

US Patent & Trademark Office

Patent Public Search | Text View

United States Patent Application Publication

20250252685

Kind Code

A1

Publication Date

August 07, 2025

Inventor(s)

Cowburn; Piers George et al.

AUGMENTED REALITY ANAMORPHOSIS SYSTEM

Abstract

Systems, methods, devices, and media for anamorphosis systems to generate and cause display of anamorphic media are disclosed. In one embodiment, an anamorphosis system is configured to identify a set of features of a space, determine relative positions of the set of features, determine a perspective of the mobile device within the space based on the relative positions of the set of features, retrieve anamorphic media based on the location of the mobile device, and apply the anamorphic media to a presentation of the space at the mobile device. The anamorphic media may include media items such as images and videos, configured such that the media items are only visible from one or more specified perspectives. The anamorphic media may include a stylized text string projected onto surfaces of a space such that the stylized text string is correctly displayed when viewed through a user device from a specified perspective.

Inventors: Cowburn; Piers George (London, GB), Pan; Qi (London, GB), Pilipski; Eitan (Los Angeles, CA)

Applicant: Snap Inc. (Santa Monica, CA)

Family ID: 66767745

Appl. No.: 19/187623

Filed: April 23, 2025

Related U.S. Application Data

parent US continuation 18119112 20230308 PENDING child US 19187623

parent US continuation 16277626 20190215 parent-grant-document US 11861795 child US 18119112

parent US continuation 15436363 20170217 parent-grant-document US 10319149 child US 16277626

Publication Classification

Int. Cl.: **G06T19/00** (20110101); **G06T7/73** (20170101); **G06T13/20** (20110101); **G06T15/20** (20110101)

U.S. Cl.:

CPC **G06T19/006** (20130101); **G06T7/73** (20170101); **G06T7/74** (20170101); **G06T13/20** (20130101); **G06T15/20** (20130101); G06T2200/24 (20130101); G06T2207/30244 (20130101)

Background/Summary

PRIORITY CLAIM [0001] This application is a continuation of U.S. patent application Ser. No. 18/119,112, filed Mar. 8, 2023, which application is a continuation of and claims the benefit of priority of U.S. patent application Ser. No. 16/277,626, filed Feb. 15, 2019, now issued as U.S. Pat. No. 11,861,795, which application is a continuation of and claims the benefit of priority of U.S. patent application Ser. No. 15/436,363, filed on Feb. 17, 2017, now issued as U.S. Pat. No. 10,319,149, which applications are hereby incorporated by reference herein in their entirety.

TECHNICAL FIELD

[0002] Embodiments of the present disclosure relate generally to mobile computing technology and, more particularly, but not by way of limitation, to the presentation of augmented and virtual reality displays.

BACKGROUND

[0003] Augmented reality (AR) is a live direct or indirect view of a physical, real-world environment whose elements are supplemented, or “augmented,” by a computer-generated sensory input such as sound, video, graphics, or the like. As a result, the technology functions to enhance a user's perception of reality.

[0004] Anamorphosis is a distorted projection or perspective requiring the viewer to use a special device or occupy a special vantage point (or both) to reconstitute an image. Anamorphic media may be generated through a process of greatly distorting an image only to have it revealed either from a single vantage point or from its reflection on a mirrored surface. For example, artists have created anamorphic art at physical locations, wherein the anamorphic art is only viewable from a single perspective, and appears distorted from all other vantage points.

Description

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0005] 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.

[0006] FIG. 1 is a block diagram showing an example messaging system for exchanging data (e.g., messages and associated content) over a network in accordance with some embodiments, wherein the messaging system includes an augmented reality anamorphosis system.

[0007] FIG. 2 is block diagram illustrating further details regarding a messaging system, according to example embodiments.

[0008] FIG. 3 is a schematic diagram illustrating data that may be stored in the database of the messaging server system, according to certain example embodiments.

[0009] FIG. 4 is a schematic diagram illustrating a structure of a message, according to some embodiments, generated by a messaging client application for communication.

[0010] FIG. 5 is a schematic diagram illustrating an example access-limiting process, in terms of which access to content (e.g., an ephemeral message, and associated multimedia payload of data) or a content collection (e.g., an ephemeral message story) may be time-limited (e.g., made ephemeral) in accordance with some embodiments.

[0011] FIG. 6 is a block diagram illustrating various modules of an anamorphosis system, according to certain example embodiments.

[0012] FIG. 7 is a flowchart illustrating various operations of the anamorphosis system in generating and causing display of anamorphic media within a presentation of a space at a graphical user interface (GUI), according to certain example embodiments.

[0013] FIG. 8 is a diagram illustrating various operations of the anamorphosis system in causing display of anamorphic media in a presentation of a space, according to certain example embodiments.

[0014] FIG. 9 is a flowchart illustrating various operations of the anamorphosis system in retrieving anamorphic media to be displayed in a presentation of a space, according to certain example embodiments.

[0015] FIG. 10 is an example of anamorphic media displayed in a presentation of a space, according to certain example embodiments.

[0016] FIG. 11 is an example of anamorphic media displayed in a presentation of a space, according to certain example embodiments.

[0017] FIG. 12 is an example of anamorphic media displayed in a presentation of a space, according to certain example embodiments.

[0018] FIG. 13 is a block diagram illustrating a representative software architecture, which may be used in conjunction with various hardware architectures herein described and used to implement various embodiments.

[0019] FIG. 14 is a block diagram illustrating components of a machine, 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.

DETAILED DESCRIPTION

[0020] Reference will now be made in detail to specific example embodiments for carrying out the inventive subject matter of the present disclosure. In the following description, specific details are set forth in order to provide a thorough understanding of the subject matter. It shall be appreciated that embodiments may be practiced without some or all of these specific details.

[0021] Disclosed are anamorphosis systems to generate and cause display of anamorphic media (e.g., distorted text or other anamorphically distorted images) within a presentation of a space (e.g., a room or a defined environment). The anamorphosis systems are configured to identify a set of features of a space (e.g., walls and objects), determine relative positions of the set of features, determine a perspective of the mobile device within the space based on the relative positions of the set of features in the space, retrieve anamorphic media based on the location of the mobile device, and apply the anamorphic media to a presentation of the space at the mobile device. For example, in a bare rectangular room, an example anamorphosis system on a mobile device may use an image sensor to capture one or more images of the walls of the room, identify the walls and their relative positions, and then apply anamorphic text as an augmented reality addition to images output on a display of the mobile device. The anamorphic text may, for example, be presented clearly on the display when the mobile device is in one perspective (e.g., phone position) and displayed as distorted to the point of being unreadable from another perspective (e.g., position of the mobile device).

[0022] Anamorphic media may include media items such as images and videos, configured such that the media items are only visible from one or more specified perspectives. The anamorphic

media may also include a stylized text string projected onto surfaces of a space such that the stylized text string is correctly displayed when viewed through a user device from a specified perspective.

[0023] The anamorphic media may be generated through a process of distorting a media item by stretching, expanding, splitting, projecting, or otherwise altering the media item such that the media item may be revealed to a user from a single vantage point or through a viewing apparatus or surface (e.g., a cylindrical, conical, or flat mirror). Thus, if the anamorphic media is viewed from a perspective other than an intended perspective, the anamorphic media may appear distorted or unclear.

[0024] The anamorphosis system may include a video and image capture system to perform functionality that includes at least recording and presenting images of a space, and a graphical interface configured to display a presentation of the space. In some example embodiments, to apply the anamorphic media to a presentation of a space, the anamorphosis system generates a surface model of a space. The surface model of the space is a topographical representation of the space that includes three-dimensional depictions of features, surfaces, contours, and shapes that make up the space. For example, the surface model may include a wire-mesh representation of a three-dimensional view of the space. To generate the surface model, the anamorphosis system may apply various computer vision techniques to digital images and videos of the space. For example, the anamorphosis system may acquire, process, and analyze a digital image and/or video of the space to generate the surface model.

[0025] In further embodiments, the anamorphosis system may access a surface model database that includes surface models of spaces, organized based on geolocation coordinates of corresponding locations depicted by the surface models. The anamorphosis system determines a location of a mobile device (e.g., based on geographic position sensors), and retrieves the corresponding surface model from the surface model database based on the location.

[0026] The anamorphosis system identifies a set of features of the space to determine a perspective of the mobile device. The set of features may include distinguishing points or features such as contours in the space, markings, or other features that may be used as graphical markers. For example, the distinguishing points or features may include landmarks as well as identifiable objects such as windows and/or doors. Having identified the set of features of the space, the anamorphosis system determines relative positions of each of the distinguishing points or features relative to one another. For example, the relative positions may indicate distances between the distinguishing points or features. The anamorphosis system thereby determines a perspective of the mobile device based on the relative positions of the distinguishing points or features. The perspective indicates a representation of the space relative to the mobile device.

[0027] In some example embodiments, the anamorphosis system retrieves anamorphic media to be applied to the presentation of the space, based on the location of the mobile device. In some example embodiments, the anamorphic media may only be available to one or more specified users (e.g., based on user identifiers). For example, upon detecting a mobile device at a location the anamorphosis system may retrieve an anamorphic media assigned to the location based on location data coordinates.

[0028] Upon retrieving the anamorphic media to be displayed in the presentation of the space, the anamorphosis system applies transformations to the anamorphic media based on the perspective of the mobile device, and displays the transformed anamorphic media in the presentation of the space. In this way, the anamorphic media may appear differently based on the perspective of the mobile device.

[0029] Consider an illustrative example from a user perspective. A user of a mobile device causes display of a space corresponding to a current location of the mobile device (e.g., the user points a camera of the mobile device at a space adjacent to them). Based on location data retrieved from the mobile device, the anamorphosis system determines the location of the mobile device, and in

response retrieves corresponding anamorphic media to be displayed in the presentation of the space. The anamorphosis system receives a set of features of the space and relative positions of the set of features within the space from the mobile device, and determines a perspective of the mobile device based on the set of features and the relative positions. The anamorphosis system thereby applies a transformation to the anamorphic media based on the perspective of the mobile device. The anamorphosis system displays the transformed anamorphic media in the presentation of the space, for viewing by the user. As the user moves within the space, the anamorphosis system recalculates the perspective of the user, and updates the display of the anamorphic media within the presentation of the space in real-time. Thus, as the user views the anamorphic media item within the presentation of the space, the user may adjust their perspective until the anamorphic media is correctly displayed.

[0030] In some example embodiments, users may generate and tag their own anamorphic media to locations. A user at a location may generate anamorphic media to be displayed in a space, and assign the anamorphic media to a position in the space. For example, the user may specify the position of the anamorphic media by selecting landmarks and/or distinguishing features of the space, and specifying a viewing perspective of the anamorphic media. The anamorphosis system may store the anamorphic media in a database to be retrieved at a later time.

[0031] 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 client devices **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).

[0032] Accordingly, each 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 applications **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 which may include or be used as anamorphic media).

[0033] 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**, it will be appreciated that 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 client device **102** has a sufficient processing capacity.

[0034] 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**. In some embodiments, this data includes, message content, client device information, geolocation information, media annotation and overlays, message content persistence conditions, social network information, and live event information, as examples. In other embodiments, other data is used. Any such data may be used as part of or to generate anamorphic media in accordance with different embodiments described herein. Data exchanges within the messaging system **100** are invoked and controlled through functions available via user interfaces (UIs) of the messaging client application **104**.

[0035] 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(s) **118**, which facilitates access to a database(s) **120** in which is stored data associated with messages processed by the application server **112**.

[0036] Dealing specifically with the Application Program Interface (API) server **110**, this server 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 **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 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, opening and application event (e.g., relating to the messaging client application **104**).

[0037] The application server **112** hosts a number of applications and subsystems, including a messaging server application **114**, an image processing system **116**, a social network system **122**, and an anamorphosis system **124**. 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.

[0038] 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**.

[0039] 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** within the database(s) **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. The anamorphosis system **124** provides functionality to generate and cause display of anamorphic media within a presentation of a space.

[0040] The application server **112** is communicatively coupled to one or more database server(s) **118**, which facilitates access to a database(s) **120** in which is stored data associated with messages processed by the messaging server application **114**.

[0041] 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 subsystems, namely an ephemeral timer system **202**, a collection management system **204** and an annotation system **206**.

[0042] 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 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 SNAPCHAT story), selectively display and enable access to messages and associated content such as anamorphic media via the messaging client application **104**. Further details regarding the operation of the ephemeral timer system **202** are provided below.

[0043] The collection management system **204** is responsible for managing collections of media (e.g., collections of text, image video and audio data). In some examples, a collection of content (e.g., messages, including anamorphic media, images, video, text and audio) may be organized into an “event gallery” or an “event story.” Such a collection may be made available for a specified time period, such as the duration of an event to which the content relates. For example, content such as anamorphic media displayed at specific locations 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**.

[0044] The collection management system **204** furthermore includes a curation 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 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.

[0045] 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 (e.g., a SNAPCHAT 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 anamorphic media, pictures, texts, logos, animations, and sound effects. An example of a visual effect includes color overlaying, or projecting an anamorphic media item over a presentation depicting a space. 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 including text that can be overlaid on top of a photograph or video stream generated 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(s) **120** and accessed through the database server(s) **118**.

[0046] 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.

[0047] 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

[0048] FIG. **3** is a schematic diagram **300** illustrating data which may be stored in the database(s) **120** of the messaging server system **108**, according to certain example embodiments. While the content of the database(s) **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).

[0049] The database(s) **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).

[0050] 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.

[0051] The database(s) **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 (e.g., anamorphic media items) that are displayed as overlaid on an image or video during presentation to a recipient user. For example, the overlay may include an anamorphic media item displayed within a presentation of a space, such that the anamorphic media item appears to be projected over a set of three dimensional surfaces of a space, following the contours of the surfaces of the space. Filters may be of various types, including a 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.

[0052] Other annotation data that may be stored within the image table **308** is so-called “lens” data. A “lens” may be a real-time special effect and sound that may be added to an image or a video.

[0053] 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 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**.

[0054] A story table **306** stores data regarding collections of messages and associated image, video or audio data, which are compiled into a collection (e.g., a SNAPCHAT 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 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.

[0055] A collection may also constitute a “live story,” which is a collection of content from

multiple users that is created manually, automatically or using a combination of manual and automatic techniques. For example, a “live story” may constitute a curated stream of user-submitted content from various locations and events. Users, whose client devices have location services enabled and are at a common location event at a particular time may, for example, be presented with an option, via a user interface of the messaging client 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.

[0056] 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 to a specific organization or other entity (e.g., is a student on the university campus).

[0057] FIG. **4** is a schematic diagram illustrating a structure of a message **400**, according to some in some embodiments, generated by a messaging client application **104** for communication to a further messaging client application **104** or the messaging server application **114**. The content of a particular message **400** is used to populate the message table **314** stored within the database(s) **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 the following components: [0058] A message identifier **402**: a unique identifier that identifies the message **400**. [0059] 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**. [0060] A message image payload **406**: image data, captured by a camera component of a client device **102** or retrieved from memory of a client device **102**, and that is included in the message **400**. [0061] A message video payload **408**: video data, captured by a camera component or retrieved from a memory component of the client device **102** and that is included in the message **400**. [0062] A message audio payload **410**: audio data, captured by a microphone or retrieved from the memory component of the client device **102**, and that is included in the message **400**. [0063] 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**. [0064] 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**. [0065] 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 the message image payload **406**, or a specific video in the message video payload **408**). [0066] 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, multiple images within the message image payload **406** may each be associated with multiple content collections using identifier values. [0067] 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 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. [0068] A message sender identifier **422**: an identifier (e.g., a messaging system identifier, email address or device identifier) indicative of a user of the client device **102** on which the message **400** was

generated and from which the message **400** was sent [0069] 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.

[0070] 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**.

[0071] 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 including anamorphic media) or a content collection (e.g., an ephemeral message story **504**) may be time-limited (e.g., made ephemeral). For example, an ephemeral message **502** may include an anamorphic media item which may be displayed for a period of time specified by the story timer **514**.

[0072] 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, where the messaging client application **104** is a SNAPCHAT application client, 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**.

[0073] 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.

[0074] The ephemeral message **502** is shown in FIG. 5 to be included within an ephemeral message story **504** (e.g., a personal SNAPCHAT story, or an event story). The ephemeral message story **504** has an associated story duration parameter **508**, a value of which determines a time-duration for which the ephemeral message story **504** is presented and accessible to users of the messaging system **100**. The story duration parameter **508**, for example, may be the duration of a music concert, where the ephemeral message story **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 story duration parameter **508** when performing the setup and creation of the ephemeral message story **504**.

[0075] Additionally, each ephemeral message **502** within the ephemeral message story **504** has an associated story 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 story **504**. Accordingly, a particular ephemeral message story **504** may “expire” and become inaccessible within the context of the ephemeral message story **504**, prior to the ephemeral message story **504** itself expiring in terms of the story duration parameter **508**. The story duration parameter **508**, story participation parameter **510**, and message receiver identifier **424** each provide input to a story timer **514**, which operationally determines, firstly, whether a particular ephemeral message **502** of the ephemeral message story **504** will be displayed to a particular receiving user and, if so,

for how long. Note that the ephemeral message story **504** is also aware of the identity of the particular receiving user as a result of the message receiver identifier **424**.

[0076] Accordingly, the story timer **514** operationally controls the overall lifespan of an associated ephemeral message story **504**, as well as an individual ephemeral message **502** included in the ephemeral message story **504**. In one embodiment, each and every ephemeral message **502** within the ephemeral message story **504** remains viewable and accessible for a time-period specified by the story duration parameter **508**. In a further embodiment, a certain ephemeral message **502** may expire, within the context of ephemeral message story **504**, based on a story 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 the context of the ephemeral message story **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 story **504**.

[0077] The ephemeral timer system **202** may furthermore operationally remove a particular ephemeral message **502** from the ephemeral message story **504** based on a determination that it has exceeded an associated story participation parameter **510**. For example, when a sending user has established a story participation parameter **510** of 24 hours from posting, the ephemeral timer system **202** will remove the relevant ephemeral message **502** from the ephemeral message story **504** after the specified 24 hours. The ephemeral timer system **202** also operates to remove an ephemeral message story **504** either when the story participation parameter **510** for each and every ephemeral message **502** within the ephemeral message story **504** has expired, or when the ephemeral message story **504** itself has expired in terms of the story duration parameter **508**.

[0078] In certain use cases, a creator of a particular ephemeral message story **504** may specify an indefinite story duration parameter **508**. In this case, the expiration of the story participation parameter **510** for the last remaining ephemeral message **502** within the ephemeral message story **504** will determine when the ephemeral message story **504** itself expires. In this case, a new ephemeral message **502**, added to the ephemeral message story **504**, with a new story participation parameter **510**, effectively extends the life of an ephemeral message story **504** to equal the value of the story participation parameter **510**.

[0079] Responsive to the ephemeral timer system **202** determining that an ephemeral message story **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 story **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** 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**.

[0080] FIG. **6** is a block diagram **600** illustrating components of the anamorphosis system **124**, that configure the anamorphosis system **124** to cause display of anamorphic media in a presentation of a space, according to various example embodiments. The anamorphosis system **124** is shown as including a location module **602**, a presentation module **604**, an identification module **606**, and an anamorphosis module **608**, all, or some, 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 processors **610** (e.g., by configuring such one or more processors to perform functions described for that module) and hence may include one or more of the processors **610**.

[0081] Any one or more of the modules described may be implemented using hardware alone (e.g., one or more of the processors **610** of a machine) or a combination of hardware and software. For example, any module described of the anamorphosis system **124** may physically include an

arrangement of one or more of the processors **610** (e.g., a subset of or among the one or more processors of the machine) configured to perform the operations described herein for that module. As another example, any module of the engagement tracking system **610** may include software, hardware, or both, that configure an arrangement of one or more processors **610** (e.g., among the one or more processors of the machine) to perform the operations described herein for that module. Accordingly, different modules of the anamorphosis system **124** may include and configure different arrangements of such processors **610** or a single arrangement of such processors **610** at different points in time. Moreover, any two or more modules of the anamorphosis system **124** 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.

[0082] FIG. 7 is a flowchart illustrating various operations of the anamorphosis system **124** in performing a method **700** for causing display of anamorphic media in a presentation of a space, according to certain example embodiments. Operations of the method **700** may be performed by the modules described above with respect to FIG. 6. As shown in FIG. 7, the method **700** includes one or more operations **702**, **704**, **706**, **708**, **710**, **712**, and **714**.

[0083] Operation **702** may be performed by the presentation module **604**. At operation **702**, the presentation module **604** causes display of a presentation of a space within a graphical user interface (GUI) of a mobile device (e.g., client device **102**). The mobile device may include a camera that captures images of a surrounding area. The images may thereby be displayed at within a GUI displayed on the mobile device.

[0084] Operation **704** may be performed by the location module **602**. At operation **704**, the location module **602** determines a location of the mobile device, wherein the location corresponds to the space displayed in the GUI. The location module **602** may determine the location based on GPS coordinates, or in some example embodiments, based on the images of the space captured by the camera of the mobile device. For example, the location module **602** may determine the location of the mobile device based on image recognition. The location module **602** may compare the images collected by the camera of the mobile device (e.g., client device **102**) with a catalog of preloaded images depicting locations. Based on the comparison, the location module **602** determines a location of the mobile device.

[0085] Operation **706** may be performed by the identification module **606**. At operation **706**, the identification module **606** identifies a set of features of the space. The set of features may include landmarks or other distinguishing features, such as windows, doors, wall outlets, identifying markings (e.g., a painted “X”), edges of walls, or the like. For example, the identification module **606** may employ computer vision and/or feature detection techniques known to persons of ordinary skill in the art, wherein the identification module **606** may collect image data that include visual images (e.g., through a camera element of the client device **102**). In some example embodiments, the identification module **606** identifies at least three distinct features.

[0086] Operation **708** may be performed by the identification module **606**. At operation **708**, the identification module **606** determines relative positions of each of the set of features identified, based on relative positions of the set of features. In some example embodiments, the identification module **606** may determine distances between each features, and a position of each feature in the display. For example, the identification module **606** may apply triangulation techniques to determine relative positions of each of the set of features.

[0087] Operation **710** may be performed by the identification module **606**. At operation **710**, the identification module determines a perspective of the mobile device based on the relative positions of each of the set of features. The perspective of the mobile device indicates a position and vantage point of the mobile device at the location.

[0088] Operation **712** may be performed by the anamorphosis module **608**. At operation **712**, the

anamorphosis module **608** retrieves anamorphic media based on the location of the mobile device. The anamorphic media includes images and video that may be displayed in a presentation of a space, and which appear distorted unless viewed from a specific viewing point in the location. [0089] In some example embodiments, the anamorphosis module **608** may access a database of anamorphic media that includes anamorphic media categorized based on location, and wherein each anamorphic media item is to be viewed from a specific viewing point at a corresponding location.

[0090] The anamorphic media may include an image to be displayed at a location, wherein the image is only discernable if viewed from a specific position at the location. For example, the anamorphic media may include a stylized text string wherein the text string is not legible unless viewed from a specific viewing location (e.g., from a specified perspective), or in further embodiments, the anamorphic media may include a video or animation that plays once the user views the anamorphic media from a specific perspective.

[0091] Operation **714** may be performed by the presentation module **604**. At operation **714**, the presentation module **604** causes display of the anamorphic media in the presentation of the space based on the perspective of the mobile device.

[0092] FIG. **8** is a diagram illustrating various operations of the anamorphosis system **124** in performing a method **800** for causing display of the anamorphic media in the presentation of the space, according to certain example embodiments. Operations of the method **800** may be performed by the modules described above with respect to FIG. **6**. As shown in FIG. **8**, the method **800** includes one or more operations **802**, **804**, and **806** that may be performed as part (e.g., a precursor task, a subroutine, or a portion) of operation **714** of the method **700**, according to some example embodiments.

[0093] Operation **802** may be performed by the anamorphosis module **608**. At operation **802**, the anamorphosis module **608** accesses a surface model of the space based on the location of the mobile device, wherein the surface model includes a three-dimensional representation of the space, such as a wire-mesh form.

[0094] In some example embodiments, the surface model may be generated by the anamorphosis module **608** based on computer vision. The surface model may be a geometric representation that includes a three-dimensional representation of a space based on a set of vertices and edges that together form polygons depicting the space. In further embodiments, the anamorphosis module **608** may access a surface model database that includes a set of pre-generated surface models categorized based on location data.

[0095] Operation **804** may be performed by the anamorphosis module **608**. At operation **804**, the anamorphosis module **608** causes display of the anamorphic media in the presentation of the space based on the three-dimensional representation of the space as depicted by the surface model and the perspective of the mobile device.

[0096] FIG. **9** is a flowchart illustrating various operations of the anamorphosis system **124** in performing a method **900** receiving anamorphic media, according to certain example embodiments. Operations of the method **900** may be performed by the modules described above with respect to FIG. **6**. As shown in FIG. **9**, the method **900** includes one or more operations **902**, **904**, and **906** that may be performed as part (e.g., a precursor task, a subroutine, or a portion) of the method **700**, according to some example embodiments.

[0097] At operation **902**, the anamorphosis system **124** receives anamorphic media from a client device **102** (e.g., a second mobile device of a second user). In some example embodiments, a second user may submit media data to be converted into anamorphic media by the anamorphosis system **124**. For example, the user may provide the anamorphosis system **124** with media data (e.g., pictures, videos), as well as location data indicating a location in which to assign the media data, and positioning data to indicate a position to display the media data in a presentation of a space corresponding to the location. The positioning data may include a perspective specified by

the user, wherein the perspective may be defined by relative positions of a set of features in a space.

[0098] Consider an illustrative example from a user perspective. A user may provide the anamorphosis system **124** with media data that includes a media item such as a digital image or video to be converted and displayed as anamorphic media at a specified location. The user may tag a media item with location data, and specify a display configuration of the media item in a presentation of the location. For example, the user may specify that the media item is to be displayed so that it is viewable from a specific viewing location (e.g., based on location data and a specified perspective of the client device **102**). Upon receiving the display configuration, the anamorphosis system **124** may apply a transformation to the media item in order to generate the anamorphic media. The transformation may include stretching, distorting, or altering the media item, such that the media item may be projected onto a surface of the space, and be visible from a perspective specified by the user. For example, the anamorphosis system **124** may apply transformations to the media item such that the media item is projected onto various surfaces on a space.

[0099] In some example embodiments, the second user may provide an input to the anamorphosis system **124** specifying that the anamorphic media is only visible/made available to “friends,” or “connections” of the second user within a social media platform. In further embodiments, the second user may specify that the anamorphic media is only available/displayed to a first user (e.g., based on a user identifier of the first user).

[0100] At operation **904**, the anamorphosis system **124** assigns the anamorphic media to the location based on location data such as GPS coordinates. For example, the user may provide the anamorphosis system **124** with GPS coordinates of the location and in response, the anamorphosis system **124** may geo-tag the anamorphic media to the location.

[0101] At operation **906**, the anamorphosis system **124** detects a client device **102** of a user at the location. For example, the anamorphosis system **124** may maintain a geofence around the location and detect a mobile device as the mobile device transgresses a threshold of the geofence.

[0102] At operation **908**, the anamorphosis system **124** retrieves the anamorphic media in response to detecting the user at the location, and causes display of the anamorphic media in a presentation of the space within the mobile device of the user. The display of the anamorphic media may vary based on the perspective of the user.

[0103] FIG. **10** is an example of anamorphic media **1020** displayed in a presentation **1000** of a space **1010**, according to certain example embodiments. The presentation **1000** may be displayed within a GUI at a client device **102** according to the method **700** of FIG. **7**. As shown in FIG. **10**, the anamorphosis system **124** may display the anamorphic media **1020** in the presentation **1000** based on a perspective of a client device **102** displaying the presentation **1000**. As the perspective of the client device **102** changes (e.g., the user moves to a different viewing location), the anamorphosis system **124** may alter the display of the anamorphic media **124** based on the changes in the perspective.

[0104] FIG. **11** is an example of anamorphic media **1120** displayed in a presentation **1100** of a space **1110**, from a first perspective, according to certain example embodiments. The presentation **1100** may be displayed within a GUI at a client device **102** according to the method **700** of FIG. **7**. As shown in FIG. **11**, the anamorphosis system **124** may display the anamorphic media **1120** in the presentation **1100** based on a perspective of a client device **102** displaying the presentation **1100**. As the perspective of the client device **102** changes (e.g., the user moves to a different viewing location), the anamorphosis system **124** may alter the display of the anamorphic media **124** based on the changes in the perspective.

[0105] FIG. **12** is an example of anamorphic media **1120** displayed in a presentation **1200** of a space **1110**, from a second perspective, according to certain example embodiments. The presentation **1200** may be displayed within a GUI at a client device **102** according to the method

700 of FIG. 7. As shown in FIG. 12, the anamorphosis system **124** may display the anamorphic media **1120** in the presentation **1200** based on a perspective of a client device **102** displaying the presentation **1200**. As the perspective of the client device **102** changes (e.g., the user moves to a different viewing location), the anamorphosis system **124** may alter the display of the anamorphic media **124** based on the changes in the perspective.

Software Architecture

[0106] FIG. 13 is a block diagram illustrating an example software architecture **1306**, which may be used in conjunction with various hardware architectures herein described. FIG. 13 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 architecture **1306** may execute on hardware such as machine **1300** of FIG. 13 that includes, among other things, processors **1304**, memory **1314**, and I/O components **1318**. A representative hardware layer **1352** is illustrated and can represent, for example, the machine **1300** of FIG. 13. The representative hardware layer **1352** includes a processing unit **1354** having associated executable instructions **1304**. Executable instructions **1304** represent the executable instructions of the software architecture **1306**, including implementation of the methods, components and so forth described herein. The hardware layer **1352** also includes memory and/or storage modules memory/storage **1356**, which also have executable instructions **1304**. The hardware layer **1352** may also comprise other hardware **1358**.

[0107] In the example architecture of FIG. 13, the software architecture **1306** may be conceptualized as a stack of layers where each layer provides particular functionality. For example, the software architecture **1306** may include layers such as an operating system **1302**, libraries **1320**, applications **1316** and a presentation layer **1314**. Operationally, the applications **1316** and/or other components within the layers may invoke application programming interface (API) API calls **1308** through the software stack and receive a response as in response to the API calls **1308**. 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 **1318**, while others may provide such a layer. Other software architectures may include additional or different layers.

[0108] The operating system **1302** may manage hardware resources and provide common services. The operating system **1302** may include, for example, a kernel **1322**, services **1324** and drivers **1326**. The kernel **1322** may act as an abstraction layer between the hardware and the other software layers. For example, the kernel **1322** may be responsible for memory management, processor management (e.g., scheduling), component management, networking, security settings, and so on. The services **1324** may provide other common services for the other software layers. The drivers **1326** are responsible for controlling or interfacing with the underlying hardware. For instance, the drivers **1326** include display drivers, camera drivers, Bluetooth® drivers, flash 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.

[0109] The libraries **1320** provide a common infrastructure that is used by the applications **1316** and/or other components and/or layers. The libraries **1320** provide functionality that allows other software components to perform tasks in an easier fashion than to interface directly with the underlying operating system **1302** functionality (e.g., kernel **1322**, services **1324** and/or drivers **1326**). The libraries **1320** may include system libraries **1344** (e.g., C standard library) that may provide functions such as memory allocation functions, string manipulation functions, mathematical functions, and the like. In addition, the libraries **1320** may include A PI libraries **1346** 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 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 **1320** may also include a wide variety of other libraries **1348** to provide many other APIs to the applications **1316** and other software components/modules.

[0110] The frameworks/middleware **1318** (also sometimes referred to as middleware) provide a higher-level common infrastructure that may be used by the applications **1316** and/or other software components/modules. For example, the frameworks/middleware **1318** may provide various graphic user interface (GUI) functions, high-level resource management, high-level location services, and so forth. The frameworks/middleware **1318** may provide a broad spectrum of other APIs that may be utilized by the applications **1316** and/or other software components/modules, some of which may be specific to a particular operating system **1302** or platform.

[0111] The applications **1316** include built-in applications **1338** and/or third-party applications **1340**. Examples of representative built-in applications **1338** 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 **1340** 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 **1340** may invoke the API calls **1308** provided by the mobile operating system (such as operating system **1302**) to facilitate functionality described herein.

[0112] The applications **1316** may use built in operating system functions (e.g., kernel **1322**, services **1324** and/or drivers **1326**), libraries **1320**, and frameworks/middleware **1318** to create user interfaces to interact with users of the system. Alternatively, or additionally, in some systems interactions with a user may occur through a presentation layer, such as presentation layer **1314**. In these systems, the application/component “logic” can be separated from the aspects of the application/component that interact with a user.

[0113] FIG. **14** is a block diagram illustrating components of a machine **1400**, 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. **14** shows a diagrammatic representation of the machine **1400** in the example form of a computer system, within which instructions **1410** (e.g., software, a program, an application, an applet, an app, or other executable code) for causing the machine **1400** to perform any one or more of the methodologies discussed herein may be executed. As such, the instructions **1410** may be used to implement modules or components described herein. The instructions **1410** transform the general, non-programmed machine **1400** into a particular machine **1400** programmed to carry out the described and illustrated functions in the manner described. In alternative embodiments, the machine **1400** operates as a standalone device or may be coupled (e.g., networked) to other machines. In a networked deployment, the machine **1400** 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 **1400** 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, head mounted device, VR goggles), 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 **1410**, sequentially or otherwise, that specify actions to be taken by machine **1400**. Further, while only a single machine **1400** is illustrated, the term “machine” shall also be taken to include a collection of machines that individually or jointly execute the instructions **1410** to perform any one or more of the

methodologies discussed herein.

[0114] The machine **1400** may include processors **1404**, memory memory/storage **1406**, and I/O components **1418**, which may be configured to communicate with each other such as via a bus **1402**. The memory/storage **1406** may include a memory **1414**, such as a main memory, or other memory storage, and a storage unit **1416**, both accessible to the processors **1404** such as via the bus **1402**. The storage unit **1416** and memory **1414** store the instructions **1410** embodying any one or more of the methodologies or functions described herein. The instructions **1410** may also reside, completely or partially, within the memory **1414**, within the storage unit **1416**, within at least one of the processors **1404** (e.g., within the processor's cache memory), or any suitable combination thereof, during execution thereof by the machine **1400**. Accordingly, the memory **1414**, the storage unit **1416**, and the memory of processors **1404** are examples of machine-readable media.

[0115] The I/O components **1418** 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 **1418** that are included in a particular machine **1400** 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 **1418** may include many other components that are not shown in FIG. **14**. The I/O components **1418** 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 **1418** may include output components **1426** and input components **1428**. The output components **1426** 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 **1428** 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.

[0116] In further example embodiments, the I/O components **1418** may include biometric components **1430**, motion components **1434**, environmental environment components **1436**, or position components **1438** among a wide array of other components. For example, the biometric components **1430** 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 **1434** may include acceleration sensor components (e.g., accelerometer), gravitation sensor components, rotation sensor components (e.g., gyroscope), and so forth. The environment components **1436** 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 components, pressure sensor components (e.g., barometer), acoustic sensor components (e.g., one or more microphones that detect background noise), proximity sensor components (e.g., infrared sensors that detect nearby objects), gas sensors (e.g., gas detection sensors to detection concentrations of hazardous gases for safety or to measure pollutants in the atmosphere), or other components that may provide indications, measurements, or signals corresponding to a surrounding physical environment. The position components **1438** may include location sensor components (e.g., a Global Position system (GPS) receiver component), altitude sensor components (e.g.,

altimeters or barometers that detect air pressure from which altitude may be derived), orientation sensor components (e.g., magnetometers), and the like.

[0117] Communication may be implemented using a wide variety of technologies. The I/O components **1418** may include communication components **1440** operable to couple the machine **1400** to a network **1432** or devices **1420** via coupling **1422** and coupling **1424** respectively. For example, the communication components **1440** may include a network interface component or other suitable device to interface with the network **1432**. In further examples, communication components **1440** may include wired communication components, wireless communication components, cellular communication components, Near Field Communication (NFC) components, Bluetooth® components (e.g., Bluetooth® Low Energy), Wi-Fi® components, and other communication components to provide communication via other modalities. The devices **1420** may be another machine or any of a wide variety of peripheral devices (e.g., a peripheral device coupled via a Universal Serial Bus (USB)).

[0118] Moreover, the communication components **1440** may detect identifiers or include components operable to detect identifiers. For example, the communication components **1440** 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 **1440**, 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.

Glossary

[0119] “ANAMORPHOSIS” in this context refers to distortions and transformations applied to a media items such as images and videos, such that the media items appear normal when viewed from a particular point or through a suitable viewing device, mirror, or lens.

[0120] “PERSPECTIVE” in this context refers to a viewing angle of a user at a particular location.

[0121] “CARRIER SIGNAL” in this context 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 medium to facilitate communication of such instructions. Instructions may be transmitted or received over the network using a transmission medium via a network interface device and using any one of a number of well-known transfer protocols.

[0122] “CLIENT DEVICE” in this context 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), smart phones, tablets, ultra books, 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.

[0123] “COMMUNICATIONS NETWORK” in this context 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

type 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 (1×RTT), 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.

[0124] “EMPHEMERAL MESSAGE” in this context 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.

[0125] “MACHINE-READABLE MEDIUM” in this context refers to a component, device or other tangible media able to store instructions and data temporarily or permanently and may include, but is not be limited to, random-access memory (RAM), read-only memory (ROM), buffer memory, flash memory, optical media, magnetic media, cache memory, other types of storage (e.g., Erasable Programmable Read-Only Memory (EEPROM)) and/or any suitable combination thereof. The term “machine-readable medium” should be taken to include a single medium or multiple media (e.g., a centralized or distributed database, or associated caches and servers) able to store instructions. The term “machine-readable medium” shall also be taken to include any medium, or combination of multiple media, that is capable of storing instructions (e.g., code) for execution by a machine, such that the instructions, when executed by one or more processors of the machine, cause the machine to perform any one or more of the methodologies described herein. Accordingly, a “machine-readable medium” refers to a single storage apparatus or device, as well as “cloud-based” storage systems or storage networks that include multiple storage apparatus or devices. The term “machine-readable medium” excludes signals per se.

[0126] “COMPONENT” in this context refers to a device, physical entity or logic having boundaries defined by function or subroutine calls, branch points, application program interfaces (APIs), or other technologies that provide for the partitioning or modularization of particular processing or control functions. Components may be combined via their interfaces with other components to carry out a machine process. A component may be a packaged functional hardware unit designed for use with other components and a part of a program that usually performs a particular function of related functions. Components may constitute either software components (e.g., code embodied on a machine-readable medium) or hardware components. A “hardware component” is a tangible unit capable of performing certain operations and may be configured or arranged in a certain physical manner. In various 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 portion) as a hardware component that operates to perform certain operations as described herein. A hardware component may also be implemented mechanically, electronically, or any suitable combination thereof. For example, a hardware component may include dedicated circuitry or logic that is permanently configured to perform certain operations. A hardware component may be a special-purpose processor, such as a Field-Programmable Gate Array (FPGA) or an Application Specific Integrated Circuit (ASIC). A hardware component may also include programmable logic or circuitry that is temporarily configured by software to perform certain operations. For example, a hardware component may include software executed by a general-purpose processor or other programmable processor. Once configured by such software, hardware components become

specific machines (or specific components of a machine) uniquely tailored to perform the configured functions and are no longer general-purpose processors. It will be appreciated that the decision to implement a hardware component mechanically, in dedicated and permanently configured circuitry, or in temporarily configured circuitry (e.g., configured by software) may be driven by cost and time considerations. Accordingly, the phrase “hardware component” (or “hardware-implemented component”) should be understood to encompass a tangible entity, be that an entity that is physically constructed, permanently configured (e.g., hardwired), or temporarily configured (e.g., programmed) to operate in a certain manner or to perform certain operations described herein. Considering 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 processor, the general-purpose processor may be configured as respectively different special-purpose processors (e.g., comprising different hardware components) at different times. Software accordingly configures a particular processor or processors, for example, to constitute a particular hardware component at one instance of time and to constitute a different hardware component at a different instance of time. Hardware components can provide information to, and receive information from, other hardware components. Accordingly, the described hardware components may be regarded as being communicatively coupled. Where multiple hardware components exist contemporaneously, communications may be achieved through signal transmission (e.g., over appropriate circuits and buses) between or among two or more of the hardware components. In 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 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 Application Program Interface (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.

[0127] “PROCESSOR” in this context 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.

[0128] “TIMESTAMP” in this context refers to a sequence of characters or encoded information identifying when a certain event occurred, for example giving date and time of day, sometimes accurate to a small fraction of a second.

Claims

1. A system comprising: a memory; and at least one hardware processor couple to the memory and comprising instructions that cause the system to perform operations comprising: causing display of a presentation of a space within a graphical user interface (GUI) of a client device; determining a perspective of the client device relative to a landmark within the presentation of the space; accessing a surface model that comprises a topographical representation of the space; transforming media content based on the topographical representation of the space; and presenting the transformed media content within the presentation of the space within the GUI based on the perspective of the client device relative to the landmark.
2. The system of claim 1, wherein the operations further comprise: identifying the landmark based on a set of image features; determining a location of the client device based on the landmark; and selecting the media content based on the location.
3. The system of claim 2, wherein the identifying the landmark based on the set of image features includes: receiving an input that identifies the set of image features; and identifying the landmark responsive to the input.
4. The system of claim 1, wherein the transforming the media content includes: receiving an input that selects the media content from among a collection of media content.
5. The system of claim 1, wherein the transforming the media content includes: receiving, from the client device, an input that specifies a display configuration of the media content; and wherein the transforming the media content is based on the display configuration.
6. The system of claim 1, wherein the transforming the media content includes applying a distortion to the media content.
7. The system of claim 1, wherein the surface model comprises a wire mesh that comprises the topographical representation of the space.
8. The system of claim 1, wherein the operations further comprise: accessing the media content that comprises access criteria, the access criteria comprising: one or more user identifiers that include a user identifier associated with the client device; and a display configuration that specifies a perspective to orient a display of the media content.
9. The system of claim 1, wherein the transformed media content appears distorted from perspectives other than the perspective of the client device relative to the landmark.
10. A method comprising: causing display of a presentation of a space within a graphical user interface (GUI) of a client device; determining a perspective of the client device relative to a landmark within the presentation of the space; accessing a surface model that comprises a topographical representation of the space; transforming media content based on the topographical representation of the space; and presenting the transformed media content within the presentation of the space within the GUI based on the perspective of the client device relative to the landmark.

- 11.** The method of claim 10, further comprising: identifying the landmark based on a set of image features; determining a location of the client device based on the landmark; and selecting the media content based on the location.
 - 12.** The method of claim 11, wherein the identifying the landmark based on the set of image features includes: receiving an input that identifies the set of image features; and identifying the landmark responsive to the input.
 - 13.** The method of claim 10, wherein the transforming the media content includes: receiving an input that selects the media content from among a collection of media content.
 - 14.** The method of claim 10, wherein the transforming the media content includes: receiving, from the client device, an input that specifies a display configuration of the media content; and wherein the transforming the media content is based on the display configuration.
 - 15.** The method of claim 10, wherein the transforming the media content includes applying a distortion to the media content.
 - 16.** The method of claim 10, wherein the surface model comprises a wire mesh that comprises the topographical representation of the space.
 - 17.** The method of claim 10, further comprising: accessing the media content that comprises access criteria, the access criteria comprising: one or more user identifiers that include a user identifier associated with the client device; and a display configuration that specifies a perspective to orient a display of the media content.
 - 18.** The method of claim 10, wherein the transformed media content appears distorted from perspectives other than the perspective of the client device relative to the landmark.
 - 19.** A non-transitory machine-readable storage medium comprising instructions that, when executed by one or more processors of a machine, cause the machine to perform operations including: causing display of a presentation of a space within a graphical user interface (GUI) of a client device; determining a perspective of the client device relative to a landmark within the presentation of the space; accessing a surface model that comprises a topographical representation of the space; transforming media content based on the topographical representation of the space; and presenting the transformed media content within the presentation of the space within the GUI based on the perspective of the client device relative to the landmark.
 - 20.** The non-transitory machine-readable storage medium of claim 19, further comprising: identifying the landmark based on a set of image features; determining a location of the client device based on the landmark; and selecting the media content based on the location.
-