



(51) International Patent Classification:
G06Q 30/02 (2012.01) G06Q 30/06 (2012.01)

(21) International Application Number:
PCT/US2021/052665

(22) International Filing Date:
29 September 2021 (29.09.2021)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
63/085,935 30 September 2020 (30.09.2020) US

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

(72) Inventors; and
(71) Applicants: LUO, Jean [US/US]; 3000 31st Street, Santa Monica, California 90405 (US). ZHOU, Zhehao [CN/US]; 3000 31st Street, Santa Monica, California 90405 (US).

(74) Agent: PERDOK, Monique M. et al.; Schwegman Lundberg & Woessner, P.A., P.O. Box 2938, Minneapolis, Minnesota 55402 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, IT, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, 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, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) Title: DETERMINING LIFETIME VALUES OF USERS

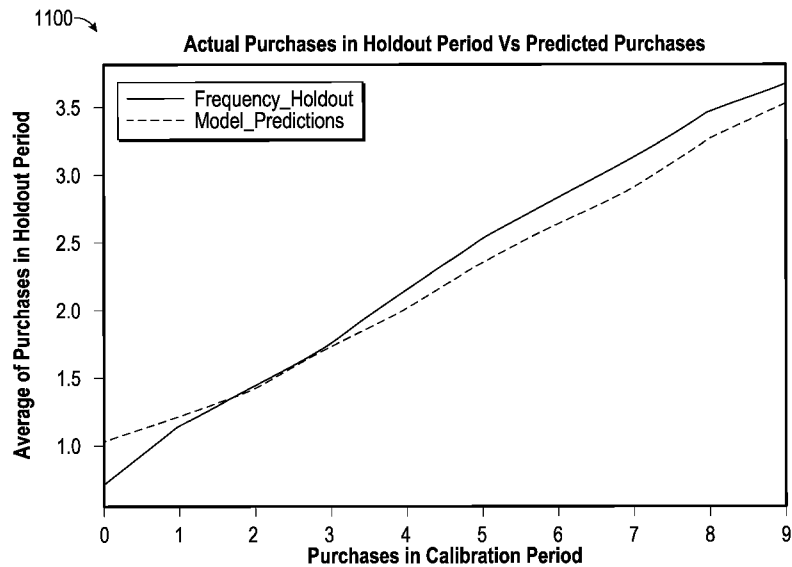


FIG. 11

(57) Abstract: The subject technology determines a transaction recency and frequency distribution. The subject technology determines a future monetary value. The subject technology determines monthly active users (MAU) and penetration for global users and users in a specific country. The subject technology predicts monetization values for the users in the specific country. The subject technology determines a lifetime value of the users in the specific country based at least in part on the monetization values.



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))*

DETERMINING LIFETIME VALUES OF USERS

PRIORITY CLAIM

5 [0001] This application claims the benefit of priority of U.S. Provisional Patent Application No. 63/085,935, filed September 30, 2020, which is hereby incorporated by reference herein in its entirety for all purposes.

BACKGROUND

10 [0002] With the increased use of digital images, affordability of portable computing devices, availability of increased capacity of digital storage media, and increased bandwidth and accessibility of network connections, digital images have become a part of the daily life for an increasing number of people.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

15 [0003] 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.

[0004] FIG. 1 is a diagrammatic representation of a networked environment in which the present disclosure may be deployed, in accordance with some example embodiments.

20 [0005] FIG. 2 is a diagrammatic representation of a messaging client application, in accordance with some example embodiments.

[0006] FIG. 3 is a diagrammatic representation of a data structure as maintained in a database, in accordance with some example embodiments.

25 [0007] FIG. 4 is a diagrammatic representation of a message, in accordance with some example embodiments.

[0008] FIG. 5 is a flowchart for an access-limiting process, in accordance with some example embodiments.

30 [0009] FIG. 6 is a schematic diagram illustrating a structure of the message annotations, as described in FIG. 4, including additional information corresponding to a given message, according to some embodiments.

- [0010] FIG. 7 is a block diagram illustrating various modules of an annotation system, according to certain example embodiments.
- [0011] FIG. 8 illustrates an example chart for monthly active users (MAU) and penetration of global camera users, in accordance with some embodiments.
- 5 [0012] FIG. 9 illustrates an example chart for users of AR content generators MAU and penetration, in accordance with some embodiments.
- [0013] FIG. 10 illustrates an example chart for Sample Full Time Period Model Validation, in accordance with some embodiments.
- [0014] FIG. 11 illustrates an example chart for Sample Holdout Period Model
10 Validation, in accordance with some embodiments.
- [0015] FIG. 12 illustrates examples of user interfaces, and presenting content items generated by an AR content generator in the messaging client application (or the messaging system), according to some embodiments.
- [0016] FIG. 13 illustrates examples of user interfaces, and presenting content
15 items generated by an AR content generator in the messaging client application (or the messaging system), according to some embodiments.
- [0017] FIG. 14 illustrates examples of user interfaces and, and presenting content items generated by an AR content generator in the messaging client application (or the messaging system), according to some embodiments.
- 20 [0018] FIG. 15 is a flowchart illustrating a method, according to certain example embodiments.
- [0019] FIG. 16 is a flowchart illustrating a method, according to certain example embodiments.
- [0020] FIG. 17 is block diagram showing a software architecture within which
25 the present disclosure may be implemented, in accordance with some example embodiments.
- [0021] FIG. 18 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, in
30 accordance with some example embodiments.

DETAILED DESCRIPTION

[0022] Users with a range of interests from various locations can capture digital images of various subjects and make captured images available to others via networks, such as the Internet. To enhance users' experiences with digital images and provide various features, enabling computing devices to perform image processing operations on various objects and/or features captured in a wide range of changing conditions (e.g., changes in image scales, noises, lighting, movement, or geometric distortion) can be challenging and computationally intensive.

[0023] As referred to herein, the phrase "augmented reality experience," "augmented reality content item," "augmented reality content generator" includes or refers to various image processing operations corresponding to an image modification, filter, Lenses, media overlay, transformation, and the like, as described further herein.

[0024] 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 application 104. Each messaging client application 104 is communicatively coupled to other instances of the messaging client application 104 and a messaging server system 108 via a network 106 (e.g., the Internet).

[0025] A messaging client application 104 is able to communicate and exchange data with another messaging client application 104 and with the messaging server system 108 via the network 106. The data exchanged between messaging client application 104, and between a messaging client application 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).

[0026] The messaging server system 108 provides server-side functionality via the network 106 to a particular messaging client application 104. While certain functions of the messaging system 100 are described herein as being performed by either a messaging client application 104 or by the messaging server system 108, the location of certain functionality either within the messaging client

application 104 or the messaging server system 108 is 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 application 104 where a
5 client device 102 has a sufficient processing capacity.

[0027] The messaging server system 108 supports various services and operations that are provided to the messaging client application 104. Such operations include transmitting data to, receiving data from, and processing data generated by the messaging client application 104. This data may include,
10 message content, client device information, geolocation information, media annotation 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 application 104.

[0028] Turning now specifically to the messaging server system 108, an Application Program Interface (API) server 110 is coupled to, and provides a programmatic interface to, an application server 112. The application server 112 is communicatively coupled to a database server 118, which facilitates access to a database 120 in which is stored data associated with messages processed by the
20 application server 112.

[0029] The Application Program Interface (API) server 110 receives and transmits message data (e.g., commands and message payloads) between the client device 102 and the application server 112. Specifically, the Application Program Interface (API) server 110 provides a set of interfaces (e.g., routines and protocols) that can be called or queried by the messaging client application
25 104 in order to invoke functionality of the application server 112. The Application Program Interface (API) server 110 exposes various functions supported by the application server 112, including account registration, login functionality, the sending of messages, via the application server 112, from a
30 particular messaging client application 104 to another messaging client application 104, the sending of media files (e.g., images or video) from a messaging client application 104 to the messaging server application 114, and for possible access by another messaging client application 104, the setting 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 adding and deletion of friends to a social graph, the location of friends within a social graph, and opening an application event (e.g., relating to the messaging client application 104).

[0030] The application server 112 hosts a number of applications and subsystems, including a messaging server application 114, an image processing system 116 and a social network system 122. The messaging server application 114 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 application 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, by the messaging server application 114, to the messaging client application 104. Other processor and memory intensive processing of data may also be performed server-side by the messaging server application 114, in view of the hardware requirements for such processing.

[0031] The application server 112 also includes an image processing system 116 that is dedicated to performing various image processing operations, typically with respect to images or video received within the payload of a message at the messaging server application 114.

[0032] The social network system 122 supports various social networking functions services, and makes these functions and services available to the messaging server application 114. To this end, the social network system 122 maintains and accesses an entity graph 304 (as shown in FIG. 3) within the database 120. Examples of functions and services supported by the social network system 122 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.

[0033] The application server 112 is communicatively coupled to a database server 118, which facilitates access to a database 120 in which is stored data associated with messages processed by the messaging server application 114.

[0034] FIG. 2 is block diagram illustrating further details regarding the messaging system 100, according to example embodiments. Specifically, the messaging system 100 is shown to comprise the messaging client application 104 and the application server 112, which in turn embody a number of some
5 subsystems, namely an ephemeral timer system 202, a collection management system 204 and an annotation system 206.

[0035] The ephemeral timer system 202 is responsible for enforcing the temporary access to content permitted by the messaging client application 104 and the messaging server application 114. To this end, the ephemeral timer
10 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 display and enable access to messages and associated content via the messaging client application 104. Further details regarding the operation of the ephemeral timer system 202 are provided below.

[0036] The collection management system 204 is responsible for managing
15 collections of media (e.g., collections of text, image video and audio data). In some examples, 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
20 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 application 104.

[0037] The collection management system 204 furthermore includes a curation
25 interface 208 that allows a collection manager to manage and curate a particular collection of content. For example, the curation interface 208 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
30 management system 204 employs machine vision (or image recognition technology) and content rules to automatically curate a content collection. In certain embodiments, compensation may be paid to a user for inclusion of user-generated content into a collection. In such cases, the curation interface 208

operates to automatically make payments to such users for the use of their content.

[0038] The annotation system 206 provides various functions that enable a user to annotate or otherwise modify or edit media content associated with a message. For example, the annotation system 206 provides functions related to the generation and publishing of media overlays for messages processed by the messaging system 100. The annotation system 206 operatively supplies a media overlay or supplementation (e.g., an image filter) to the messaging client application 104 based on a geolocation of the client device 102. In another example, the annotation system 206 operatively supplies a media overlay to the messaging client application 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) at the client device 102. For example, the media overlay may include text 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 annotation system 206 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 120 and accessed through the database server 118.

[0039] In one example embodiment, the annotation system 206 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 annotation system 206 generates a media overlay that includes the uploaded content and associates the uploaded content with the selected geolocation.

[0040] In another example embodiment, the annotation system 206 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 annotation system 206 associates the media overlay of a highest bidding merchant with a corresponding geolocation for a predefined amount of time.

[0041] FIG. 3 is a schematic diagram illustrating data structures 300 which may be stored in the database 120 of the messaging server system 108, according to certain example embodiments. While the content of the database 120 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).

[0042] The database 120 includes message data stored within a message table 314. The entity table 302 stores entity data, including an entity graph 304. Entities for which records are maintained within the entity table 302 may include individuals, corporate entities, organizations, objects, places, events, etc. Regardless of 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).

[0043] The entity graph 304 furthermore 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.

[0044] The database 120 also stores annotation data, in the example form of filters, in an annotation table 312. Filters for which data is stored within the annotation table 312 are associated with and applied to videos (for which data is stored in a video table 310) and/or images (for which data is stored in an image table 308). 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 gallery of filters presented to a sending user by the messaging client application 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 application 104, based on geolocation information determined by a GPS unit of the client device 102. Another type of filter is a data filter, which may be selectively presented to a sending user by the messaging client application 104, based on other inputs or information gathered by the client device 102 during the message creation process. Example 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.

[0045] Other annotation data that may be stored within the image table 308 are augmented reality content generators (e.g., corresponding to applying Lenses, augmented reality experiences, or augmented reality content items). An augmented reality content generator may be a real-time special effect and sound that may be added to an image or a video.

[0046] As described above, augmented reality content generators, augmented reality content items, overlays, image transformations, AR images and similar terms refer to modifications that may be made to videos or images. This includes real-time modification which modifies an image as it is captured using a device sensor and then displayed on a screen of the device 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 device with access to multiple augmented reality content generators, a user can use a single video clip with multiple augmented reality content generators to see how the different augmented reality content generators will modify the stored clip. For example, multiple augmented reality content generators that apply different pseudorandom movement models can be applied to the same content by selecting different augmented reality content generators 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 device 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 generators 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.

[0047] Data and various systems using augmented reality content generators 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 embodiments, different methods for achieving such transformations may be used. For example, some embodiments 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 embodiments, 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 embodiments, 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 generators 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.

[0048] 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.

[0049] In some embodiments, 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 of 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 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 method, a background of the modified object can be changed or distorted as well by tracking and modifying the background.

[0050] In one or more embodiments, transformations changing some areas of an object using its elements can be performed by calculating of 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 which are based on a request for modification; and modifying or distorting the elements of an area or object. In various embodiments, 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.

[0051] In some embodiments of a computer animation model to transform image data using face detection, the face is detected on an image with 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.

[0052] In other embodiments, other methods and algorithms suitable for face detection can be used. For example, in some embodiments, 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. In 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 embodiments, 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.

[0053] In some embodiments, 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 of 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 matchers to form a stronger overall classifier. The entire search is repeated at each level in an image pyramid, from coarse to fine resolution.

[0054] Embodiments of 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.

[0055] In some example embodiments, 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 application 104 operating on the client device 102. The transform system operating within the messaging client application 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 which 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). In some embodiments, a modified image or video stream may be presented in a graphical user interface displayed on the mobile client device as soon as the image or video stream is captured and a specified modification is selected. The transform 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.

[0056] In some embodiments, 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 embodiments, 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 browse 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 embodiments,

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.

5 [0057] In some example embodiments, a graphical processing pipeline architecture is provided that enables different augmented reality experiences (e.g., AR content generators) to be applied in corresponding different layers. Such a graphical processing pipeline provides an extensible rendering engine for providing multiple augmented reality experiences that are included in a
10 composite media (e.g., image or video) for rendering by the messaging client application 104 (or the messaging system 100).

[0058] As mentioned above, the video table 310 stores video data which, in one embodiment, is associated with messages for which records are maintained within the message table 314. Similarly, the image table 308 stores image data
15 associated with messages for which message data is stored in the entity table 302. The entity table 302 may associate various annotations from the annotation table 312 with various images and videos stored in the image table 308 and the video table 310.

[0059] A story table 306 stores data regarding collections of messages and
20 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 302). A user may create a ‘personal story’ in the form of a collection of content that has been created and sent/broadcast by that user. To
25 this end, the user interface of the messaging client application 104 may include an icon that is user-selectable to enable a sending user to add specific content to his or her personal story.

[0060] A collection may also constitute a ‘live story,’ which is a collection of
30 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 application 104, to contribute content to a particular live story. The live story may be identified to the user by the messaging client application 104, based on his or her location. The end result is a ‘live story’ told from a community perspective.

5 [0061] 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 embodiments, a contribution to a location story may require a second degree of authentication to verify that the end user belongs
10 to a specific organization or other entity (e.g., is a student on the university campus).

[0062] FIG. 4 is a schematic diagram illustrating a structure of a message 400, according to some embodiments, generated by a messaging client application 104 for communication to a further messaging client application 104 or the
15 messaging server application 114. The content of a particular message 400 is used to populate the message table 314 stored within the database 120, accessible by the messaging server application 114. 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 server 112. The message 400 is shown to include
20 the following components:

[0063] A message identifier 402: a unique identifier that identifies the message 400.

[0064] A 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.

25 [0065] A 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.

[0066] A message video payload 408: video data, captured by a camera component or retrieved from a memory component of the client device 102 and
30 that is included in the message 400.

[0067] A 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.

- [0068]** A message annotations 412: annotation data (e.g., filters, stickers or other enhancements) that represents annotations to be applied to message image payload 406, message video payload 408, or message audio payload 410 of the message 400.
- 5 **[0069]** A 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 application 104.
- 10 **[0070]** A 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
- 15 the message image payload 406, or a specific video in the message video payload 408).
- [0071]** A message story identifier 418: identifier values identifying one or more content collections (e.g., ‘stories’) with which a particular content item in the message image payload 406 of the message 400 is associated. For example,
- 20 multiple images within the message image payload 406 may each be associated with multiple content collections using identifier values.
- [0072]** A 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
- 25 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.
- [0073]** A message sender identifier 422: an identifier (e.g., a messaging system
- 30 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

[0074] A 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.

[0075] 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 308. Similarly, values within the message video payload 408 may point to data stored within a video table 310, values stored within the message annotations 412 may point to data stored in an annotation table 312, values stored within the message story identifier 418 may point to data stored in a story table 306, 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 302.

[0076] 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).

[0077] 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 application 104. In one embodiment, 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.

[0078] The message duration parameter 506 and the message receiver identifier 424 are shown to be inputs to a message timer 512, 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 512 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.

[0079] 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.

[0080] Additionally, each ephemeral message 502 within the ephemeral message group 504 has an associated group participation parameter 510, 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 510, 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.

[0081] 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 embodiment, 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 embodiment, a certain

ephemeral message 502 may expire, within the context of ephemeral message group 504, based on a group participation parameter 510. 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
5 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.

10 **[0082]** 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 510. For example, when a sending user has established a group participation parameter 510 of 24 hours from posting, the ephemeral timer
15 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 either when the group participation parameter 510 for each and every ephemeral message 502 within the ephemeral message group 504 has expired, or when the ephemeral
20 message group 504 itself has expired in terms of the group duration parameter 508.

[0083] 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 510 for the last remaining
25 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 510, effectively extends the life of an ephemeral message group 504 to equal the value of the group participation parameter 510.

30 **[0084]** 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 application 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 application 104. Similarly, when the ephemeral timer system 202 determines that the message duration parameter 506 for a particular ephemeral message 502
5 has expired, the ephemeral timer system 202 causes the messaging client application 104 to no longer display an indicium (e.g., an icon or textual identification) associated with the ephemeral message 502.

[0085] As described above, media overlays, such as Lenses, overlays, image transformations, AR images and similar terms refer to modifications that may be
10 made to videos or images. This includes real-time modification which modifies an image as it is captured using a device sensor and then displayed on a screen of the device 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 device with access to multiple media overlays (e.g., Lenses), a user can use a
15 single video clip with multiple Lenses to see how the different Lenses will modify the stored clip. For example, multiple Lenses that apply different pseudorandom movement models can be applied to the same content by selecting different Lenses for the content. Similarly, real-time video capture may be used with an illustrated modification to show how video images
20 currently being captured by sensors of a device 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 Lenses will look within different
25 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.

[0086] Data and various systems to use Lenses or other such transform systems to modify content using this data can thus involve detection of objects
30 (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 embodiments, different methods for achieving such transformations may be used. For example, some embodiments 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 embodiments, 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 embodiments, neural network analysis of video frames may be used to place images, models, or textures in content (e.g. images or frames of video). Lens data thus refers 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.

[0087] 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.

[0088] In some embodiments, 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 of 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 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 method a background of the modified object can be changed or distorted as well by tracking and modifying the background.

5 [0089] In one or more embodiments, transformations changing some areas of an object using its elements can be performed by calculating of 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
10 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:
15 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 which are based on a request for modification; and modifying or distorting the elements of an area or object. In various embodiments, any combination of such modifications or other similar modifications may be used. For certain models to be animated,
20 some characteristic points can be selected as control points to be used in determining the entire state-space of options for the model animation.

[0090] In some embodiments of a computer animation model to transform image data using face detection, the face is detected on an image with use of a specific face detection algorithm (e.g. Viola-Jones). Then, an Active Shape
25 Model (ASM) algorithm is applied to the face region of an image to detect facial feature reference points.

[0091] In other embodiments, other methods and algorithms suitable for face detection can be used. For example, in some embodiments, features are located using a landmark which represents a distinguishable point present in most of the
30 images under consideration. For facial landmarks, for example, the location of the left eye pupil may be used. In 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

embodiments, 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
5 shape is the mean of the aligned training shapes.

[0092] In some embodiments, 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 of the image texture
10 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 matchers to form a stronger overall classifier. The entire search is repeated at each level in an image pyramid, from coarse to fine resolution.

[0093] Embodiments of a transformation system can capture an image or video stream on a client device and perform complex image manipulations locally on a client device such as client device 102 while maintaining a suitable
15 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
20 implemented by a convolutional neural network that has been configured to execute efficiently on a client device.

[0094] In some example embodiments, 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
25 a neural network operating as part of a messaging client application 104 operating on the client device 102. The transform system operating within the messaging client application 104 determines the presence of a face within the
30 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 which 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). In some embodiments, a modified image or video stream may be presented in a graphical user interface displayed on the mobile client device as soon as the image or video stream is captured and a specified modification is selected. The transform 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.

[0095] In some embodiments, 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 embodiments, 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 browse 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 embodiments, 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.

[0096] In some example embodiments, a graphical processing pipeline architecture is provided that enables different media overlays to be applied in corresponding different layers. Such a graphical processing pipeline provides an

extensible rendering engine for providing multiple augmented reality content generators that are included in a composite media (e.g., image or video) for rendering by the messaging client application 104 (or the messaging system 100).

5 **[0097]** As discussed herein, the subject infrastructure supports the creation and sharing of interactive messages with interactive effects throughout various components of the messaging system 100. In an example, to provide such interactive effects, a given interactive message may include image data along with 2D data, or 3D data. The infrastructure as described herein enables other
10 forms of 3D and interactive media (e.g., 2D media content) to be provided across the subject system, which allows for such interactive media to be shared across the messaging system 100 and alongside photo and video messages. In example embodiments described herein, messages can enter the system from a live camera or via from storage (e.g., where messages with 2D or 3D content or
15 augmented reality (AR) effects (e.g., 3D effects, or other interactive effects are stored in memory or a database). In an example of an interactive message with 3D data, the subject system supports motion sensor input and manages the sending and storage of 3D data, and loading of external effects and asset data.

[0098] As mentioned above, an interactive message includes an image in
20 combination with a 2D effect, or a 3D effect and depth data. In an example embodiment, a message is rendered using the subject system to visualize the spatial detail / geometry of what the camera sees, in addition to a traditional image texture. When a viewer interacts with this message by moving a client device, the movement triggers corresponding changes in the perspective the
25 image and geometry are rendered at to the viewer.

[0099] In an embodiment, the subject system provides AR effects (which may include 3D effects using 3D data, or interactive 2D effects that do not use 3D data) that work in conjunction with other components of the system to provide particles, shaders, 2D assets and 3D geometry that can inhabit different 3D-
30 planes within messages. The AR effects as described herein, in an example, are rendered in a real-time manner for the user.

[00100] As mentioned herein, a gyro-based interaction refers to a type of interaction in which a given client device's rotation is used as an input to change

an aspect of the effect (e.g., rotating phone along x-axis in order to change the color of a light in the scene).

[00101] As mentioned herein, an augmented reality content generator refers to a real-time special effect and/or sound that may be added to a message and
5 modifies image and/or 3D data with an AR effects and/other 3D content such as 3D animated graphical elements, 3D objects (e.g., non-animated), and the like.

[00102] The following discussion relates to example data that is stored in connection with such a message in accordance to some embodiments.

[00103] FIG. 6 is a schematic diagram illustrating a structure of the message annotations 412, as described above in FIG. 4, including additional information
10 corresponding to a given message, according to some embodiments, generated by the messaging client application 104.

[00104] In an embodiment, the content of a particular message 400, as shown in FIG. 3, including the additional data shown in FIG. 6 is used to populate the
15 message table 314 stored within the database 120 for a given message, which is then accessible by the messaging client application 104. As illustrated in FIG. 6, message annotations 412 includes the following components corresponding to various data:

- augmented reality (AR) content identifier 652: identifier of an AR
20 content generator utilized in the message
- message identifier 654: identifier of the message
- asset identifiers 656: a set of identifiers for assets in the message. For example, respective asset identifiers can be included for assets that are
25 determined by the particular AR content generator. In an embodiment, such assets are created by the AR content generator on the sender side client device, uploaded to the messaging server application 114, and utilized on the receiver side client device in order to recreate the message. Examples of typical assets include:

- * The original still RGB image(s) captured by the camera
- * The post-processed image(s) with AR content generator
30 effects applied to the original image

- augmented reality (AR) content metadata 658: additional metadata associated with the AR content generator corresponding to the AR identifier 652, such as:
 - AR content generator category: corresponding to a type or classification for a particular AR content generator
 - AR content generator carousel index
 - carousel group: This can be populated and utilized when eligible post-capture AR content generators are inserted into a carousel interface. In an implementation, a new value “AR_DEFAULT_GROUP” (e.g., a default group assigned to a particular AR content generator can be added to the list of valid group names, and other selected AR content generators can be included this group.
- capture metadata 660 corresponding to additional metadata, such as:
 - camera image metadata
 - camera intrinsic data
 - focal length
 - principal point
 - other camera information (e.g., camera position)
 - sensor information
 - gyroscopic sensor data
 - position sensor data
 - accelerometer sensor data
 - other sensor data
 - location sensor data

[00105] FIG. 7 is a block diagram 700 illustrating various modules of an annotation system 206, according to certain example embodiments. The annotation system 206 is shown as including an image data receiving module 702, a sensor data receiving module 704, an image data processing module 706, an augmented reality (AR) effects module 708, a rendering module 710, and a sharing module 712. The various modules of the annotation system 206 are configured to communicate with each other (e.g., via a bus, shared memory, or a switch). Any one or more of these modules may be implemented using one or more computer processors 720 (e.g., by configuring such one or more computer

processors to perform functions described for that module) and hence may include one or more of the computer processors 720 (e.g., a set of processors provided by the client device 102).

[00106] Any one or more of the modules described may be implemented using hardware alone (e.g., one or more of the computer processors 720 of a machine (e.g., machine 1800) or a combination of hardware and software. For example, any described module of the annotation system 206 may physically include an arrangement of one or more of the computer processors 720 (e.g., a subset of or among the one or more computer processors of the machine (e.g., machine 1800) configured to perform the operations described herein for that module. As another example, any module of the annotation system 206 may include software, hardware, or both, that configure an arrangement of one or more computer processors 720 (e.g., among the one or more computer processors of the machine (e.g., machine 1800) to perform the operations described herein for that module. Accordingly, different modules of the annotation system 206 may include and configure different arrangements of such computer processors 720 or a single arrangement of such computer processors 720 at different points in time. Moreover, any two or more modules of the annotation system 206 may be combined into a single module, and the functions described herein for a single module may be subdivided among multiple modules. Furthermore, according to various example embodiments, modules described herein as being implemented within a single machine, database, or device may be distributed across multiple machines, databases, or devices.

[00107] The image data receiving module 702 receives images and depth data captured by a client device 102. For example, an image is a photograph captured by an optical sensor (e.g., camera) of the client device 102. An image includes one or more real-world features, such as a user's face or real-world object(s) detected in the image. In some embodiments, an image includes metadata describing the image.

[00108] The sensor data receiving module 704 receives sensor data from a client device 102. Sensor data is any type of data captured by a sensor of the client device 102. In an example, sensor data can include motion of the client device 102 gathered by a gyroscope, touch inputs or gesture inputs from a touch

sensor (e.g., touchscreen), GPS, or another sensor of the client device 102 that describes a current geographic location and/or movement of the client device 102. As another example, sensor data may include temperature data indicating a current temperature as detected by a sensor of the client device 102. As another
5 example, the sensor data may include light sensor data indicating whether the client device 102 is in a dark or bright environment.

[00109] The image data processing module 706 performs operations on the received image data. For example, various image processing operations are performed by the image data processing module 706, which are discussed further
10 herein.

[00110] The AR effects module 708 performs various operations based on algorithms or techniques that correspond to animations and/or providing visual and/or auditory effects to the received image data, which is described further herein. In an embodiment, a given augmented reality content generator can
15 utilize the AR effects module 708 to perform operations to render AR effects (e.g., including 2D effects or 3D effects) and the like.

[00111] The rendering module 710 performs rendering of the message for display by the messaging client application 104 based on data provided by at least one of the aforementioned modules. In an example, the rendering module
20 710 utilizes a graphical processing pipeline to perform graphical operations to render the message for display. The rendering module 710 implements, in an example, an extensible rendering engine which supports multiple image processing operations corresponding to respective augmented reality content generators.

[00112] In some implementations, the rendering module 710 provide a graphics system that renders two-dimensional (2D) objects or objects from a three-dimensional (3D) world (real or imaginary) onto a 2D display screen. Such a graphics system (e.g., one included on the client device 102) includes a graphics processing unit (GPU) in some implementations for performing image
30 processing operations and rendering graphical elements for display.

[00113] In an implementation, the GPU includes a logical graphical processing pipeline, which can receive a representation of a 2D or 3D scene and provide an output of a bitmap that represents a 2D image for display. Existing application

programming interfaces (APIs) have implemented graphical pipeline models. Examples of such APIs include the Open Graphics Library (OpenGL) API and the METAL API. The graphical processing pipeline includes a number of stages to convert a group of vertices, textures, buffers, and state information into an
5 image frame on the screen. In an implementation, one of the stages of the graphical processing pipeline is a shader, which may be utilized as part of a particular augmented reality content generator that is applied to an input frame (e.g., image or video). A shader can be implemented as code running on a specialized processing unit, also referred to as a shader unit or shader processor,
10 usually executing several computing threads, programmed to generate appropriate levels of color and/or special effects to fragments being rendered. For example, a vertex shader processes attributes (position, texture coordinates, color, etc.) of a vertex, and a pixel shader processes attributes (texture values, color, z-depth and alpha value) of a pixel. In some instances, a pixel shader is
15 referred to as a fragment shader.

[00114] It is to be appreciated that other types of shader processes may be provided. In an example, a particular sampling rate is utilized, within the graphical processing pipeline, for rendering an entire frame, and/or pixel shading is performed at a particular per-pixel rate. In this manner, a given electronic
20 device (e.g., the client device 102) operates the graphical processing pipeline to convert information corresponding to objects into a bitmap that can be displayed by the electronic device.

[00115] The sharing module 712 generates the message for storing and/or sending to the messaging server system 108. The sharing module 712 enables
25 sharing of messages to other users and/or client devices of the messaging server system 108.

[00116] The augmented reality content generator module 714 cause display of selectable graphical items that, in an embodiment, are presented in a carousel arrangement. By way of example, the user can utilize various inputs to rotate the
30 selectable graphical items onto and off of the display screen in manner corresponding to a carousel providing a cyclic view of the graphical items. The carousel arrangement allows multiple graphical items to occupy a particular graphical area on the display screen. In an example, augmented reality content

generators can be organized into respective groups for including on the carousel arrangement thereby enabling rotating through augmented reality content generators by group.

5 [00117] In embodiments described herein, by using depth and image data, 3D face and scene reconstruction can be performed that adds a Z-axis dimension (e.g., depth dimension) to a traditional 2D photos (e.g., X-axis and Y-axis dimensions). This format enables the viewer to interact with the message, changing the angle/perspective in which the message is rendered by the subject system, and affecting particles and shaders that are utilized in rendering the
10 message.

[00118] In an example, viewer interaction input comes from movement (e.g., from a movement sensor of the device displaying the message to the viewer) whilst viewing the message, which in turn is translated to changes in perspective for how content, particles and shaders are rendered. Interaction can also come
15 from onscreen touch gestures and other device motion.

[00119] In embodiments of such user interfaces, selectable graphical items may be presented in a carousel arrangement in which a portion or subset of the selectable graphical items are visible on a display screen of a given computing device (e.g., the client device 102). By way of example, the user can utilize
20 various inputs to rotate the selectable graphical items onto and off of the display screen in manner corresponding to a carousel providing a cyclic view of the graphical items. The carousel arrangement as provided in the user interfaces therefore allow multiple graphical items to occupy a particular graphical area on the display screen.

25 [00120] In an example, respective AR experiences corresponding to different AR content generators can be organized into respective groups for including on the carousel arrangement thereby enabling rotating through media overlays by group. Although a carousel interface is provided as an example, it is appreciated that other graphical interfaces may be utilized. For example, a set of augmented
30 reality content generators can include graphical list, scroll list, scroll graphic, or another graphical interface that enables navigation through various graphical items for selection, and the like. As used herein a carousel interface refers to display of graphical items in an arrangement similar to a circular list, thereby

enabling navigation, based on user inputs (e.g., touch or gestures), through the circular list to select or scroll through the graphical items. In an example, a set of graphical items may be presented on a horizontal (or vertical) line or axis where each graphical item is represented as a particular thumbnail image (or icon, avatar, and the like). At any one time, some of the graphical items in the carousel interface may be hidden. If the user wants to view the hidden graphical items, in an example, the user may provide a user input (e.g., touch, gesture, and the like) to scroll through the graphical items in a particular direction (e.g., left, right, up, or down, and the like). Afterward, a subsequent view of the carousel interface is displayed where an animation is provided or rendered to present one or more additional graphical items for inclusion on the interface, and where some of the previously presented graphical items may be hidden in this subsequent view. In an embodiment, in this manner the user can navigate through the set of graphical items back and forth in a circular fashion. Thus, it is appreciated that the carousel interface can optimize screen space by displaying only a subset of images from a set of graphical items in a cyclic view.

[00121] As described herein, augmented reality content generators can included on the carousel arrangement (or another interface as discussed above) thereby enabling rotating through augmented reality content generators. Further, augmented reality content generators can be selected for inclusion based on various signals including, for example, time, date, geolocation, metadata associated with the media content, and the like.

[00122] In some implementations, the messaging system 100 determines a lifetime value (LTV) analysis for users of augmented reality content generators based on hard-coded heuristics, but this approach may not be scalable as the number of users increases within the messaging system 100. The subject technology provides new techniques to determine the current and future monetization state of users of augmented reality content generators, and determine the monetization potential of users of augmented reality content generators, which are driven to the messaging system 100 by viral (e.g., popular in usage) augmented reality content generators. LTV as referred to herein can include revenue from online advertisements (e.g., ads), story ads and sponsored creative tools.

[00123] The messaging system 100 (or components thereof) provides techniques to address the following technical challenges:

- 5 1. Bridge the gap of the top line number of monetized users of augmented reality content generators
 - In an example, 12% of users of augmented reality content generators monthly active users (MAU) in the US are monetized on a daily basis, compared to 60% overall level in the US;
 - In an example, globally, 2% of users of augmented reality content generators MAU are monetized on a daily basis while it is 33% at overall level;
- 10 2. Increase the LTV of users of augmented reality content generators or the % of users of augmented reality content generators LTV of all users
 - In an example, users of augmented reality content generators LTV is 13% all users LTV in the US
 - In an example, users of augmented reality content generators LTV is 8% all users LTV globally
- 15 3. Graduate users of augmented reality content generators into other highly monetized and more similar cohorts
 - As users of augmented reality content generators are more passive and have fewer friends compared to other highly engaged cohorts, determine ways to transition them into more similar cohorts, which in an example can be more than 10 times monetized on the platform globally
- 20
- 25

[00124] The messaging system 100 (or components thereof), in particular, implements the following approaches to address the above:

- 30 1. Seasonality (e.g., where data can experience regular and predictable changes that recur every calendar year or other time period) can impact the prediction of customer average transaction value. The messaging system 100 (or components thereof) decomposes time series transaction data into trend and seasonality. In an example, the seasonality issue can be resolved more naturally if the messaging system 100 (or components thereof) utilizes a longer period, and has a good user level revenue forecasting model from the business.
- 35 2. The messaging system 100 (or components thereof) uses the same methodology to check the LTV of any experiments designed to increase the monetized value of users of augmented reality content generators and quantify the impact from monetization perspective
- 40 3. The messaging system 100 (or components thereof) incorporates current analysis into further development of clustering analysis for better understanding of cohort LTV.

4. The messaging system 100 (or components thereof) uses the same methodology to model LTV for other user personas.

5 [00125] In some implementations, the messaging system 100 (or components thereof) provides and utilizes a model to determine LTV of users. In an example, a key assumption behind the model is to consider the user-level transaction or ads as a non-contractual scenario, and the goal is to model users based on past transaction behaviors from both transaction recency/frequency distribution and transaction monetary value distribution.

10 [00126] To model transaction recency/frequency distribution, the messaging system 100 (or components thereof) implements a model based on a BG/NBD (Beta Geometric Negative Binomial Distribution) model after the modification from a Pareto/NBD model. A goal is to make sure customers, either as an individual or as a whole, follow certain behaviors to have the desired outcome.

15 In an implementation, the following are key assumptions for the model:

1. the number of transactions made by a customer follows a Poisson process with transaction rate λ
2. Heterogeneity (e.g., populations, samples or results are different) in λ follows a gamma distribution
- 20 3. After any transaction, a customer becomes inactive with probability p . Therefore the point at which the customer “drops out” is distributed across transactions according to a (shifted) geometric distribution
4. Heterogeneity in p follows a beta distribution
- 25 5. The transaction rate λ and the dropout probability p vary independently across customers

[00127] To model the future monetary value, the messaging system 100 (or components thereof) follows a Gamma-Gamma model (e.g., where the monetary value of a user’s given transaction varies randomly around their average transaction value, average transaction values vary across users but do not vary over time for any given individual (user), or distribution of average transaction values across users is independent of the transaction process) to predict the monetary values, and the user profile follows the following general guidelines:

- 35 1. The monetary value of a customer’s given transaction varies randomly around their average transaction value.

2. Average transaction values vary across customers but do not vary over time for any given individual.
3. The distribution of average transaction values across customers is independent of the transaction process.

5

[00128] FIG. 8 illustrates an example chart for monthly active users (MAU) and penetration of global camera users, in accordance with some embodiments.

[00129] As shown in chart 800, users of AR content generators have a much higher penetration in MAU, increasing from 25% to over 30% after AR content generators for facial effects. For the US market, chart 800 shows a very similar trend, which shows the users of AR content generators' penetration from 8% to over 10%.

[00130] FIG. 9 illustrates an example chart for users of AR content generators MAU and penetration, in accordance with some embodiments.

[00131] As shown in chart 900, although users of AR content generators account for almost one third of the MAU, the % of monetized MAU on a daily basis is really low. Only around 12% for users of AR content generators in the United States (US) are monetized on a daily basis while it is close to 60% of MAU in the US at the overall level. There is still much room to grow this number over time given the size of this group of users is likely to increase over time.

[00132] For the customer LTV analysis, the discussion further herein mostly focuses on the US market as it is the most monetized market and will give a better understanding in terms of future monetization expectations and directions. Because of the size of the users for analysis, sampling is utilized to speed up the training data preparation process. For the monetization value prediction, a discount rate of 5% is assumed.

[00133] As shown in Table 1 below, the average users of AR content generators' LTV is around 30% lower after such users have used the AR content generators for facial effects [\$2.42 post vs. \$3.43 pre]. As also shown in Table 1, this is driven by much lower LTV of new users of AR content generators compared to pre-AR content generators for facial effects [\$2.09 new post vs. \$3.43 pre], which drags down the average LTV for all users of AR content

generators post AR content generators for facial effects (e.g., after such users have used the AR content generators for facial effects).

Cohort	US	Global
users of AR content generators Pre-AR content generators for facial effects	\$3.43	
users of AR content generators Post-AR content generators for facial effects	\$2.42	\$0.86
All users Post-AR content generators for facial effects	\$18.05	\$10.85
New users of AR content generators Post-AR content generators for facial effects	\$2.09	
All Discover Watcher	\$18.48	\$11.23

5 Table 1. Summarized average LTV for different user cohort at the end of year 2 assuming discount rate at 0.05

10 **[00134]** At the same time as shown in Table 2 below, the cumulative revenue for all users is slightly higher post AR content generators for facial effects [**\$16.88M post vs. \$16.73M pre**]. AR content generators for facial effects may not have materially improved the overall monetization opportunity for users of AR content generators, with only 1% revenue increase, but there is a room to improve the monetization likelihood for lightly or newly engaged users of AR content generators, and a clustering analysis can be performed for users of AR content generators pre and post AR content generators for facial effects.

User Cohort	users of AR content generators Pre-AR content generators for facial effects US	users of AR content generators Post-AR content generators for facial effects US	users of AR content generators Post-AR content generators for facial effects Globally
Cumulative LTV at end of 2 years (million)	\$16.73	\$16.88	\$24.60

20 Table 2. Summarized cumulative LTV for different users of AR content generators cohort at the end of year 2 assuming discount rate at 0.05

[00135] As shown in the Table 1 above, globally, the average LTV is about 36% of the LTV in the US [\$0.86 global vs. \$2.42 US] but given its much larger volume in the international market, it adds around 46% more revenue [\$24.60M vs. \$16.88M] to the cumulative revenue. Even though US users of AR content
5 generators have been monetized relatively well, its LTV is only 13% of overall users in the US [\$2.42 users of AR content generators vs. \$18.05 all users].

[00136] In the following discussion related to model validation, the messaging system 100 (or components thereof) uses the timeframe of 09/01/19 and 12/31/19 to build the model and validate. This is done to separate users from the
10 impact of AR content generators for facial effects in the summer and wait for new users to be qualified for seeing ads. In an embodiment, there two ways to check how models perform:

1. Use the whole time period to check how the actual is compared to model for repeat transactions, and compare the difference
- 15 2. Have a short holdout period from this period to test model accuracy for user transaction frequency

[00137] In some embodiments, as shown in FIG. 10, the messaging system 100 (or components thereof) can determine Actual and Model comparison at different transaction frequency.

20 **[00138]** FIG. 10 illustrates an example chart 1000 for Sample Full Time Period Model Validation, in accordance with some embodiments.

[00139] In an embodiment, the messaging system 100 (or components thereof) uses 09/01/19 to 11/30/19 for calibration, and 12/01/19 and 12/31/19 for holdout testing.

25 **[00140]** FIG. 11 illustrates an example chart for Sample Holdout Period Model Validation, in accordance with some embodiments.

[00141] In this example, chart 1100 shows the cumulative error rate up to purchase frequency of 9, which shows the most traffic of data, in the calibration period. The model is reasonably accurate except for a large cohort of users like
30 all users in US post-AR content generators for facial effects. The accuracy can be improved by modeling a subset or cohort of users individually and using a longer period for training and testing. In an example, the purchase frequency up to a value of 9 in chart 1100 corresponds to a cumulative transaction number

over the calibration period (e.g., 09/01/19 to 11/31/19), which was set to this upper limit to provide a more accurate picture of underlying errors. In an example, the cumulative error rate being under a predetermined percentage (e.g., 5%) indicates that the model is accurate.

5

Cohort	US	Global
users of AR content generators Pre-AR content generators for facial effects	-0.04%	
users of AR content generators Post-AR content generators for facial effects	-1.5%	-4.3%
All users Post-AR content generators for facial effects	23.2%	0.2%
New users of AR content generators Post-AR content generators for facial effects	8.8%	
All users Post-AR content generators for facial effects Discover Watcher	-34.7%	-31.5%

Table 3. Cumulative holdout error up to frequency of 9 in calibration period

10 **[00142]** FIG. 12 illustrates examples of user interfaces 1200, and presenting content items generated by an AR content generator in the messaging client application 104 (or the messaging system 100), according to some embodiments.

[00143] As shown, user interfaces 1200 includes AR content items that can be selected and rendered based at least in part on the aforementioned LTV techniques.

15 **[00144]** FIG. 13 illustrates examples of user interfaces 1300, and presenting content items generated by an AR content generator in the messaging client application 104 (or the messaging system 100), according to some embodiments.

20 **[00145]** As shown, user interfaces 1300 includes AR content items that can be selected and rendered based at least in part on the aforementioned LTV techniques.

[00146] FIG. 14 illustrates examples of user interfaces 1400 and 1450, and presenting content items generated by an AR content generator in the messaging

client application 104 (or the messaging system 100), according to some embodiments.

[00147] As shown, user interfaces 1400 and 1450 includes AR content items that can be selected and rendered based at least in part on the aforementioned
5 LTV techniques.

[00148] FIG. 15 is a flowchart illustrating a method 1500, according to certain example embodiments. The method 1500 may be embodied in computer-readable instructions for execution by one or more computer processors such that the operations of the method 1500 may be performed in part or in whole by
10 the messaging system 100; accordingly, the method 1500 is described below by way of example with reference thereto. However, it shall be appreciated that at least some of the operations of the method 1500 may be deployed on various other hardware configurations and the method 1500 is not intended to be limited to the messaging system 100, and can be performed by the messaging client
15 application 104, or components of the annotation system 206.

[00149] At operation 1502, the messaging system 100 determines a transaction recency and frequency distribution.

[00150] At operation 1504, the messaging system 100 determines a future monetary value.

20 [00151] At operation 1506, the messaging system 100 determines monthly active users (MAU) and penetration for global users and users in a specific country.

[00152] At operation 1508, the messaging system 100 predicts monetization values for the users in the specific country.

25 [00153] At operation 1510, the messaging system 100 determines a lifetime value of the users in the specific country based at least in part on the monetization values. In an embodiment, determining the lifetime value comprises validating a model of the lifetime value based a period of time, determining a holdout period from the period of time to test model accuracy for
30 user transaction frequency, and comparing a first set of values corresponding to actual purchases during the holdout period with a second set of values corresponding to purchases in a calibration period to determine a cumulative error rate.

[00154] FIG. 16 is a flowchart illustrating a method 1600, according to certain example embodiments. The method 1600 may be embodied in computer-readable instructions for execution by one or more computer processors such that the operations of the method 1600 may be performed in part or in whole by the messaging system 100; accordingly, the method 1600 is described below by way of example with reference thereto. However, it shall be appreciated that at least some of the operations of the method 1600 may be deployed on various other hardware configurations and the method 1600 is not intended to be limited to the messaging system 100, and can be performed by the messaging client application 104, or components of the annotation system 206.

[00155] At operation 1602, the messaging system 100 determines monthly active users (MAU) and penetration for global users and users in a specific country. In an embodiment, the messaging system 100 determines that users of the at least one AR content generator has a higher penetration in MAU. The messaging system determines that a percentage of monetized users on a daily basis is lower than a percentage of the users of at least one AR content generator.

[00156] At operation 1604, the messaging system 100 predicts monetization values for the users in the specific country.

[00157] At operation 1606, the messaging system 100 determines a lifetime value of the users in the specific country based at least in part on the monetization values. In an example, the messaging system 100 determines that lifetime values of users of the at least one AR content generator are lower than prior lifetime values of the users before using the at least one AR content generator, determines that cumulative revenue for users is higher after using the at least one AR content generator, and determines a set of new users that have not accessed the at least one AR content generator, the set of new users have accessed other AR content generators different than the at least one AR content generator.

[00158] At operation 1608, the messaging system 100 selects at least one augmented reality (AR) content generator based at least in part on the determined lifetime value of the users in the specific country. In an embodiment, selecting the at least one AR content generator based at least in

part on determining that the percentage of monetized users on the daily basis is lower than a percentage of the users of the at least one AR content generator.

[00159] At operation 1610, the messaging system 100 causes, at a client device, display of the at least one AR content generator. In an embodiment, the
5 messaging system provides, for display, the at least one AR content generator at a particular client device associated with at least one new user from the set of new users, wherein the set of new users are selected to improve monetization based on revenue from revenue from online advertisements and sponsored creative tools.

10 **[00160]** FIG. 17 is a block diagram illustrating an example software architecture 1706, which may be used in conjunction with various hardware architectures herein described. FIG. 17 is a non-limiting example of a software architecture and it will be appreciated that many other architectures may be implemented to facilitate the functionality described herein. The software
15 architecture 1706 may execute on hardware such as machine 1800 of FIG. 18 that includes, among other things, processors 1804, memory 1814, and (input/output) I/O components 1818. A representative hardware layer 1752 is illustrated and can represent, for example, the machine 1800 of FIG. 18. The representative hardware layer 1752 includes a processing unit 1754 having
20 associated executable instructions 1704. Executable instructions 1704 represent the executable instructions of the software architecture 1706, including implementation of the methods, components, and so forth described herein. The hardware layer 1752 also includes memory and/or storage modules memory/storage 1756, which also have executable instructions 1704. The
25 hardware layer 1752 may also comprise other hardware 1758.

[00161] In the example architecture of FIG. 17, the software architecture 1706 may be conceptualized as a stack of layers where each layer provides particular functionality. For example, the software architecture 1706 may include layers such as an operating system 1702, libraries 1720, frameworks/middleware 1718,
30 applications 1716, and a presentation layer 1714. Operationally, the applications 1716 and/or other components within the layers may invoke API calls 1708 through the software stack and receive a response as in messages 1712 to the API calls 1708. The layers illustrated are representative in nature and not all

software architectures have all layers. For example, some mobile or special purpose operating systems may not provide a frameworks/middleware 1718, while others may provide such a layer. Other software architectures may include additional or different layers.

5 **[00162]** The operating system 1702 may manage hardware resources and provide common services. The operating system 1702 may include, for example, a kernel 1722, services 1724, and drivers 1726. The kernel 1722 may act as an abstraction layer between the hardware and the other software layers. For example, the kernel 1722 may be responsible for memory management,
10 processor management (e.g., scheduling), component management, networking, security settings, and so on. The services 1724 may provide other common services for the other software layers. The drivers 1726 are responsible for controlling or interfacing with the underlying hardware. For instance, the drivers 1726 include display drivers, camera drivers, Bluetooth® drivers, flash
15 memory drivers, serial communication drivers (e.g., Universal Serial Bus (USB) drivers), Wi-Fi® drivers, audio drivers, power management drivers, and so forth depending on the hardware configuration.

[00163] The libraries 1720 provide a common infrastructure that is used by the applications 1716 and/or other components and/or layers. The libraries 1720
20 provide functionality that allows other software components to perform tasks in an easier fashion than to interface directly with the underlying operating system 1702 functionality (e.g., kernel 1722, services 1724 and/or drivers 1726). The libraries 1720 may include system libraries 1744 (e.g., C standard library) that may provide functions such as memory allocation functions, string manipulation
25 functions, mathematical functions, and the like. In addition, the libraries 1720 may include API libraries 1746 such as media libraries (e.g., libraries to support presentation and manipulation of various media format such as MPREG4, H.264, MP3, AAC, AMR, JPG, PNG), graphics libraries (e.g., an OpenGL
30 framework that may be used to render 2D and 3D in a graphic content on a display), database libraries (e.g., SQLite that may provide various relational database functions), web libraries (e.g., WebKit that may provide web browsing functionality), and the like. The libraries 1720 may also include a wide variety of other libraries 1748 to provide many other APIs to the applications 1716 and other software components/modules.

[00164] The frameworks/middleware 1718 (also sometimes referred to as middleware) provide a higher-level common infrastructure that may be used by the applications 1716 and/or other software components/modules. For example, the frameworks/middleware 1718 may provide various graphic user interface (GUI) functions, high-level resource management, high-level location services, and so forth. The frameworks/middleware 1718 may provide a broad spectrum of other APIs that may be used by the applications 1716 and/or other software components/modules, some of which may be specific to a particular operating system 1702 or platform.

10 [00165] The applications 1716 include built-in applications 1738 and/or third-party applications 1740. Examples of representative built-in applications 1738 may include, but are not limited to, a contacts application, a browser application, a book reader application, a location application, a media application, a messaging application, and/or a game application. Third-party applications 1740
15 may include an application developed using the ANDROID™ or IOS™ software development kit (SDK) by an entity other than the vendor of the particular platform, and may be mobile software running on a mobile operating system such as IOS™, ANDROID™, WINDOWS® Phone, or other mobile operating systems. The third-party applications 1740 may invoke the API calls 1708
20 provided by the mobile operating system (such as operating system 1702) to facilitate functionality described herein.

[00166] The applications 1716 may use built in operating system functions (e.g., kernel 1722, services 1724 and/or drivers 1726), libraries 1720, and frameworks/middleware 1718 to create user interfaces to interact with users of
25 the system. Alternatively, or additionally, in some systems interactions with a user may occur through a presentation layer, such as presentation layer 1714. In these systems, the application/component ‘logic’ can be separated from the aspects of the application/component that interact with a user.

[00167] FIG. 18 is a block diagram illustrating components of a machine 1800,
30 according to some example embodiments, able to read instructions from a machine-readable medium (e.g., a machine-readable storage medium) and perform any one or more of the methodologies discussed herein. Specifically, FIG. 18 shows a diagrammatic representation of the machine 1800 in the

example form of a computer system, within which instructions 1810 (e.g., software, a program, an application, an applet, an app, or other executable code) for causing the machine 1800 to perform any one or more of the methodologies discussed herein may be executed. As such, the instructions 1810 may be used to implement modules or components described herein. The instructions 1810 transform the general, non-programmed machine 1800 into a particular machine 1800 programmed to carry out the described and illustrated functions in the manner described. In alternative embodiments, the machine 1800 operates as a standalone device or may be coupled (e.g., networked) to other machines. In a networked deployment, the machine 1800 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 1800 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 smart phone, a mobile device, a wearable device (e.g., a smart watch), 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 1810, sequentially or otherwise, that specify actions to be taken by machine 1800. Further, while only a single machine 1800 is illustrated, the term ‘machine’ shall also be taken to include a collection of machines that individually or jointly execute the instructions 1810 to perform any one or more of the methodologies discussed herein.

[00168] The machine 1800 may include processors 1804, including processor 1808 to processor 1812, memory/storage 1806, and I/O components 1818, which may be configured to communicate with each other such as via a bus 1802. The memory/storage 1806 may include a memory 1814, such as a main memory, or other memory storage, and a storage unit 1816, both accessible to the processors 1804 such as via the bus 1802. The storage unit 1816 and memory 1814 store the instructions 1810 embodying any one or more of the methodologies or functions described herein. The instructions 1810 may also reside, completely or partially, within the memory 1814, within the storage unit 1816, within at least one of the processors 1804 (e.g., within the processor’s cache memory), or

any suitable combination thereof, during execution thereof by the machine 1800. Accordingly, the memory 1814, the storage unit 1816, and the memory of processors 1804 are examples of machine-readable media.

[00169] The I/O components 1818 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 1818 that are included in a particular machine 1800 will depend on the type of machine. For example, portable machines such as mobile phones will likely 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 1818 may include many other components that are not shown in FIG. 18. The I/O components 1818 are grouped according to functionality merely for simplifying the following discussion and the grouping is in no way limiting. In various example embodiments, the I/O components 1818 may include output components 1826 and input components 1828. The output components 1826 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 input components 1828 may include alphanumeric input components (e.g., a keyboard, a touch screen configured to receive alphanumeric input, a photo-optical keyboard, or other alphanumeric input components), point based input components (e.g., a mouse, a touchpad, a trackball, a joystick, a motion sensor, or other pointing instrument), tactile input components (e.g., a physical button, a touch screen that provides location and/or force of touches or touch gestures, or other tactile input components), audio input components (e.g., a microphone), and the like.

[00170] In further example embodiments, the I/O components 1818 may include biometric components 1830, motion components 1834, environmental components 1836, or position components 1838 among a wide array of other components. For example, the biometric components 1830 may 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 1834 may include
5 acceleration sensor components (e.g., accelerometer), gravitation sensor components, rotation sensor components (e.g., gyroscope), and so forth. The environmental components 1836 may include, for example, illumination sensor components (e.g., photometer), temperature sensor components (e.g., one or more thermometer that detect ambient temperature), humidity sensor
10 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
15 components that may provide indications, measurements, or signals corresponding to a surrounding physical environment. The position components 1838 may 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.,
20 magnetometers), and the like.

[00171] Communication may be implemented using a wide variety of technologies. The I/O components 1818 may include communication components 1840 operable to couple the machine 1800 to a network 1832 or devices 1820 via coupling 1824 and coupling 1822, respectively. For example,
25 the communication components 1840 may include a network interface component or other suitable device to interface with the network 1832. In further examples, communication components 1840 may include wired communication components, wireless communication components, cellular communication components, Near Field Communication (NFC) components,
30 Bluetooth® components (e.g., Bluetooth® Low Energy), Wi-Fi® components, and other communication components to provide communication via other modalities. The devices 1820 may be another machine or any of a wide variety of peripheral devices (e.g., a peripheral device coupled via a USB).

[00172] Moreover, the communication components 1840 may detect identifiers or include components operable to detect identifiers. For example, the communication components 1840 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 1840, such as, location via Internet Protocol (IP) geo-location, location via Wi-Fi® signal triangulation, location via detecting a NFC beacon signal that may indicate a particular location, and so forth.

[00173] The following discussion relates to various terms or phrases that are mentioned throughout the subject disclosure.

[00174] ‘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.

[00175] ‘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
5 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
10 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.

[00176] ‘Processor’ refers to any circuit or virtual circuit (a physical circuit emulated by logic executing on an actual processor) that manipulates data values
15 according to control signals (e.g., ‘commands’, ‘op codes’, ‘machine code’, etc.) and which produces corresponding output signals that are applied to operate a machine. A processor may, for example, be 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
20 Signal Processor (DSP), an Application Specific Integrated Circuit (ASIC), a Radio-Frequency Integrated Circuit (RFIC) or any combination thereof. A processor may further be a multi-core processor having two or more independent processors (sometimes referred to as ‘cores’) that may execute instructions contemporaneously.

25 **[00177]** ‘Machine-Storage Medium’ refers to a single or multiple storage devices and/or media (e.g., a centralized or distributed database, and/or associated caches and servers) that store executable instructions, routines and/or 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
30 external to processors. Specific examples of machine-storage media, computer-storage media and/or 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
5 exclude carrier waves, modulated data signals, and other such media, at least some of which are covered under the term ‘signal medium.’

[00178] ‘Component’ refers to a device, physical entity, or logic having boundaries defined by function or subroutine calls, branch points, APIs, or other
10 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
15 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 example embodiments, one or more computer systems (e.g., a standalone
20 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
25 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
30 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),
5 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
10 operations described herein. Considering embodiments 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
15 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
20 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
25 transmission (e.g., over appropriate circuits and buses) between or among two or more of the hardware components. In embodiments 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
30 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.

5 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
10 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
15 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
20 processors, not only residing within a single machine, but deployed across a number of machines. In some example embodiments, 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 example embodiments, the processors or processor-implemented
25 components may be distributed across a number of geographic locations.

[00179] ‘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
30 or received over a network using a transmission medium via a network interface device.

[00180] ‘Computer-Readable 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.

[00181] ‘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. In the subject disclosure, a client device is also referred to as an ‘electronic device.’

[00182] ‘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.

[00183] ‘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.

[00184] ‘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.

[00185] ‘Processor’ refers to any circuit or virtual circuit (a physical circuit emulated by logic executing on an actual processor) that manipulates data values according to control signals (e.g., ‘commands’, ‘op codes’, ‘machine code’, etc.) and which produces corresponding output signals that are applied to operate a machine. A processor may, for example, be 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) or any combination thereof. A processor may further be a multi-core processor having two or more independent processors (sometimes referred to as ‘cores’) that may execute instructions contemporaneously.

[00186] ‘Machine-Storage Medium’ refers to a single or multiple storage devices and/or media (e.g., a centralized or distributed database, and/or associated caches and servers) that store executable instructions, routines and/or 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/or 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.’

5

10 **[00187]** ‘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

15 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

20 operations and may be configured or arranged in a certain physical manner. In various example embodiments, 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

25 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

30 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

5 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

10 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 embodiments 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

15 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

20 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

25 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 embodiments in which multiple hardware components are configured or instantiated at different times, communications

30 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 example embodiments, 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 example embodiments, the processors or processor-implemented components may be distributed across a number of geographic locations.

[00188] ‘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.

[00189] ‘Computer-Readable 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
5 thing and may be used interchangeably in this disclosure.

[00190] ‘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),
10 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.

[00191] ‘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
15 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.

20

CLAIMS

What is claimed is:

1. A method, comprising:
 - 5 determining, by one or more hardware processors, a transaction recency and frequency distribution;
 - determining, by the one or more hardware processors, a future monetary value;
 - 10 determining, by the one or more hardware processors, monthly active users (MAU) and penetration for global users and users in a specific country;
 - predicting, by the one or more hardware processors, monetization values for the users in the specific country; and
 - determining, by the one or more hardware processors, a lifetime value of the users in the specific country based at least in part on the monetization values.
- 15 2. The method of claim 1, wherein determining the transaction recency and frequency distribution is based on at least a number of transactions made by a customer that follows a Poisson process with a transaction rate λ .
- 20 3. The method of claim 2, wherein heterogeneity in the transaction rate λ follows a gamma distribution.
4. The method of claim 3, wherein after a transaction, a user becomes inactive with a probability p .
- 25 5. The method of claim 4, wherein a point at which the user becomes inactive is distributed across transactions according to a geometric distribution.
6. The method of claim 2, wherein heterogeneity in the probability p
- 30 follows a beta distribution.
7. The method of claim 6, wherein the transaction rate λ and the probability p varies independently across users.

8. The method of claim 1, wherein determining the future monetary value is based on a customer's given transaction varying randomly around their average transaction value, average transaction values varying across users and not
5 varying over time for any user, and a distribution of average transaction values across customers is independent of a transaction process.

9. The method of claim 1, wherein determining the lifetime value comprises:
10 validating a model of the lifetime value based a period of time; and
determining a holdout period from the period of time to test model accuracy for user transaction frequency.

10. The method of claim 9, further comprising:
15 comparing a first set of values corresponding to actual purchases during the holdout period with a second set of values corresponding to purchases in a calibration period to determine a cumulative error rate.

11. A system comprising:
20 a processor; and
a memory including instructions that, when executed by the processor, cause the processor to perform operations comprising:
determining a transaction recency and frequency distribution;
determining a future monetary value;
25 determining monthly active users (MAU) and penetration for global users and users in a specific country;
predicting monetization values for the users in the specific country; and
determining a lifetime value of the users in the specific country based at least in part on the monetization values.

30
12. The system of claim 11, wherein determining the transaction recency and frequency distribution is based on at least a number of transactions made by a customer that follows a Poisson process with a transaction rate λ .

13. The system of claim 12, wherein heterogeneity in the transaction rate λ follows a gamma distribution.
14. The system of claim 13, wherein after a transaction, a user becomes
5 inactive with a probability p .
15. The system of claim 14, wherein a point at which the user becomes inactive is distributed across transactions according to a geometric distribution.
- 10 16. The system of claim 12, wherein heterogeneity in the probability p follows a beta distribution.
17. The system of claim 16, wherein the transaction rate λ and the probability p varies independently across users.
15
18. The system of claim 11, wherein determining the future monetary value is based on a customer's given transaction varying randomly around their average transaction value, average transaction values varying across users and not varying over time for any user, and a distribution of average transaction
20 values across customers is independent of a transaction process.
19. The system of claim 11, wherein determining the lifetime value comprises:
25 validating a model of the lifetime value based a period of time; and
 determining a holdout period from the period of time to test model accuracy for user transaction frequency; and
 comparing a first set of values corresponding to actual purchases during the holdout period with a second set of values corresponding to purchases in a calibration period to determine a cumulative error rate.
30

20. A non-transitory computer-readable medium comprising instructions, which when executed by a computing device, cause the computing device to perform operations comprising:
- determining a transaction recency and frequency distribution;
 - 5 determining a future monetary value;
 - determining monthly active users (MAU) and penetration for global users and users in a specific country;
 - predicting monetization values for the users in the specific country; and
 - determining a lifetime value of the users in the specific country based at
- 10 least in part on the monetization values.

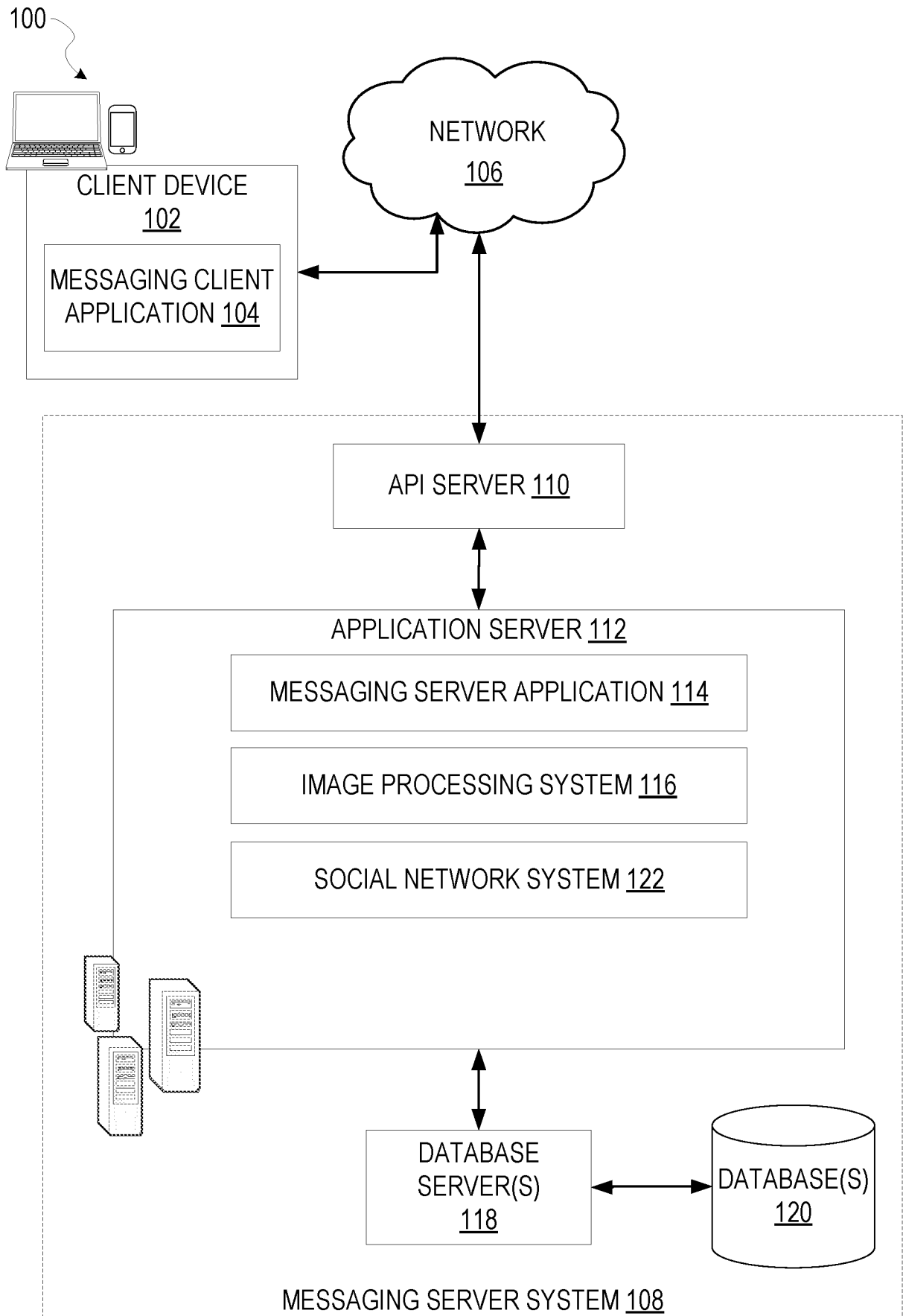


FIG. 1

100

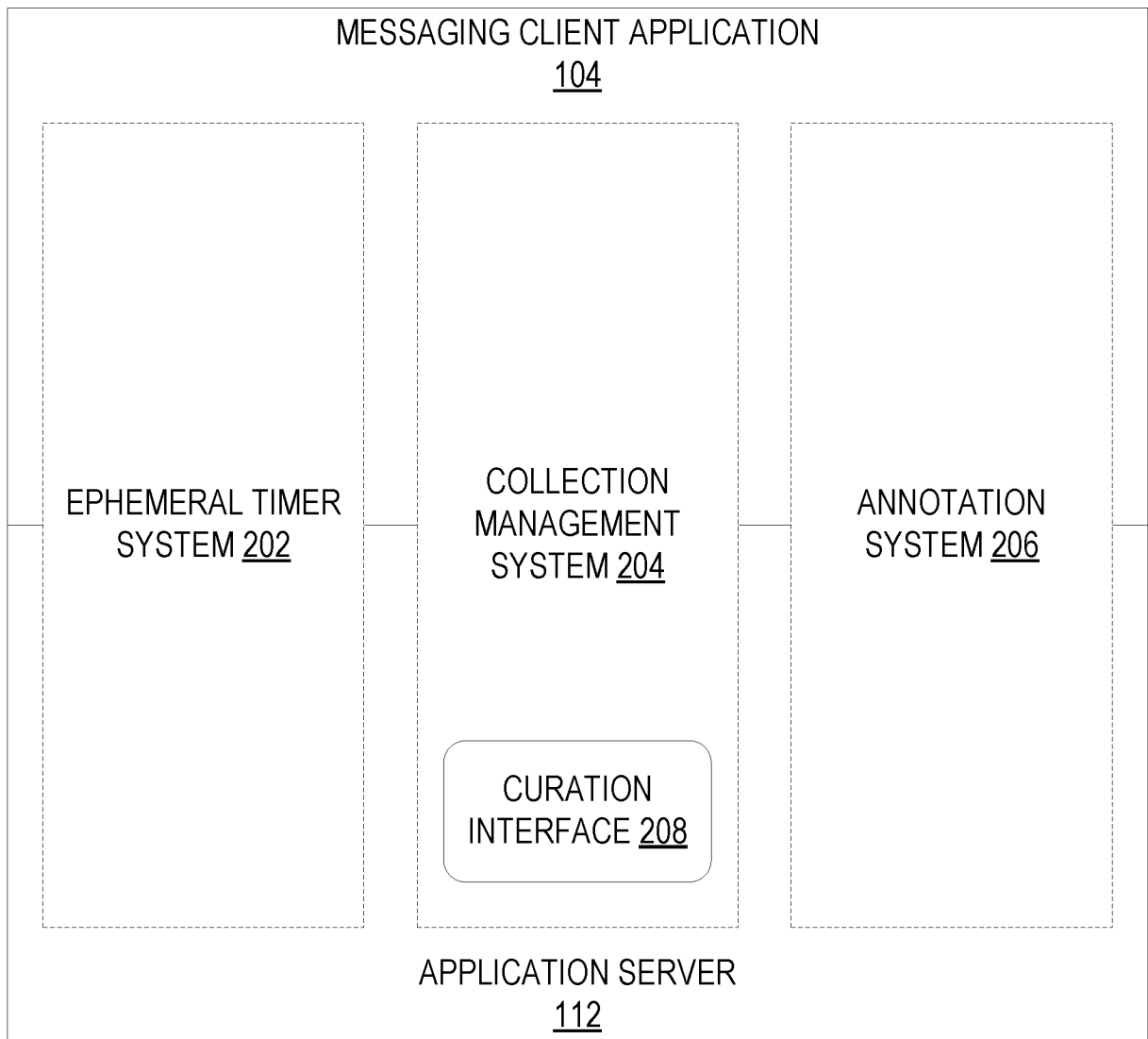



FIG. 2

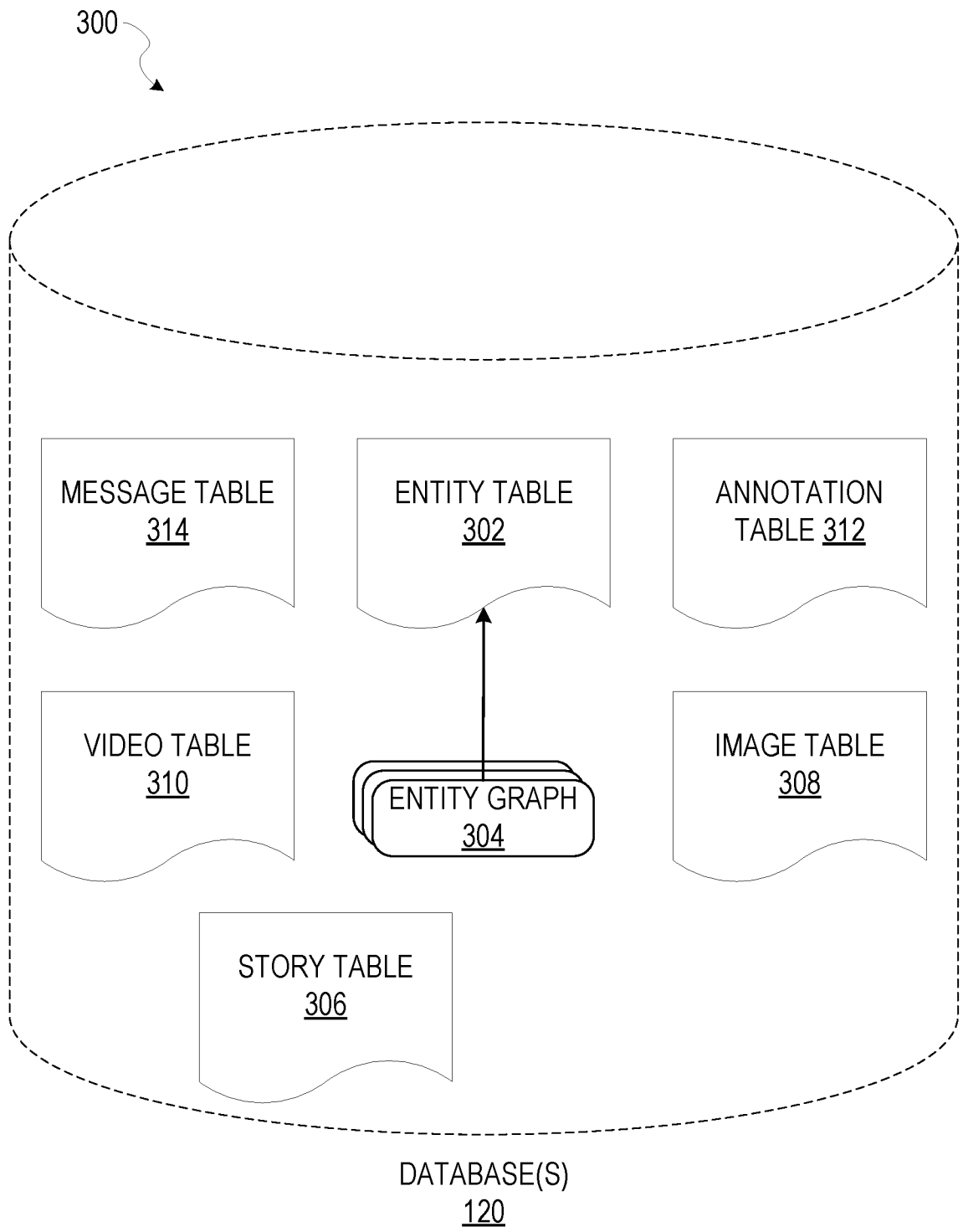


FIG. 3

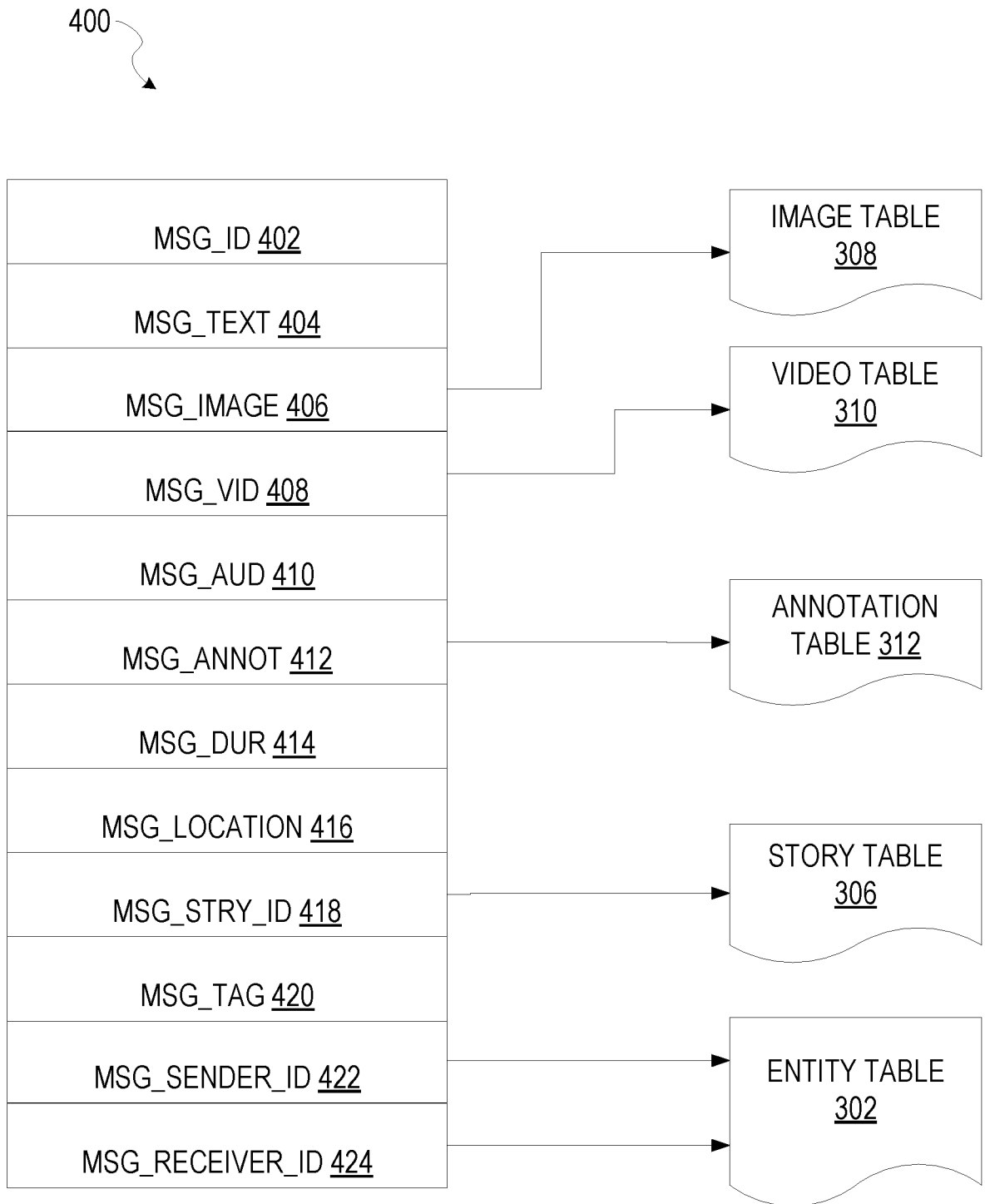


FIG. 4

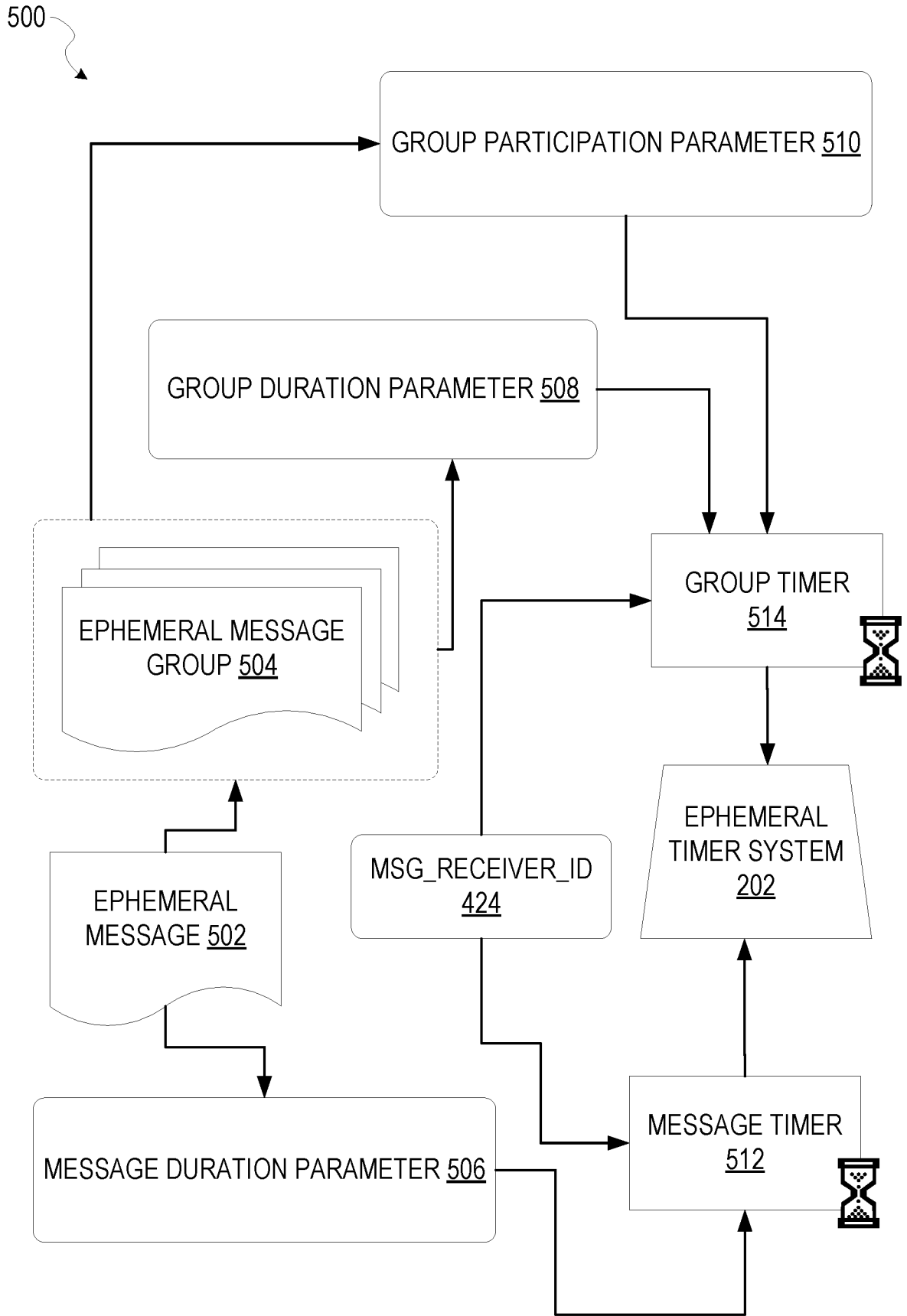


FIG. 5

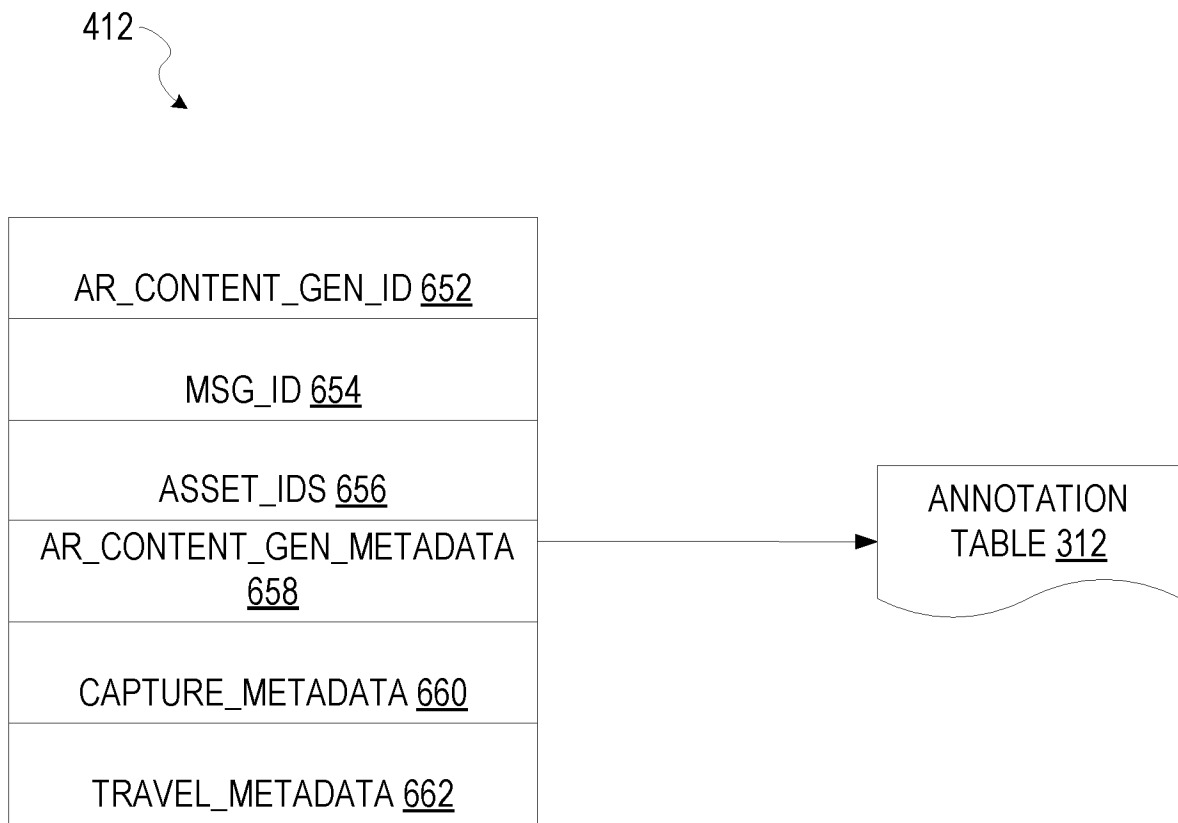


FIG. 6

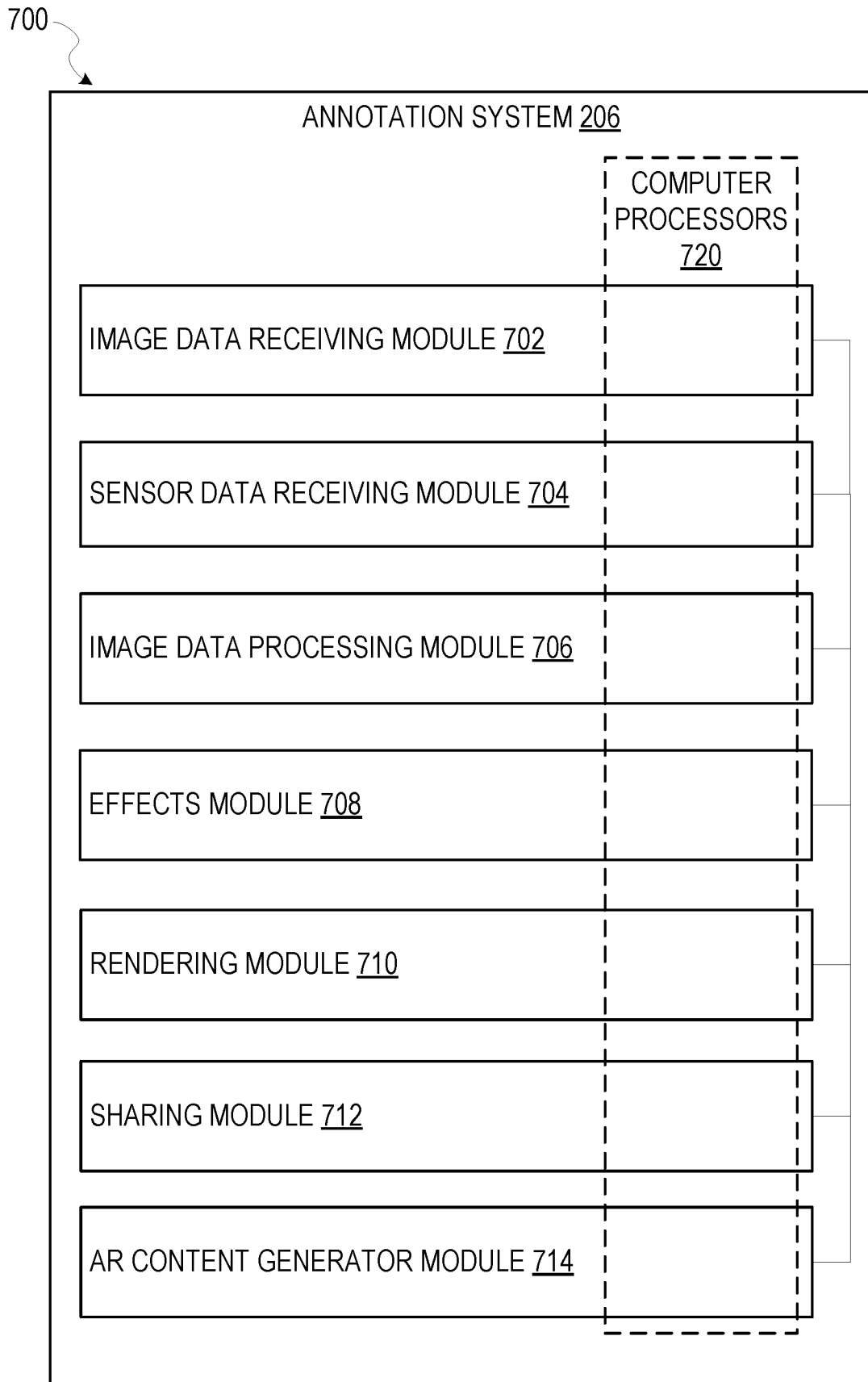


FIG. 7

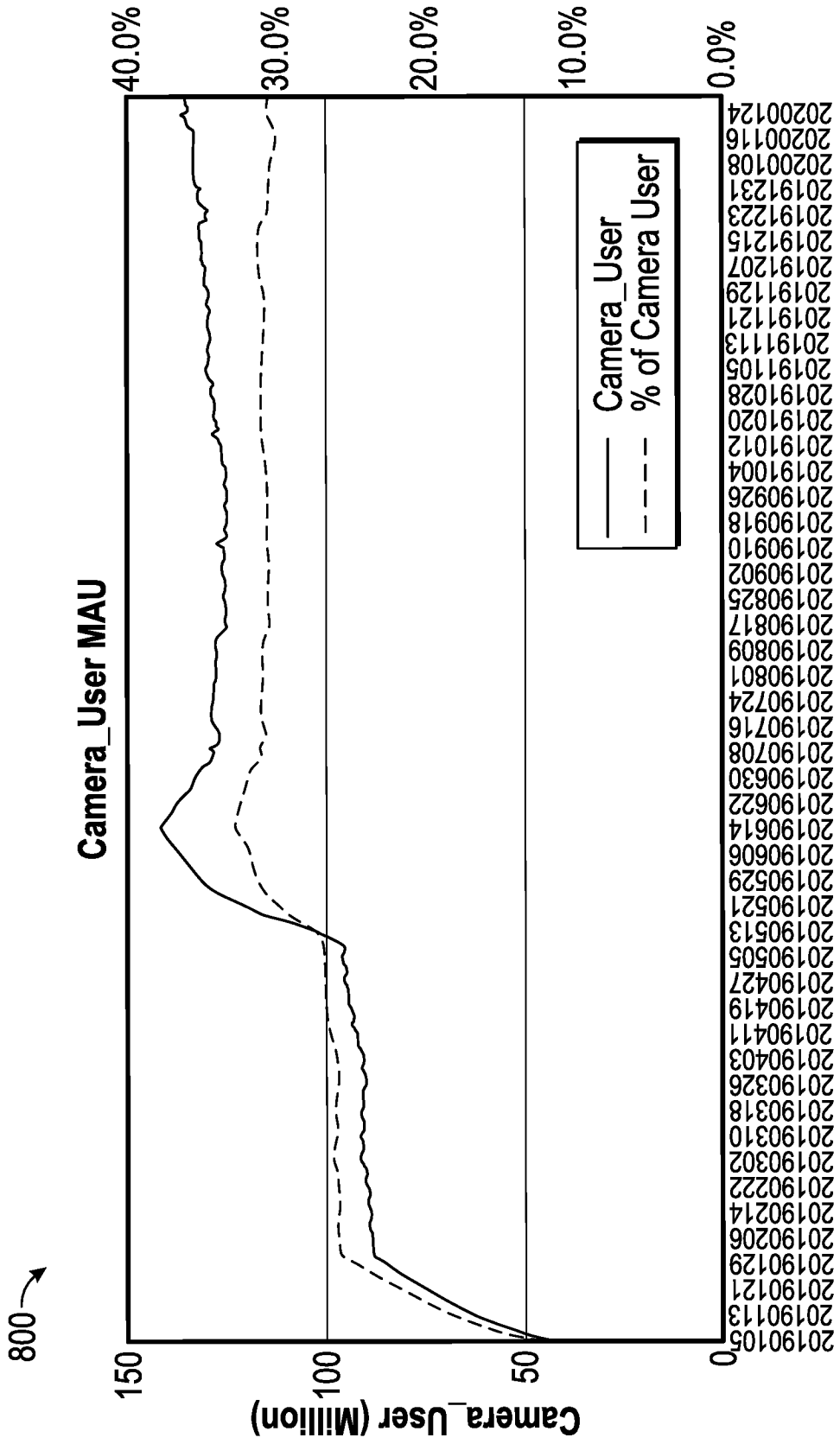


FIG. 8

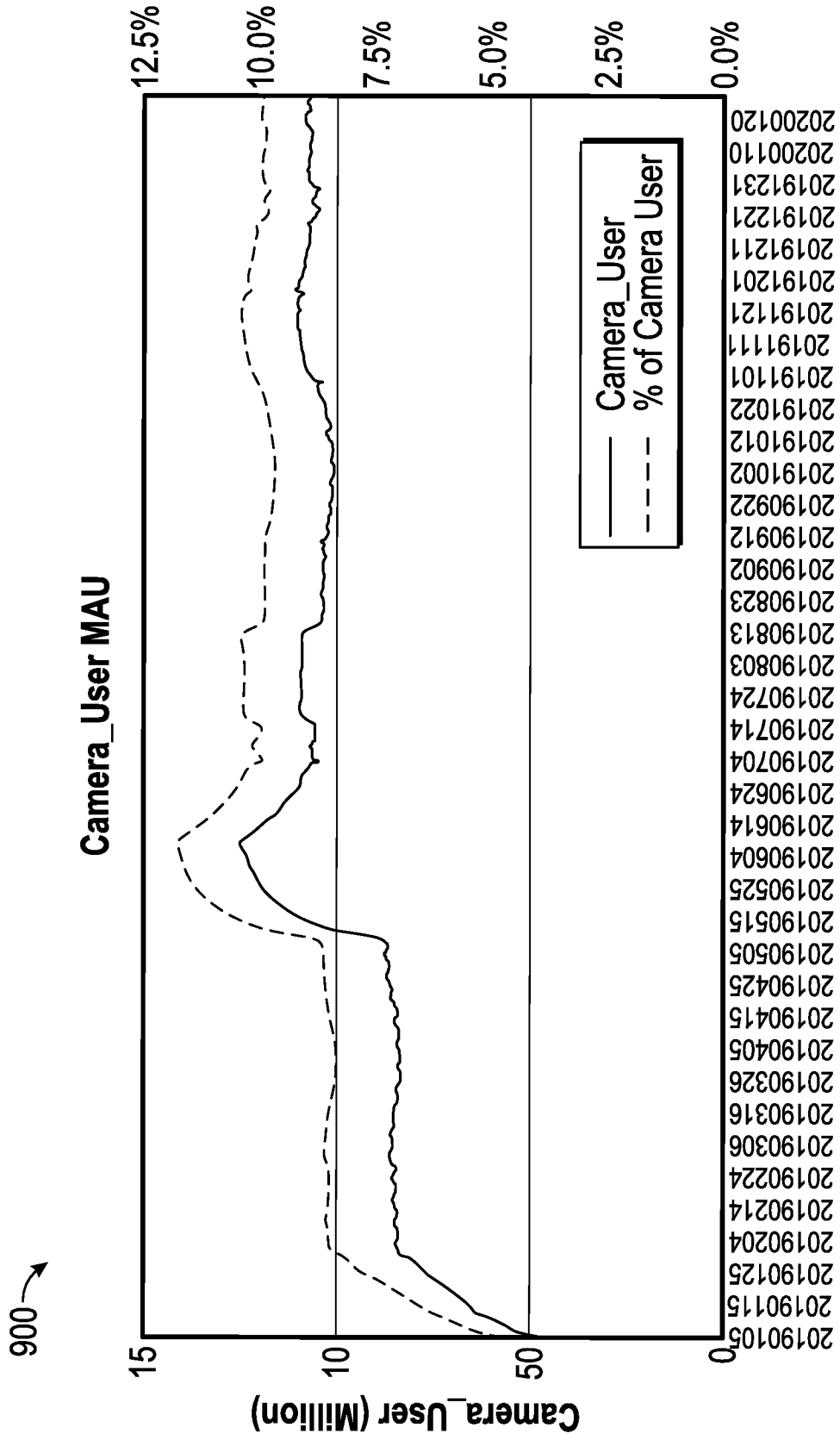


FIG. 9

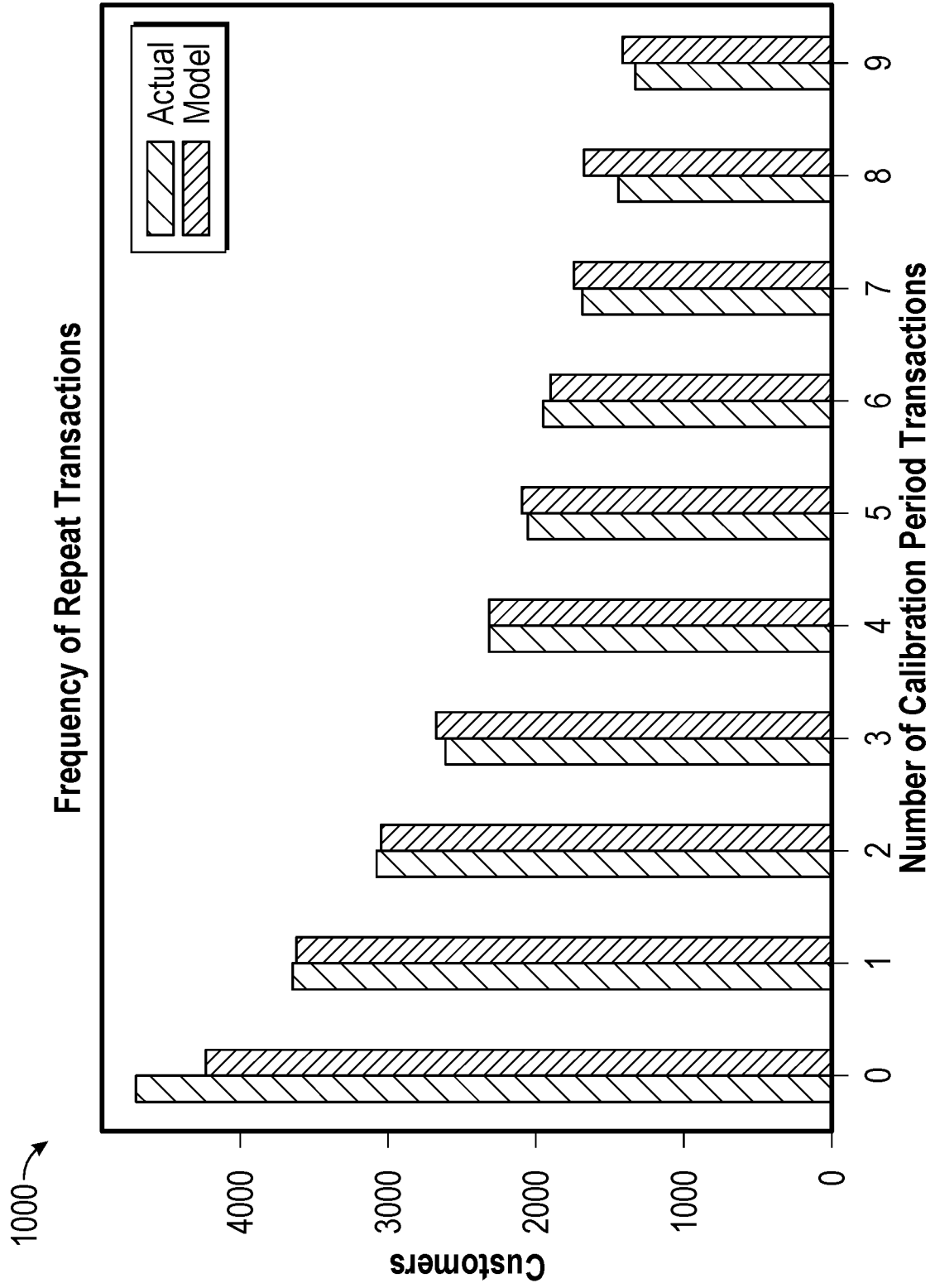


FIG. 10

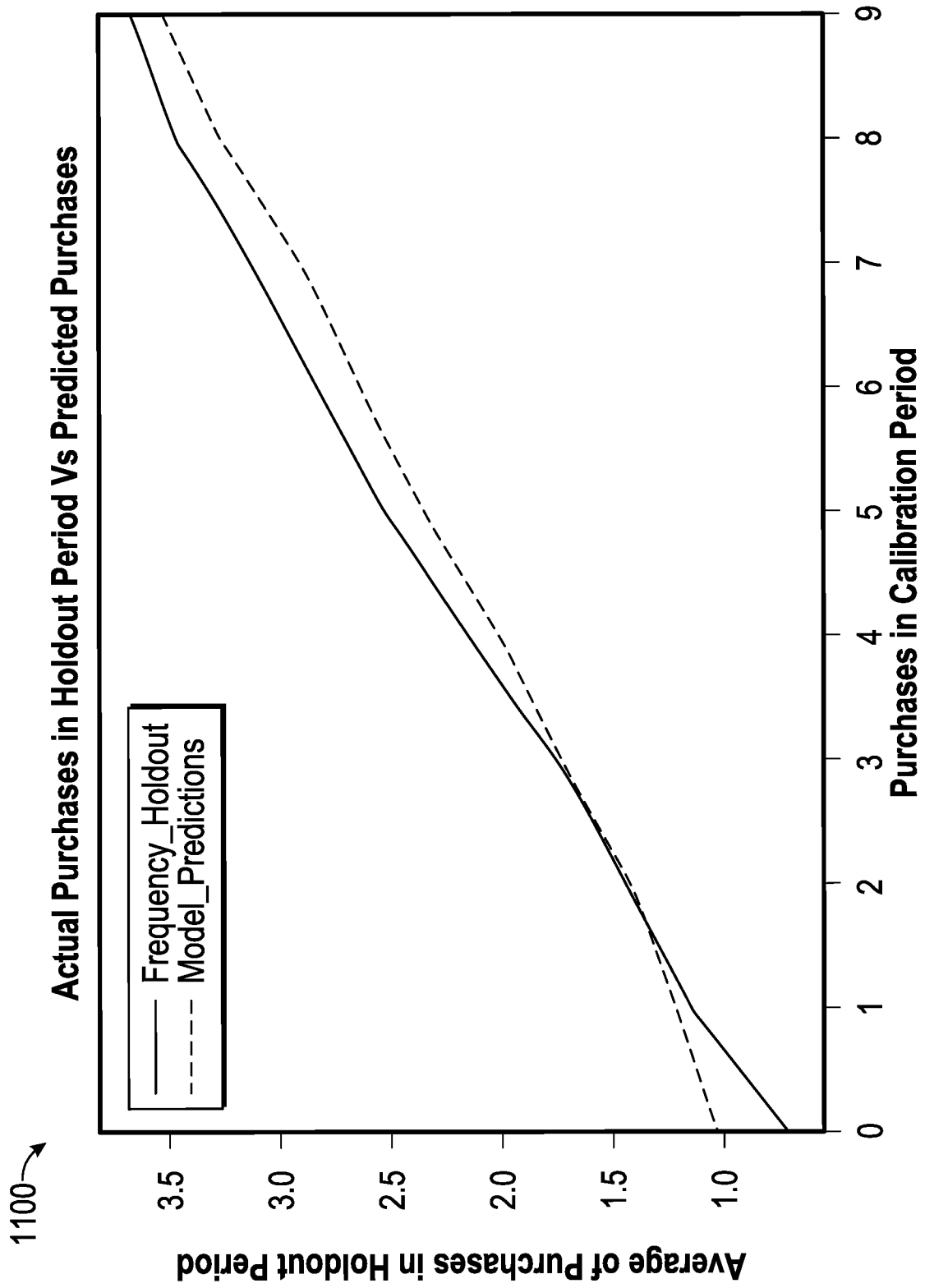


FIG. 11

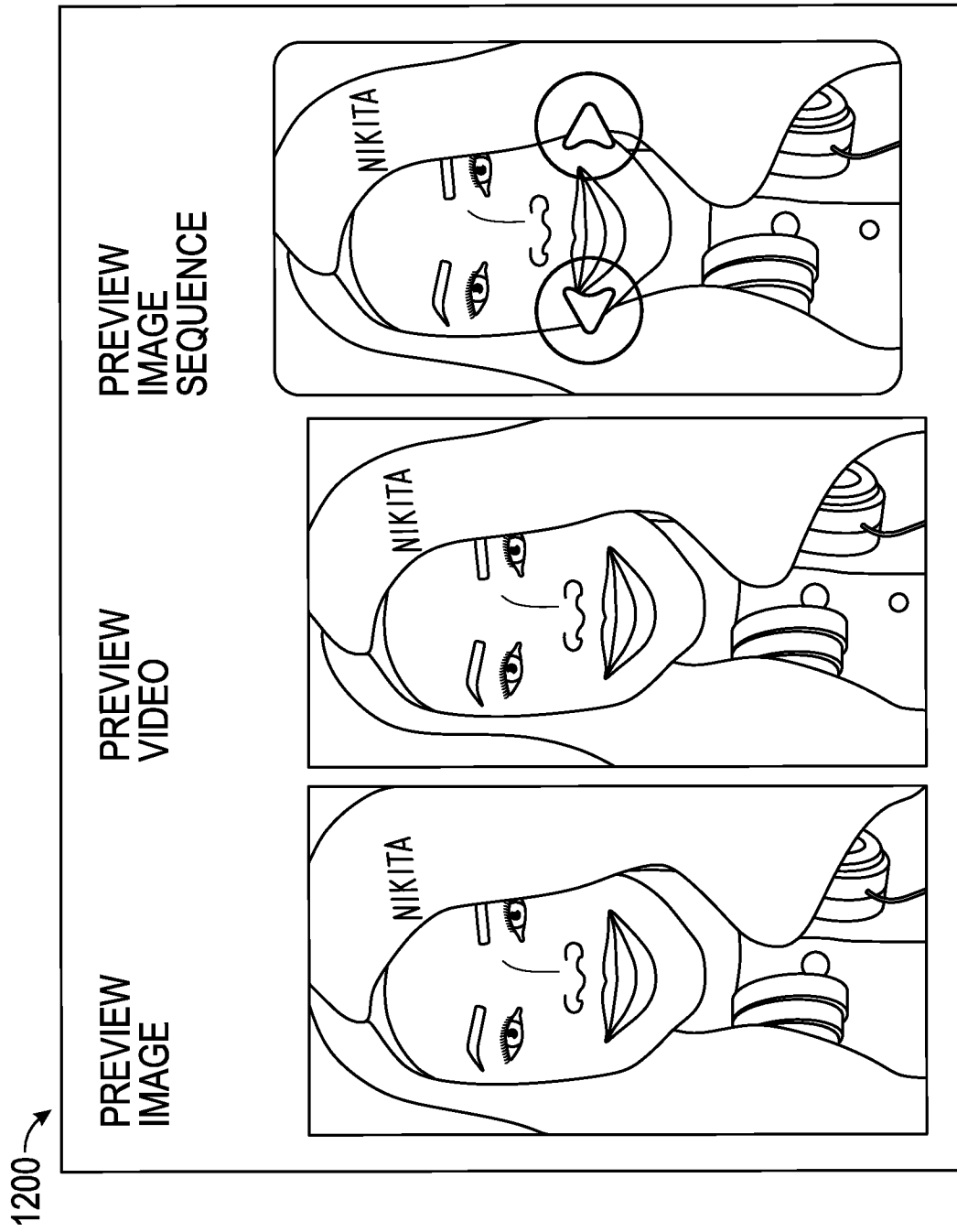


FIG. 12

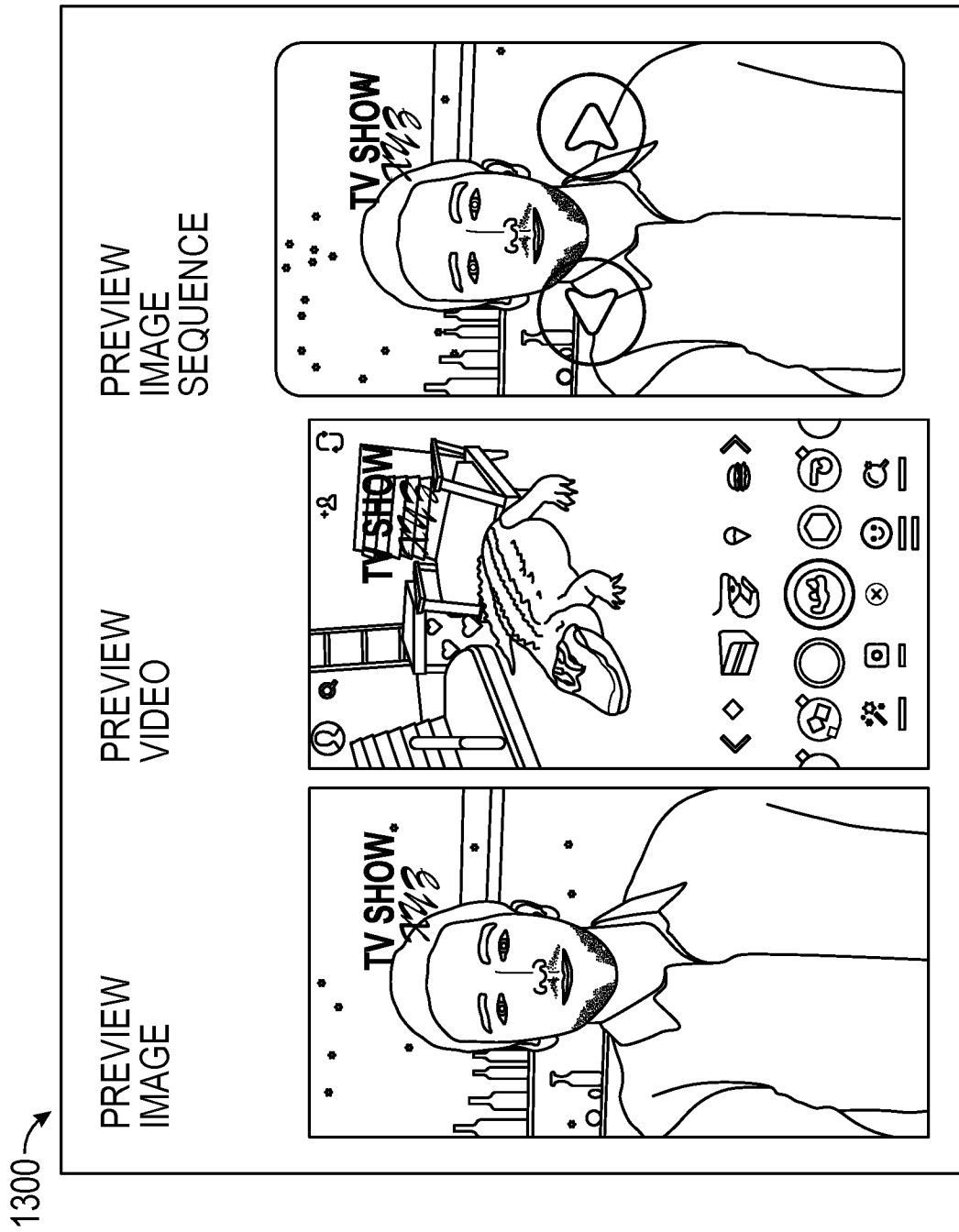


FIG. 13

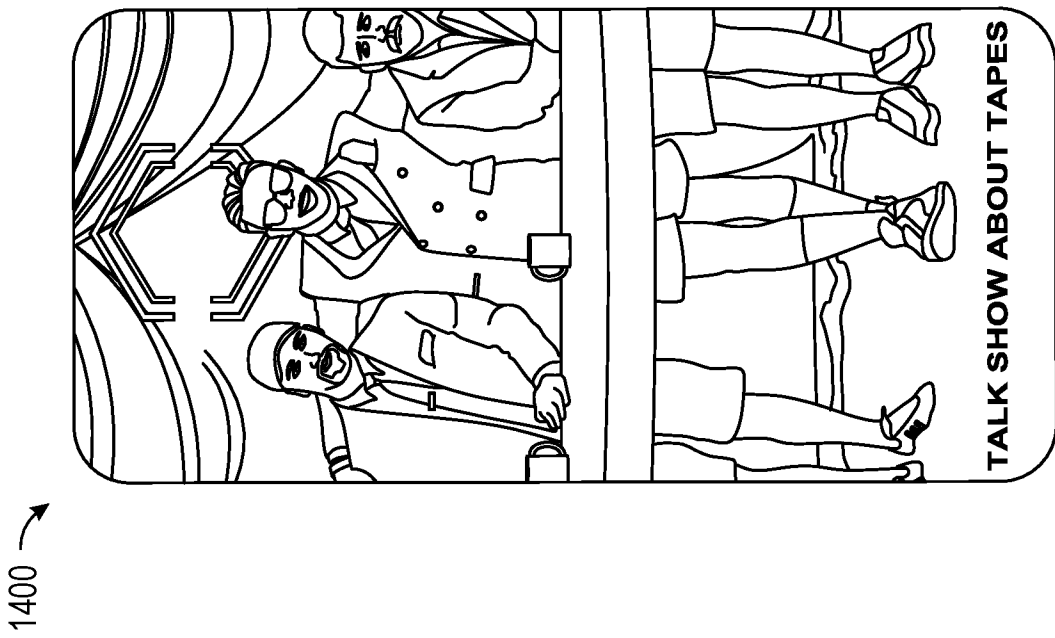
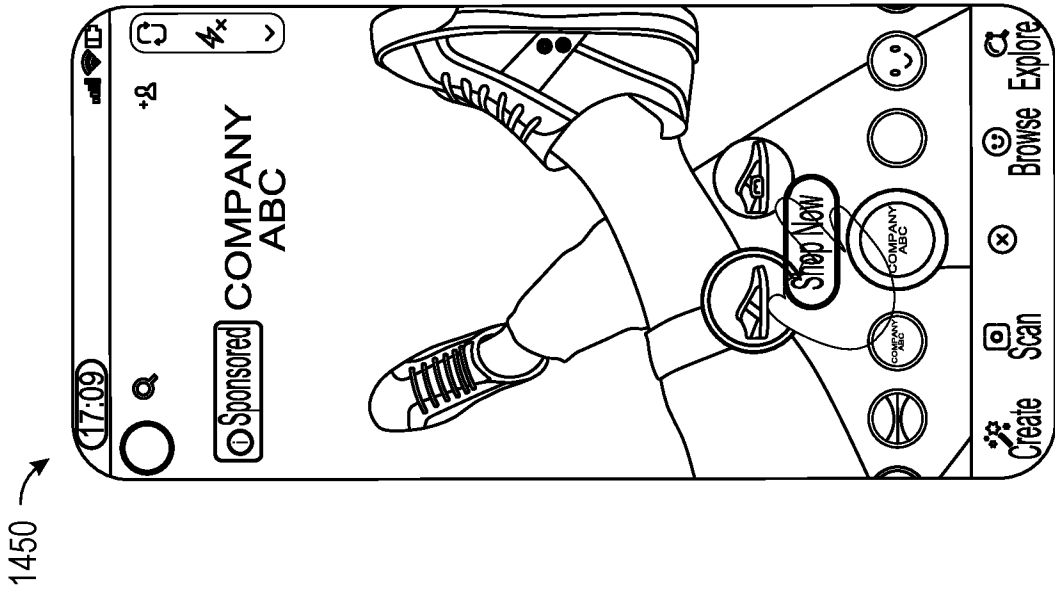


FIG. 14

1500

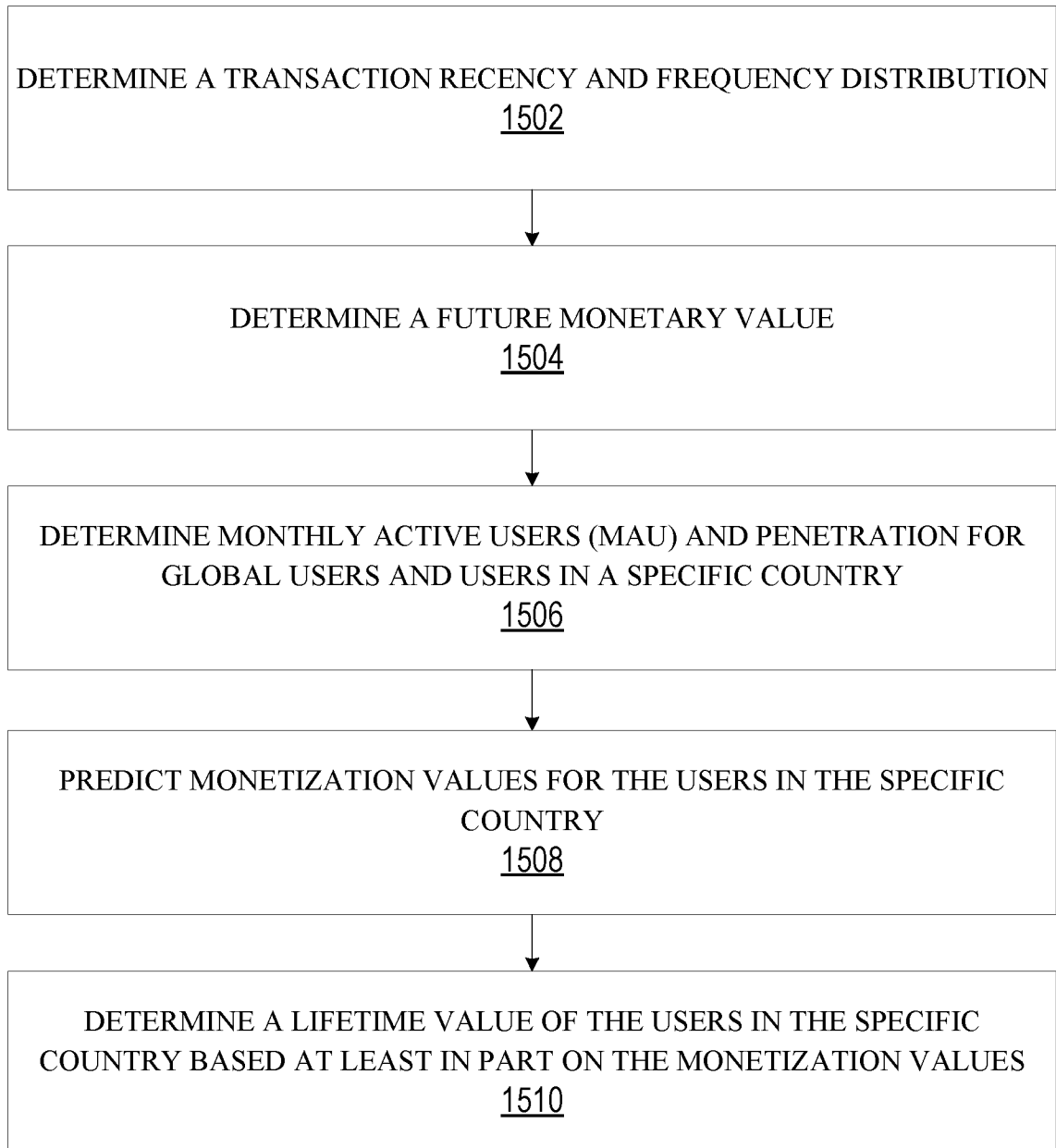


FIG. 15

1600

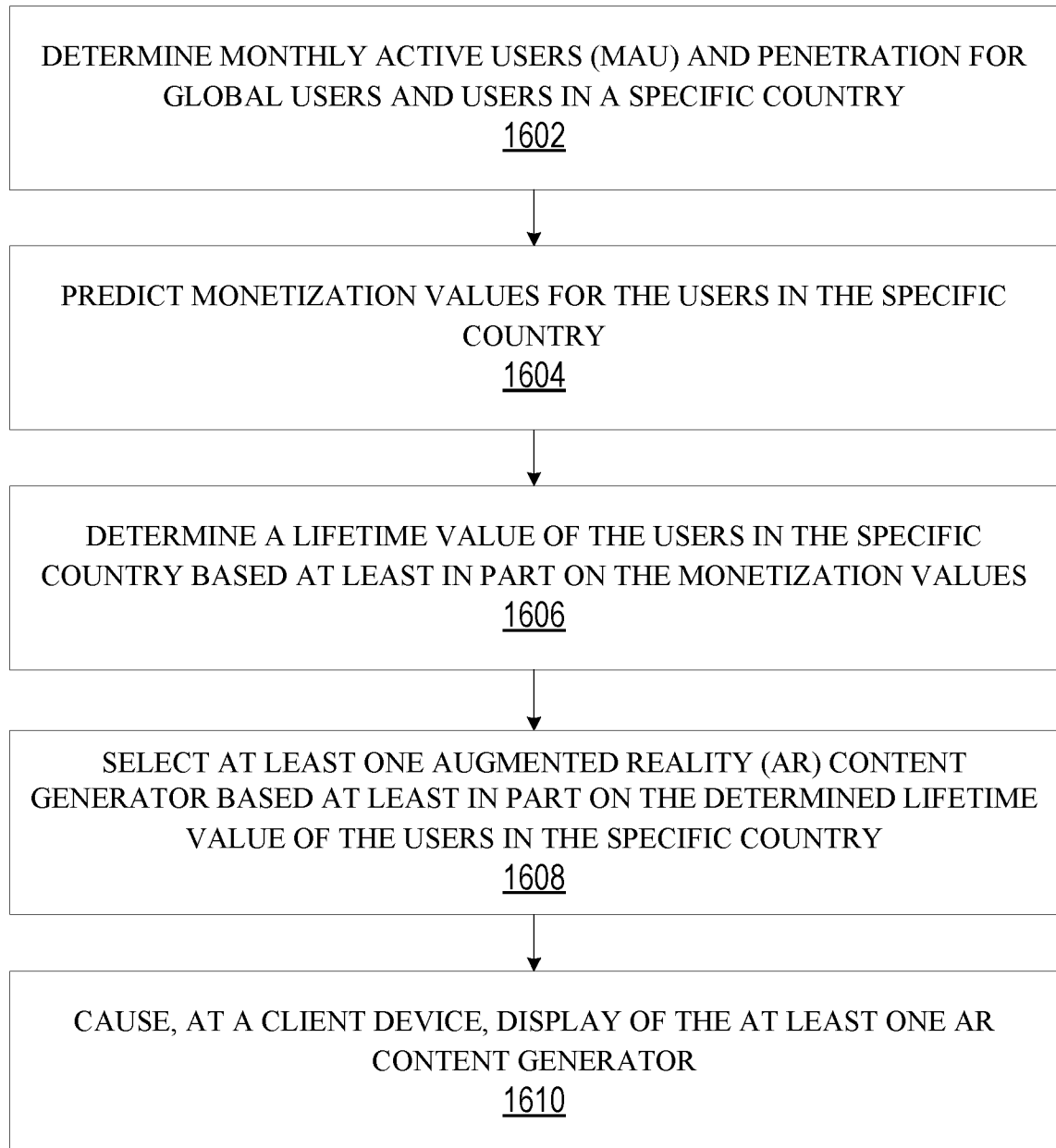


FIG. 16

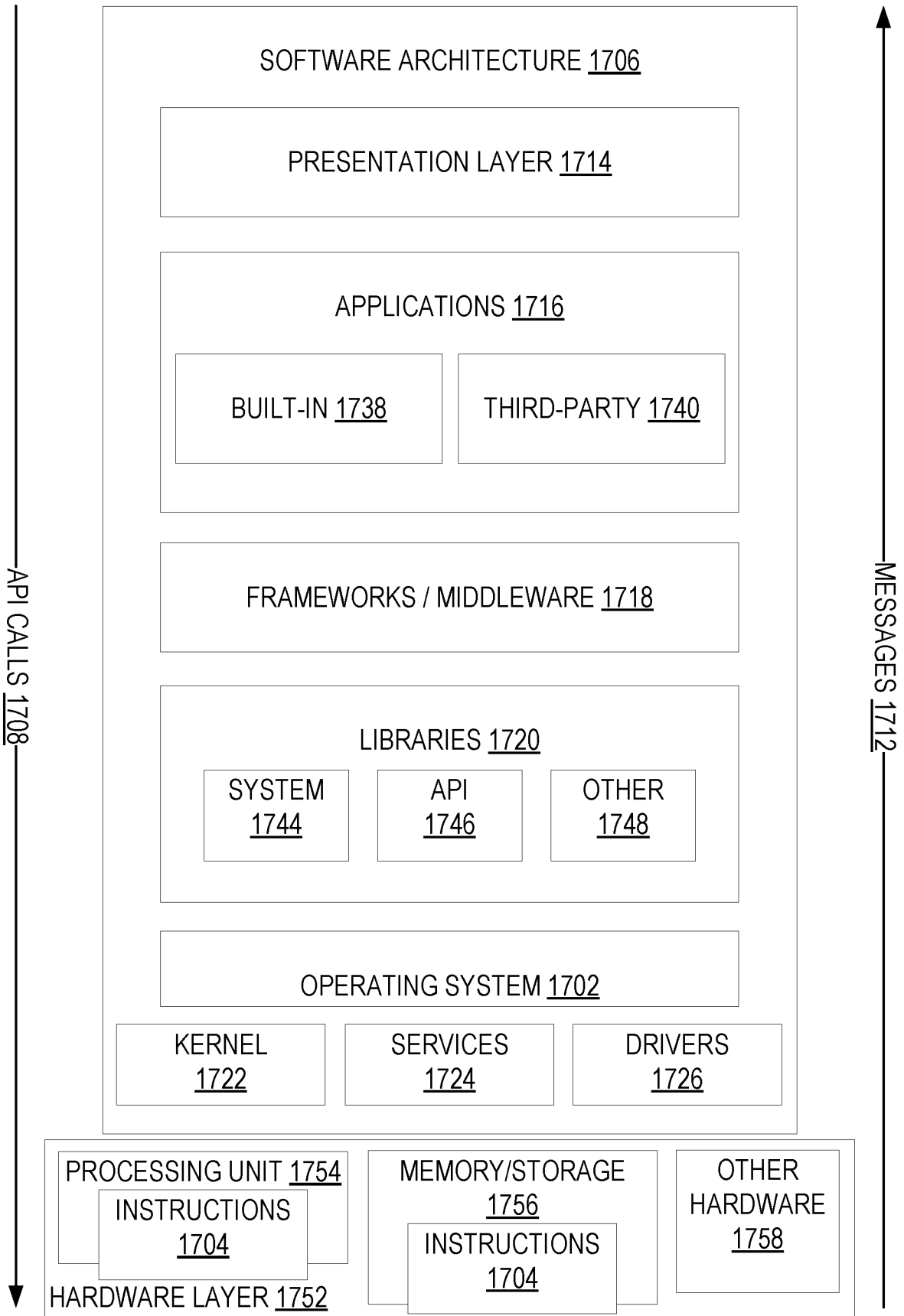


FIG. 17

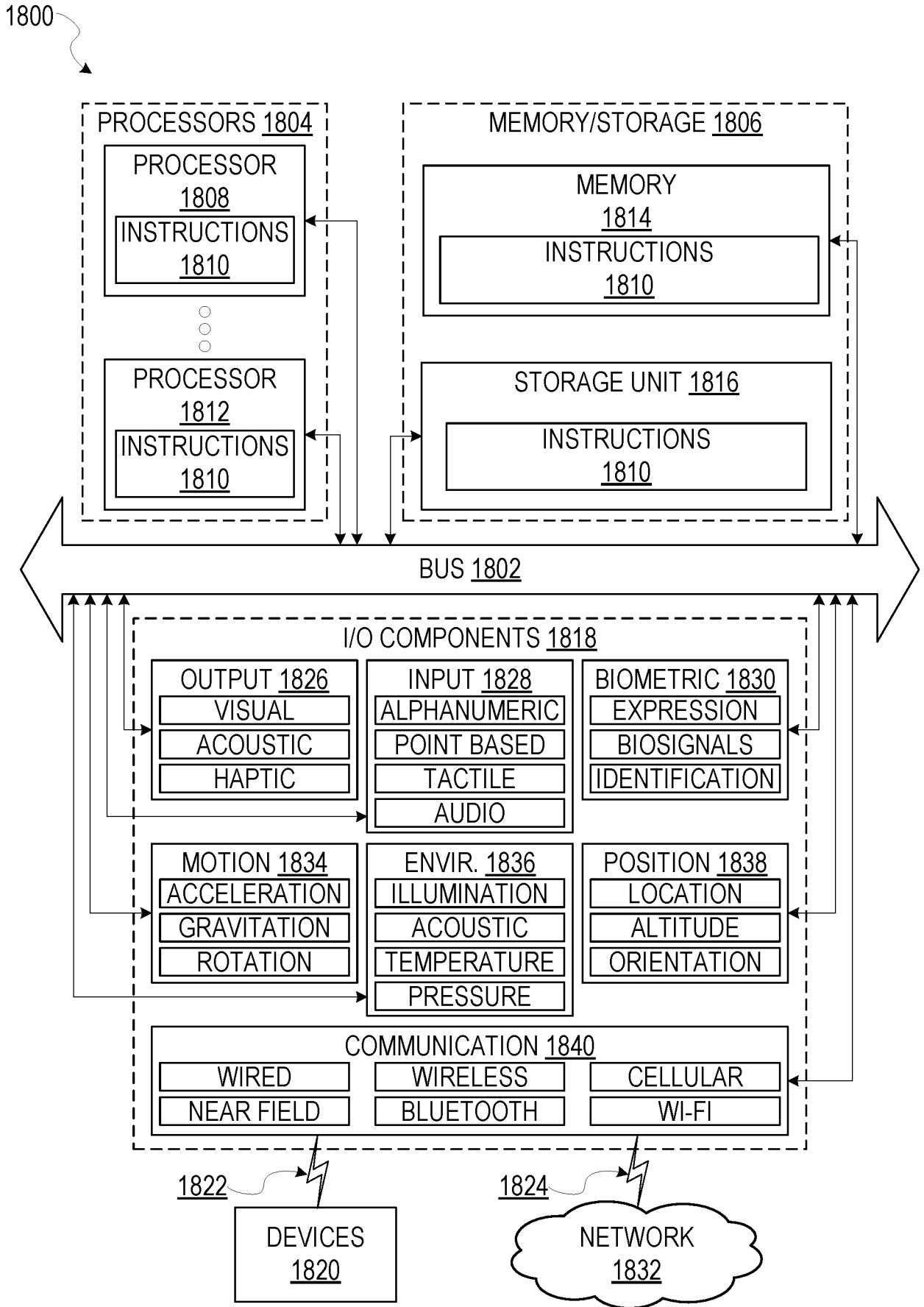


FIG. 18

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2021/052665

A. CLASSIFICATION OF SUBJECT MATTER G06Q 30/02(2012.01)i; G06Q 30/06(2012.01)i		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) G06Q 30/02(2012.01); A63F 13/61(2014.01); G06F 11/36(2006.01); G06F 17/27(2006.01); G06Q 20/40(2012.01); G06Q 30/00(2006.01)		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched Korean utility models and applications for utility models Japanese utility models and applications for utility models		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) eKOMPASS(KIPO internal) & Keywords: lifetime value, transaction, active user, monetization, compare		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2019-0180319 A1 (UNITY IPR APS) 13 June 2019 (2019-06-13) See paragraphs 20-25, claim 1 and figures 1-3.	1-20
Y	US 2014-0278967 A1 (HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.) 18 September 2014 (2014-09-18) See paragraphs 12-20, claim 1 and figures 2-3.	1-20
Y	US 2011-0258049 A1 (JOREY RAMER et al.) 20 October 2011 (2011-10-20) See paragraphs 110, 1168-1274, 1421, 1522 and figure 1.	1-20
Y	US 2015-0170147 A1 (CELLCO PARTNERSHIP (D/B/A VERIZON WIRELESS)) 18 June 2015 (2015-06-18) See claim 1 and figure 1.	10
A	US 2018-0143891 A1 (ACCENTURE GLOBAL SOLUTIONS LIMITED) 24 May 2018 (2018-05-24) See the entire document.	1-20
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> 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 "D" document cited by the applicant in the international application "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 21 January 2022		Date of mailing of the international search report 21 January 2022
Name and mailing address of the ISA/KR Korean Intellectual Property Office 189 Cheongsa-ro, Seo-gu, Daejeon 35208, Republic of Korea Facsimile No. +82-42-481-8578		Authorized officer PARK, Hye Lyun Telephone No. +82-42-481-3463

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/US2021/052665

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2019-0180319	A1	13 June 2019	WO	2019-115718	A1	20 June 2019
US	2014-0278967	A1	18 September 2014	None			
US	2011-0258049	A1	20 October 2011	EP	2531969	A2	12 December 2012
				JP	2013-519162	A	23 May 2013
				KR	10-2013-0009754	A	23 January 2013
				US	10038756	B2	31 July 2018
				US	10592930	B2	17 March 2020
				US	2007-0060099	A1	15 March 2007
				US	2007-0060109	A1	15 March 2007
				US	2007-0060114	A1	15 March 2007
				US	2007-0060129	A1	15 March 2007
				US	2007-0060136	A1	15 March 2007
				US	2007-0060173	A1	15 March 2007
				US	2007-0061197	A1	15 March 2007
				US	2007-0061198	A1	15 March 2007
				US	2007-0061211	A1	15 March 2007
				US	2007-0061229	A1	15 March 2007
				US	2007-0061242	A1	15 March 2007
				US	2007-0061243	A1	15 March 2007
				US	2007-0061244	A1	15 March 2007
				US	2007-0061245	A1	15 March 2007
				US	2007-0061247	A1	15 March 2007
				US	2007-0061300	A1	15 March 2007
				US	2007-0061301	A1	15 March 2007
				US	2007-0061302	A1	15 March 2007
				US	2007-0061317	A1	15 March 2007
				US	2007-0061328	A1	15 March 2007
				US	2007-0061331	A1	15 March 2007
				US	2007-0061332	A1	15 March 2007
				US	2007-0061333	A1	15 March 2007
				US	2007-0061334	A1	15 March 2007
				US	2007-0061335	A1	15 March 2007
				US	2007-0061336	A1	15 March 2007
				US	2007-0061363	A1	15 March 2007
				US	2007-0073722	A1	29 March 2007
				US	2007-0073723	A1	29 March 2007
				US	2007-0094042	A1	26 April 2007
				US	2007-0100650	A1	03 May 2007
				US	2007-0100651	A1	03 May 2007
				US	2007-0100652	A1	03 May 2007
				US	2007-0100653	A1	03 May 2007
				US	2007-0100805	A1	03 May 2007
				US	2007-0100806	A1	03 May 2007
				US	2007-0118533	A1	24 May 2007
				US	2007-0168354	A1	19 July 2007
				US	2007-0192318	A1	16 August 2007
				US	2007-0239724	A1	11 October 2007
				US	2007-0260635	A1	08 November 2007
				US	2007-0288427	A1	13 December 2007
				US	2008-0009268	A1	10 January 2008

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/US2021/052665

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
		US 2008-0214148 A1	04 September 2008
		US 2008-0214149 A1	04 September 2008
		US 2008-0214150 A1	04 September 2008
		US 2008-0214151 A1	04 September 2008
		US 2008-0214152 A1	04 September 2008
		US 2008-0214153 A1	04 September 2008
		US 2008-0214154 A1	04 September 2008
		US 2008-0214155 A1	04 September 2008
		US 2008-0214156 A1	04 September 2008
		US 2008-0214157 A1	04 September 2008
		US 2008-0214162 A1	04 September 2008
		US 2008-0214166 A1	04 September 2008
		US 2008-0214204 A1	04 September 2008
		US 2008-0215428 A1	04 September 2008
		US 2008-0215429 A1	04 September 2008
		US 2008-0215475 A1	04 September 2008
		US 2008-0215557 A1	04 September 2008
		US 2008-0215623 A1	04 September 2008
		US 2008-0242279 A1	02 October 2008
		US 2008-0270220 A1	30 October 2008
		US 2009-0029687 A1	29 January 2009
		US 2009-0222329 A1	03 September 2009
		US 2009-0234711 A1	17 September 2009
		US 2009-0234745 A1	17 September 2009
		US 2009-0234861 A1	17 September 2009
		US 2009-0240568 A1	24 September 2009
		US 2009-0240569 A1	24 September 2009
		US 2009-0240586 A1	24 September 2009
		US 2010-0063877 A1	11 March 2010
		US 2010-0076845 A1	25 March 2010
		US 2010-0076994 A1	25 March 2010
		US 2010-0082431 A1	01 April 2010
		US 2010-0094878 A1	15 April 2010
		US 2011-0143731 A1	16 June 2011
		US 2011-0143733 A1	16 June 2011
		US 2011-0177799 A1	21 July 2011
		US 2012-0173365 A1	05 July 2012
		US 2012-0173369 A1	05 July 2012
		US 2012-0209705 A1	16 August 2012
		US 2012-0209706 A1	16 August 2012
		US 2012-0209707 A1	16 August 2012
		US 2012-0209708 A1	16 August 2012
		US 2012-0209709 A1	16 August 2012
		US 2012-0209710 A1	16 August 2012
		US 2012-0215602 A1	23 August 2012
		US 2012-0215612 A1	23 August 2012
		US 2012-0215622 A1	23 August 2012
		US 2012-0215623 A1	23 August 2012
		US 2012-0215624 A1	23 August 2012
		US 2012-0215625 A1	23 August 2012

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/US2021/052665

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
		US 2012-0215626 A1	23 August 2012
		US 2012-0215635 A1	23 August 2012
		US 2012-0215639 A1	23 August 2012
		US 2012-0215640 A1	23 August 2012
		US 2012-0330750 A1	27 December 2012
		US 2013-0053005 A1	28 February 2013
		US 7548915 B2	16 June 2009
		US 7577665 B2	18 August 2009
		US 7603360 B2	13 October 2009
		US 7660581 B2	09 February 2010
		US 7676394 B2	09 March 2010
		US 7702318 B2	20 April 2010
		US 7752209 B2	06 July 2010
		US 7769764 B2	03 August 2010
		US 7860871 B2	28 December 2010
		US 7912458 B2	22 March 2011
		US 8027879 B2	27 September 2011
		US 8103545 B2	24 January 2012
		US 8131271 B2	06 March 2012
		US 8195133 B2	05 June 2012
		US 8229914 B2	24 July 2012
		US 8290810 B2	16 October 2012
		US 8302030 B2	30 October 2012
		US 8311888 B2	13 November 2012
		US 8364521 B2	29 January 2013
		US 8364540 B2	29 January 2013
		US 8615719 B2	24 December 2013
		US 8660891 B2	25 February 2014
		US 8666376 B2	04 March 2014
		US 8688671 B2	01 April 2014
		US 8832100 B2	09 September 2014
		US 8989718 B2	24 March 2015
		US 9201979 B2	01 December 2015
		US 9471925 B2	18 October 2016
		WO 2007-033358 A2	22 March 2007
		WO 2007-033358 A3	22 November 2007
		WO 2011-094734 A2	04 August 2011
		WO 2011-094734 A3	19 January 2012
US 2015-0170147 A1	18 June 2015	US 9508075 B2	29 November 2016
US 2018-0143891 A1	24 May 2018	AU 2017-258970 A1	07 June 2018
		AU 2019-201142 A1	07 March 2019
		US 10353799 B2	16 July 2019