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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
04 June 2020 (04.06.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/113020 A1

(51) International Patent Classification:
G06K 9/00 (2006.01)

(21) International Application Number:
PCT/US2019/063632

(22) International Filing Date:
27 November 2019 (27.11.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/772,770 29 November 2018 (29.11.2018) US

(71) Applicant: **GOOGLE LLC** [US/US]; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US).

(72) Inventors: **BADR, Ibrahim**; Brandschenkestrasse 110, 8002 Zurich (CH). **CHUNG, Edgar**; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US).

(74) Agent: **PROBST, Joseph J.** et al.; Dority & Manning, P.A., P.O. Box 1449, Greenville, South Carolina 29602 (US).

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

(54) Title: PROVIDING CONTENT RELATED TO OBJECTS DETECTED IN IMAGES

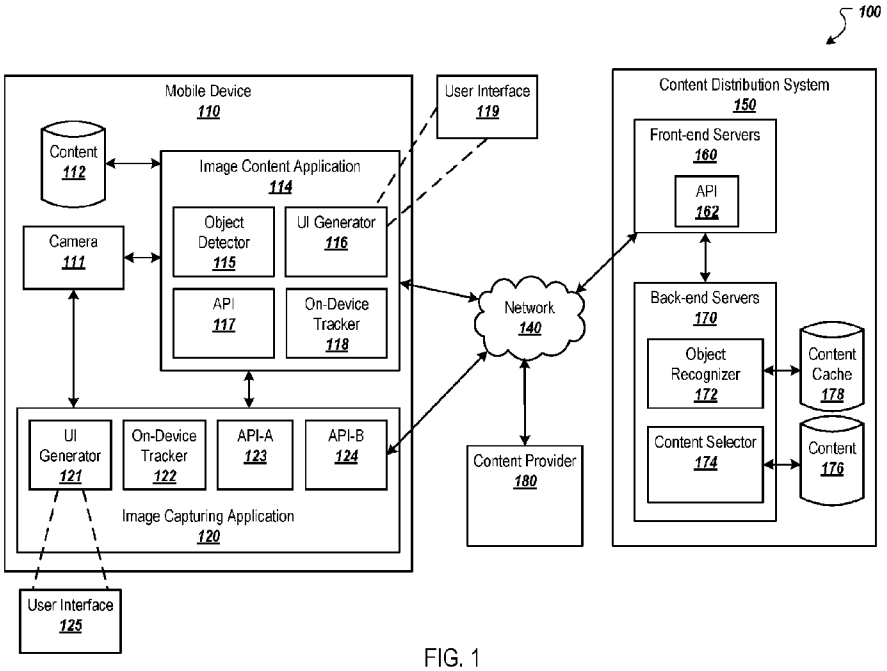

FIG. 1

(57) Abstract: Methods, systems, and apparatus for recognizing objects and providing content related to the recognized objects are described. In one aspect, a method includes capturing, by a user device, first image data of a first image displayed in a viewfinder of a camera on the user device. An application operating on the user device sends the first image data to a server. The application receives, from the server, data corresponding to object(s) depicted by the first image. The data includes, for each object, content related to the object and a location within the first image to present the content. After receiving the data, subsequent image data of a subsequent image displayed in the viewfinder is captured. The application determines, for each object within the viewfinder of the camera for which data was received, a subsequent location of the object based on the location in the data received from the server.

WO 2020/113020 A1

[Continued on next page]

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

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

Published:

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

PROVIDING CONTENT RELATED TO OBJECTS DETECTED IN IMAGES

BACKGROUND

[0001] Computer visual analysis techniques can be used to detect and recognize objects in images. For example, optical character recognition (OCR) techniques can be used to recognize text in images and edge detection techniques can be used to detect objects (e.g., products, landmarks, animals, etc.) in images. Content related to the detected objects can be provided to a user, e.g., a user that captured the image in which the object is detected.

SUMMARY

[0002] This specification describes technologies relating to presenting content related to objects recognized in images.

[0003] In general, one innovative aspect of the subject matter described in this specification can be embodied in a method performed by one or more data processing apparatus of a user device, the method including capturing, by the user device, first image data of a first image displayed in a viewfinder of a camera on the user device. The first image can be displayed by an application operating on the user device. The application can send the first image data to a server system that is external to the user device. The application can receive, from the server system, data corresponding to one or more objects depicted by the first image. The data can include, for each object, content related to the object and a location within the first image to present the content on the viewfinder. After receiving the data corresponding to the one or more objects depicted by the first image, subsequent image data of a subsequent image displayed in the viewfinder on the user device can be captured. The subsequent image data can be different from the first image data. The application can determine, for each object within the viewfinder of a camera for which data was received from the server, a subsequent location based on the location provided by the data received from the server system. The subsequent location can be different from the location provided by the data received from the server system and corresponds to the object. For each object within the viewfinder of the camera for which data was received from the server and for which

a corresponding subsequent location was determined, content related to the object can be presented in the viewfinder. Other implementations of this aspect include corresponding apparatus, systems, and computer programs, configured to perform the actions of the methods, encoded on computer storage devices.

[0004] These and other implementations can each optionally include one or more of the following features. In some aspects, sending the first image data to a server system that is external to the user device can include detecting, by the user device, the presence of one or more objects in the image data. Sending the first image data to a server system that is external to the user device can be based upon the detecting.

[0005] In some aspects, detecting the presence of one or more objects in the image data can include processing the image data using a coarse classifier. Sending the first image data can include selecting image data of the first image data in which the presence of one or more objects is determined and transmitting the selected image data.

[0006] In some aspects, the application can include an application programming interface (API) that determines the subsequent location for each object in the viewfinder for which data was received from the server and causes the application to present the content for each object in the viewfinder. The API can include an on-device tracker that tracks location of objects depicted in the viewfinder of the camera.

[0007] In some aspects, the API is configured to obtain, from the user device, context data indicative of a context in which the first image was captured. The context data can include at least one of a geographic location of the client device at the time the image was captured or data identifying the application. The server can select the content for each of the one or more objects based on the context data. The server system can use the context data to disambiguate, for at least one object depicted by the image, between multiple potential objects that match the at least one object.

[0008] In some aspects, the server system determines, for a given object depicted by the first image, a category of the object. In response to determining the category of the object, at least a portion of the first image that depicts the given object can be sent to a content provider that recognizes objects of the determined category and that provides

content related to objects of the determined category. The content provider can provide content related to the given object. The content for the given object can be included in the data corresponding to one or more objects depicted by the first image provided to the application.

[0009] Some aspects can include presenting, by the application, an interface control for a second application that presents content related to objects depicted in images. The application can detect user interaction with the interface control and, in response, cause the user device to launch the second application, capture additional image data of an additional image displayed in a viewfinder of a camera, and provide the additional image data to the second application. The second application can obtain content related to one or more objects depicted by the additional image and present the additional image and the content related to the one or more objects depicted by the additional image.

[0010] In some aspects, the data corresponding to one or more objects depicted by the first image is generated by an object recognizer. The object recognizer can include a machine learning model to recognize objects in image data received from the mobile device.

[0011] The subject matter described in this specification can be implemented in particular embodiments so as to realize one or more of the following advantages. Content related to objects recognized in images can be syndicated in various applications that do not have the capability to recognize objects and/or identify content related to the recognized objects, allowing such applications to provide content that may be helpful to users and/or allow users to access content that the user may otherwise not be able to access. Code of an application that presents content related to objects in images can be implemented in another application to control processes for determining when to present content, requesting (or selecting content), and/or presenting the content such that the content is presented at appropriate times and in appropriate ways. Special purpose application programming interfaces (APIs) enable the applications to send images and context data to another system that recognizes the objects and

provides content and data specifying where in the image to present the content to the application.

[0012] Various features and advantages of the foregoing subject matter are described below with respect to the figures. Additional features and advantages are apparent from the subject matter described herein and the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] FIG. 1 is a block diagram of an example environment in which a mobile device presents content related to objects recognized in images.

[0014] FIG. 2 depicts a sequence of example screen shots of user interfaces that present content related to objects recognized in an image.

[0015] FIG. 3 depicts another sequence of example screen shots of a user interfaces that present an image.

[0016] FIG. 4 is a flow chart of an example process for presenting content related to objects recognized in an image.

[0017] FIG. 5 is a flow chart of an example process for selecting content related to an object recognized in an image received from an application and providing content for presentation by the application.

[0018] FIG. 6 is a flow chart of an example process for launching an application to present content related to an object recognized in an image captured by a different application.

[0019] FIG. 7 is a flow chart of an example process for obtaining content related to an object recognized in an image and presenting the content.

[0020] Like reference numbers and designations in the various drawings indicate like elements.

DETAILED DESCRIPTION

[0021] In general, systems and techniques described herein can recognize objects in images and present content related to the objects. A system (e.g., application and/or server of a content distribution system) can recognize objects in images captured by an application, e.g., an image capturing application, and select content related to the objects. The system can provide the content for presentation by the image capturing application. For example, the image capturing application can include a special purpose application programming interface (API) configured to send images captured from a viewfinder of a camera, and optionally context data for the image, to the system. The context data can include geographic location data, e.g., that specifies a location of a mobile device or the location at which an image was captured, an identifier for the application, time of day, etc.

[0022] The system can recognize objects in the images and select content based on the objects and/or the context data. The system can then provide the content to the image capturing application, e.g., with instruction data instructing the image capturing application where in the image to present the content. The image capturing application can present the content in the image and/or a viewfinder from which the image was captured. The API of the application can determine a current location of the objects in the viewfinder and present the content for the objects in at or near the objects in the viewfinder, or at another location specified by the instruction data.

[0023] In some implementations, the system that recognizes objects in images and provides content for the objects is operated by a different entity than an entity that distributes the application, e.g., the entities can be different organizations that develop and distribute different applications. The entity that operates the system can provide the API to other entities for integration in their applications so that the applications can request, from the system, content related to objects depicted in images captured by the applications.

[0024] In another example, the application can include a user interface control that, when interacted with, causes a mobile device on which the application is installed to launch another application, e.g., an image content application, and causes the application to provide an image (e.g., an image being presented by the application) to the image content application. The image content application can then recognize objects in the image, select content related to the recognized objects (or send the image to another system that recognizes the object and/or selects the content), and present the selected content. The application can also provide, to the image content application, context data that can be used by the application (or other system) to select the content that is presented.

[0025] The system, or image content application, can also request content from other entities, e.g., other content providers that can recognize objects in images and provide content related to the recognized objects. For example, the system can recognize a type of object in an image (e.g., that the image includes a bottle of wine). However, the system may not be able to recognize the actual object (e.g., the brand or type of wine) or have access to details about the object. The system can provide the image (or the portion that includes the object of interest) to the content provider with a request for content related to (e.g., information about) the object. The content provider can provide the content to the system and the system can cause the content to be presented by the image capturing application or the image content application.

[0026] FIG. 1 is a block diagram of an example environment 100 in which a mobile device 110 presents content related to objects recognized in images. The mobile device 110 is an electronic device that is capable of requesting and receiving resources over a data communication network 140, such as a local area network (LAN), a wide area network (WAN), the Internet, a mobile network, or a combination thereof. Example mobile devices 110 include smartphones, tablet computers, and wearable devices (e.g., smart watches). The environment 100 can include many mobile devices 110.

[0027] The mobile device 110 can include applications, e.g., native applications developed for a particular platform. For example, the mobile device 110 includes an image content application 114 and an image capturing application 120. This image

content application 114 may be developed and distributed by an entity (e.g., an organization) that operates a content distribution system 150 that provides content related to images. In some implementations, the image capturing application 120 may be developed and distributed by a different entity that is unaffiliated with the entity that operates the content distribution system 150. For example, the image capturing application 120 can be an application that presents images and the entity that develops and distributes the image capturing application 120 may want to present content provided by the content distribution system 150 with the images. In some implementations, the mobile device 110 may include only one of the two applications 114 and 120. For example, as described below, the mobile device 110 may not include the image content application 114. In some implementations, the image content application 114 and the image capturing application 120 can be developed and/or distributed by the same entity.

[0028] In general, the image content application 114 can present content related to objects depicted in images. The image content application 114 can control a camera 111 of the mobile device 110 to capture images or present a viewfinder of the camera 111. For example, the image content application 114 may be a dedicated application for controlling the camera 111, a camera-first application that controls the camera 111 for use with other features of the application 114, or another type of application that can access and control the camera 111. The image content application 114 can present the viewfinder of the camera 111 in user interfaces 119 of the camera application 114.

[0029] In general, the image content application 114 enables a user to view content (e.g., information or user experiences) related to objects depicted in the viewfinder of the camera 111 and/or view content related to objects depicted in images stored on the mobile device 110 or stored at another location accessible by the mobile device 110. The viewfinder is a portion of the mobile device's display that presents a live image of the scene that is in the field of the view of the camera's lens. As the user moves the camera 111 (e.g., by moving the mobile device 110), the viewfinder is updated to present the current field of view of the lens.

[0030] The image content application 114 includes an object detector 115, a user interface generator 116, an API 117, and an on-device tracker 118. The object detector 115 can detect objects in the viewfinder using edge detection and/or other object detection techniques. In some implementations, the object detector 115 includes a coarse classifier that determines whether an image includes an object in one or more particular classes (e.g., categories) of objects. For example, the coarse classifier may detect that an image includes an object of a particular class, with or without recognizing the actual object.

[0031] The coarse classifier can detect the presence of a class of objects based on whether or not the image includes (e.g., depicts) one or more features that are indicative of the class of objects. The coarse classifier can include a light-weight model to perform a low computational analysis to detect the presence of objects within its class(es) of objects. For example, the coarse classifier can detect, for each class of objects, a limited set of visual features depicted in the image to determine whether the image incudes an object that falls within the class of objects. In a particular example, the coarse classifier can detect whether an image depicts an object that is classified in one or more of the following classes: text, barcode, landmark, media object (e.g., album cover, movie poster, etc.), or artwork object (e.g., painting, sculpture, etc.). For barcodes, the coarse classifier can determine whether the image includes parallel lines with different widths.

[0032] In some implementations, the coarse classifier uses a trained machine learning model (e.g., a convolutional neural network) to determine whether images includes objects in one or more classes based on visual features of the images. For example, the machine learning model can be trained using labeled images that are labeled with their respective class(es) of objects. The machine learning model can be trained to classify images into zero or more of a particular set of classes of objects. The machine learning model can receive, as inputs, data related to the visual features of an image and output a classification into zero or more of the classes of objects in the particular set of classes of objects.

[0033] The coarse classifier can output data specifying whether a class of object has been detected in the image. The coarse classifier can also output a confidence value that indicates the confidence that the presence of a class of object has been detected in the image and/or a confidence value that indicates the confidence that an actual object, e.g., the Eiffel Tower, is depicted in the image. The object detector may therefore perform initial processing on a mobile device to determine whether an object is represented in image data. If it is determined that an object is represented in the image data, image data may be transmitted to a server for further processing. Use of bandwidth for transmitting of image data may therefore be restricted by use of a light-weight classifier, with only image data determined by the light-weight classifier being transmitted for further processing.

[0034] The object detector 115 can receive image data representing the field of view of the camera 111, e.g., what is being presented in the viewfinder, and detect the presence of one or more objects in the image data. If at least one object is detected in the image data, the image content application 114 can provide (e.g., transmit) the image data to a content distribution system 150 over the network 140. As described below, the content distribution system 150 can recognize objects in the image data and provide content related to the objects to the mobile device 110.

[0035] The image content application 114 can receive the content from the content distribution system 150 and present the content in a user interface 119 generated by the user interface generator 116. In some implementations, the user interface generator 116 generates a user interface that presents the image represented by the image data and the content received from the content distribution system 150 for the image. For example, the user interface generator 116 can generate a user interface that presents content selected for a particular object depicted in the image adjacent to the object in the image. The content can be presented in an overlay adjacent to (or within a threshold number of pixels of) the object. In another example, the content can be presented below or adjacent to the image.

[0036] The user interface generator 116 can also generate and update user interfaces that present the viewfinder of the camera 111 and the content received from the content

distribution system 150. In this example, the user interface generator 116 can present content selected for a particular object near the object in the live image of the viewfinder. For example, the user may move the mobile device 110 such that the location of the object in the viewfinder changes over time (and may even leave the field of view of the camera 111). The user interface generator 116 can interact with the on-device tracker 118 to track the location of objects in the viewfinder.

[0037] The on-device tracker 118 can track the movement of objects detected by the object detector 115 across multiple images. For example, the object detector 115 can output data specifying the location (e.g., pixel coordinates) of an object detected in an image. The pixel coordinates can include coordinates for multiple locations along the perimeter of the object, e.g., to outline the object. The on-device tracker 118 can then monitor the movement of the object based on the movement of the visual content depicted by the pixels at that location (e.g., within the pixel coordinates) across subsequent images.

[0038] To track the movement of the object, the on-device tracker 118 can analyze each subsequent image to identify the location of the visual content presented in the pixel coordinates of the first image in which the object was detected, similar to the way the on-device tracker 118 tracks the movement of individual pixels and groups of pixels. For example, the on-device tracker 118 can determine the visual characteristics of each pixel within the pixel coordinates for the object in the first image and the relative orientation of the pixels (e.g., distance and direction between pairs of pixels). In each subsequent image, the on-device tracker 118 can attempt to identify pixels having the same (or similar) visual characteristics and the same (or similar) orientation. For example, the orientation of an object within the viewfinder may change based on a change in orientation of the mobile device 110, a change in orientation of the object, and/or a change in distance between the mobile device 110 and the object. Thus, the on-device tracker 118 may determine that a group of pixels in a subsequent image matches the object if the distance between each pair of pixels is within a threshold of the distance between the pair of pixels in previous images.

[0039] In another example, the on-device tracker 118 can determine that a group of pixels in a subsequent image matches the object if the group of pixels has the same (or similar) shape and visual characteristics (e.g., color and intensity) irrespective of orientation or size. For example, the user may rotate the mobile device 110 or move closer to the object in an attempt to capture a better image of the object. In this example, the size and orientation of the object may change within the viewfinder, but the shape and color should remain the same or similar.

[0040] In another example, the on-device tracker 118 can identify edges of the object based on the pixel coordinates and track the movement of the edges from image to image. If the mobile device 110 is substantially still, the edges will likely not move much between successive images. Thus, the on-device tracker 118 can locate the edge in a subsequent image by analyzing the pixels near (e.g., within a threshold number of pixels of) the edge in the previous image.

[0041] The content distribution system 150 includes one or more front-end servers 160 and one or more back-end servers 170. The front-end servers 160 can receive image data from mobile devices, e.g., the mobile device 110. The front-end servers 160 can provide the image data to the back-end servers 170. The back-end servers 170 can identify (e.g., select) content related to objects recognized in the image data and provide the content to the front-end servers 160. In turn, the front-end servers 160 can provide the content to the mobile device 110 from which the image data was received.

[0042] The back-end servers 170 include an object recognizer 172 and a content selector 174. The object recognizer 172 can process image data received from mobile devices and recognize objects, if any, in the image data. The object recognizer 172 can use computer vision and/or other object recognition techniques (e.g., edge matching, pattern recognition, greyscale matching, gradient matching, etc.) to recognize objects in image data.

[0043] In some implementations, the object recognizer 172 uses a trained machine learning model (e.g., a convolutional neural network) to recognize objects in image data received from the mobile devices. For example, the machine learning model can be trained using labeled images that are labeled with their respective objects. The

machine learning model can be trained to recognize and output data identifying objects depicted in images represented by the image data. The machine learning model can receive, as inputs, data related to the visual features of an image and output data identifying objects depicted in the image. The object recognizer 172 may therefore perform processing that is computationally more expensive than the processing performed by the object detector 115.

[0044] The object recognizer 172 can also output a confidence value that indicates the confidence that the image depicts the recognized object. For example, the object recognizer 172 can determine a confidence level for each object recognized in the image based on a level of match between features of the object and the features of the image.

[0045] In some implementations, the object recognizer 172 includes multiple object recognizer modules, e.g., one for each class of objects that recognizes objects in its respective class. For example, the object recognizer 172 can include a text recognizer module that recognizes text (e.g., recognizes characters, words, etc.) in image data, a barcode recognizer module that recognizes (e.g., decodes) barcodes (including QR codes) in image data, a landmarks recognizer module that recognizes landmarks in image data, and/or other object recognizer modules that recognize a particular class of objects.

[0046] In some implementations, the image content application 114 provides, with the image data for an image, data specifying the location (e.g., pixel coordinates) within the image where a particular object or class of object was detected or selected by a user (e.g., a tap target). This can increase the speed at which objects are recognized by enabling the object recognizer 172 to focus on the image data for that location and/or by enabling the object recognizer 172 to use the appropriate object recognizer module (e.g., only the one for the class of object specified by the data received from the image content application 116) to recognize the object(s) in the image data. This also reduces the amount of computational resources that would be used by the other object recognition modules.

[0047] The content selector 174 can select content to provide to the image content application 114 for each object recognized in the image data. The content can include information related to the object (e.g., text that includes the name of the object and/or facts about the object), visual treatments (e.g., other images or videos of the object or of related objects), links to resources related to the object (e.g., links to web pages or application pages at which the user can purchase the object or view additional information about the object), or experiences related to the object (augmented reality video, playing music in response to recognizing a singer or poster of a singer), and/or other appropriate content. For example, if the object is a barcode, the selected content may include a text-based caption that includes the name of the product that corresponds to the barcode and information about the product, a link to a web page or application page at which the user can purchase the product, and/or an image of the product.

[0048] The content selector 174 can also select visual treatments that present text related to a recognized object. The visual treatments can be in the form of a text caption that can be presented at the object in the viewfinder. The text included in the captions can be based on a ranking of facts about the object, e.g., more popular facts may be ranked higher. The content selector 174 can select one or more of the captions for a recognized object to provide to the mobile device 110 based on the ranking.

[0049] The content selector 174 can select the text for a caption based on the level of confidence output by the object recognizer 172. For example, if the level of confidence is high (e.g., greater than a threshold), the text can include a popular fact about the object or the name of the object. If the level of confidence is low (e.g., less than a threshold), the text can indicate that the object might be what the object recognizer 172 detected (e.g. "this might be a golden retriever").

[0050] The content selector 174 can also select interactive controls based on the object(s) recognized in the image. For example, if the object recognizer 172 detects a phone number in the image, the content selector 174 can select a click-to-call user interface control (e.g., icon) that, when interacted with, causes the mobile device 110 to call the recognized phone number.

[0051] The content can be stored in a content data storage unit 178, which can include hard drives, flash memory, fast access memory, or other data storage devices. In some implementations, the content data storage unit 178 includes an index that specifies, for each object and/or type of object, content that can be provided for the object or type of object. The index can increase the speed at which content is selected for an object or type of object.

[0052] After the content is selected, the content can be provided to the mobile device 110 from which the image data was received, stored in a content cache 178 of the content distribution system 150, and/or stored at the top of a memory stack of the front-end servers 160. In this way, the content can be quickly presented to the user in response to the user requesting the content. If the content is provided to the mobile device 110, the image content application 114 can store the content in a local cache or other fast access memory of the mobile device 110. For example, the image content application 114 can store the content for an object with a reference to the object so that the image content application 114 can identify the appropriate content for the object in response to determining to present the content for the object.

[0053] In some implementations, the image capturing application 120 can invoke the image content application 114 to present content related to an image provided by the image capturing application 120. The API 117 of the image content application 114 defines methods and protocols that allow the image capturing application 120 to communicate with the image content application 114 using API calls. The image capturing application 120 can include a corresponding API, API-A 123. The APIs 117 and 123 can be special purpose APIs that define what data can be sent between the two applications 114 and 120. For example, the APIs can specify a particular set of data fields that can be populated with data and included in the API calls.

[0054] The image content application 114 and/or the image capturing application 130 can also store content and other data securely, e.g., using encryption, to prevent other applications from altering the data or pretending to be the image content application 114 or the image capturing application 120. For example, one of the APIs of the image

capturing application 120, e.g., one provided by the content distribution system 150, can provide these security features.

[0055] The image content application 114 and/or the content distribution system 150 can include security features that control what applications can launch or otherwise make API calls to the image content application 114 or to the content distribution system 150. For example, the image content application 114 can include a whitelist of package names for applications that are eligible to connect to the image content application 114, e.g., as a client of the image content application 114. When the image content application 114 receives an API call or request to connect to or launch the image content application 114, the image content application 114 can compare, to the whitelist, the package name for the application that sent the API call or request. If the package name matches a package name in the whitelist, the image content application 114 can allow the application to connect and make API calls to the image content application 114. If not, the image content application 114 can block the application from making API calls to the image content application 114. The front-end servers 160 of the content distribution 150 can handle requests and/or API calls in a similar manner.

[0056] The image content application 114 and/or the content distribution system 140 can also perform signature checks for applications that attempt to access or launch the image content application 114 or that attempt to access the content distribution system 150. For example, each application that is eligible to access the image content application 114 or the content distribution system 150 may have to be signed by an entity that develops the image content application 114 or another entity that verifies the authenticity of applications. When an application attempts to access or launch the image content application 114, the image content application 114 can compare a digital signature of the application to the digital signature generated by the entity that develops the image content application 114. If the signatures match, the application is the same application that was signed by the entity and the image content application 114 can allow the application to connect and make API calls to the image content application 114. If not, the application could be a fraudulent application and the image content application 114 can block the application from making API calls to the image content application 114. In some implementations, the application may be required to provide

its signature with each API call to the image content application 114. The front-end servers 160 of the content distribution system 150 can handle requests and/or API calls in a similar manner. These security features can prevent attacks on the image content application and/or the content distribution system 150.

[0057] One example API call that can be sent from the image capturing application 120 to the image content application 114 is a request to present content for an image provided by the image capturing application 120. For example, the image capturing application 120 may also be able to control the camera 111 to capture images. The image capturing application 120 may also be an image storage application (e.g., a gallery application). The image capturing application 120 can generate an API call that includes an image and optionally context data. The context data can include location data (e.g., GPS coordinates of the mobile device 110), an identifier of the image capturing application 120 generating the API call, a location of a tap target (e.g., pixel coordinates within the image of a location in the image selected by a user), an identifier for the user, the time and/or date when the image was captured, and/or other appropriate context data related to the image.

[0058] The image capturing application 120 includes a user interface generator 121 that generates user interfaces 125 that present images (and optionally content related to the images as described below). In some implementations, the user interfaces 125 include an icon (or other selectable user interface control) for the image content application 114. For example, the user interface control can be presented when one or more images are being presented by a user interface 125. The user interface control can be for invoking the image content application 114 to present content related to an image that is being presented by the image capturing application 120. When the image capturing application 120 detects a user interaction with (e.g., selection of) the user interface control, the image capturing application 120 can cause the image content application 114 to launch if not already launched. The image capturing application 120 can also generate and send the API call (with the image and optionally context data) to the image content application 114. If the image capturing application 120 is presenting a live image in a viewfinder, user interaction with the user interface control can cause

the image capturing application 120 to capture a still image of the live image and provide the image to the image content application114.

[0059] The user interface generator 116 of the image content application 114 can generate a user interface that presents the image included in the API call and content related to one or more objects recognized in the image. For example, in response to receiving the API call, the object detector 115 of the image content application 114 can determine whether the image depicts one or more objects and, if so, send the image to the content distribution system 150. The image content application 114 can receive content related to the object(s) from the content distribution system 150 and present the content with the image in the user interface 119. In some implementations, the image content application 114 can send the image to the content distribution system 150 without attempting to detect objects in the image.

[0060] The image content application 114 can also send, to the content distribution system 150, the context data received from the image capturing application 120. As described below, the object recognizer 172 can use the context data to recognize objects in the image and/or the content selector 174 can use the context data to select content related to objects recognized in the image.

[0061] In some implementations, the image content application 114 includes the object recognizer 172 (or a scaled down version of the object recognizer 172) and the content selector 174 (or a scaled down version of the object recognizer 172). In this example, the image content application 114 can recognize objects in the image received from the image capturing application 120, select content for each object, and present the content with the image.

[0062] If the image content application 114 is not installed on the mobile device 110, the image capturing application 120 can prompt the user to install the image content application 114 in response to detecting user interaction with the user interface control. For example, the image capturing application 120 can present a link to another application (e.g., an application store) from which the image content application can be downloaded.

[0063] Another API call that can be sent from the image capturing application 120 to the image content application 114 is a request to present the user interface control. For example, the image content application 114 may only present content for images under certain circumstances or may only be capable of supporting certain images or context. This API call can include a request to present the user interface control and context data, e.g., location data for the mobile device 110, an identifier of the image capturing application 120 generating the API call, an identifier for the user, the time and/or date when the image was captured, a language being presented by the image capturing application 120, version of the image capturing application 120, capabilities or model of the mobile device 110, and/or other appropriate context data. For example, the image content application 114 may not support all languages and therefore may not allow the user interface control to be presented within images that include text in unsupported languages.

[0064] The image content application 114 can determine whether the user interface control should be presented based on the context data. The image content application 114 can then generate an API call to the image capturing application 120 with data specifying whether the user interface control can be presented. If the data indicates that the user interface control can be presented, the user interface generator 121 can present the user interface control in the user interface 125.

[0065] The image capturing application 120 also includes an on-device tracker 122. The on-device tracker 122 can operate the same as, or similar to, the on-device tracker 118. In other examples, the mobile device 110 may include an on-device tracker that is shared by the two applications 114 and 120. As described below, the on-device tracker 122 can be used to track the location of objects in a viewfinder of the camera 111 presented in user interfaces 125 generated by the user interface generator 121 so that content received from the content distribution system 150 can be presented with the objects in the viewfinder.

[0066] The image capturing application 120 also includes an API-B 124 that enables the image capturing application 120 to transmit API calls with requests for content to the front-end servers 160 of the content distribution system 150 and to receive content from

the front-end servers 160. The front-end servers 160 include a special purpose API 162 for receiving API calls from the image capturing application 120. The API 162 can specify a particular set of data fields that can be populated with data and included in the API calls.

[0067] The data fields of an API call for a request for content generated by the image capturing application 120 can include a field for an image data defining an image and one or more fields for context data. The context data fields can include respective fields for location data (e.g., GPS coordinates of the mobile device 110), an identifier of the image capturing application 120 generating the API call, a location of a tap target (e.g., pixel coordinates within the image of a location in the image selected by a user), an identifier for the user, a language for the user (e.g., the current language being used on the mobile device 110), the time and/or date when the image was captured, an identifier for a previous application used on the mobile device 110, previous images presented in the viewfinder (e.g., images of the same objects at different angles), information extracted from the previous images (e.g., object tracking information, text identified in the previous images, etc.), depth of view information, and/or other appropriate context data related to the image or the mobile device 110. The API-B 124 can be implemented such that there is no added latency or excessive RAM usage to store content received from the front-end servers 160 of the content distribution system 150.

[0068] The front-end servers 160 can transmit, to the mobile device 110, content related to objects recognized in the image using one or more API calls. In some implementations, the front-end servers 160 send a ranked list of content that is ranked based on relatedness to the object, a confidence that the object in the image is the recognized object, the image capturing application 120 that sent the image, and/or other appropriate data.

[0069] The front-end servers 160 can also transmit, using one or more API calls, data specifying the location in the image where the content is to be presented. For example, the object recognizer 172 can determine the pixel coordinates of a recognized object (and/or other objects) in the image. The content selector 174 can specify that certain content should be presented near the recognized object and certain content can be

presented adjacent to the image or elsewhere in the user interface 125. For content that is to be presented near the object, the front-end servers 160 can send, to the image capturing application 120, data specifying the pixel coordinates of the object in the image. The front-end servers 160 can also send data specifying either the pixel coordinates where the content should be presented, data specifying a maximum number of pixels form the object the content should be presented, or data specifying that the content should be presented adjacent to the object.

[0070] If the image capturing application 120 is presenting the viewfinder of the camera 111 in the user interface 125, the location of the object can move as the user moves the mobile device 110. The image capturing application 120 can interact with the on-device tracker 122 to determine the current location of the object and present the content in the appropriate place near the object in the viewfinder.

[0071] In some implementations, user interaction with content provided by the content distribution system 150 within the image capturing application 120 can cause the mobile device 110 to open an application (e.g., the image content application 114 or another application) of the entity that operates the content distribution system 150 and distributes the image content application 114. For example, the content can include link (e.g., deep links) to applications that present additional content related to the objects or to the provided content.

[0072] In some implementations, the image capturing application 120 includes code or files of the image content application 114 that the image capturing application 120 can use to perform operations that would normally be performed by the image content application 120. For example, the image capturing application 120 can include a library of the image content application 114 and/or the API-B 124 can include code that performs the operations. In this example, application developers can include the functionality and/or ability to communicate with the content distribution system 150 by downloading the library or API-B 124, including the library or API-B 124 in the developer's application and/or configuring the functionality of the library or API-B 124 for use with the application.

[0073] The image capturing application 120 can use the library or API-B 124 to determine when to send an image to the content distribution system 150. For example, the library or API-B 124 can be used to implement the functions of the object recognizer 115 of the image content application 114. In this example, when the image capturing application 120 is going to present an image in the user interface 125, the image capturing application 120 can use the library or API-B 124 to determine whether the image includes an object and, if so, send the image and context data to the content distribution system 150.

[0074] In another example, the image capturing application 120 can use the library to API-B 124 to track objects detected in images and to present content related to the objects near the objects. For example, the library or API-B 124 can implement the functions of the on-device tracker 122. The image capturing application 120 can also use the library or API-B 124 to determine where and how to present the content, e.g., using a set of rules. For example, the rules may specify that certain types of content should be presented near the object and other types of content can be presented below or adjacent to the image.

[0075] In some implementations, the image content application 114 can perform some pre-processing prior to receiving a request (e.g., an API call) from the image capturing application 120. For example, if the image content application 114 detects that the image capturing application 120 is running or is depicting an image, e.g., when the icon is being presented by the image capturing application 120. The image content application 114 can start some services prior to receiving an image from the image capturing application 120. For example, the image content application 114 can initiate its search services and/or initiate a new session prior to the user selecting the icon to launch the image content application 114. This can reduce latency in launching the image content application 114 and presenting the image and results by the image content application 114.

[0076] Another way to reduce latency is for the image content application 114 and the image capturing application 120 to share memory. For example, the applications can use a shared memory approach to send images from the image capturing application

120 to the image content application 114. In this way, the image content application 114 can receive an image from the image capturing application 120 almost instantaneously. For example, the image content application 114 can provide, with an API call, metadata that identifies the image or the memory location of the image.

[0077] In some implementations, the content distribution system 150 can request content from one or more content providers 180. Each content provider 180 can provide content related to objects of a particular type or category. For example, one content provider may specialize in recognizing wines based on images of wine bottles and providing content (e.g., data identifying the wine, information about the wine, images of the wine bottles, and so on) related to the recognized wines.

[0078] If the object recognizer 172 determines that an image includes a particular type of object, but cannot recognize the actual object, the object recognizer 172 can send (via the front-end servers 160) the image to the appropriate content provider 180 with a request for content related to the object. The content provider 180 can recognize the object and provide content related to the object to the front-end servers 160. In turn, the front-end servers 160 can send the content to the image content application 114 or image capturing application 120 that sent the image to the content distribution system 150.

[0079] In some implementations, the image content application 114 can request content from the content providers 180 in a similar manner, e.g., in implementations in which the image content application 114 selects content related to objects. In some implementations, applications, e.g., native applications, provided by the content providers 180 can recognize objects and provide data identifying the objects and/or content related to the objects. Such application can be installed on the mobile device 110. In this example, the image content application 114 can request content from the applications on the mobile device 110 rather than sending a request to remote servers or the content providers 180, which can increase the speed at which content is presented.

[0080] FIG. 2 depicts a sequence 210 of example screen shots 210 and 220 of a user interface 211 that presents content related to objects recognized in an image 213. The

example user interface 211 is generated and presented by a photo sharing application. In the screen shot 210, the user interface 211 presents a viewfinder 212 of a camera which includes a live image 213 of the scene in front of the camera. The user interface 211 also includes a content user interface control 214 for an image content application and share user interface control 215 for sharing the image 213 (e.g., within the photo sharing application, using a messaging application, using a social networking application, or using another appropriate application).

[0081] In this example, the photo sharing application can use a special purpose API to send the image 213 (e.g., a still image captured from the viewfinder 212) to a content distribution system, such as the content distribution system of FIG. 1. The photo sharing application can include an on-device tracker, e.g., as part of the API, that tracks the location of groups of pixels (or detected objects) as they move within or in an out of the viewfinder 212. While waiting for the content, the photo sharing application can use the on-device tracker to track the location of groups of pixels (or objects).

[0082] The photo sharing application can receive, from the content distribution system, content related to one or more objects recognized in the image 213 and data specifying where the content is to be presented within the user interface 211. For example, as shown in the screen shot 220, the user interface 211 has been updated to present a content element 223 that identifies a gray chair in the updated image 222 and a content element 224 that identifies a gray couch in the updated image 222. In this example, the content distribution system has recognized the gray chair and the gray couch and selected captions with text that identify the chair and couch.

[0083] The content elements 223 and 224 are presented over a portion of their respective objects in the user interface. For example, the content distribution system can send data specifying the location where each content element is to be presented in the image 222. The data specifying the location can be at different levels of granularity. For example, the location data may specify that the content element 223 is to be presented within a threshold number of pixels of the gray chair. In another example, the location data may specify that the content element 223 is to be presented over the top of the gray chair or to one side of the gray chair.

[0084] FIG. 3 depicts another sequence 300 of example screen shots 310 and 320 of a user interfaces 311 and 312 that present an image. The example user interface 311 is generated and presented by a photo sharing application. In the screen shot 310, he user interface 311 presents an image 312 (e.g., a still image or live image of a viewfinder), a content user interface control 314 for an image content application and a share user interface control 315 for sharing the image 312.

[0085] When the photo sharing application detects a user interaction with (e.g., section of) the content user interface control 314, the photo sharing application and cause an image content application (e.g., the image content application 114 of FIG. 1 to launch on the same mobile device as the photo sharing application Is executing. The photo sharing application can also send the image 312 and context data to the image content application, e.g., using one or more API calls.

[0086] As shown in the screen shot 320, the image content application can generate a user interface 321 that presents the image 312 and content elements 323-327 that include content related to objects recognized in the image 312. For example, the image content application can receive the image 312 and context data from the photo sharing application and provide the image 312 and context data to a content distribution system (e.g., the content distribution system 150 of FIG. 1). The content distribution system can recognize objects in the image 312 and select the content elements 323-327 or their respective content based on the objects and optionally the context data.

[0087] In this example, the content distribution system has recognized the gray chair and the gray couch and selected a content element 323 that identifies the gray chair and a content element 324 that identifies the gray couch. The content distribution system has also selected the content element 325 that indicates that the furniture in the image 312 is sold by an Example Furniture Store, e.g., by determining that both the gray chair and the gray couch are products sold by the Example Furniture Store.

[0088] The content distribution system has also provided content element 326 and 327 that include links (e.g., application deeplinks) to shop for furniture or view similar furniture, respectively. If the image content application detects user interaction with (e.g., selection of) the content element 326, the image content application can cause a

shopping application to launch and navigate to a page within the application where the user can shop for furniture. If the image content application detects user interaction with (e.g., selection of) the content element 327, the image content application can cause another application (e.g., a web browser) to launch and show information or content (e.g., images) of similar furniture.

[0089] In this example, the content distribution system may have provided data specifying that the content elements 323 and 324 are to be presented near their respective objects and that the content elements 325-327 are to be presented below or adjacent to the image 312. In another example, the content distribution system can provide data specifying which, if any, object each portion of content is related. The content distribution system can also indicate that the content elements 325-327 are related to multiple objects or to the image 312 as a whole. The image content application can then determine to present the content elements 323 and 324 that are related to particular objects near the objects and to present the content elements 325-327 adjacent to the image 312 or another appropriate location.

[0090] FIG. 4 is a flow chart of an example process 400 for presenting content related to objects recognized in an image. Operations of the process 400 can be performed, for example, by one or more data processing apparatus, such as a user device, e.g., the mobile device 110 of FIG. 1. Operations of the process 400 can also be implemented as instructions stored on a non-transitory computer readable medium. Execution of the instructions cause one or more data processing apparatus to perform operations of the process 400.

[0091] The user device captures first image data of a first image displayed in a viewfinder of a camera on the user device (402). The first image can displayed by an application operating on the user device. For example, the application can be present the viewfinder of the camera in a user interface of the application.

[0092] The application sends the first image data to a server system that is external to the user device (404). For example, the application can send the first image data to a server of a content distribution system that recognizes objects depicted in images and provides content related to the objects for presentation by the application. As described

above, the application can include an API that sends the image and optionally context data for the image to a server of a content distribution system.

[0093] The application receives, from the server system, data corresponding to one or more objects depicted by the first image (406). The server system can recognize the one or more objects using one or more object recognition techniques. For each recognized object, the server system can select content related to the object. The server system can also determine a location within the first image to present the content. An example process for recognizing objects, selecting content for objects, and determining a location to present content is illustrated in FIG. 5 and described below. The data provided by the server system can include, for each recognized object, content related to the object and the location within the first image to present the content on the viewfinder.

[0094] After receiving the data corresponding to the one or more objects depicted by the first image, the application can capture subsequent image data of a subsequent image displayed in the viewfinder on the user device (408). The subsequent image data can be different from the first image data. For example, the user device may be moved by a user resulting in the scene in the viewfinder of the camera being different form when the first image data was captured. The subsequent image data can represent the subsequent image being presented in the viewfinder when, or at some time after, the data is received from the server system.

[0095] The application determines, for each object within the viewfinder of a camera for which data was received from the server, a subsequent location based on the location provided by the data received from the server system (410). For example, the application can use an on-device tracker to track the location of objects or groups of pixels within the viewfinder. The application can determine, for each location received from the server system, a current location in the viewfinder that corresponds to the received location if the received location is within the subsequent image. For example, the server system can provide, for an object, a caption for the object and location data specifying that the caption should be presented over a particular portion of the object. The application can use the on-device tracker to determine whether that portion of the

object is presented in the subsequent image and, if so, the location of that portion of the object in the subsequent image.

[0096] For each object within the viewfinder of the camera for which data was received from the server and for which a corresponding subsequent location was determined, content related to the object is presented in the viewfinder (412). The content can be presented in the location specified by the data received from the server system.

[0097] FIG. 5 is a flow chart of an example process 500 for selecting content related to an object recognized in an image received from an application and providing content for presentation by the application. Operations of the process 500 can be performed, for example, by one or more data processing apparatus, such as the content distribution system 150 of FIG. 1. Operations of the process 500 can also be implemented as instructions stored on a non-transitory computer readable medium. Execution of the instructions cause one or more data processing apparatus to perform operations of the process 500.

[0098] Image data representing an image is received from an application (502). For example, an application operating on a user device, e.g., a mobile device can capture an image using a camera of the user device and provide the image data representing the image to a content distribution system using one or more API calls. The application can also provide context data. As described above, the context data can include location data (e.g., GPS coordinates of the mobile device), an identifier of the application generating the API call, a location of a tap target (e.g., pixel coordinates within the image of a location in the image selected by a user), an identifier for the user, the time and/or date when the image was captured, and/or other appropriate context data related to the image.

[0099] One or more objects are recognized in the image (504). For example, as described above, an object recognizer can use a trained machine learning model to recognize objects in image data received from user devices. The object recognizer can also use computer vision and/or other object recognition techniques (e.g., edge

matching, pattern recognition, greyscale matching, gradient matching, etc.) to recognize objects in image data.

[00100] In some implementations, the object recognizer can use the context data to recognize objects in the image. For example, the object recognizer can use the location at which an image was captured to disambiguate objects in the image. In a particular example, if the context data indicates that the image was captured near the Eiffel Tower, the object recognizer can increase a confidence score that an object that appears to be the Eiffel Tower is actually the Eiffel Tower. The object recognizer can then determine what object is depicted in the image based on the increased confidence score for the Eiffel Tower and confidence scores for other objects that the object in the image may be.

[00101] The object recognizer can use the identifier of the application to recognize the object(s) in the image. For example, if the application is directed to a particular topic or category, the object recognizer can increase the confidence score for objects that are related to or directed to the particular topic or category. In a particular example, if the application is directed to bird watching, the object recognizer can increase the confidence scores for birds based on the data identifying the application as a bird watching application.

[00102] In another example, if a particular object (or particular type of object) is commonly recognized in images received from the application (e.g., at least a threshold number times), the object recognizer can increase the confidence score that represents a likelihood that an object depicted in an image received from the application is the particular object (or particular type of object).

[00103] Content is selected for the recognized object(s) (506). For example, as described above, the content can include information related to the object (e.g., text that includes the name of the object and/or facts about the object), visual treatments (e.g., other images or videos of the object or of related objects), links to resources related to the object (e.g., links to web pages or application pages at which the user can purchase the object or view additional information about the object), or experiences related to the

object (augmented reality video, playing music in response to recognizing a singer or poster of a singer), and/or other appropriate content.

[00104] In some implementations, the content is selected based on the context data. For example, the content may be selected based on the location at which the image was captured. In a particular example, if the location corresponds to a tourist destination or museum, content that includes a description or additional information about an object may be selected instead of additional images of the object. If the image was captured at a location that is far from the location of the object (e.g., an image of the Eiffel Tower is captured in the U.S.), additional images of the object can be selected.

[00105] Content can be selected based on the identifier of the application. For example, the entity that develops and distributes the application can specify the types of content provided for presentation by the application. In another example, historical data can show that users of a particular application interact more frequently with certain types of content than other types of content or that the users prefer certain types of content.

[00106] The content can be ranked and the ranking can be provided to the application. For example, the content for each object can be ranked separately if there are multiple objects recognized in the image. The content for each object can be ranked based on the relevancy of the content to the object, a number of user interactions the content has received when provided with images of the object (and/or when presented by the application), and/or other appropriate data. The application can select, from the ranked content, which content to present with the image.

[00107] A location to present each portion of content is identified (508). For example, content that identifies an object, e.g., a caption that identifies the object, may be presented near (or partially over) the object in an image that includes the object. Other content, e.g., captions that include links to other content, can be presented adjacent to the image or another appropriate location that is not necessarily near the object. For example, this type of content may be presented under the image in a designated area.

[00108] The location for each portion of content can be determined based on the ranking of the content identified for the object. For example, higher ranked content can be presented near (or partially over) the object and lower ranked content can be presented adjacent to the image.

[00109] The location for each portion of content can be a group of pixels, e.g., pixel coordinates for a group of pixels at or near the object. For example, the content distribution system can determine that there are no objects depicted in the image to the left of a recognized object and that this area is large enough for the portion of content. In response, the content distribution system can select this location for presenting the portion of content. The content distribution system can specify, as the location data, the pixel coordinates of the location in the image. In another example, the content distribution system can specify, as the location data, the number of pixels in one or more directions from a particular part of the recognized object. For example, the location data may specify that the portion of content is to be presented ten pixels above and ten pixels to the left of a top left pixel of the object.

[00110] The content and data specifying the location to present the content is sent to the application (510). For example, a content distribution system can send the content and the location data using one or more API calls.

[00111] The application can present the content at the specified location (512). For example, as described above, the application can have an on-device tracker (or access to an on-device tracker) that tracks the location of groups of pixels or objects. When the content is received, the application can determine, based on the on-device tracker's location of the groups of pixels or objects, a current location of each object and present the content based on the current location. For example, if caption is provided for a presentation in an overlay next to the object, the application can determine the current location of that object and present the caption next to the object.

[00112] If the location data specifies pixel coordinates for the content, the application can interact with the on-device tracker to determine the current location of the pixel coordinates. For example, the on-device tracker can determine what object or group of pixels was presented at the pixel coordinates in the image send to the content

distribution system. The on-device tracker can then determine the current location of the group of pixels in the live (or updated) image.

[00113] FIG. 6 is a flow chart of an example process 600 for launching an application to present content related to an object recognized in an image captured by a different application. Operations of the process 600 can be performed, for example, by one or more data processing apparatus, such as the mobile device 110 of FIG. 1. Operations of the process 600 can also be implemented as instructions stored on a non-transitory computer readable medium. Execution of the instructions cause one or more data processing apparatus to perform operations of the process 600.

[00114] Interaction with an interface control is detected at a first application (602). The interface control can be a control that requests opening of a second application different from the first application and for presentation of content related to an image (e.g., a live image of a viewfinder) being presented by the first application. For example, the first application can present the user interface control for invoking the second application.

[00115] An image is captured in response to the user interaction (604). For example, the first application can capture a still image of the live image in response to detecting the user interaction.

[00116] Context data is identified (606). For example, the first application can identify the context data in response to detecting the user interaction. As described above, the context data can include location data (e.g., GPS coordinates of the mobile device), an identifier of the application generating the API call, a location of a tap target (e.g., pixel coordinates within the image of a location in the image selected by a user), an identifier for the user, the time and/or date when the image was captured, and/or other appropriate context data related to the image.

[00117] The second application is launched in response to the user interaction (608). For example, the mobile device can launch the second application in response to the user interaction. If the second application is not installed on the mobile device, the mobile device can prompt the user to download the second application.

[00118] The first application provides the image and the context data to the second application (610). The second application can obtain content related to objects depicted in the image, optionally using the context data. For example, the second application can provide the image and context data to a content distribution system with a request for content related to objects recognized in the image. The content distribution system can then recognize the objects, select content related to the objects, and provide the content to the second application. In another example, the second application can recognize the objects in the image and select content related to the objects in a similar manner as the content distribution system.

[00119] The second application presents the image and the content in an interface of the second application (612). For example, the second application can present the content for each object near the object in the image, e.g., using overlays that include the content as described above.

[00120] In some implementations, the second application presents the image and content when the second application opens. For example, the second application can wait to present any content in the user interface until content for the objects in the image has been received.

[00121] FIG. 7 is a flow chart of an example process 700 for obtaining content related to an object recognized in an image and presenting the content. Operations of the process 700 can be performed, for example, by one or more data processing apparatus, such as the content distribution system 150 and/or the mobile device 110 of FIG. 1. Operations of the process 700 can also be implemented as instructions stored on a non-transitory computer readable medium. Execution of the instructions cause one or more data processing apparatus to perform operations of the process 700.

[00122] Image data representing an image is received (702). For example, the image data can be received by an image content application of a mobile device, e.g., from another application such as an image capturing application different from the image content application. In another example the image data can be received by a content distribution system, e.g., from an image content application or another application.

[00123] One or more objects are detected in the image (704). For example, the objects can be detected using vision analysis, machine learning models, coarse classifiers, or other appropriate techniques as described above.

[00124] The image data is sent to a content provider with a request for content related to the object(s) (706). In some implementations, the image data is sent to one or more content providers in response to detecting an object, but not being able to recognize the object. For example, if the content distribution system (or the image content application) determines that a type or category of object may be depicted in the image, the content distribution system (or the image content application) can send the image data to a content provider that recognizes objects of that type or category.

[00125] The content distribution system (or image content application) can also provide data specifying the location of the object in the image so that the content provider can focus its object recognition on that portion of the image. In another example, the content distribution system (or image content application) provides only the portion of the image that includes the object.

[00126] The content provider can attempt to recognize the object in the image. If successful, the content provider can provide data identifying the object and/or content related to the object, e.g., images of the object, information about the object, links to additional information about the object, and so on). The content distribution system (or image content application) can also identify (e.g., select) content based on the identity of the object. If unsuccessful, the content provider can provide data indicating that the content provider was not able to recognize the object.

[00127] The content is received from the content provider (708). In some cases, multiple types of objects may be detected in an image. In these cases, the image data can be provided to multiple different content providers, e.g., one or more for each type of object. Each content provider can provide content related to the object(s) recognized by the provider.

[00128] The content is presented (710). If the content is received by the image content application, the image content application can present the content (e.g., with the image as described above) or provide the content to the image capturing application for

presentation (e.g., if the image data was received from the image capturing application. If the content is received by the content distribution system, the content distribution system can provide the content to the application from which the image data was received.

[00129] Embodiments of the subject matter and the operations described in this specification can be implemented in digital electronic circuitry, or in computer software, firmware, or hardware, including the structures disclosed in this specification and their structural equivalents, or in combinations of one or more of them. Embodiments of the subject matter described in this specification can be implemented as one or more computer programs, i.e., one or more modules of computer program instructions, encoded on computer storage medium for execution by, or to control the operation of, data processing apparatus. Alternatively or in addition, the program instructions can be encoded on an artificially generated propagated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus for execution by a data processing apparatus. A computer storage medium can be, or be included in, a computer-readable storage device, a computer-readable storage substrate, a random or serial access memory array or device, or a combination of one or more of them. Moreover, while a computer storage medium is not a propagated signal, a computer storage medium can be a source or destination of computer program instructions encoded in an artificially generated propagated signal. The computer storage medium can also be, or be included in, one or more separate physical components or media (e.g., multiple CDs, disks, or other storage devices).

[00130] The operations described in this specification can be implemented as operations performed by a data processing apparatus on data stored on one or more computer-readable storage devices or received from other sources.

[00131] The term “data processing apparatus” encompasses all kinds of apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, a system on a chip, or multiple ones, or combinations, of the foregoing. The apparatus can include special purpose logic

circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit). The apparatus can also include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, a cross-platform runtime environment, a virtual machine, or a combination of one or more of them. The apparatus and execution environment can realize various different computing model infrastructures, such as web services, distributed computing and grid computing infrastructures.

[00132] A computer program (also known as a program, software, software application, script, or code) can be written in any form of programming language, including compiled or interpreted languages, declarative or procedural languages, and it can be deployed in any form, including as a stand alone program or as a module, component, subroutine, object, or other unit suitable for use in a computing environment. A computer program may, but need not, correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program in question, or in multiple coordinated files (e.g., files that store one or more modules, sub programs, or portions of code). A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[00133] The processes and logic flows described in this specification can be performed by one or more programmable processors executing one or more computer programs to perform actions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit).

[00134] Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital computer. Generally, a processor will receive

instructions and data from a read only memory or a random access memory or both. The essential elements of a computer are a processor for performing actions in accordance with instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Moreover, a computer can be embedded in another device, e.g., a mobile telephone, a personal digital assistant (PDA), a mobile audio or video player, a game console, a Global Positioning System (GPS) receiver, or a portable storage device (e.g., a universal serial bus (USB) flash drive), to name just a few. Devices suitable for storing computer program instructions and data include all forms of non volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and CD ROM and DVD-ROM disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

[00135] To provide for interaction with a user, embodiments of the subject matter described in this specification can be implemented on a computer having a display device, e.g., a CRT (cathode ray tube) or LCD (liquid crystal display) monitor, for displaying information to the user and a keyboard and a pointing device, e.g., a mouse or a trackball, by which the user can provide input to the computer. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, or tactile input. In addition, a computer can interact with a user by sending documents to and receiving documents from a device that is used by the user; for example, by sending web pages to a web browser on a user's client device in response to requests received from the web browser.

[00136] Embodiments of the subject matter described in this specification can be implemented in a computing system that includes a back end component, e.g., as a data server, or that includes a middleware component, e.g., an application server, or

that includes a front end component, e.g., a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the subject matter described in this specification, or any combination of one or more such back end, middleware, or front end components. The components of the system can be interconnected by any form or medium of digital data communication, e.g., a communication network. Examples of communication networks include a local area network ("LAN") and a wide area network ("WAN"), an inter-network (e.g., the Internet), and peer-to-peer networks (e.g., ad hoc peer-to-peer networks).

[00137] The computing system can include clients and servers. A client and server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other. In some embodiments, a server transmits data (e.g., an HTML page) to a client device (e.g., for purposes of displaying data to and receiving user input from a user interacting with the client device). Data generated at the client device (e.g., a result of the user interaction) can be received from the client device at the server.

[00138] While this specification contains many specific implementation details, these should not be construed as limitations on the scope of any inventions or of what may be claimed, but rather as descriptions of features specific to particular embodiments of particular inventions. Certain features that are described in this specification in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[00139] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system components in the embodiments described above should not be understood as requiring such separation in all embodiments, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products.

[00140] Thus, particular embodiments of the subject matter have been described. Other embodiments are within the scope of the following claims. In some cases, the actions recited in the claims can be performed in a different order and still achieve desirable results. In addition, the processes depicted in the accompanying figures do not necessarily require the particular order shown, or sequential order, to achieve desirable results. In certain implementations, multitasking and parallel processing may be advantageous.

CLAIMS

What is claimed is:

1. A method performed by a user device, the method comprising:

capturing, by the user device, first image data of a first image displayed in a viewfinder of a camera on the user device, the first image displayed by an application operating on the user device;

sending, from the application, the first image data to a server system that is external to the user device;

receiving, by the application and from the server system, data corresponding to one or more objects depicted by the first image, wherein the data includes, for each object:

content related to the object; and

a location within the first image to present the content on the viewfinder; and

after receiving the data corresponding to the one or more objects depicted by the first image, capturing subsequent image data of a subsequent image displayed in the viewfinder on the user device, the subsequent image data being different from the first image data;

determining, by the application, for each object within the viewfinder of a camera for which data was received from the server:

a subsequent location based on the location provided by the data received from the server system, wherein the subsequent location is different from the location provided by the data received from the server system, and corresponds to the object; and

for each object within the viewfinder of the camera for which data was received from the server and for which a corresponding subsequent location was determined, presenting, in the viewfinder, content related to the object.

2. The method of claim 1, wherein sending the first image data to a server system that is external to the user device comprises:

detecting, by the user device, the presence of one or more objects in the image data;

wherein sending the first image data to a server system that is external to the user device is based upon the detecting.

3. The method of claim 2, wherein detecting the presence of one or more objects in the image data comprises processing the image data using a coarse classifier.

4. The method of claim 3, wherein sending the first image data comprises:

selecting image data of the first image data in which the presence of one or more objects is determined; and

transmitting the selected image data.

5. The method of claim 1, wherein the application comprises an application programming interface (API) that determines the subsequent location for each object in the viewfinder for which data was received from the server and causes the application to present the content for each object in the viewfinder.

6. The method of claim 5, wherein the API includes an on-device tracker that tracks location of objects depicted in the viewfinder of the camera.

7. The method of claim 5, wherein:

the API is configured to obtain, from the user device, context data indicative of a context in which the first image was captured, the context data comprising at least one of a geographic location of the client device at the time the image was captured or data identifying the application; and

the server selects the content for each of the one or more objects based on the context data.

8. The method of claim 7, wherein the server system uses the context data to disambiguate, for at least one object depicted by the image, between multiple potential objects that match the at least one object.

9. The method of claim 1, wherein the server system:

determines, for a given object depicted by the first image, a category of the object;

in response to determining the category of the object, sending at least a portion of the first image that depicts the given object to a content provider that recognizes objects of the determined category and that provides content related to objects of the determined category;

receiving, from the content provider, content related to the given object; and

including the content for the given object in the data corresponding to one or more objects depicted by the first image provided to the application.

10. The method of claim 1, further comprising:

presenting, by the application, an interface control for a second application that presents content related to objects depicted in images;

detecting, by the application, user interaction with the interface control and, in response:

causing the user device to launch the second application;

capturing additional image data of an additional image displayed in a viewfinder of a camera; and

providing the additional image data to the second application, wherein the second application obtains content related to one or more objects depicted by the additional image and presents the additional image and the content related to the one or more objects depicted by the additional image.

11. The method of claim 1, wherein the data corresponding to one or more objects depicted by the first image is generated by an object recognizer.

12. The method of claim 11, wherein the object recognizer comprises a machine learning model to recognize objects in image data received from the mobile device.

13. The method of claim 1, wherein:

the server system compares at least one of a digital signature of the application or a package name of the application to a whitelist; and

provides the data corresponding to the one or more objects in response to the digital signature or package name matching a corresponding signature or corresponding package name in the whitelist.

14. A system comprising:

one or more data processing apparatus of a user device; and

a memory storage apparatus in data communication with the one or more data processing apparatus, the memory storage apparatus storing instructions executable by the one or more data processing apparatus and that upon such execution cause the one or more data processing apparatus to perform operations comprising:

capturing, by the user device, first image data of a first image displayed in a viewfinder of a camera on the user device, the first image displayed by an application operating on the user device;

sending, from the application, the first image data to a server system that is external to the user device;

receiving, by the application and from the server system, data corresponding to one or more objects depicted by the first image, wherein the data includes, for each object:

content related to the object; and

a location within the first image to present the content on the viewfinder; and

after receiving the data corresponding to the one or more objects depicted by the first image, capturing subsequent image data of a subsequent image displayed in the viewfinder on the user device, the subsequent image data being different from the first image data;

determining, by the application, for each object within the viewfinder of a camera for which data was received from the server:

a subsequent location based on the location provided by the data received from the server system, wherein the subsequent location is different from the location provided by the data received from the server system, and corresponds to the object; and

for each object within the viewfinder of the camera for which data was received from the server and for which a corresponding subsequent location was determined, presenting, in the viewfinder, content related to the object.

15. The system of claim 14, wherein sending the first image data to a server system that is external to the user device comprises:

detecting, by the user device, the presence of one or more objects in the image data;

wherein sending the first image data to a server system that is external to the user device is based upon the detecting.

16. The system of claim 15, wherein detecting the presence of one or more objects in the image data comprises processing the image data using a coarse classifier.

17. The system of claim 16, wherein sending the first image data comprises:

selecting image data of the first image data in which the presence of one or more objects is determined; and

transmitting the selected image data.

18. The system of claim 14, wherein the application comprises an application programming interface (API) that determines the subsequent location for each object in the viewfinder for which data was received from the server and causes the application to present the content for each object in the viewfinder.

19. The system of claim 18, wherein the API includes an on-device tracker that tracks location of objects depicted in the viewfinder of the camera.

20. The system of claim 8, wherein:

the API is configured to obtain, from the user device, context data indicative of a context in which the first image was captured, the context data comprising at least one of a geographic location of the client device at the time the image was captured or data identifying the application; and

the server selects the content for each of the one or more objects based on the context data.

21. The system of claim 20, wherein the server system uses the context data to disambiguate, for at least one object depicted by the image, between multiple potential objects that match the at least one object.

22. The system of claim 14, wherein the server system:

determines, for a given object depicted by the first image, a category of the object;

in response to determining the category of the object, sending at least a portion of the first image that depicts the given object to a content provider that recognizes objects of the determined category and that provides content related to objects of the determined category;

receiving, from the content provider, content related to the given object; and

including the content for the given object in the data corresponding to one or more objects depicted by the first image provided to the application.

23. The system of claim 14, wherein the operations comprise:

presenting, by the application, an interface control for a second application that presents content related to objects depicted in images;

detecting, by the application, user interaction with the interface control and, in response:

causing the user device to launch the second application;

capturing additional image data of an additional image displayed in a viewfinder of a camera; and

providing the additional image data to the second application, wherein the second application obtains content related to one or more objects depicted by the additional image and presents the additional image and the content related to the one or more objects depicted by the additional image.

24. The system of claim 14, wherein the data corresponding to one or more objects depicted by the first image is generated by an object recognizer.

25. The system of claim 24, wherein the object recognizer comprises a machine learning model to recognize objects in image data received from the mobile device.

26. A non-transitory computer storage medium encoded with a computer program, the program comprising instructions that when executed by data processing apparatus cause the data processing apparatus to perform operations comprising:

capturing, by a user device, first image data of a first image displayed in a viewfinder of a camera on the user device, the first image displayed by an application operating on the user device;

sending, from the application, the first image data to a server system that is external to the user device;

receiving, by the application and from the server system, data corresponding to one or more objects depicted by the first image, wherein the data includes, for each object:

content related to the object; and

a location within the first image to present the content on the viewfinder; and

after receiving the data corresponding to the one or more objects depicted by the first image, capturing subsequent image data of a subsequent image displayed in the

viewfinder on the user device, the subsequent image data being different from the first image data;

determining, by the application, for each object within the viewfinder of a camera for which data was received from the server:

a subsequent location based on the location provided by the data received from the server system, wherein the subsequent location is different from the location provided by the data received from the server system, and corresponds to the object; and

for each object within the viewfinder of the camera for which data was received from the server and for which a corresponding subsequent location was determined, presenting, in the viewfinder, content related to the object.

27. The non-transitory computer storage medium of claim 26, wherein sending the first image data to a server system that is external to the user device comprises:

detecting, by the user device, the presence of one or more objects in the image data;

wherein sending the first image data to a server system that is external to the user device is based upon the detecting.

28. The non-transitory computer storage medium of claim 27, wherein detecting the presence of one or more objects in the image data comprises processing the image data using a coarse classifier.

29. The non-transitory computer storage medium of claim 28, wherein sending the first image data comprises:

selecting image data of the first image data in which the presence of one or more objects is determined; and

transmitting the selected image data.

30. The non-transitory computer storage medium of claim 26, wherein the application comprises an application programming interface (API) that determines the subsequent

location for each object in the viewfinder for which data was received from the server and causes the application to present the content for each object in the viewfinder.

31. The non-transitory computer storage medium of claim 30, wherein the API includes an on-device tracker that tracks location of objects depicted in the viewfinder of the camera.

32. The non-transitory computer storage medium of claim 30, wherein:

the API is configured to obtain, from the user device, context data indicative of a context in which the first image was captured, the context data comprising at least one of a geographic location of the client device at the time the image was captured or data identifying the application; and

the server selects the content for each of the one or more objects based on the context data.

33. The non-transitory computer storage medium of claim 32, wherein the server system uses the context data to disambiguate, for at least one object depicted by the image, between multiple potential objects that match the at least one object.

34. The non-transitory computer storage medium of claim 26, wherein the server system:

determines, for a given object depicted by the first image, a category of the object;

in response to determining the category of the object, sending at least a portion of the first image that depicts the given object to a content provider that recognizes objects of the determined category and that provides content related to objects of the determined category;

receiving, from the content provider, content related to the given object; and

including the content for the given object in the data corresponding to one or more objects depicted by the first image provided to the application.

35. The non-transitory computer storage medium of claim 26, wherein the operations comprise:

presenting, by the application, an interface control for a second application that presents content related to objects depicted in images;

detecting, by the application, user interaction with the interface control and, in response:

causing the user device to launch the second application;

capturing additional image data of an additional image displayed in a viewfinder of a camera; and

providing the additional image data to the second application, wherein the second application obtains content related to one or more objects depicted by the additional image and presents the additional image and the content related to the one or more objects depicted by the additional image.

36. The non-transitory computer storage medium of claim 26, wherein the data corresponding to one or more objects depicted by the first image is generated by an object recognizer.

37. The non-transitory computer storage medium of claim 36, wherein the object recognizer comprises a machine learning model to recognize objects in image data received from the mobile device.

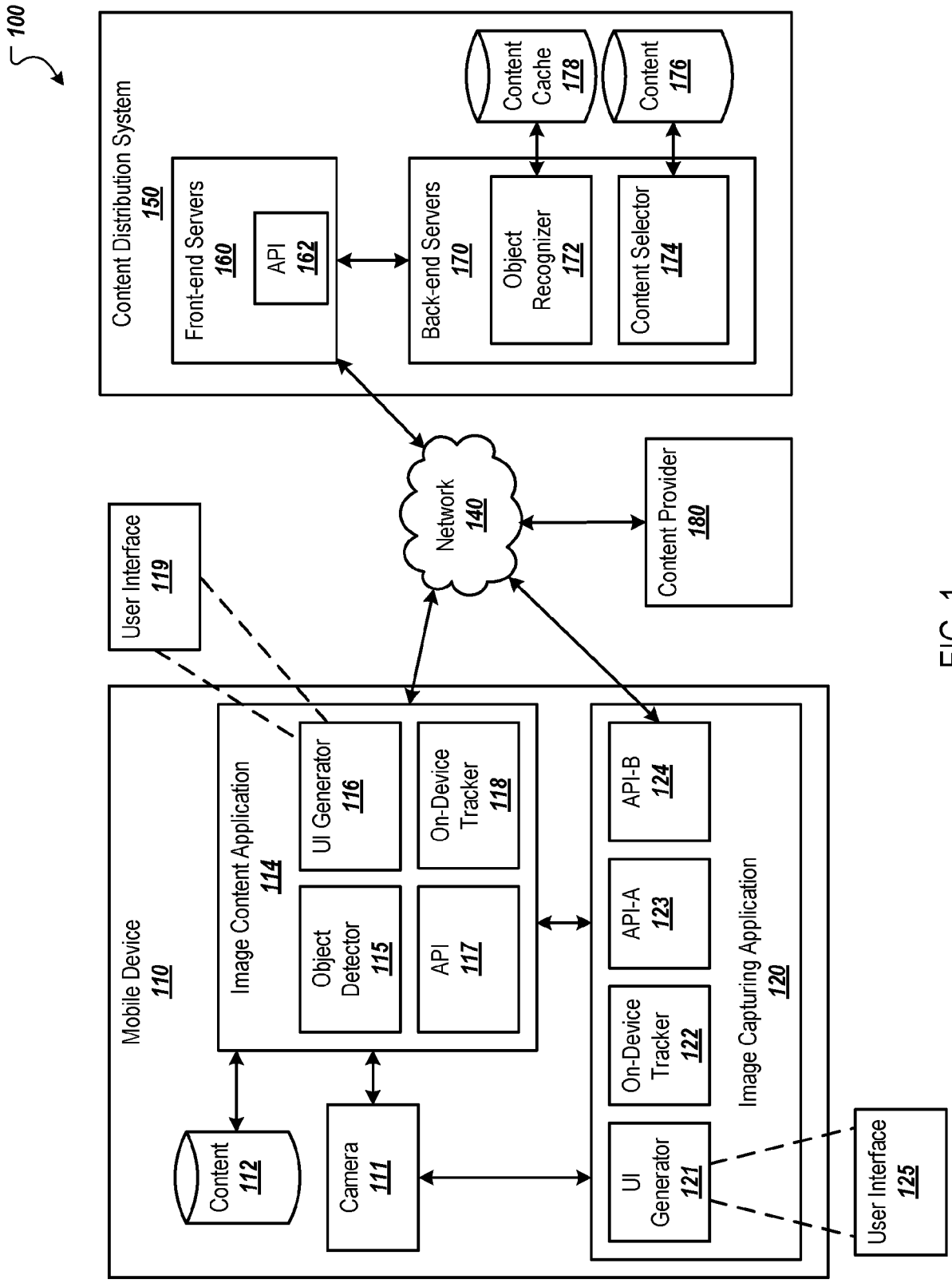
100
Content Distribution System
150
Front-end Servers
160
API
162
Back-end Servers
170
Object Recognizer
172
Content Selector
174
Content Cache
178
Content
176
Network
140
Content Provider
180
User Interface
119
Mobile Device
110
Image Content Application
114
UI Generator
116
On-Device Tracker
118
Object Detector
115
API
117
Content
112
Camera
111
Image Capturing Application
120
API-B
124
API-A
123
On-Device Tracker
122
UI Generator
121
User Interface
125

FIG. 1

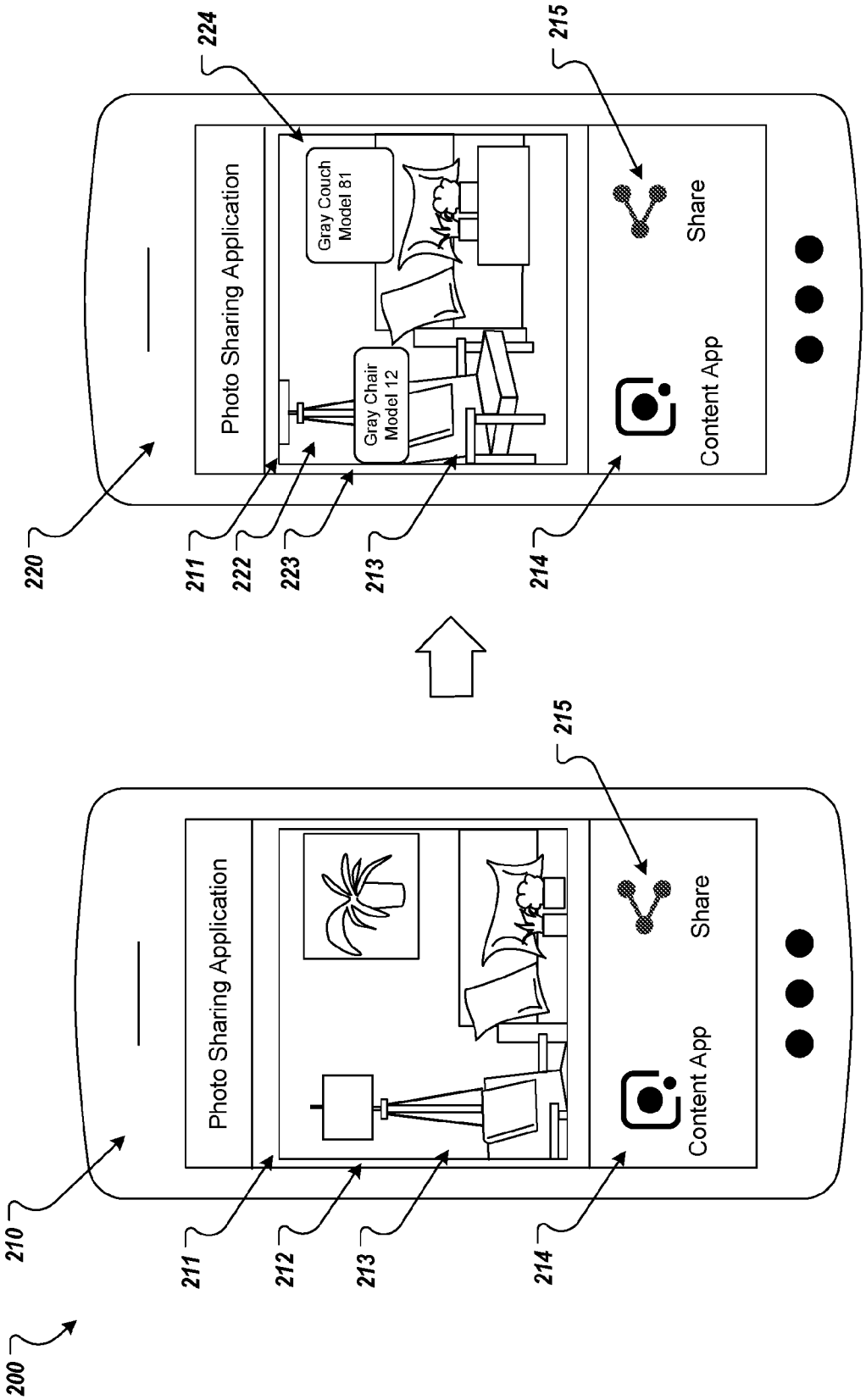
200
210
211
212
213
214
215
Photo Sharing Application
Content App
Share
220
211
222
223
213
214
224
215
Gray Couch Model 81
Gray Chair Model 12
Photo Sharing Application
Content App
Share

FIG. 2

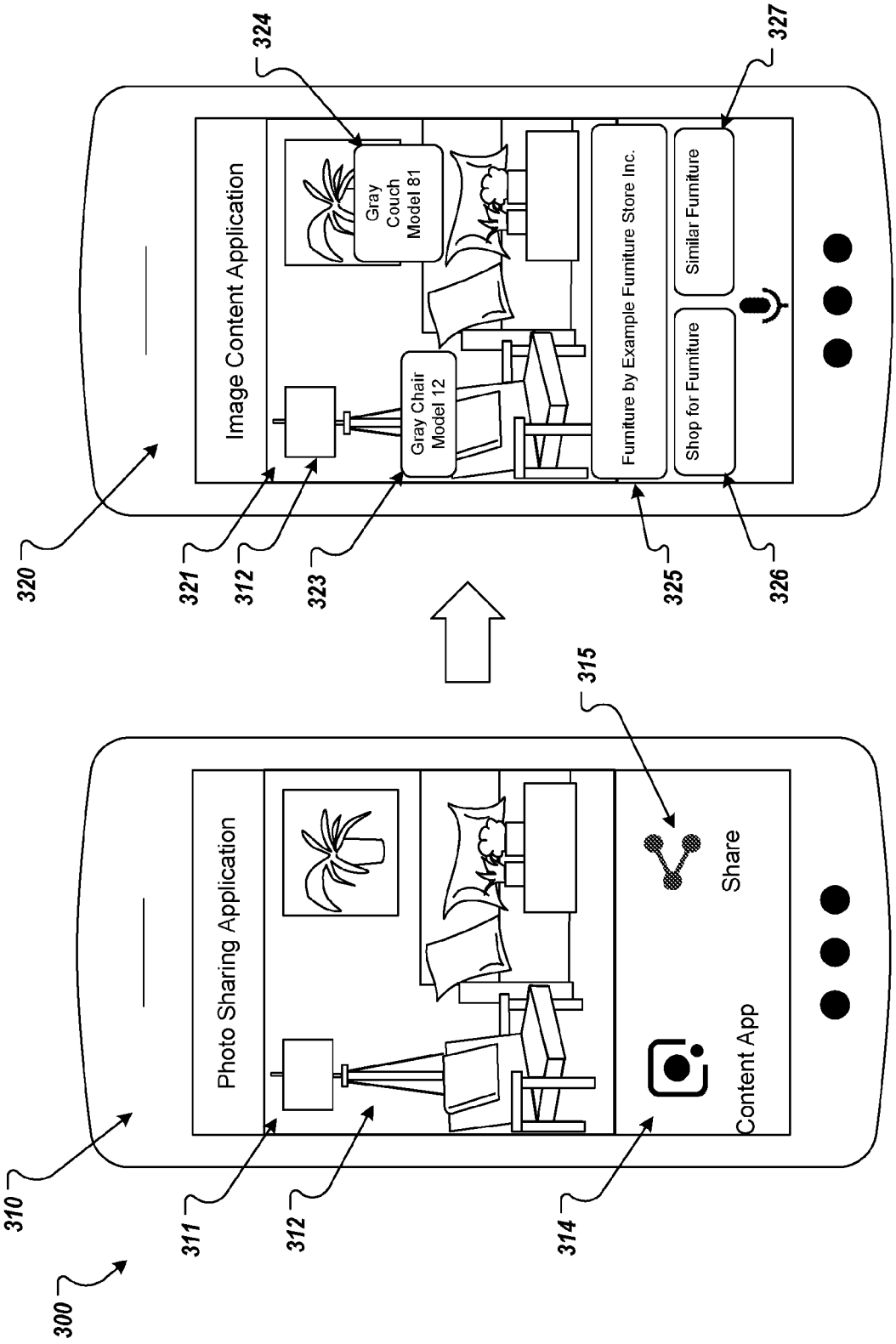
Photo Sharing Application
Share
Content App
Image Content Application
Gray Couch Model 81
Gray Chair Model 12
Furniture by Example Furniture Store Inc.
Shop for Furniture
Similar Furniture
300
310
311
312
314
315
320
321
312
323
324
325
326
327

FIG. 3

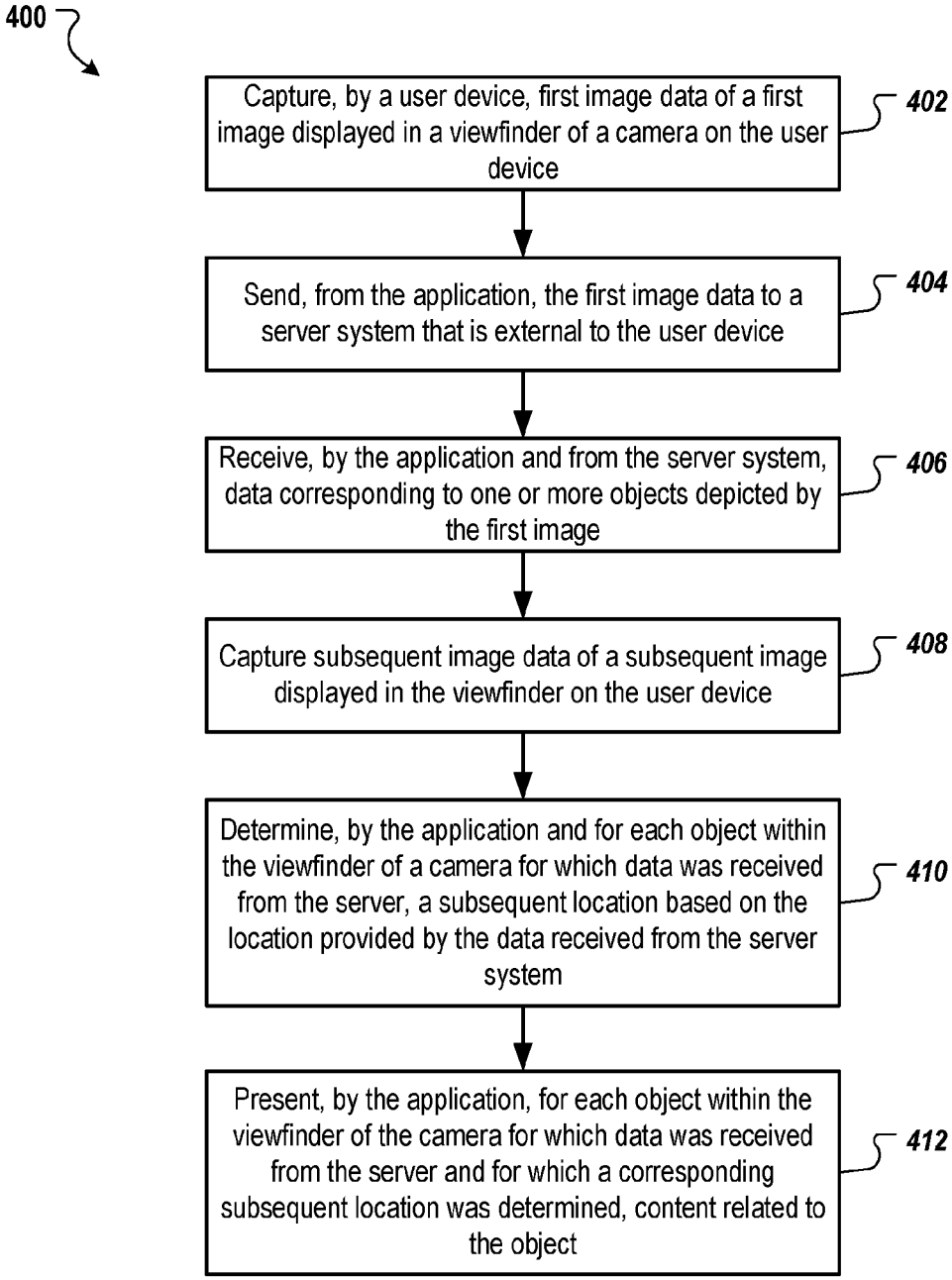
400
Capture, by a user device, first image data of a first image displayed in a viewfinder of a camera on the user device
402
Send, from the application, the first image data to a server system that is external to the user device
404
Receive, by the application and from the server system, data corresponding to one or more objects depicted by the first image
406
Capture subsequent image data of a subsequent image displayed in the viewfinder on the user device
408
Determine, by the application and for each object within the viewfinder of a camera for which data was received from the server, a subsequent location based on the location provided by the data received from the server system
410
Present, by the application, for each object within the viewfinder of the camera for which data was received from the server and for which a corresponding subsequent location was determined, content related to the object
412

FIG. 4

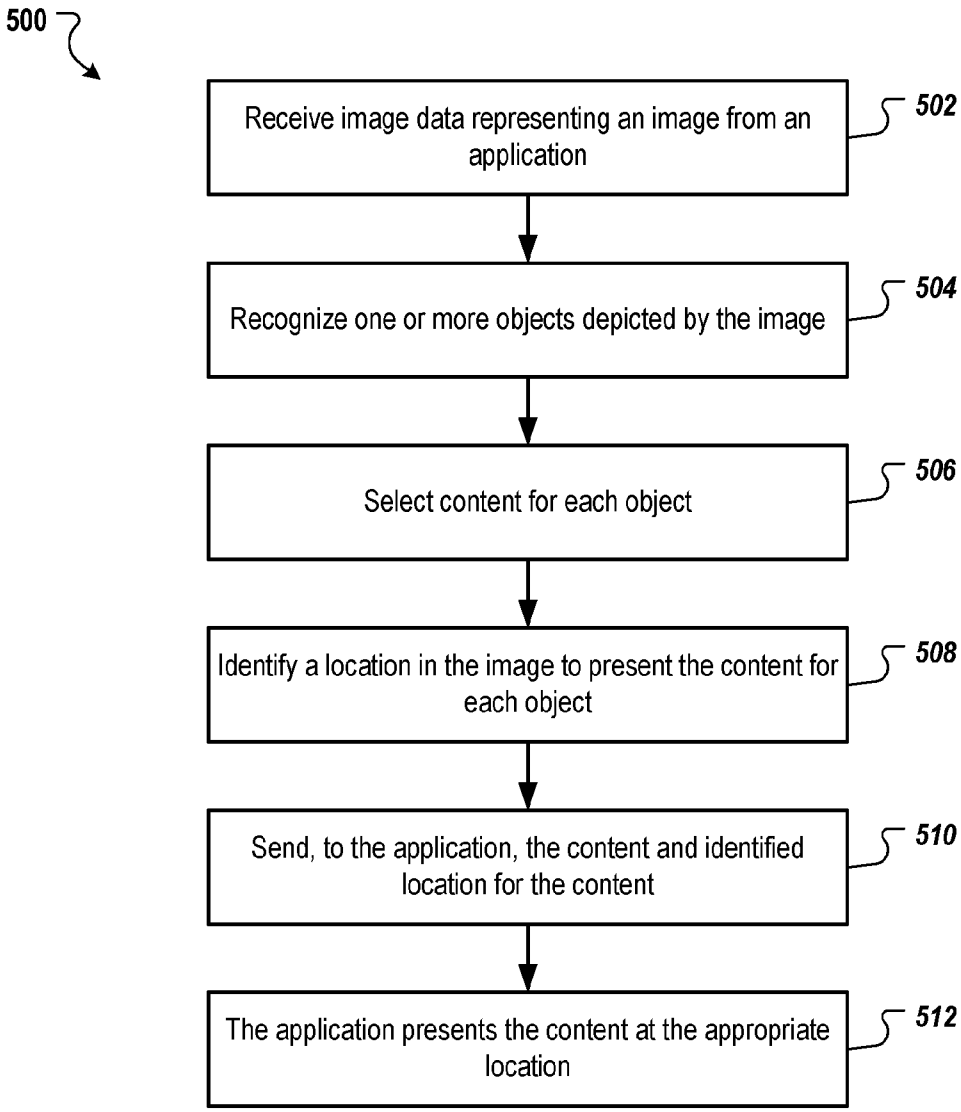
500
Receive image data representing an image from an application
502
Recognize one or more objects depicted by the image
504
Select content for each object
506
Identify a location in the image to present the content for each object
508
Send, to the application, the content and identified location for the content
510
The application presents the content at the appropriate location
512

FIG. 5

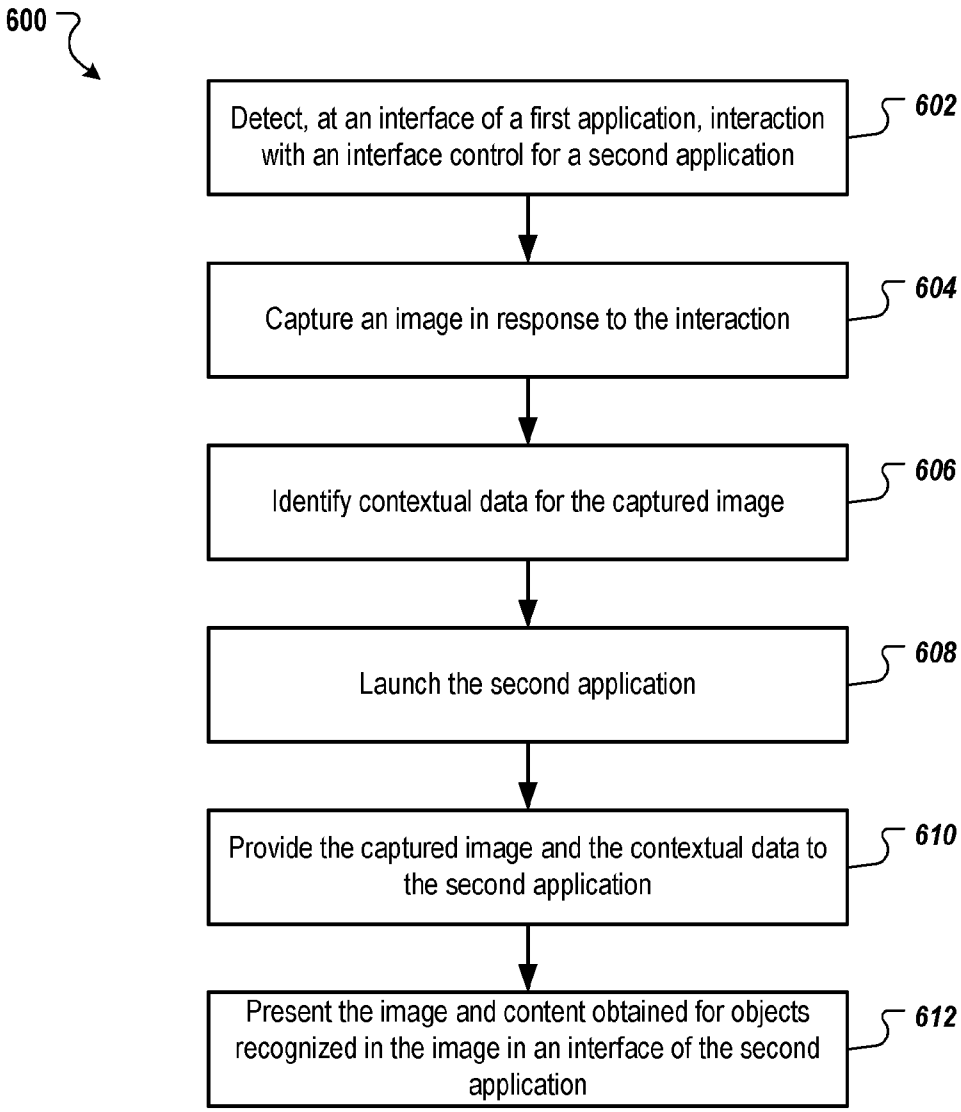
600
Detect, at an interface of a first application, interaction with an interface control for a second application
602
Capture an image in response to the interaction
604
Identify contextual data for the captured image
606
Launch the second application
608
Provide the captured image and the contextual data to the second application
610
Present the image and content obtained for objects recognized in the image in an interface of the second application
612

FIG. 6

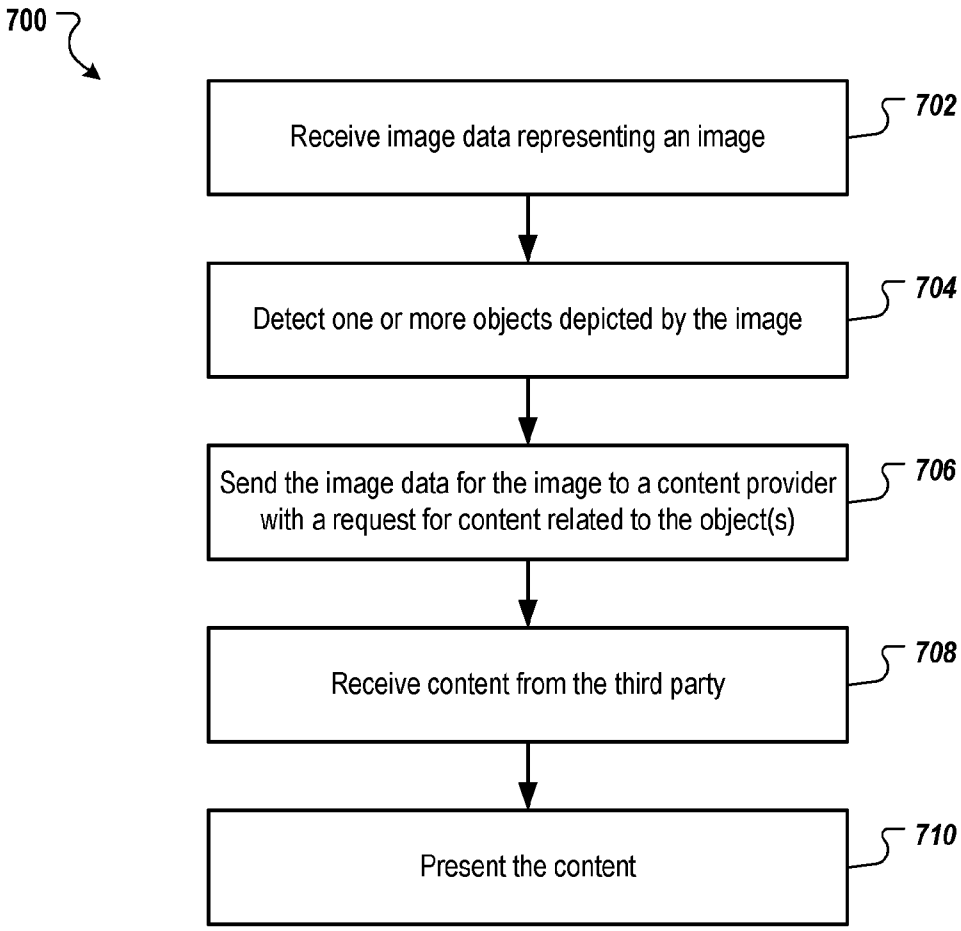
700
Receive image data representing an image
702
Detect one or more objects depicted by the image
704
Send the image data for the image to a content provider with a request for content related to the object(s)
706
Receive content from the third party
708
Present the content
710

FIG. 7

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2019/063632

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06K9/00
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06K

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2013/114849 A1 (PENGELLY ROBERT [US] ET AL) 9 May 2013 (2013-05-09) paragraph [0041] - paragraph [0048]; figures 5, 6 ----- -/--	1-7, 9-12, 14-20, 22-32, 34-37

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
6 March 2020	16/03/2020

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

2

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

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Tech Insider: "Google Showed Off A Camera App That Identifies Real-World Objects", YouTube, 17 May 2017 (2017-05-17), pages 1-2, XP054980289, Retrieved from the Internet: URL:https://www.youtube.com/watch?v=bwYkQzlF5bk&list=PLfUTGA4z85UHMmuL0niZK7WcDquqwcgBn&index=7&t=0s [retrieved on 2020-03-06] the whole document -----	1,14,26
A	US 9 685 004 B2 (APPLE INC [US]) 20 June 2017 (2017-06-20) figures 2-5 -----	1,14,26

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2019/063632

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2013114849 A1	09-05-2013	US 2013114849 A1	09-05-2013
		US 2016358030 A1	08-12-2016
US 9685004 B2	20-06-2017	EP 2901413 A1	05-08-2015
		US 2015235424 A1	20-08-2015
		US 2017301141 A1	19-10-2017
		WO 2014048497 A1	03-04-2014