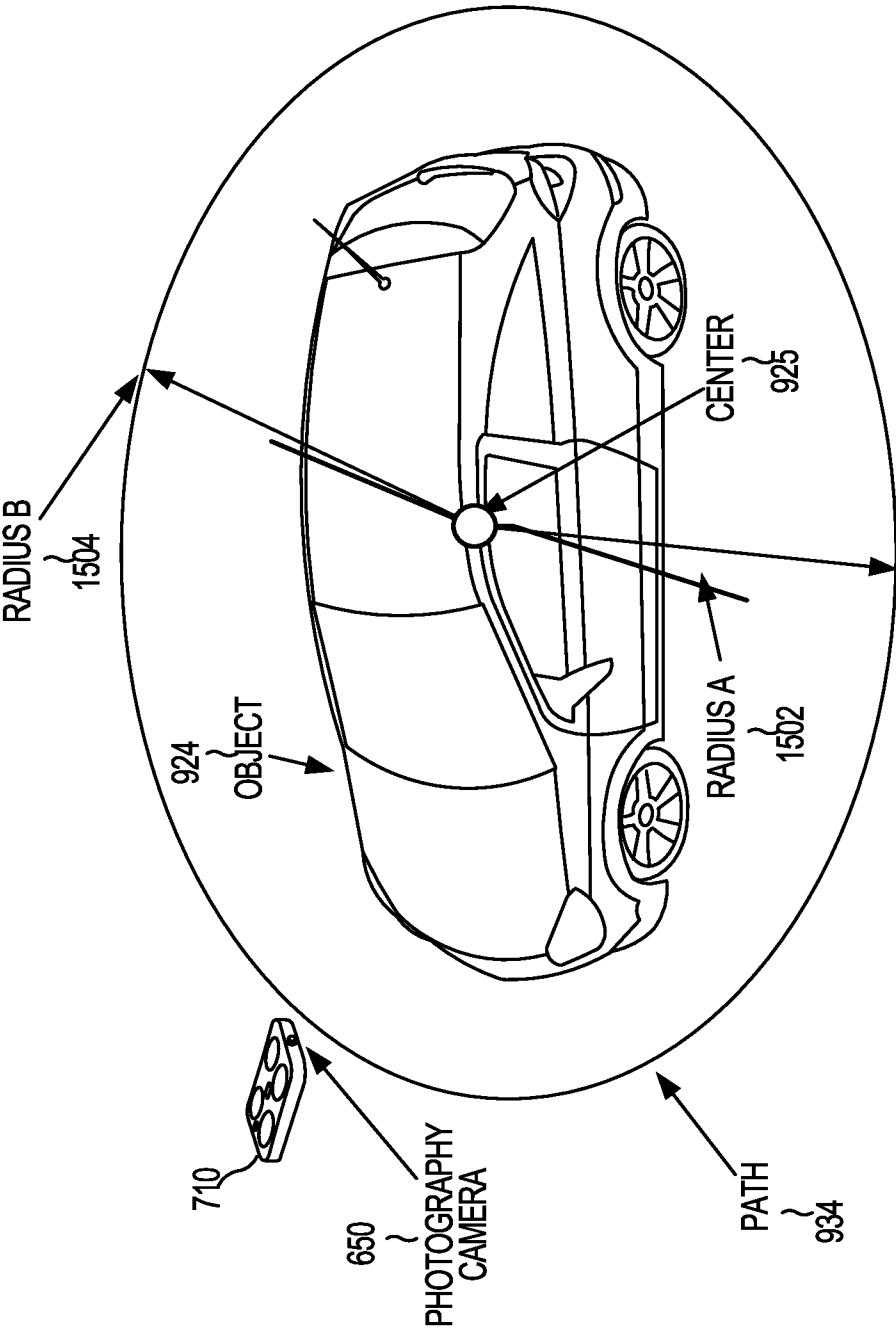


FIG. 17

1600

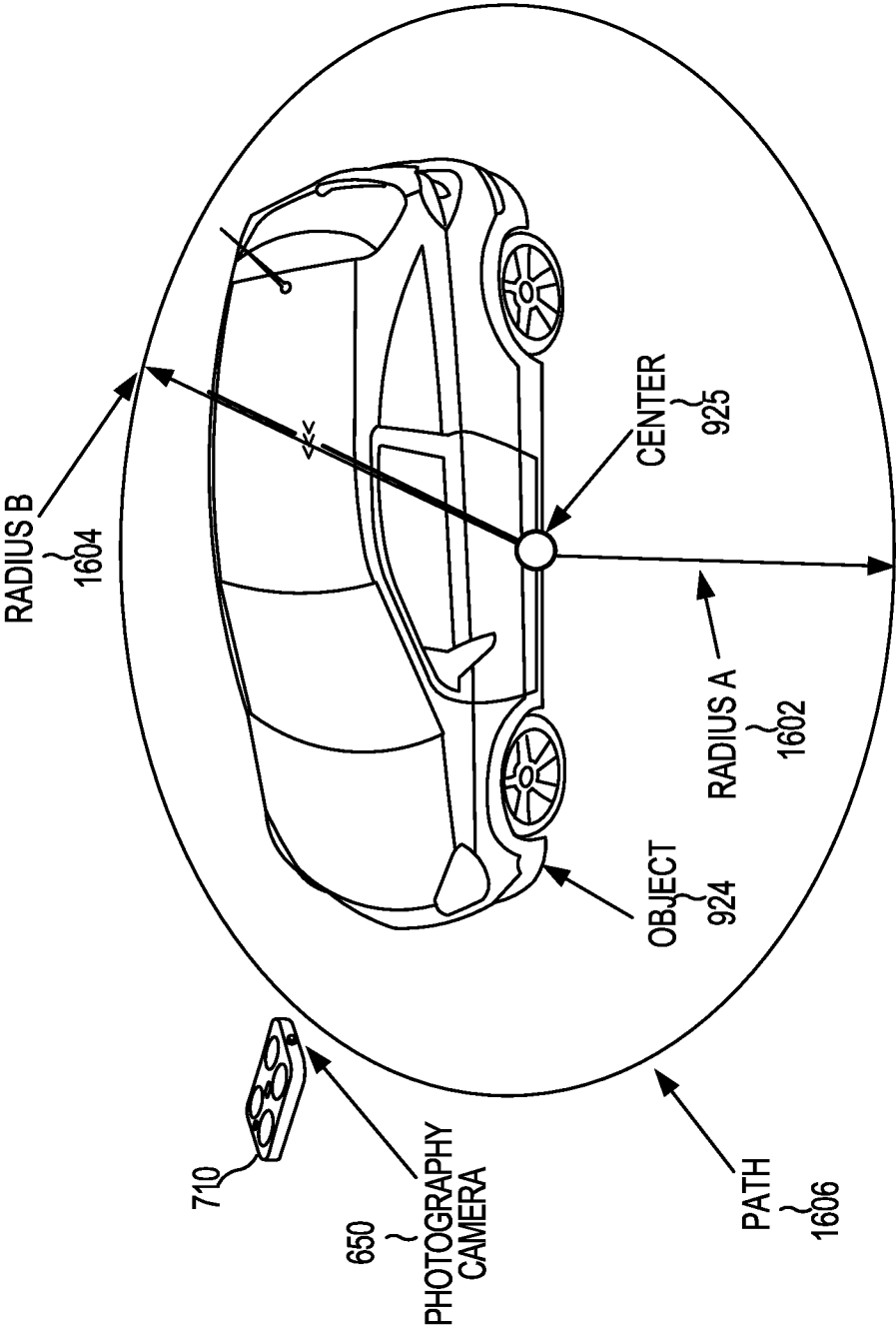
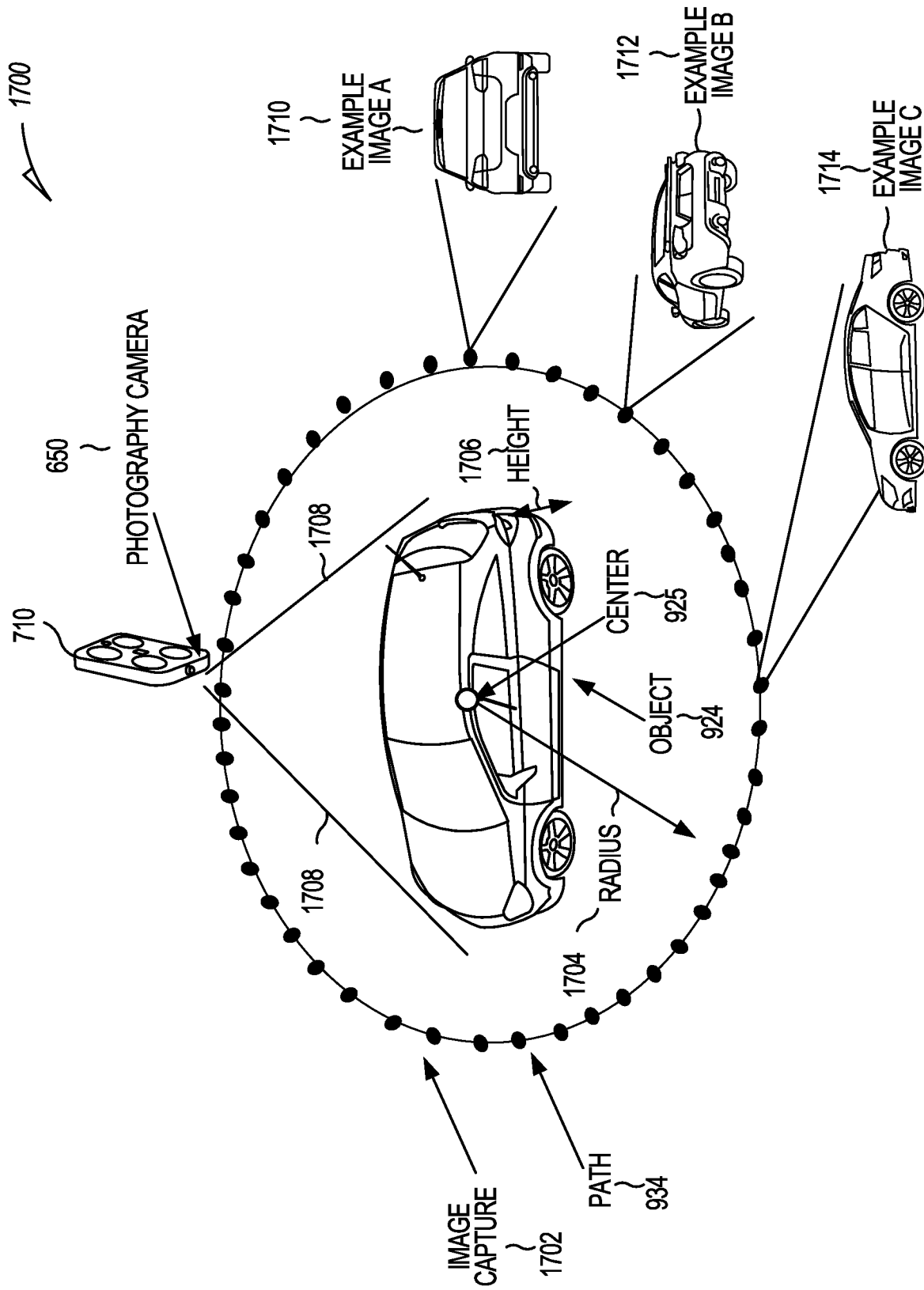


FIG. 18



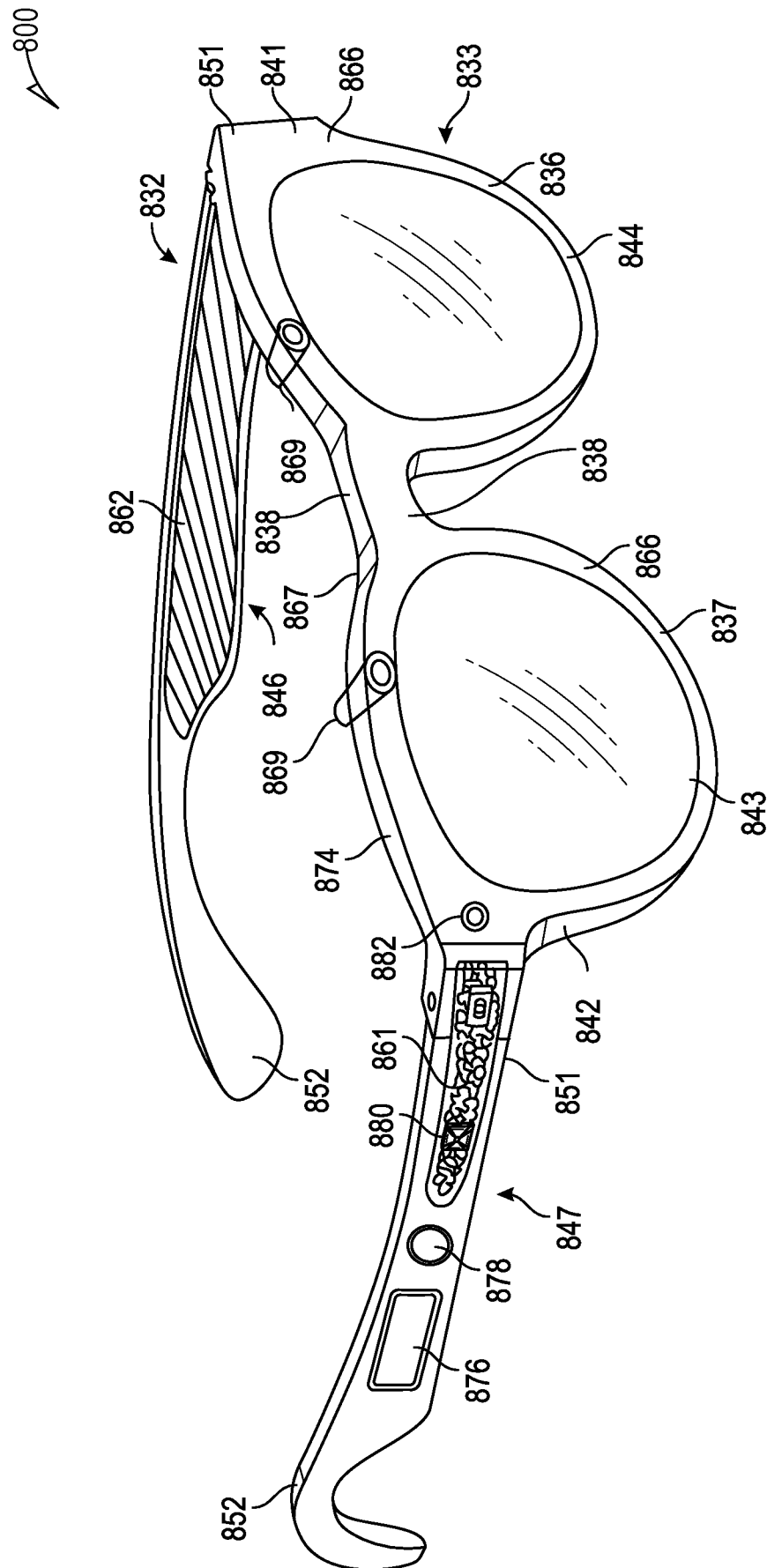


FIG. 20

21 / 24

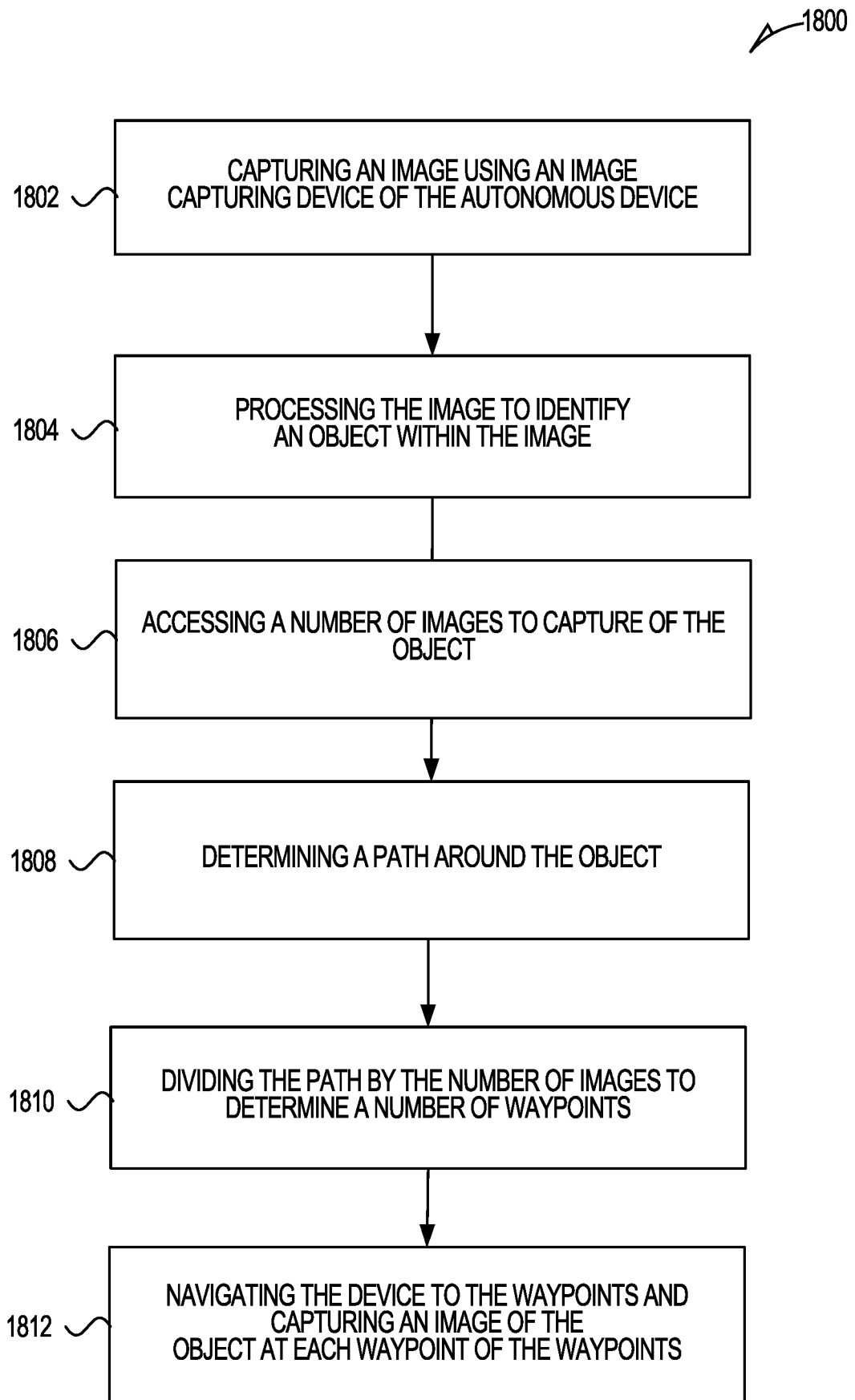


FIG. 21

22 / 24

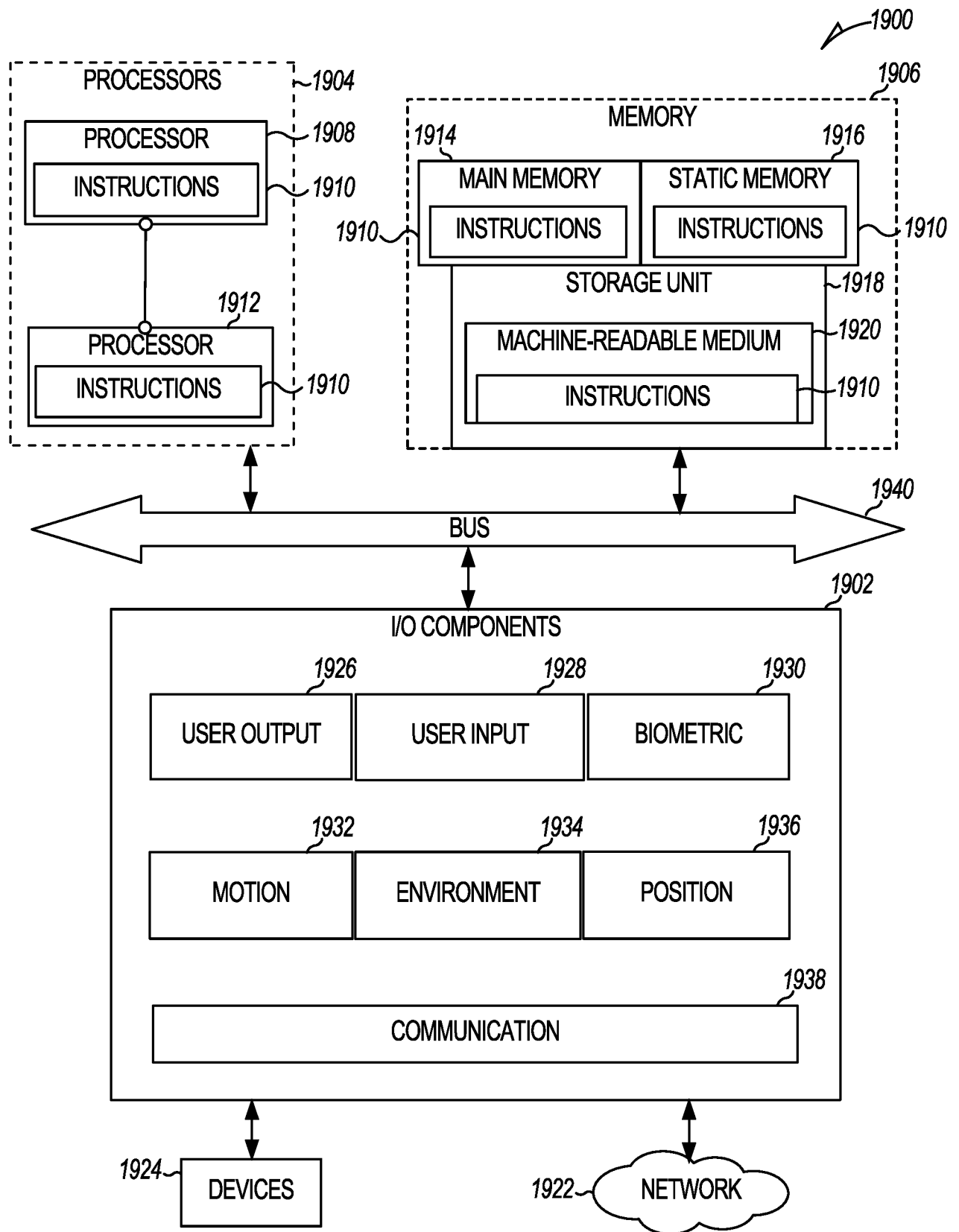


FIG. 22

23 / 24

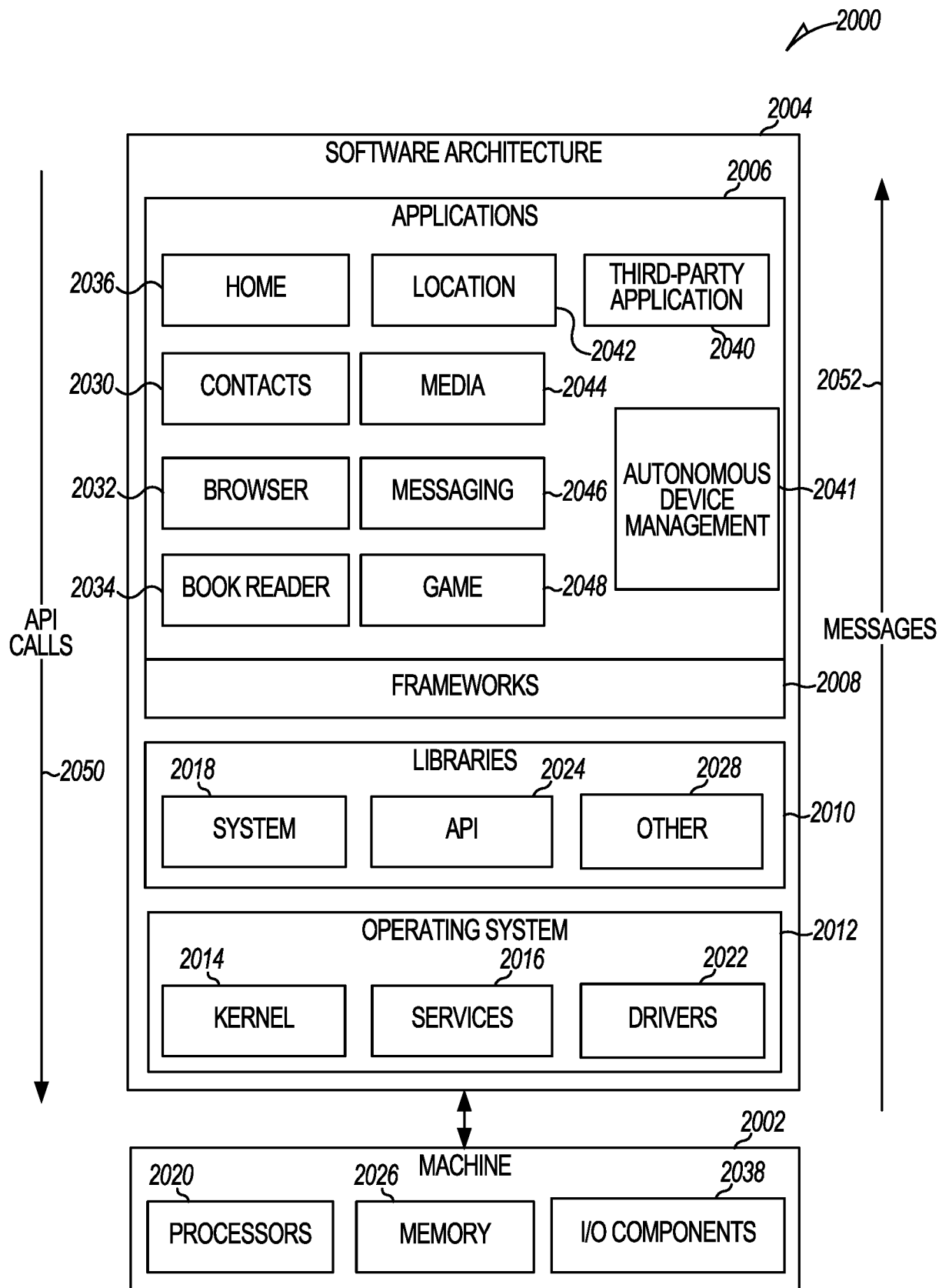


FIG. 23

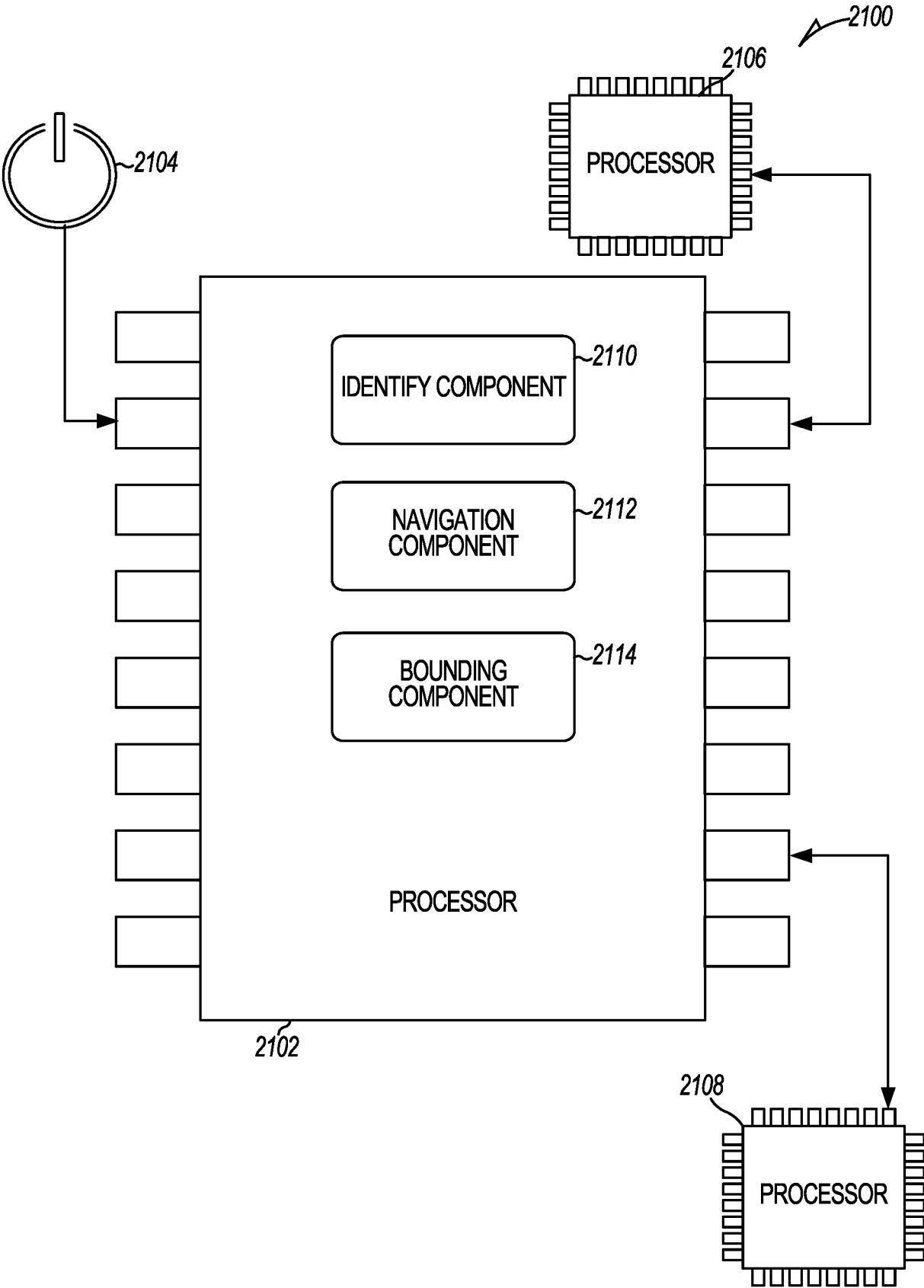


FIG. 24

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2024/024618

A. CLASSIFICATION OF SUBJECT MATTER INV. G06V20/17 G06V20/52 G06V20/56 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) G06V Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal, WPI Data		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	FREIMUTH HENK ET AL: "A Framework for Automated Acquisition and Processing of As-Built Data with Autonomous Unmanned Aerial Vehicles", SENSORS, vol. 19, no. 20, 17 October 2019 (2019-10-17), page 4513, XP093135222, CH ISSN: 1424-8220, DOI: 10.3390/s19204513 abstract sections 1, 2, 3 -----	1 - 20
A	US 2018/149138 A1 (THIERCELIN ARNAUD [US] ET AL) 31 May 2018 (2018-05-31) abstract -----	1 - 20
☐ 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 27 June 2024		Date of mailing of the international search report 11/07/2024
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Philips, Petra

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2024/024618

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2018149138 A1	31-05-2018	US 2018149138 A1	31-05-2018
		US 2020332776 A1	22-10-2020
