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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
28 February 2013 (28.02.2013)

WIPO | PCT

(10) International Publication Number
WO 2013/028279 A1

(51) International Patent Classification:
G06K 9/22 (2006.01) *G06K 9/00* (2006.01)

(21) International Application Number:
PCT/US2012/046816

(22) International Filing Date:
14 July 2012 (14.07.2012)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/525,628 19 August 2011 (19.08.2011) US
13/549,339 13 July 2012 (13.07.2012) US

(71) Applicant *(for all designated States except US)*: **QUALCOMM Incorporated** [US/US]; ATTN: International IP Administration, 5775 Morehouse Drive, San Diego, California 92121-1714 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **KULKARNI, Tejas Dattatraya** [IN/US]; 6650 Wisteria Way, San Jose, California 95129 (US). **LIU, Bocong** [CN/US]; 5775 Morehouse Drive, San Diego, California 92121 (US). **NANDWANI, Ankur B** [IN/US]; 5775 Morehouse Drive, San Diego, California 92121 (US). **TASESKI, Justin E** [US/US]; 5775 Morehouse Drive, San Diego, California 92121 (US). **YULE, Benjamin J** [US/US]; 5775 Morehouse Drive, San Diego, California 92121 (US). **KALEAS, Dimosthenis** [US/US]; 5775 Morehouse Drive, San Diego, California 92121 (US). **FORUTANPOUR, Babak** [US/US]; 5775 Morehouse Drive, San Diego, California 92121 (US).

(74) Agent: **SURYADEVARA, Omkar K.**; SILICON VALLEY PATENT GROUP LLP, 4010 Moorpark Avenue, Suite 210, San Jose, California 95117 (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, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, 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.

(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, 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, ML, MR, NE, SN, TD, TG).

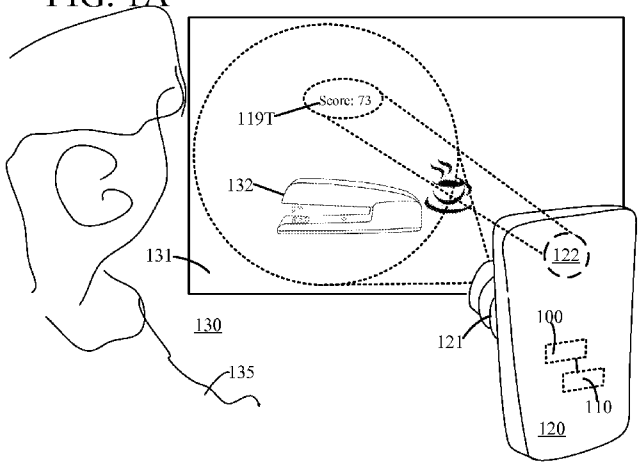
Declarations under Rule 4.17:

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

[Continued on next page]

(54) Title: USE OF ASSOCIATION OF AN OBJECT DETECTED IN AN IMAGE TO OBTAIN INFORMATION TO DISPLAY TO A USER

FIG. 1A



(57) Abstract: Camera(s) capture a scene, including an object that is portable. An image of the scene is processed to segment therefrom a portion corresponding to the object, which is identified from among a set of predetermined real world objects. An identifier of the object is used, with a set of associations between object identifiers and user identifiers, to obtain a user identifier that identifies a user at least partially from among a set of users. The user identifier may identify a group of users that includes the user ("weak identification") or alternatively the user identifier may identify the user uniquely ("strong identification") in the set. The user identifier is used either alone or in combination with user input to obtain and store in memory, information to be output to the user. At least a portion of the obtained information is thereafter output, e.g. displayed by projection into the scene.

WO 2013/028279 A1

— *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))*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

Use Of Association Of An Object Detected In An Image To Obtain Information To Display To A User

CROSS-REFERENCE TO PROVISIONAL APPLICATION

[0001] This application claims priority from U.S. Provisional Application No. 61/525,628 filed on August 19, 2011 and entitled "Projection of Information Onto Real World Objects or Adjacent Thereto", which is assigned to the assignee hereof and which is incorporated herein by reference in its entirety.

CROSS-REFERENCE TO PARENT APPLICATION

[0002] This application claims priority from U.S. Application No. 13/549,339 filed on July 13, 2012 and entitled " Use Of Association Of An Object Detected In An Image To Obtain Information To Display To A User", which is assigned to the assignee hereof and which is incorporated herein by reference in its entirety.

CROSS-REFERENCE TO RELATED APPLICATIONS

[0003] This application is also related to U.S. Application No 13/549,388, Attorney Docket No. Q111570U2os, filed on July 13, 2012, and entitled "Dynamic Selection of Surfaces In Real World For Projection of Information Thereon" which is assigned to the assignee hereof and which is incorporated herein by reference in its entirety.

BACKGROUND

[0004] It is well known to use a projector to project information for use via hand gestures by a user. For details on such prior art, see an article by Mistri, P., Maes, P., Chang, L. "WUW – Wear Ur World – A wearable Gestural Interface," CHI 2009, April 4-9, 2009, Boston, MA, USA, 6 pages that is incorporated by reference herein in its entirety.

[0005] Computer recognition of hand gestures of the type described above raises several issues, such as lighting conditions and robustness. Several such issues are addressed by use of Time of Flight cameras, e.g. as described in the article entitled "Picture Browsing and Map Interaction using a Projector Phone" by Andrew Greaves, Alina Hang, and Enrico Rukzio, MobileHCI 2008, September 2–5, 2008, Amsterdam, the Netherlands 4 pages. For additional information on such background on identifying hand gestures, see Mitra and Acharya, "Gesture Recognition: A Survey", IEEE

transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews, Vol. 37, No. 3, May 2007, 14 pages that is incorporated by reference herein in its entirety.

[0006] In prior art of the type described above, traditional approaches appear to require a user to explicitly look for information that the user desires, and which is not personalized automatically for the user. Requiring the user to explicitly look for desired information can be non-trivial when the information is being projected. Thus, what is needed is an improved way to obtain information that may be of interest to a user, as described below.

SUMMARY

[0007] In several embodiments, one or more cameras capture a scene that includes an object in real world that is sufficiently small to be carried by a human hand ("portable"), such as a stapler or a book. Thereafter, an image of the scene from the one or more cameras is processed to detect therein a portion corresponding to the object, which is recognized from among a set of pre-selected real world objects. An identifier of the object is then used, with a set of associations that associate object identifiers and identifiers of users, to obtain a user identifier that identifies a user at least partially among a set of users. In some embodiments, the user identifier identifies the user generically, as belonging to a particular group of users (also called "weak identification") among several such groups. In other embodiments, the user identifier identifies a single user uniquely (also called "strong identification"), among all such users in the set. The user identifier (obtained, as noted above, by use of an association and the object identifier) is thereafter used in several such embodiments, either alone or in combination with user-supplied information, to obtain and store in memory, information to be output to the user. At least a portion of the obtained information is thereafter output, for example by projection into the scene.

BRIEF DESCRIPTION OF THE DRAWING

[0008] FIG. 1A illustrates a portable real world object 132 (e.g. a stapler) being imaged by a camera 121, and its use by processor 100 in an electronic device 120 to cause a

projection of information into scene 130 by performing acts illustrated in FIG. 1C, in several embodiments described herein.

[0009] FIG. 1B illustrates another portable real world object 142 (e.g. a tape dispenser) being imaged by another electronic device 150 in a manner similar to electronic device 120 of FIG. 1A, to display information 119T on a screen 152 of device 150, also by performing the acts of FIG. 1C in several embodiments described herein.

[0010] FIG. 1C illustrates in a high-level flow chart, acts performed by one or more processors 100 (e.g. in one of electronic devices 120 and 150 of FIGs. 1A and 1B), to use an association of an object detected in an image to obtain information to display, in some embodiments described herein.

[0011] FIG. 1D illustrates in a high-level block diagram, a set of associations 111 in a memory 110 used by processor 100 of FIG. 1C, in some of the described embodiments.

[0012] FIG. 1E illustrates association of portable real world object 132 with group 2 by user 135 making a hand gesture with two outstretched fingers adjacent to portable real world object 132, in certain embodiments.

[0013] FIG. 1F illustrates use of the association of FIG. 1E to obtain and display information 188 for group 2, in certain embodiments.

[0014] FIG. 1G illustrates changing the association of FIG. 1E by replacing group 1 with group 2 by user 135 making another hand gesture with index finger 136 pointing at portable real world object 132, in such embodiments.

[0015] FIG. 1H illustrates use of the association of FIG. 1G to obtain and display information 189A and 189B for group 1, in such embodiments.

[0016] FIGs. 1I and 1J illustrate colored beams 163 and 164 of blue color and green color respectively projected on to object 132 by a projector 122 in certain embodiments.

[0017] FIG. 2A illustrates in a high-level flow chart, acts performed by one or more processors 100 to use an association of an object detected in an image in combination with user input, to obtain information to display, in some embodiments described herein.

[0018] FIG. 2B illustrates in a high-level block diagram, a set of associations 211 in a memory 110 used by processor 100 of FIG. 2A, in some of the described embodiments.

[0019] FIG. 2C illustrates real world object 132 in scene 230 imaged by camera 121, for use in projection of information in certain embodiments.

[0020] FIG. 2D illustrates processor 100 coupled to memory 110 that contains image 109 as well as an address 291 (in a table 220) used to obtain information for display in some embodiments.

[0021] FIG. 2E illustrates display of information in the form of a video 295 projected adjacent to object 132 in some embodiments.

[0022] FIGs. 3A and 3F illustrate in high-level flow charts, acts performed by one or more processors 100 to use an association of an object detected in an image with a single person, to obtain information to display specific to that person, in some embodiments described herein.

[0023] FIG. 3B illustrates in a high-level block diagram, a set of associations 311 in a memory 110 used by processor 100 of FIGs. 3A and 3F, in some of the described embodiments.

[0024] FIG. 3C illustrates, a user interface included in information projected adjacent to object 132 and in some embodiments.

[0025] FIGs. 3D and 3E illustrate, a user's personalized information included in information projected adjacent to object 132 (in FIG. 3D) and onto object 132 (in FIG. 3E) in some embodiments.

[0026] FIG. 4A illustrates, in a high-level block diagram, a processor 100 coupled to a memory 110 in a mobile device 120 of some embodiments.

[0027] FIG. 4B illustrates in a high-level flow chart, acts performed by processor 100 of FIG. 4A in projecting information into scene 130 in several embodiments.

[0028] FIGs. 4C and 4D illustrate in intermediate-level flow charts, acts performed by processor 100 in projecting information in certain embodiments.

[0029] FIG. 5 illustrates, in a high-level block diagram, a mobile device 120 of several embodiments.

DETAILED DESCRIPTION

[0030] In accordance with the described embodiments, one or more device(s) 120 (FIG. 1A) use one or more cameras 121 (and/or sensors such as a microphone) to capture input e.g. one or more images 109 (FIG. 1B) from a scene 130 (FIG. 1A) that contains an object 132 (FIG. 1A). Depending on the embodiment, object 132 can be any object in the real world (in scene 130) that is portable by a human, e.g. small enough to be carried in (and/or moved by) a human hand, such as any handheld object. Examples of object 132 are a stapler, a mug, a bottle, a glass, a book, a cup, etc. Also depending on the embodiment, device(s) 120 can be any electronic device that includes a camera 121, a memory 110 and a processor 100, such as a smartphone or a tablet (e.g. iPhone or iPad available from APPLE, Inc.). For convenience, the following description refers to a single device 120 performing the method of FIG. 1C, although multiple such devices can be used to individually perform any one or more of steps 101-108, depending on the embodiment.

[0031] Accordingly, one or more captured images 109 (FIG. 1B) are initially received from a camera 121 (as per act 101 in FIG. 1C) e.g. via bus 1113 (FIG. 5) and stored in a memory 110. Processor 100 then processes (as per act 102 in FIG. 1C) the one or more images 109 to detect the presence of an object 132 (FIG. 1A) e.g. on a surface of a table 131, in a scene 130 of real world outside camera 121 (see act 102 in FIG. 1C). For example, processor 100 may be programmed to recognize a portion of image 109 corresponding to portable real world object 132, to obtain an identifier ("object identifier") 112O that uniquely identifies object 132 among a set of predetermined objects.

[0032] Processor 100 then uses the object identifier (e.g. stapler identifier 112O in FIG. 1D) that is obtained in act 102 to look up a set of associations 111 in an act 103 (FIG. 1C). The result of act 103 is an identifier of a user (e.g. user identifier 112U) who has been previously associated with portable real world object 132 (e.g. stapler). Specifically, in certain aspects of several embodiments, memory 110 that is coupled to processor 100 holds a set of associations 111 including for example an association 112

(FIG. 1D) that associates a stapler identifier 112O with user identifier 112U and another association 114 that associates a tape-dispenser identifier 114O with another user identifier 114U. Such a set of association 111 may be created in different ways depending on the embodiment, and in some illustrative embodiments an association 112 in set 111 is initialized or changed in response to a hand gesture by a user, as described below.

[0033] In certain embodiments, user identifiers 112U and 114U which are used in associations 112 and 114 do not identify a single user uniquely, among the users of a system of such devices 120. Instead, in these embodiments, user identifiers 112U and 114U identify corresponding groups of users, such as a first group 1 of users A, B and C, and a second group 2 of users X, Y and Z (users A-C and X-Z are not shown in FIG. 1D). In such embodiments (also called "weak identification" embodiments), a user identifier 112U that is obtained in act 103 (described above) is generic to several users A, B and C within the first group 1. In other embodiments (also called "strong identification" embodiments), each user identifier obtained in act 103 identifies a single user uniquely, as described below in reference to FIGs. 3A-3F.

[0034] Referring back to FIG. 1C, processor 100 of several embodiments uses a user identifier 112U looked up in act 103, to generate an address of information 119 (FIG. 5) to be output to the user and then obtains and stores in memory 110 (as per act 104 in FIG. 1C) the information 119. In certain weak identification embodiments, wherein user identifier 112U is of a group 1, information 119 to be output obtained in act 104 is common to all users within that group 1 (e.g. common to users A, B and C). For example, information 119 includes the text "Score: 73" which represents a score of this group 1, in a game being played between two groups of users, namely users in group 1 and users in group 2.

[0035] Information 119 is optionally transformed and displayed or otherwise output to user 135, in an act 105 (see FIG. 1C) as information 119T (FIG. 1A). Specifically, in some embodiments, information 119 is displayed by projection of at least a portion of the information (e.g. the string of characters "Score: 73") into scene 130 by use of a projector 122 as illustrated in FIG. 1A. In other embodiments, information 119 (or a portion thereof) may be output by act 105 (FIG. 1C) in other ways, e.g. device 150 (FIG. 1B) displaying information 119 to user 135 directly on a screen 151 (FIG. 1B)

that also displays a live video of scene 130 (e.g. by displaying image 109), thereby to provide an augmented reality (AR) display.

[0036] In still other embodiments, information 119 may be played through a speaker 1111 (FIG. 5) in device 120, or even through a headset worn by user 135. In an embodiment illustrated in FIG. 1B, device 150 is a smartphone that includes a front-facing camera 152 (FIG. 1B) in addition to a rear-facing camera 121 that captures image 109. Front-facing camera 152 (FIG. 1B) is used in some embodiments to obtain an image of a face of user 135, for use in face recognition in certain strong identification embodiments described below in reference to FIGs. 3A-3F. Device 150 (FIG. 1B) may be used in a manner similar or identical to use of device 120 as described herein, depending on the embodiment.

[0037] Processor 100 (FIG. 5) of certain embodiments performs act 103 after act 102 when associations 111 (FIG. 1D) have been previously formed and are readily available in memory 110. Associations 111 may be set up in memory 110 based on information input by user 135("user input") in any manner, as will be readily apparent in view of this detailed description. For example, user input in the form of text including words spoken by user 135 are extracted by some embodiments of processor 100 operating as a user input extractor 141E, from an audio signal that is generated by a microphone 1112 (FIG. 5) in the normal manner. In certain embodiments, user input (e.g. for use in preparing associations 111) is received by processor 100 via camera 121 in the form of still images of shapes or a sequence of frames of video of gestures of a hand 138 of a user 135 (also called social protocol), and comparing the received gestures with a library. In some "weak identification" embodiments, processor 100 is programmed to respond to user input sensed by one or more sensors in device 120, (e.g. camera 121) that detect one or more actions (e.g. gestures) by a user 135 as follows: processor 100 associates an object 132 that is selectively placed within a field of view 121F of camera 121 with an identifier (e.g. a group identifier) that depends on user input (e.g. hand gesture).

[0038] As would be readily apparent in view of this detailed description, any person, e.g. user 135 can use their hand 138 to form a specific gesture (e.g. tapping on object 132), to provide user input via camera 121 to processor 100 that in turn uses such user input in any manner described herein. For example, processor 100 may use input from

user 135 in the form of hand gestures (or hand shapes) captured in a video (or still image) by camera 121, to initialize or change a user identifier that is generic and has been associated with object 132, as illustrated in FIGs. 1E-1H, and described below. As noted above, some embodiments accept user input in other forms, e.g. audio input such as a whistling sound, and/or a drumming sound and/or a tapping sound, and/or sound of text including words "Group Two" spoken by user 135 may be used to associate an object 132 that is imaged within image 109 with a user identifier that is generic (e.g. commonly used to identify multiple users belonging to a particular group).

[0039] In some embodiments, user input extractor 141E is designed to be responsive to images from camera 121 of a user 135 forming a predetermined shape in a gesture 118, namely the shape of letter "V" of the English alphabet with hand 138, by stretching out index finger 136 and stretching out middle finger 137. Specifically, camera 121 images hand 138, with the just-described predetermined shape "V" at a location in real world that is adjacent to (or overlapping) portable real world object 132 (FIG. 1E). Moreover, processor 100 is programmed to perform act 106 (FIG. 1C) to extract from image 109 (FIG. 1D) in memory 110 this hand gesture 118 (index and middle finger images 136I and 137I outstretched in human hand image 138I in FIG. 4A). During an initialization phase, processor 100 responds to detection of such a hand gesture by forming an association (in the set of associations 111) that is thereafter used to identify person 135 (as belonging to group 2) every time this same hand gesture is recognized by processor 100.

[0040] In some embodiments, for processor 100 to be responsive to a hand gesture, user 135 is required to position fingers 136 and 137 sufficiently close to object 132 so that hand gesture fingers 136, 137 and object 132 are all imaged together within a single image 109 (which may be a still image, or alternatively a frame of video, depending on the embodiment) from camera 121. Note that in the just-described embodiments, when user 135 makes the same hand gesture, but outside the field of view 121F of camera 121 (FIG. 5), processor 100 does not detect such a hand gesture and so processor 100 does not use make or change any association, even when object 132 is detected in image 109.

[0041] In some embodiments, an act 106 is performed by processor 100 after act 102, to identify the above-described hand gesture (or any other user input depending on the embodiment), from among a library of such gestures (or other such user input) that are

predetermined and available in a database 199 on a disk (see FIG. 2D) or other non-transitory storage media. Next, in act 107, processor 100 performs a look up of a predetermined mapping 116 (FIG. 1D) based on the hand gesture (or other such user input) detected in act 106 to obtain a user identifier from the set of associations 111 (FIG. 5). In the example illustrated in FIG. 1D, two-finger gesture 118 (or other user input, e.g. whistling or drumming) is related by a mapping 116 to an identifier 114U of group 2, and therefore in an act 108 this identifier (looked-up from mapping 116) is used to form association 114 in the set 111.

[0042] On completion of act 108, processor 100 of several embodiments proceeds to obtain information 119 to be output (as per act 104, described above), followed by optional transformation and output (e.g. projection as per act 105), as illustrated in FIG. 1F. In one example shown in FIG. 1A, object 132 was associated with group 1, and therefore information 119T which is output into scene 130 includes the text string "Score: 73" which represents a score of group 1, in the game being played with users in group 2. In several such embodiments, as illustrated in FIG. 1G, the same user 135 can change a previously formed association, by making adjacent to the same portable real world object 132, a second hand gesture 117 (e.g. index finger 136 outstretched) that is different from a first hand gesture 118 (e.g. index finger 136 and middle finger 137 both stretched out). As noted above, the hand gesture is made by user 135 sufficiently close to object 132 to ensure that the gesture and object 132 are both captured in a common image by camera 121.

[0043] Such a second hand gesture 117 (FIG. 1D) is detected by processor 100 in act 106, followed by lookup of mapping 116, followed by over-writing of a first user identifier 114U that is currently included in association 114 with a second user identifier 112U, thereby to change a previously formed association. Hence, after performance of acts 104 and 105 at this stage, information including the text string "Score: 73" previously displayed (see FIG. 1F) is now replaced with new information including the text string "Score:0" which is the score of Group 2 as shown in FIG. 1G. In some embodiments, in addition to the just-described text string, one or more additional text strings may be displayed to identify the user(s). For example, the text string "Group 2" is displayed as a portion of information 188T in FIG. 1G, and while the text string "Group 1" is displayed as a portion of information 188T in FIG. 1F. In some

embodiments, information 188T is optionally transformed for display relative to information 188 that is obtained for output, resulting in multiple text strings of information 188 being displayed on different surfaces, e.g. information 189A displayed on object 132 and information 189B displayed on table 131 as shown in FIG. 1H.

[0044] Although in some embodiments a mapping 116 maps hand gestures (or other user input) to user identifiers of groups, in other embodiments each hand gesture (or other user input) may be mapped to a single user 135, thereby to uniquely identify each user ("strong identification embodiments"), e.g. as described below in reference to FIGs. 3A-3F. In several such embodiments, yet another data structure (such as an array or a linked list) identifies a group to which each user belongs, and processor 100 may be programmed to use that data structure to identify a user's group when needed for use in associations 111.

[0045] In obtaining to-be-displayed information in act 104 (FIG. 1C), processor 100 of some embodiments simply uses recognition of a user's hand gesture (or other user input) to select Group 2 from among two groups, namely Group 1 and Group 2. Specifically, although presence of portable real world object 132 is required by processor 100 in an image 109 in order to recognize a gesture, the identity of object 132 is not used in some embodiments to obtain the to-be-displayed information. However, other embodiments of processor 100 do use two identifiers based on corresponding detections in image 109 as described above, namely user identifier and object identifier, to obtain and store in memory 110 the to-be-displayed information 119 in act 104 (FIG. 1C).

[0046] Moreover, as will be readily apparent in view of this detailed description, the to-be-displayed information 119 (FIG. 5) may be obtained by processor 100 based on recognition of (1) only a hand gesture or (2) only the real world object, or (3) a combination of (1) and (2), depending on the embodiment. Also as noted above, a hand gesture is not required in certain embodiments of processor 100 that accepts other user input, such as an audio signal (generated by a microphone) that carries sounds made by a user 135 (with their mouth and/or with their hands) and recognized by processor 100 on execution of appropriate software designed to recognize such user-made sounds as user input, e.g. in signals from microphone 1112 (FIG. 5).

[0047] Although in some embodiments, a user identifier with which portable real world object 132 is associated is displayed as text, as illustrated by text string 189A in FIG. 1H, in other embodiments a user identifier may be displayed as color, as illustrated in FIGs. 1I and 1J. Specifically, for example, to begin with, object 132 in the form of a cap 132 of a bottle is selected by a user 135 to be included in an image of a scene 130 of real world being captured by a camera 121 (FIG. 5).

[0048] Initially identity 214 of bottle cap 132 is associated in a set of associations 211 (see FIG. 2B) by default with three users that are identified as a group of friends by identity 215, and this association is shown by device 120, e.g. by projecting a beam 163 of blue color light on object 132 and in a peripheral region outside of and surrounding object 132 (denoted by the word of text 'blue' in FIG. 1I, as colors are not shown in a black-and-white figure). In this example, color blue has been previously associated with the group of friends of identity 215, as the group's color. Similarly, a book's identity 212 is associated in the set of associations 211 (see FIG. 2B) by default with four users that are identified as a group (of four students) by identity 213 (e.g. John Doe, Jane Wang, Peter Hall and Tom McCue).

[0049] At this stage, a person 135 (e.g. Tom McCue) identified as a user of the group of students associates his group's identity 213 (see FIG. 2B) with object 132 (FIG. 1J) by tapping table surface 131 with index finger 136 repeatedly several times in rapid succession (i.e. performs a hand gesture to which processor 100 is programmed to recognize and suitably respond), until person 135 sees a projection of beam 164 of green light on and around object 132 (denoted by the word 'green' in FIG. 1J, as this figure is also a black-and-white figure). In this example, color green was previously associated with the group of students of identity 213, as its group color. Moreover, tapping is another form of hand gesture recognized by processor 100, e.g. on processing a camera-generated image containing the gesture and optionally user input in the form of sound, or both in some illustrative embodiments.

[0050] In certain embodiments of the type illustrated in FIG. 2A, processor 100 performs acts 201-203 that are similar or identical to acts 101-103 described above in reference to FIG. 1C. Accordingly, in act 201, one or more rear-facing cameras 121 are used to capture scene 130 (FIG. 2C) that includes real world object 132 (in the form of a book) and store image 109 (FIG. 2D) in memory 110 in a manner similar to that

described above, although in FIGs. 2C and 2D, the object 132 being imaged is a book. In performing acts 202-206 in FIG. 2A processor 100 not only recognizes object 132 as a book in image 109 (FIG. 2D) but additionally recognizes a text string 231A therein (FIG. 2C), which is identified by a hand gesture.

[0051] Specifically, in act 204 (FIG. 2A), processor 100 operates as a user-input extractor 141E (FIG. 5) that obtains input from user 135 for identifying information to be obtained for display in act 205 (which is similar to act 105 described above). For example, in act 204, user input is received in processor 100 by detection of a gesture in image 109 (FIG. 2D) in which object 132 has also been imaged by camera 121 (FIG. 2C). In this example, user 135 (FIG. 2C) makes a predetermined hand gesture, namely an index finger hand gesture 117 by stretching finger 136 of hand 138 to point to text string 231A in portable real world object 132. This index finger hand gesture 117 is captured in one or more image(s) 109 (FIG. 2D) in which object 132 is also imaged (e.g. finger 136 overlaps object 132 in the same image 109). The imaged gesture is identified by use of a library of gestures, and an procedure triggered by the index finger hand gesture 117 is performed in act 204, including OCR of an image portion identified by the gesture, to obtain as user input, the string of characters "Linear Algebra."

[0052] Subsequently, in act 205 (FIG. 2A), processor 100 operates as an object-user-input mapper 141M (FIG. 5) that uses both: (1) the user group identified in act 203 from the presence of object 132 and (2) the user input identified from a text string 231A (e.g. "Linear Algebra") detected by use of a gesture identified in act 204 (FIG. 2A), to generate an address 291 (in a table 220 in FIG. 2B). For example, a user group identified by act 203 (FIG. 2A) may be first used by object-user-input mapper 141M (FIG. 5) to identify table 220 (FIG. 2B) from among multiple such tables, and then the identified table 220 is looked up with the user-supplied information, to identify an address 291, which may be accessible on the Internet.

[0053] Such an address 291 is subsequently used (e.g. by information retriever 141R in FIG. 5) to prepare a request for fetching from Internet, a video that is associated with the string 231A. For example, in act 205 (FIG. 2A) processor 100 generates address 291 as http://ocw.mit.edu/courses/ mathematics/18-06-linear-algebra-spring-2010/video-lectures/ which is then used to retrieve information 119 (FIG. 5). Use of table 220 as just described enables a query that is based on a single common text string

231A to be mapped to different addresses, for information to be displayed to different groups of users. For example, a processor for one user A retrieves an address of the website of Stanford Distance learning course (namely http://scpd.stanford.edu/coursesSeminars/seminarsAndWebinars.jsp) from one table (e.g. customized for user A) while processor 100 for another user B retrieves another address 291 for MIT's OpenCourseware website (namely http://videolectures.net/mit_ocw/) from another table 220 (e.g. customized for user B).

[0054] Such an address 291 that is retrieved by processor 100 using a table 220, in combination with one or words in text string 231A may be used in some embodiments with an Internet-based search service, such as the website www.google.com to identify content for display to user 135. Subsequently, in act 206, processor 100 issues a request to address 291 in accordance with http protocol and obtains as information to be output, a video stream from the Internet, followed by optional transformation and projection of the information, as described below.

[0055] A result of performing the just-described method of FIG. 2A is illustrated in FIG. 2E by a video 295 shown projected (after any transformation, as appropriate) on a surface of table 131 adjacent to object 132. As noted above, video 295 has been automatically selected by processor 100 and is being displayed, based at least partially on optical character recognition (OCR) to obtain from one or more images (e.g. in video 295) of object 132, a text string 231A that has been identified by an index finger hand gesture 117. In several embodiments, no additional input is needed by processor 100 from user 135, after the user makes a predetermined hand gesture to point to text string 231A and before the video is output, e.g. no further user command is needed to invoke a video player in device 120, as the video player is automatically invoked by processor 100 to play the video stream retrieved from the Internet. Other such embodiments may require user input to approve (e.g. confirm) that the video stream is to be displayed.

[0056] Note that text string 231A is recognized from among many such strings that are included in image 109, based on the string being located immediately above a tip of index finger 136 which is recognized by processor 100 in act 204 as a portion of a predetermined hand gesture. As noted above, human finger 136 is part of a hand 138 of a human user 135 and in this example finger 136 is used in gesture 117 to identify as user input a string of text in scene 130, which is to trigger retrieval of information to be

output. Instead of index finger hand gesture 117 as illustrated in FIGs. 2C and 2D, in certain alternative embodiments user 135 makes a circling motion with finger 136 around text string 231A as a different predetermined gesture that is similarly processed. Hence, in act 204, processor 103 completes recognition of real world object 132 (in the form of a book in FIG. 2C), in this example by recognizing string 231A. Thereafter, processor 103 generates a request to a source on the Internet to obtain information to be projected in the scene for use by person 135 (e.g. based on a generic user identifier of person 135 as belonging to a group of students).

[0057] Accordingly, user interfaces in certain embodiments of the type described above in reference to FIGs. 2A-2E automatically project information 119 on real world surfaces using a projector 122 embedded in a mobile device 120. Thus, user interfaces of the type shown in FIGs. 2A-2E reverse the flow of information of prior art user interfaces which require a user 135 to explicitly look for information, e.g. prior art requires manually using a web browser to navigate to a web site at which a video stream is hosted, and then manually searching for and requesting download of the video stream. Instead, user interfaces of the type shown in FIGs. 2A-2E automatically output information that is likely to be of interest to the user, e.g. by projection on to surfaces of objects in real world, using an embedded mobile projector.

[0058] Depending on the embodiment, mobile device 120 may be any type of electronic device with a form factor sufficiently small to be held in a human hand 138 (similar in size to object 132) which provides a new way of interacting with user 135 as described herein. For example, user 135 may use such a mobile device 120 in collaboration with other users, with contextual user interfaces based on everyday objects 132, wherein the user interfaces overlap for multiple users of a specific group, so as to provide common information to all users in that specific group (as illustrated for group 1 in FIG. 1F and group 2 in FIG. 1G). Moreover, as described above in reference to FIGs. 2A-2E, processor 100 may be programmed in some embodiments to automatically contextualize a user interface, by using one or more predetermined techniques to identify and obtain for display information that a user 135 needs to view, when interacting with one or more of objects 132.

[0059] In some illustrative embodiments, in response to user 135 opening book (shown in FIG. 2C as object 132) to a specific page (e.g. page 20), processor 120 automatically

processes images of real world to identify therein a text string 231A based on its large font size relative to the rest of text on page 20. Then, processor 100 of several embodiments automatically identifies a video on "Linear Algebra" available on the Internet e.g. by use of a predetermined website (as described above in reference to act 205), and then seeks confirmation from user 135 that the identified video should be played. The user's confirmation may be received by processor 100 in a video stream that contains a predetermined gesture, e.g. user 135 waving of index finger in a motion to make a check mark (as another predetermined gesture that identifies user input).

[0060] Processor 100 is programmed in some embodiments to implement strong identification embodiments, by performing one or more acts 301-309 illustrated in FIG. 3A, by use of user identifiers that uniquely identify a single user X from among all users A-Z. In examples of such embodiments that use strong identification, information 319 (FIGs. 3D and 3E) that is obtained for display may be specific to that single user X, for example email messages that are specifically addressed to user X.

[0061] In certain embodiments of the type illustrated in FIG. 3A, processor 100 performs acts 301 and 302 that are similar or identical to acts 101 and 102 described above in reference to FIG. 1C. Thereafter, in act 303, processor 100 uses an identifier of the portable real world object with a set of associations 311 (FIG. 3B), to obtain an identifier that uniquely identifies a user of the portable real world object. Note that act 303 is similar to act 103 except for the set of associations 311 being used in act 303. Specifically, in set 311, an association 314 maps an object identifier 114O (such as a bottle cap ID) to a single user 314U (such as Jane Doe), as illustrated in FIG. 3B. Accordingly, in such embodiments, user 314U is uniquely associated with object 132 in the form of a bottle cap (FIG. 1E), i.e. no other user is associated with a bottle cap. Hence, other users may be associated with other such portable real world objects (e.g. book in FIG. 2C or stapler in FIG. 1A), but not with a bottle cap as it has already been uniquely associated with user 314U (FIG. 3B).

[0062] Thereafter, a user identifier which is obtained in act 303 is used in act 304 (FIG. 3A) to obtain and store in memory 110, information 319 that is specific to user 314U (of the portable real world object 132). This information 319 is then displayed, in act 305, similar to act 105 described above, except that the information being displayed is

specific to user 314U as shown in FIGs. 3D and 3E. For example, the to-be-displayed information 319 may be obtained in act 304 from a website www.twitter.com, specific to the user's identity. In some examples, the to-be-displayed information 319 received by processor 100 is personalized for user 135, based on user name and password authentication by the website www.twitter.com. Although the personalized information 319 illustrated in one example is from www.twitter.com other websites can be used, e.g. an email website such as http://mail.yahoo.com can be used to obtain other such information personalized for user 135.

[0063] In several strong identification embodiments of the type illustrated in FIGs. 3A and 3B, user-supplied text (for use in preparing associations 311) is received by processor 100 via an authentication (also called login) screen. Specifically, in some embodiments an act 306 is performed by processor 100 after act 302, to display an authentication screen. For example, in act 306 authentication screen 321 is projected on to table 131 adjacent to object 132 as part of information 322 as shown in FIG. 3C. In the example illustrated in FIG. 3C, processor 100 obtains the authentication screen to be displayed in act 306 from a computer (not shown) accessible on the Internet, such as a web server at the website www.twitter.com, and this screen is of a user interface such as a dialog box that prompts the user to enter their user name and password.

[0064] In some embodiments, processor 100 automatically includes adjacent to such a dialog box, a layout image 333 of a keyboard in information 322 that is projected into scene 130. Note that although only a keyboard image 333 is illustrated in FIG. 3C, a mouse may additionally be included in the projected information 322. Next, as user 135 types on keyboard image 333, in act 307 processor 100 recognizes additional images (similar to image 109 described above) and generates user input by performing Optical Character Recognition (OCR), and such user input in the form of text string(s) is stored in memory 110 and then sent to the website www.twitter.com. The same user input is also used in act 308 by some embodiments of processor 100 to identify the user (i.e. by using at least a portion of the user input as a user identifier), followed by performing authentication using a table 391 in memory 110. For example, in act 308 a user name and password received as the user-supplied text may be locally checked against table 391 by processor 100.

[0065] At this stage, the user's identity is known, and a "strong" identification embodiment is implemented by processor 100 performing act 309. Specifically, in act 309, processor 100 prepares an association 314 in set 311, to associate an identifier 114U of object 132 with an identifier 314U (e.g. name) of the user identified in the authentication screen 321 (projected adjacent to object 132), in response to user input being authenticated . Thereafter, acts 304 and 305 are performed as described above. Note that in other embodiments, a different user name and password may be used locally by processor 100. Hence, in one such example the user is authenticated two times, once by processor 100 locally (when the user enters their user name and password to login, into device 120), and another time by a remote computer (or web server) that supplies information 119 (FIG. 5) to be output (e.g. at the website www.twitter.com). In some embodiments, a single authentication is sufficient, e.g. the user name and password that were used to log into device 120 are automatically used (directly or indirectly via a table lookup) to communicate with the remote computer (or web server), to obtain information 119 (FIG. 5). In some embodiments, a user name that was used to log into device 120 is also used to identify a table 220 (among multiple such tables) used in identifying address 291, for obtaining information 119 (FIG. 5)

[0066] User-specific information 319 that is obtained in act 304 is typically displayed as per act 305 at a location adjacent to object 132 (FIG. 3D) or alternatively on object 132 itself (FIG. 3E) in order to reduce the likelihood of snooping by users other than user 135 with whom object 132 is uniquely associated. Prior to display (e.g. by projection) in act 305, such information may be optionally transformed. A specific technique for transformation that is selected for projection of user-specific personalized information can depend on a number of factors, such as the smoothness and/or shape and size and/or a dynamically computed surface normal (or gradient) of object 132, and/or resolution and legibility of information 119 that is to be projected, etc. A transformation technique that is selected may also edit information 119 to be output, e.g. truncate or abbreviate the information, omit images, or down-scale images etc. depending on the embodiment.

[0067] Although certain embodiments to implement strong identification described above use an authentication screen, other embodiments use two cameras to perform a method of the type illustrated in FIG. 3F, wherein one camera is embedded with a projector 122 in device 120 and other camera is a normal camera included in a mobile

device such as a smart phone (or alternatively external thereto, in other form factors). Hence, in the method of FIG. 3F, two cameras are operated to synchronize (or use) hand gestures with a person's face, but otherwise many of the acts are similar or identical to the acts described above in reference to FIG. 3A.

[0068] Specifically, when person 135 makes a specific hand gesture adjacent to object 132 or touches object 132 in a certain manner (e.g. taps on the object), a back-facing camera 121 (FIG. 1B) in mobile device 150 captures an image 109 of scene 130. Detection in such an image 109 of a portion that corresponds to the specific hand gesture (as per act 396 in FIG. 3F) triggers processor 100 to perform act 397 to operate a front-facing camera 152 (FIG. 1B). Front-facing camera 152 (FIG. 1B) then captures an image including a face of the user 135, and the image is then segmented (e.g. by module 141S in FIG. 5) to obtain a portion corresponding to the face which is used (in module 141S) by processor 100 performing act 398 to determine a user identifier (e.g. perform authentication).

[0069] Specifically, in act 398, processor 100 of some embodiments compares the image portion corresponding to the face to a database 199 (FIG. 2D) of faces, and on finding a match obtains from the database an identifier of the user (selected from among user identifiers of faces in the database). Hence, the user's face 120 is received by processor 100 from a front facing camera 152 and is detected as such in act 398, thereby resulting in a unique identifier for the user that may be supplied to user input extractor 141E for use in preparing an association, to associate the user identifier with an object identifier, in response to detecting a predetermined gesture adjacent to the object. Illustrative embodiments of a segmentation module 141S (FIG .5) may identify users as described by Viola and Jones in an article entitled "Robust Real-Time Face Detection", 18 pages, International Journal of Computer Vision 57(2), 137-154, 2004 that is incorporated by reference herein in its entirety. Next, act 309 is performed as described above in response to detection of the gesture, to prepare an association so that object 132 (e.g. bottle cap) is identified as belonging to person 135.

[0070] In certain embodiments, an object identifier of portable real world object 132 in image 109 (FIG. 4A) is automatically identified by processor 100 using data 445 (also called "object data", see FIG. 4A) on multiple real world objects in a database 199 that is coupled to processor 100 in the normal manner. Specifically, in an example

illustrated in FIGs. 4A and 4B, processor 100 recognizes object 132 to be a bottle cap (which is identified by an identifier ID1), based on attributes in data 441D (FIG. 4A) in database 199 matching attributes of a portion 132I of image 109 (FIG. 4A) received as per act 431 (FIG. 4B).

[0071] Hence, in some embodiments, processor 100 is programmed with software to operate as an object extractor 141O (see FIG. 5) which determines feature vectors from an image 109, and compares these feature vectors to corresponding feature vectors of objects that are previously computed and stored in a database 199, to identify an object. Comparison between feature vectors can be done differently depending on the embodiment (e.g. using Euclidean distance). On completion of comparison, object extractor 141O identifies from database 199 an object 132 that most closely matches the feature vectors from image 109, resulting in one or more object identifiers 112O, 112U (FIG. 5).

[0072] Any type of features known to a skilled artisan may be extracted from image 109 by object extractor 141O (FIG. 5), depending on the embodiment. For more information on use of such feature, see a 15-page article entitled "Scale-invariant feature transform" at "http://en.wikipedia.org/ wiki/Scale-invariant_feature_transform" as available on April 9, 2012, which is incorporated by reference herein in its entirety. In several such embodiments, processor 100 is programmed with software to identify clusters of features that vote for a common pose of an object (e.g. using the Hough transform). In several such embodiments, bins that accumulate a preset minimum number of votes (e.g. 3 votes) are identified by object extractor 141O, as object 132.

[0073] Some embodiments of object extractor 141O extract SIFT features as described in the preceding paragraph, while other embodiments use a method described by Viola and Jones in a 25-page article entitled "Robust Real-time Object Detection," in the Second International Workshop On Statistical And Computational Theories Of Vision – Modeling, Learning, Computing, And Sampling, Vancouver, Canada, July 13, 2001 that is incorporated by reference herein in its entirety. For more information on such a method, see a 3-page article entitled "Viola-Jones object detection framework" at "http://en.wikipedia. org/wiki/Viola%E2%80%93Jones_object_detection_framework" as available on April 9, 2012, which is incorporated by reference herein in its entirety.

[0074] Accordingly, features (that are determined from an image 109 as described above) are used in some embodiments by object extractor 141O (FIG. 5) to generate a geometric description of items in image 109 that is received from a camera, such as object 132. Similar or identical software may be used by processor 100 to extract from image 109, a blob 136I (FIG. 2D) of a finger in a hand gesture (and/or to recognize a user's face as described herein). As noted above, some embodiments of processor 100 use Haar features which consist of vector windows that are used to calculate edges, line features and center-surrounded features in an image 109. In certain embodiments, vector windows are run by processor 100 across portions of image 109 (FIG. 2D) and the resulting output is an integer value. If the value at a certain position in the image exceeds a certain threshold, such embodiments determine that there is a positive match. Depending on the embodiment, processor 100 uses such features (also called "feature vectors") differently depending upon the item to be recognized (object, or face, or hand gesture). Hence, depending on the embodiment, processor 100 uses vector windows that are different for objects, hand gestures, face recognition etc. Use of Haar features by processor 100 in certain embodiments has limitations, such as robustness and low fps (frames per second) due to dependency on scaling and rotation of Haar vector windows.

[0075] Various embodiments of processor 100 may be programmed with software to operate as object extractor 141O (FIG. 5) that uses other methods, such as a method described in an 8-page article entitled "Object Recognition from Local Scale-Invariant Features" by David G. Lowe, in Proceedings of the International Conference on Computer Vision, Corfu (Sept 1999), which is incorporated by reference herein in its entirety. In some embodiments, after one or more objects are identified by object extractor 141O (of the type described above), three dimensional (3D) surfaces of the object(s) are segmented by processor 100 into local regions with a curvature (or other such property) within a predetermined range, so that the regions are similar, relative to one another.

[0076] The above-described segmentation into local regions is done by processor 100 so that when information is projected on such object(s), processor 100 may be optionally programmed to operate as information transformer 141T (FIG. 5) to truncate or otherwise manipulate information 119, so that the information fits within local regions of object 132 identified by segmentation. Truncation or manipulation of content

in information 119 by processor 100 reduces or eliminates the likelihood that projection of information 119 on to object 132 will irregularly wrap between local regions which may have surface properties different from one another. Processor 100 of some embodiments segments a 3D surface of object 132 to identify local regions therein as described in a 14-page article entitled "Partitioning 3D Surface Meshes Using Watershed Segmentation" by Alan P. Mangan and Ross T. Whitaker, in IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 5, NO. 4, OCTOBER-DECEMBER 1999 which is incorporated by reference herein in its entirety.

[0077] In some embodiments, to project information 119 onto local regions of object 132, processor 100 operating as information transformer 141T calibrates camera 121 (FIG. 5) using any calibration method, such as the method described in an a 4-page article entitled "Camera Calibration Toolbox for Matlab" at "http://www.vision.caltech.edu/ bouguetj/calib_doc/" as available on April 10, 2012, which is incorporated by reference herein in its entirety. Several embodiments of information transformer 141T (FIG. 5) are programmed to determine shapes and/or surface normals of surfaces of object 132 in image 109, using one of the following two methods described in the next paragraph.

[0078] A first method uses a projection of light, e.g. as described in an 8-page article entitled "Dynamic scene shape reconstruction using a single structured light pattern" by Hiroshi Kawasaki et al , IEEE Conference on Computer Vision and Pattern Recognition 2008, which is incorporated by reference herein in its entirety. A second method also uses a projection of light, e.g. as described in a 13-page article entitled "Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming" by Li Zhang et al , in Proc. Int. Symposium on 3D Data Processing Visualization and Transmission (3DPVT), 2002, which is incorporated by reference herein in its entirety.

[0079] Referring to FIG. 4A, certain other data 442D and 443D in object data 445 (FIG. 4A) of some embodiments may include attributes to be used by object extractor 141O (FIG. 5) in identifying various other objects such as a object 132 (FIG. 2C) having identifier ID2 (FIG. 4A) or a cup (not shown) having identifier ID3, in an image 109 analyzed by processor 100 in act 432 (FIG. 4B). As will be readily apparent to the skilled artisan in view of this detailed description, on completion of act 432, one or

more of object identifiers ID1, ID2 and ID3 uniquely identify within processor 100, corresponding portable real world objects, namely a bottle cap, a book and a cup when these objects are imaged in an image 109 of a scene 130 by a camera of device 120.

[0080] Although image 109 illustrated in FIG. 4A includes a portion 132I that corresponds to the entirety of object 132, depending on the aspect of the described embodiments, another image that captures only a portion of object 132 may be sufficient for processor 100 to recognize object 132 (in act 432 of FIG. 4B). Moreover, depending on the aspect of the described embodiments, processor 100 may be programmed to operate as object extractor 141O to perform act 432 to recognize additional information captured from object 132, as described above.

[0081] Several weak identification embodiments use groups 446 in associations 448, 449, while in strong identification embodiments each of associations 448, 449 contains a single user's name as the user identifier associated with a corresponding object identifier. As will be readily apparent to the skilled artisan in view of this detailed description, certain embodiments may use both forms of identification in processor 100 operating as an information retriever 141R (FIG. 5) with different types of portable real world objects and/or different information display software 141 (FIG. 4A), depending on the programming of processor 100 (FIG. 4B).

[0082] As noted above, associations 447 (FIG. 4A) may be set up in different ways in database 199, prior to their use by processor 100, depending on the embodiment. In several embodiments, processor 100 is programmed with software to operate as information identifier 141I (FIG. 5) that extracts user input (in user input extractor 141E) and uses associations to generate an address 291 of information to be output (in object-user-input mapper 141M). Specifically, in some embodiments, processor 100 is programmed with software to operate as user input extractor 141E to process an image 109 received in act 431 or to process additional image(s) received in other such acts or to process other information captured from scene 130 (e.g. audio signal of a drumming sound or a tapping sound, and/or whistling sound made by a user 135), so as to recognize a user's hand gesture in act 439A for use in initialization of an association. Optionally, after completion of act 439A, processor 100 may return to act 432 (described above).

[0083] As noted above, depending on the embodiment, image 109 (or additional such images that are later captured) may include image portions corresponding to one or more human fingers, e.g. index finger 136 (FIG. 1E) and middle finger 137 are parts of a human hand 138, of a person 135. Processor 100 is programmed to operate as user input extractor 141E (FIG. 5) in act 439A to use predetermined information (not shown in FIG. 4A) in database 199 to recognize in image 109 (FIG. 4A) in memory 110 certain user gestures (e.g. index and middle finger images 136I and 137I outstretched in human hand image 138I), and then use a recognized gesture to identify person 135 (e.g. as belonging to a specific group), followed by identification of information to be output.

[0084] Depending on the embodiment, processor 100 may perform different acts after act 439A, to identify user 135 as per act 433, e.g. in user input extractor 141E. For example, in several embodiments, processor 100 is programmed to perform an act 439D to recognize a face of the user 135 in another image from another camera. In several such embodiments, mobile device 120 includes two cameras, namely a rear camera 121 that images object 132 and a front camera 152 that images a user's face. Moreover, such embodiments store in non-transitory storage media, feature vectors for human faces ("face features") in a database similar or identical to feature vectors for hand gestures ("gesture features"). Accordingly, processor 100 is programmed (by instructions in one or more non-transitory computer readable storage media, such as a disk or ROM) to compare a portion of an image segmented therefrom and corresponding to a face of user 138, with a database of faces. A closest match resulting from the comparison identifies to processor 100 a user identifier, from among user identifiers of multiple faces in the database. Processor 100 then associates (e.g. in act 439C in FIG. 4B) this user identifier with an object identifier e.g. by user input extractor 141E (FIG. 5) preparing an association, in response to detection of a predetermined hand gesture adjacent to object 132.

[0085] An act 439D (FIG. 4B) may be followed by processor 100 performing another additional act, such as 439E to synchronize (or otherwise use) recognition of the user's face with the user gesture. Alternatively, in other embodiments, a user gesture recognized in act 439A may be used to identify person 135 by user input extractor 141E looking up a table 451 of FIG. 4A (similar to mapping 116 in FIG. 1D) as per act 439B (FIG. 4B). After acts 439D and 439E or alternatively after act 439B, processor 100

performs act 439C (see FIG. 4B) to associate a user identifier of person 135 with portable real world object 132. A user identifier that is used in act 439C depends on whether strong or weak identification is implemented by processor 100, for real world object 132.

[0086] Referring back to FIG. 4B, after performance of act 434 to obtain to-be-projected information 119 using at least the user identifier, processor 100 is optionally programmed to operate as information transformer 141T to perform act 435. Specifically, in act 435, processor 100 of mobile device 120 identifies from among a group of predetermined techniques 461, 462, a specific technique 461 to transform the obtained information 188 for projection into scene 130. For example, transformation technique 461 (FIG. 4A) is to project on object 132, whereas transformation technique 462 is to project adjacent to object 132, and one of these techniques is identified prior to projection of information 188. In some embodiments, the specific technique 461 is selected (and therefore identified) automatically based on one or more surface properties of real world object 132 as determined by processor 100, such as surface roughness (or smoothness), orientation of surface normal, color, opacity, etc, whereas in other embodiments processor 100 uses user input (e.g. in the form of spoken words) that explicitly identify a specific technique to be used.

[0087] Then, as per act 436 in FIG. 4B, information transformer 141T uses the specific technique that was identified (e.g. on-object technique 461) to transform information 188 (or 189) in memory 110, and then supply transformed information 188T (or 189T) resulting from use of the specific technique to a projector 122 (FIG. 1E). Next, as per act 437 in FIG. 4B, projector 122 of mobile device 120 projects on object 132 in scene 130 the transformed information 188T (or 189T), which is thereby displayed on object 132 as illustrated in FIG. 1G.

[0088] Various embodiments of information transformer 141T (FIG. 5) may perform any steps using one or more transformation techniques 461 and 462 described above. Moreover, as will be readily apparent in view of this detailed description, any other transformation technique can be used to prepare information 188 for output to a user. For example, as illustrated in FIG. 1H, information 189 may be transformed by another

technique into a first component 189A of transformed information 189T that is projected onto object 132 (namely the text string "Group 1"), and a second component 189B of the transformed information 189T that is projected adjacent to object 132 (namely the text string "Score 0").

[0089] Referring back to FIG. 4B, in act 437 processor 100 operates projector 122 (FIG. 5) to project the transformed information 188T (or 189T) on to object 132 that is identified by the object identifier that was obtained in act 432 (described above). Thereafter, in act 438, processor 100 operating as user input extractor 141E receives and processes additional images in a manner similar to that described above, e.g. to receive user input 482 and store it in memory 110 (FIG. 4A), by recognizing one or more parts of an additional image that includes transformed information 188T, 189T projected into scene 130. The parts of the additional image that are recognized may be, for example, another hand gesture 117 in which only one finger namely index finger 136 is outstretched as illustrated in FIG. 1H. On recognition of this hand gesture 117 (only index finger outstretched), processor 100 operating as user input extractor 141E may determine that the user is now part of Group 1, and therefore now obtains information 189 of Group 1 (see FIG. 4A) in act 434 (FIG. 4B), followed by output to the user (e.g. by transformation and projection) as per one or more of acts 435-437 described above.

[0090] In some embodiments, processor 100 uses recognition of a bottle cap as the portable real world object 132 to invoke execution of information transformer 141T (e.g. instructions to perform acts 431-438 and 439A-439E) from among multiple such softwares 141O, 141I, 141R and 141T. Note that in some embodiments, software 141 is generic to multiple objects, although in other embodiments software 141 (also called information display software) is customized for and individually associated with corresponding objects, such as, for example, a book software 442S (FIG. 4A) associated with a book identified by ID2 and a cup software 443S (FIG. 4A) associated with a cup identified by ID3 as described above in reference to FIG. 4A.

[0091] Note that although table 451 is described above for use with user input in the form of a hand gesture to identify a user, such a table 451 can alternatively be looked up by processor 100 using an identifier of object 132, depending on how table 451 is set up, in various embodiments described herein. Use of table 451 with object identifiers

enables "strong" identification in some embodiments of information identifier 141I, wherein a person 135 identifies to processor 100 (ahead of time), an object 132 that is to be uniquely associated with his/her identity. Other embodiments of processor 100 use both an object identifier as well as user input to look up another table 220, which enables "weak" identification as described herein.

[0092] Some embodiments implement one or more acts of FIG. 4B by performing one or more acts of the type described below in reference to FIGs. 4C and 4D. Note that the acts of FIGs. 4C and 4D described below can alternatively be performed in other embodiments that do not perform any act of FIG. 4B. Some embodiments of processor 100 are programmed to operate as object extractor 141O to track portable real world objects that may be temporarily occluded from view of rear-facing camera 121 that captures image 109, by performing an act 411 (FIG. 4C) to check the image 109 for presence of each object in a set of objects (in database 199 of FIG. 4A) and to identify a subset of these objects as being initially present in image 109.

[0093] Moreover in act 412, processor 100 of some embodiments adds an identifier of an object 132 in the subset to a list 498 (FIG. 4A), and starts a timer for that identifier, and the timer starts incrementing automatically from 0 at a preset frequency (e.g. every millisecond). Therefore, if there are multiple identifiers in list 498 for multiple objects, then correspondingly multiple timers are started, by repeated performance of act 412 for each object in the subset. Next, in act 413, processor 100 checks if list 498 is empty. As list 498 was just populated, it is not empty at this stage and therefore processor 100 performs act 414 (FIG. 4C) and then returns to act 413. Whenever list 498 becomes empty, processor goes from act 413 via the yes branch to act 411 (described above).

[0094] In act 414 (FIG. 4C), additional images are captured into memory 110, and processed by processor 100 in the manner described herein. For example, processor 100 scans through list 498 to check if each object identified in the list 498 is found in the additional image. When an identifier in list 498 identifies an object 132 that is recognized to be present in the additional image, processor 100 resets the timer (started in act 412 as noted above) which starts incrementing automatically again from 0. When an identifier in list 498 identifies an object that is not recognized in the additional image, and if its timer has reached a preset limit (e.g. 10,000 milliseconds) then processor 100 removes the identifier from the list and stops the corresponding timer.

[0095] Accordingly, when an object 132 is absent from view of camera 121 for more than the preset limit, the object 132 is no longer used in the manner described above, to retrieve and display information. Thus, use of a timer as illustrated in FIG. 4C and described above reduces the likelihood that a display of information (e.g. the user's emails) is interrupted when object 132 that triggered the information display is accidentally occluded from view of camera 121, e.g. for the period of time identified in the preset limit. Hence, when a user 135 inadvertently places a hand 138 or other such item between a camera 121 and object 132, the output of information 119T is not stopped, until the preset limit of time passes, which can eliminate jitter from a projection or other display of information 119T, as described herein. The preset limit of time in some embodiments is set by processor 100, based on one or more input(s) from user 135, and hence a different value can be set by user 135 depending on location e.g. a first limit used at a user's home (or office) and a second limit (lower than the first limit) in a public location.

[0096] Referring to FIG. 4D, some embodiments of processor 100 are optionally programmed to operate as information transformer 141T to perform act 421 (FIG. 4D) to compute a value of a property of object 132 identified in image 109, such as size of a surface, shape of the surface, orientation of surface normal, surface smoothness, etc. Then in act 422, processor 100 checks if the property's value satisfies a predetermined test on feasibility for projection on to object 132. For example, processor 100 may check if the surface area of a surface of object 132 is large enough to accommodate the to-be-projected information, and/or if object 132 has a color that is sufficiently neutral for use as a display, and/or a normal at the surface of object 132 is oriented within a preset range relative to a projector 122. Such feasibility tests are designed ahead of time, and programmed into processor 100 to ensure that the to-be-projected information is displayed in manner suitable for user 135 e.g. font size is legible. Numerous other feasibility tests, for information projection on to an object 132 in real world, will be readily apparent, in view of this detailed description.

[0097] If the answer in act 422 is yes, processor 100 goes to act 423 and uses a first technique 461 (FIG. 4A) to generate transformed information 119T (FIG. 5) for projection on to object 132, followed by act 437 (FIG. 4B). Depending on the embodiment, first technique 461 may transform information 119 based on an orientation

(in the three angles, pitch, yaw and roll) of a surface of object 132 relative to orientation of projector 122 to ensure legibility when information 119T is rendered on object 132. A specific manner in which information 119 is transformed can be different in different embodiments, and in some embodiments there is no transformation e.g. when the information 119 is to be displayed on a screen of mobile device 120 (as shown in FIG. 1B). If the answer in act 422 is no, processor 100 goes to act 424 and uses a second technique 462 (FIG. 4A) to generate transformed information 119T for projection adjacent to (but not on to) object 132. After performing one of acts 423 and 424, processor 100 then goes to act 437 (described above in reference to FIG. 4B).

[0098] In some of the described embodiments, one or more of acts 421-424 described above are performed as described in U.S. Application No 13/549,388, Attorney Docket No. Q111570U2os, filed on July 13, 2012, and entitled "Dynamic Selection of Surfaces In Real World For Projection of Information Thereon" which has been incorporated by reference above.

[0099] Note that although the description of certain embodiments refers to processor 100 being a part of a mobile device 120 as an illustrative example, in other embodiments such a processor 100 may be partially or wholly included in one or more other processor(s) and/or other computer(s) that interoperate(s) with such a mobile device 120, e.g. by exchanging information therewith via a cellular link or a WiFi link. Moreover, although one camera 121 is shown in FIG. 1E, depending on the embodiment, one or more cameras (see FIG. 5) may be used. Hence, although certain acts illustrated in FIG. 4B are described for some embodiments as being performed by mobile device 120, some or all of acts in FIG. 4B may be performed by use of one or more computers and/or one or more processors and/or one or more cameras. Therefore, it is to be understood that several such embodiments use one or more devices to perform such act(s), either alone or in combination with one another.

[00100] Processor 100 which is programmed with software in memory 110 as described above in reference to FIG. 4B and/or FIGs. 4C and 4D may be included in a mobile device 120 as noted above. Mobile device 120 may be any device that includes a projector 122 and/or a camera 121, and device 120 may include additional parts that are normally used in any hand held device, e.g. sensors, such as accelerometers,

gyroscopes or the like, which may be used in one or more acts described above, e.g. in determining the pose of mobile device 120 relative to object 132 in the real world.

[00101] In performing the method of FIG. 4B to project information into a scene as described above, there might be different interaction metaphors used. User input 482 that is generated from images captured by camera 121 (FIG. 1E) allows a user to reach into scene 130 and manipulate real world object 132 directly, as opposed to on-screen based interaction, where users interact by directly touching a screen 151 of mobile device 150 (FIG. 1B). Specifically, when image-based user supplied information is obtained as input, methods of the type described above in reference to FIG. 4B enable a user to use his hands in scene 130 with information projected into the real world, as the user is supplying input which changes the information being projected into scene 130.

[00102] User interfaces in information projected into a scene as described herein can have a broad range of applications. Specifically, projected user interfaces can be used to generate user input 482 (FIG. 4A) by projecting information 322 including screen 321 and keyboard image 333 (FIG. 3C) similar to real world typing using a real keyboard. A projected user interface allows a user to supply input to select between different software for execution and display of projected information and/or select between different groups of users to play games, and in general to specify various parameters to software being executed by a processor 100 that generates the information which is projected into scene 130 (e.g. see FIG. 1G).

[00103] Hence, several embodiments of mobile device 120 as described herein reverse a flow of information between (A) user interfaces and (B) user input (relative to a conventional flow). Specifically, in several examples of the type noted above in reference to FIGs. 2A-2C, instead of users explicitly looking for information to be displayed, several embodiments of device 120 automatically obtain and display interactive information, e.g. by projection on real world surfaces using an embedded mobile projector. Other embodiments may display information as described herein on a screen that is supported on an eye-glass frame worn by a user, for example. Still other embodiments may display information as described herein on a screen that forms an integral portion of a smart phone (such as a touch screen), in the normal manner.

[00104] Various descriptions of implementation details of some embodiments of mobile device 120 are merely illustrative and not limiting. For example, depending on the embodiment, any method may be used by mobile device 120 to receive input from a user, e.g. an IR camera may be used to receive user input in the form of hand gestures. Moreover, various types of hand gesture recognition systems may be implemented in several embodiments of a mobile device 120 as described herein. In certain embodiments, an embedded projector in mobile device 120 projects a cell phone's normal display on everyday surfaces such as a surface of a wall or a surface of a desk, with which a user interacts using hand gestures.

[00105] It should be understood that mobile device 120 may be any electronic device that is portable by hand, such as a cellular or other wireless communication device, personal communication system (PCS) device, personal navigation device (PND), Personal Information Manager (PIM), Personal Digital Assistant (PDA), laptop, tablet, or an eye glass frame that supports a display to be worn on a person's face, a headset, a camera, or other suitable mobile device that is capable of imaging scene 130 and/or projecting information into scene 130. In some embodiments, a single device 120 includes both camera 121 and projector 122 whereas in other embodiments one such device includes camera 121 and another such device includes projector 122 and both devices communicate with one another either directly or via a computer (not shown).

[00106] In several embodiments, a prototype of mobile device 120 is built with custom hardware (PCB) board taped onto the back of a smartphone (e.g. GALAXY NEXUS available from Samsung Electronics Co. Ltd). One such approach performs computation in the infrared (IR) spectrum. In some such embodiments, hand and body tracking is robust and very accurate, although additional hardware may be integrated within such a smartphone 120 used for display of information. Some embodiments of device 120 use IR sensors (e.g. in an IR camera) that have been proven to work on commercially successful platforms, such as the Xbox Kinect. Certain embodiments of device 120 implement augmented reality (AR) applications using marker patterns, such as checkerboard for camera calibration and detection of objects within a scene of real world, followed by use of object identifiers to display information, as described herein.

[00107] Depending on the embodiment, mobile device 120 may be programmed with software 141 that uses a mobile-projector system in combination with a camera. An embedded projector 122 is used in such embodiments to display information 119T on everyday surfaces such as a wall, with which users interact using hand gestures. Also in some embodiments, mobile device 120 is operatively coupled to an external IR camera 1006 that tracks an IR laser stylus (not shown), or gloves with one or more IR LEDs 1121, 1122 (FIG. 5) mounted at the finger tips (also called IR gloves).

[00108] External IR camera 1006 is used in some embodiments in a manner similar or identical to receipt of IR images and tracking of objects within the images by use of an IR camera in a remote control device (also called "Wiimote" or "Wii Remote") for gaming console Wii, available from Nintendo Co. Ltd.. So, IR camera 1006 may be used in some embodiments as described in a section entitled "Tracking Your Fingers with the Wiimote" in a 2-page article available at http://johnnylee.net/ projects/ wii/ as available on April 9, 2012, which is incorporated by reference herein in its entirety. Alternatively, some non-IR embodiments of device 120 use one or more normal RGB (red-green-blue) CMOS cameras 121 (FIG. 5) to capture an image of scene 130 including object 132.

[00109] An object extractor 141O in a mobile device 120 of the type described herein may use any known object recognition method, based on "computer vision" techniques. Such a mobile device120 may also include means for controlling operation of a real world object 132 (that may be electronic) in response to user input of the type described above such as a toy equipped with an IR or RF transmitter or a wireless a transmitter enabled to receive and/or transmit one or more signals over one or more types of wireless communication networks such as the Internet, WiFi, cellular wireless network or other network.

[00110] As illustrated in FIG. 5, mobile device 120 may additionally include a graphics engine 1004 to generate information 119 to be output, an image processor 1005 to process image(s) 109 and/or transform information 119, and a read only memory (ROM) 1007 to store firmware and/or constant data. Mobile device 120 may also include a disk 1008 to store software and/or database 199 for use by processor 100. Mobile device 120 may further include a wireless transmitter and receiver 1010 and/or

any other communication interfaces 1009, sensors 1003, a touch screen 1001 or other screen 1002, a speaker 1111 and a microphone 1112.

[00111] Some embodiments of user input extractor 141E (FIG. 5) sense user input in the form of hand gestures in images generated by an infra-red (IR) camera that tracks an IR laser stylus or gloves with IR LEDs (also called IR gloves), while certain other embodiments of user input extractor 141E sense hand gestures using an existing camera (e.g. in a normal cell phone) that captures images of a user's fingers. Specifically, in some embodiments (e.g. IntuoIR), an external PCB board 1130 (FIG. 5) having mounted thereon an ARM Cortex processor (not shown) is interfaced with an IR camera 1006 (FIG. 5) and a Bluetooth module (not shown). Hand tracking data from IR camera 1006 is sent via Bluetooth to a smartphone 1140 in device 120 that has a touch screen 1001 (e.g. HTC Explorer available from HTC Corporation).

[00112] Hence, certain embodiments of device 120 include PCB board 1130 mounted on and operatively coupled to a smartphone 1140 (FIG. 5). PCB board 1130 includes IR camera 1006, such as mbed LPC1768 available from Foxconn, e.g. Mbed as described at http://mbed.org/nxp/lpc1768/ and a Bluetooth chipset e.g. BlueSMiRF Gold as described at http://www.sparkfun.com/products/582. Mbed (ARM processor) is used in some embodiments of PCB 1130 to collect data from IR camera 1006 and transmit co-ordinates of brightest points to smartphone 1140 (e.g. via Bluetooth link 1131).

[00113] In some embodiments, user input captured in images by a camera is extracted therefrom by the smartphone in device 120 performing gesture recognition on data received from an infra-red (IR) sensor, as described in an article entitled "iGesture: A General Gesture Recognition Framework" by Signer et al,, In Proc. ICDAR '07, 5 pages which is incorporated by reference herein in its entirety.

[00114] Hence, some embodiments of user input extractor 141E (FIG. 5) operate with a user wearing infra-red (IR) gloves that are identified in another image generated by an IR camera. An IR camera of such embodiments may be externally coupled to a smartphone in mobile device 120 in some embodiments while in other embodiments the IR camera is built into the smartphone. Some embodiments operate with the user 135 using an IR laser stylus 1135 whose coordinates are detected by device 120 in any

manner known in the art. Still other embodiments of user input extractor 141E (FIG. 5) receive user input in other forms as noted above, e.g. as audio input from microphone 1112.

[00115] In certain embodiments, user input extractor 141E (FIG. 5) processes a frame of video captured by a camera to obtain user input in the form of hand gestures, by segmenting each image into one or more areas of interest, such as a user's hands. Any known method can be modified for use in user input extractor 141E as described herein, to remove background noise, followed by identification of a portion of the image which contains the user's hand, which is then used to generate a binary image (also called a "blob"). A next step in some embodiments of user input extractor 141E is to calculate locations (e.g. coordinates) of the user's fingers within the blob.

[00116] An IR camera 1006 is not used in certain embodiments wherein a normal RGB camera is used instead to generate one or more images 109 which contain user input. The user input is extracted from images 109 by user input extractor 141E (FIG. 5) performing one of two methods as follows. A first method is of the type described in a 4-page article entitled "HandVu: Vision-based Hand Gesture Recognition and User Interface" at http://www.movesinstitute.org/~kolsch /HandVu/HandVu.html" as available on April 9, 2012, which is incorporated by reference herein in its entirety. A second method is of the type described in another 4-page article entitled "A Robust Method for Hand Gesture Segmentation and Recognition Using Forward Spotting Scheme in Conditional Random Fields" by Mahmoud Elmezain, Ayoub Al-Hamadi, and Bernd Michaelis, in International Conference on Pattern Recognition, 2010, which is incorporated by reference herein in its entirety. Hence, such embodiments that use an existing RGB camera in a normal smartphone may use a combination of skin segmentation, graph cut and recognition of hand movement, to detect hand gestures.

[00117] For recognition of hand gestures, some embodiments of user input extractor 141E (FIG. 5) are designed to use a supervised learning approach in an initialization phase of device 120. In the supervised learning approach, user input extractor 141E learns different gestures from input binary images (e.g. consisting of a user's hands) during initialization, and generates a mathematical model to be used to identify gestures in images generated during normal operation (after the initialization phase) by using Support Vector Machines (SVM) of the type known in the prior art.

Experimental results show that several methods of the type presented herein work well in real time and under changing illumination conditions.

[00118] As noted above, some embodiments use an infrared (IR) camera 1006 (FIG. 5) to extract portions of an image 109 that correspond to a user's fingers as blobs. In several embodiments, a user holds an IR light source, such as a laser pointer or alternatively the user wears IR gloves. Specifically, in certain embodiments, a user wears on a hand 138 (FIG. 1E) a glove (not shown) with an IR LED 1121, 1122 (FIG. 5) on each finger 136, 137 (FIG. 1E). Detection of position of one IR LED 1121 (FIG. 5) on left index finger 136 (FIG. 1E) by IR camera 1006 (FIG. 5) is used with detection of another IR LED 1122 (FIG. 5) on middle finger 137 (FIG. 1E) also by the IR camera 1006 (FIG. 5) to identify as a blob, a human hand image 138I (FIG. 4A) in image 109.

[00119] After device 120 detects one or more such blob(s), gesture recognition is performed by processor 100 executing software to operate as user input extractor 141E (FIG. 5) as described herein. Specifically, in some embodiments of mobile device 120, co-ordinates of IR LEDs 1121 and 1122 generated by IR camera 1006 are used by processor 100 to identify a blob (e.g. human hand image 138I in FIG. 4A) corresponding to a hand 138 in an image 109 from an RGB camera 121 (which is thereby additionally used), and the blob is then used by user input extractor 141E to generate features (also called feature vectors such as a "swipe" gesture or a "thumbs up" gesture) that are then matched to corresponding features in a database, to identify as user input, a hand gesture in image 109.

[00120] Some embodiments extract blobs in two-dimensional (2D) space due to limitations inherent in design (for a similar setup, see the 2-page article described above in reference to http://johnnylee.net/projects/wii/). Certain embodiments of user input extractor 141E (FIG. 5) perform blob detection in 3D space, using a depth camera. In some embodiments (called "IntuoIR"), images from an IR camera 1006 are used to translate hand movements to specific co-ordinates within native applications running on a smartphone included in device 120. Several embodiments implement a simple camera calibration technique, similar to the techniques described in the 2-page article at http://johnnylee.net/projects/wii/.

[00121] Some embodiments of mobile device 120 generate a depth map of a scene by use of a 3D Time-of-Flight (TOF) camera of the type known in the prior art. A Time-of-Flight camera is used in certain embodiments, to measure a phase difference between photons coming onto a sensor, which in turn provides a distance between the sensor and objects in the scene.

[00122] Other embodiments of device 120 also use Time-of-flight cameras, e.g. as described by Freedman in US Patent Publication 2010/0118123 entitled "Depth Mapping using Projected Patterns" which is incorporated by reference herein in its entirety. Such embodiments of device 120 may use projector 122 (FIG. 5) to shine an infrared (IR) light pattern on object 132 in the scene. A reflected light pattern is observed in such embodiments by a depth camera 121, which generates a depth map. Hence, in some embodiments of mobile device 120, a depth map is used to enhance segmentation of image to identify areas that contain a user's face, the user's hand and/or one or more objects in an image received from an RGB camera 121 (which is thereby additionally used).

[00123] A device 120 of some embodiments uses projector 122 to project information 119T onto a surface of object 132, followed by capture of a user's finger movements as follows. By using a laser pointer (or gloves with IR LEDs on fingers) and the IR camera, such embodiments implement a motion capture system. Depending on a region where information is projected, an IR camera 1006 is calibrated. After camera calibration, certain embodiments generate a one-to-one mapping between the screen resolution of device 120 and the user's hand movements. When there is an IR light point inside a projected display area, the IR camera 1006 captures the brightest IR point. The coordinates of such points are processed inside an application layer or kernel layer. Based on the processed data, the user input is determined by processor 100.

[00124] Experimental results for a method using IR camera 1006 coupled to or included in device 120 of some embodiments were obtained with the number of samples or "frame rate" of the IR camera 1006 at approximately 120 samples per second, for use in gesture recognition in real time. Two factors that affect the performance of some embodiments of mobile device 120 are: distance (between IR camera 1006 and the IR LEDs 1121, 1122) and light conditions. When IR camera 1006 faces high intensity light source, false coordinates are generated 87.18% of time. And when a prototype of

the type described herein (FIG. 5) is present in ambient environment, there is nearly no noise observed. More importantly, if a room is uniformly lit at any intensity, the noise is close to 0%. Several of these tests were performed with an IR light source 1121 directly facing the IR camera 1006.

[00125] The above description presented certain methods for hand gesture recognition that are used in some embodiments of the type described herein. As noted above, one of the methods (IntuoIR) is based on tracking IR light sources, e.g. IR LEDs 1121 and 1122 (FIG. 5) mounted at finger tips of a glove. One limitation of some embodiments is a need for external hardware (i.e. hardware not present in a conventional smartphone), such as an IR camera 1006 (FIG. 5). Certain embodiments of mobile device 120 use one or more Infrared time-of-flight (TOF) camera(s) 1006 instead of or in addition to a CMOS infrared camera 1006. In some such embodiments, background noise may be present in images 109 being captured and filtered by device 120. Such embodiments may utilize a frame buffer of a screen in mobile device 120 and perform stereo correspondence to reduce such noise. Several embodiments of device 120 implement any known techniques to reduce background noise arising from use of stereo cameras (e.g. 3D cameras) 121.

[00126] Mobile device 120 of several described embodiments may also include means for remotely controlling a real world object which may be a toy, in response to user input e.g. by use of transmitter in transceiver 1010, which may be an IR or RF transmitter or a wireless a transmitter enabled to transmit one or more signals over one or more types of wireless communication networks such as the Internet, WiFi, cellular wireless network or other network. Of course, mobile device 120 may include other elements, such as a read-only-memory 1007 which may be used to store firmware for use by processor 100.

[00127] Also, depending on the embodiment, various functions of the type described herein may be implemented in software (executed by one or more processors or processor cores) or in dedicated hardware circuitry or in firmware, or in any combination thereof. Accordingly, depending on the embodiment, any one or more of object extractor 141O, information identifier 141I, information retriever 141R, information transformer 141T and segmentation module 141S illustrated in FIG. 5 and

described above can, but need not necessarily include, one or more microprocessors, embedded processors, controllers, application specific integrated circuits (ASICs), digital signal processors (DSPs), and the like. The term processor is intended to describe the functions implemented by the system rather than specific hardware. Moreover, as used herein the term "memory" refers to any type of computer storage medium, including long term, short term, or other memory associated with the mobile platform, and is not to be limited to any particular type of memory or number of memories, or type of media upon which memory is stored.

[00128] Hence, methodologies described herein may be implemented by various means depending upon the application. For example, these methodologies may be implemented in firmware in ROM 1007 (FIG. 5) or software, or hardware or any combination thereof. For a hardware implementation, the processing units may be implemented within one or more application specific integrated circuits (ASICs), digital signal processors (DSPs), digital signal processing devices (DSPDs), programmable logic devices (PLDs), field programmable gate arrays (FPGAs), processors, controllers, micro-controllers, microprocessors, electronic devices, other electronic units designed to perform the functions described herein, or a combination thereof. For a firmware and/or software implementation, the methodologies may be implemented with modules (e.g., procedures, functions, and so on) that perform the functions described herein.

[00129] Any machine-readable medium tangibly embodying computer instructions may be used in implementing the methodologies described herein. For example, software 141 (FIG. 5) may include program codes stored in memory 110 and executed by processor 100. Memory 110 may be implemented within or external to the processor 100. If implemented in firmware and/or software, the functions may be stored as one or more computer instructions or code on a computer-readable medium. Examples include nontransitory computer-readable storage media encoded with a data structure (such as a sequence of images) and computer-readable media encoded with a computer program (such as software 141 that can be executed to perform the method of FIGs. 1C, 2A, 3A, 3F, and 4B-4D).

[00130] Computer-readable media includes physical computer storage media. A

storage medium may be any available medium that can be accessed by a computer. By way of example, and not limitation, such computer-readable media can comprise RAM, ROM, Flash Memory, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other medium that can be used to store program code in the form of software instructions (also called "processor instructions" or "computer instructions") or data structures and that can be accessed by a computer; disk and disc, as used herein, includes compact disc (CD), laser disc, optical disc, digital versatile disc (DVD), floppy disk and blu-ray disc where disks usually reproduce data magnetically, while discs reproduce data optically with lasers. Combinations of the above should also be included within the scope of computer-readable media.

[00131] Although the present invention is illustrated in connection with specific embodiments for instructional purposes, the present invention is not limited thereto. Hence, although item 120 shown in FIG. 5 of some embodiments is a mobile device, in other embodiments item 120 is implemented by use of form factors that are different, e.g. in certain other embodiments item 120 is a mobile platform (such as a tablet, e.g. iPad available from Apple, Inc.) while in still other embodiments item 120 is any electronic device or system. Illustrative embodiments of such an electronic device or system 120 may include multiple physical parts that intercommunicate wirelessly, such as a processor and a memory that are portions of a stationary computer, such as a lap-top computer, a desk-top computer, or a server computer communicating over one or more wireless link(s) with sensors and user input circuitry enclosed in a housing that is small enough to be held in a hand.

[00132] Although several aspects are illustrated in connection with specific embodiments for instructional purposes, various embodiments of the type described herein are not limited thereto. Various adaptations and modifications may be made without departing from the scope of the described embodiments. Therefore, the spirit and scope of the appended claims should not be limited to the foregoing description.

CLAIMS

What is claimed is:

1. A method comprising:

receiving from a camera an image of a scene, the scene comprising an object in real world, the object being portable by hand;

processing the image using data on a plurality of predetermined portable real world objects, to detect at least a portion of the image corresponding to the object and obtain an object identifier identifying the object uniquely among the plurality;

one or more processors using the object identifier with at least an association in a set of associations to lookup a user identifier that identifies a user at least partially among a plurality of users; and

using at least the user identifier to obtain and store in computer memory, information to be output to the user identified by the user identifier.

2. The method of claim 1 wherein:

the user identifier identifies from among the plurality of users, a first group of users including the user;

the information to be output is common to all users in the first group; and

the plurality of users includes at least a second group of users different from the first group.

3. The method of claim 1 further comprising:

detecting in the image, a gesture adjacent to the object;

the one or more processors using the gesture to look up the user identifier from a mapping between hand gestures and user identifiers; and

preparing the association in the set, to associate the user identifier with the object identifier, in response to the detecting.

4. The method of claim 1 wherein:

the user identifier uniquely identifies the user among the plurality of users; and

the information is specific to the user.

5. The method of claim 1 further comprising:

displaying an authentication screen adjacent to the object in the scene;

using user input in the authentication screen to determine the user identifier; and

preparing the association in the set, to associate the user identifier with the object identifier, in response to the user input being authenticated.

6. The method of claim 1 wherein the portion is hereinafter first portion, and the image is hereinafter first image, the method further comprising:

segmenting from a second image obtained from a second camera, a second portion thereof corresponding to a face of the user;

comparing the second portion with a database of faces, to obtain the user identifier selected from among user identifiers of faces in the database; and

preparing the association in the set, to associate the user identifier with the object identifier, in response to detecting a predetermined gesture adjacent to the object.

7. The method of claim 1 further comprising:

projecting at least a portion of the information into the scene.

8. The method of claim 1 wherein said processing the image comprising:

checking the image for presence of each object among the plurality of predetermined portable real world objects, to identify a subset of the plurality of predetermined portable real world objects;

identifying in a list each object in the subset and starting a timer for the each object in the subset; and

while the list is not empty, repeatedly capturing additional images of the scene and scanning through the list to determine whether each object identified therein is present in each additional image and if so resetting the timer for the each object in the list and removing an object from the list when the timer for the object reaches a preset limit; and

when the list is empty, returning to the checking.

9. One or more non-transitory computer readable storage media comprising:

instructions to one or more processors to receive an image of a scene from a camera, the scene comprising an object in real world, the object being portable by hand;

instructions to the one or more processors to process the image using data on a plurality of predetermined portable real world objects, to obtain an object identifier uniquely identifying the object from among the plurality;

instructions to the one or more processors to use the object identifier with at least an association in a set of associations to obtain a user identifier that identifies a user of the object at least partially among a plurality of users; and

instructions to the one or more processors to obtain and store in computer memory, information to be output to the user, by using at least the user identifier.

10. The one or more non-transitory computer readable storage media of claim 9 wherein:

the user identifier identifies from among the plurality of users, a first group of users including the user; and

the information is common to all users in the first group.

11. The one or more non-transitory computer readable storage media of claim 9 further comprising:

instructions to the one or more processors to detect a gesture adjacent to the object in the scene;

instructions to the one or more processors to use the gesture to look up the user identifier from a mapping of hand gestures to user identifiers; and

instructions to the one or more processors to prepare the association in the set, to associate the user identifier with the object identifier.

12. The one or more non-transitory computer readable storage media of claim 9 wherein:

the user identifier uniquely identifies the user in the plurality of users; and

the information is unique to the user.

13. The one or more non-transitory computer readable storage media of claim 9 further comprising:

instructions to display an authentication screen adjacent to the object in the scene; and

instructions to use user input in the authentication screen to determine the user identifier; and

instructions to prepare the association in the set, to associate the user identifier with the object identifier, in response to detecting the user input to be authenticated.

14. The one or more non-transitory computer readable storage media of claim 9 wherein the portion is hereinafter first portion, and the image is hereinafter first image, the one or more non-transitory computer readable storage media further comprising:

instructions to the one or more processors to segment from a second image obtained from a second camera, a second portion thereof corresponding to a face of the user;

instructions to the one or more processors to compare the second portion with a database of faces to identify the user identifier from among user identifiers of faces in the database; and

instructions to the one or more processors to prepare the association in the set, to associate the user identifier with the object identifier, in response to detecting the user performing a predetermined gesture adjacent to the object.

15. The one or more non-transitory computer readable storage media of claim 9 wherein:

the instructions to output comprise instructions to project the portion of the information into the scene.

16. The one or more non-transitory computer readable storage media of claim 9 wherein the instructions to the one or more processors to process the image comprise:

instructions to check the image for presence of each object among the plurality of predetermined portable real world objects, to identify a subset of the plurality of predetermined portable real world objects;

instructions to identify in a list each object in the subset and starting a timer for the each object in the subset; and

while the list is not empty, instructions to repeatedly capture additional images of the scene and scanning through the list to determine whether each object identified

therein is present in each additional image and if so resetting the timer for the each object in the list and removing an object from the list when the timer for the object reaches a preset limit; and

instructions responsive to the list being empty, to return to the instructions to check.

17. One or more devices comprising:

a camera;

one or more processors, operatively coupled to the camera;

memory, operatively coupled to the one or more processors; and

software held in the memory that when executed by the one or more processors, causes the one or more processors to:

receive an image of a scene from the camera, the scene comprising an object in real world, the object being portable by hand;

process the image using data on a plurality of predetermined portable real world objects, to obtain an object identifier uniquely identifying the object from among the plurality;

use the object identifier with at least an association in a set of associations to obtain a user identifier that identifies a user of the object at least partially among a plurality of users; and

obtain and store in the memory, information to be displayed, by using at least the user identifier.

18. The one or more devices of claim 17 wherein the software further causes the one or more processors to:

detect a gesture adjacent to the object in the scene;

use the gesture to look up the user identifier from a mapping of hand gestures to user identifiers; and

prepare the association in the set, to associate the user identifier with the object identifier.

19. The one or more devices of claim 17 wherein the software further causes the one or more processors to:

display an authentication screen adjacent to the object in the scene; and

use user input in the authentication screen to determine the user identifier; and

prepare the association in the set, to associate the user identifier with the object identifier, in response to detecting the user input to be authenticated.

20. The one or more devices of claim 17 wherein the portion is hereinafter first portion, and the image is hereinafter first image and the software further causes the one or more processors to:

segment from a second image obtained from a second camera, a portion thereof corresponding to a face of the user;

compare the second portion with a database of faces to identify the user identifier from among user identifiers of faces in the database; and

prepare the association in the set, to associate the user identifier with the object identifier, in response to detecting the user performing a predetermined gesture adjacent to the object.

21. The one or more devices of claim 17 wherein the software that causes the one or more processors to process the image comprises software to cause the one or more processors to:

check the image for presence of each object in the plurality of predetermined portable real world objects, to identify a subset therein as being present in the image;

add an identifier of each object in the subset to a list, starting a timer for each identifier of each object in the subset; and

while the list is not empty, repeatedly scan through the list to determine if each object identified therein is present in each additional image of the scene and if so resetting the timer for each identifier in the list and removing any identifier from the list when the timer for said any identifier reaches a preset limit; and

when the list is empty, repeat the check.

22. A system comprising a processor coupled to a memory and a camera, the system comprising:

means for processing an image of a scene received from the camera, the scene comprising an object in real world, the object being portable by hand, the image being processed by using data on a plurality of predetermined portable real world objects, to obtain an object identifier uniquely identifying the object from among the plurality;

means for using the object identifier with at least an association in a set of associations to obtain a user identifier that identifies a user of the object at least partially among a plurality of users; and

means for obtaining and storing in the memory, information to be displayed, by using at least the user identifier.

23. The system of claim 22 further comprising:

means for detecting a gesture adjacent to the object in the scene;

means for using the gesture to look up the user identifier from a mapping of hand gestures to user identifiers; and

means for preparing the association in the set, to associate the user identifier with the object identifier.

24. The system of claim 22 further comprising:

means for displaying an authentication screen adjacent to the object in the scene;

means for using user input in the authentication screen to determine the user identifier; and

means for preparing the association in the set, to associate the user identifier with the object identifier, in response to detecting the user input to be authenticated.

FIG. 1A

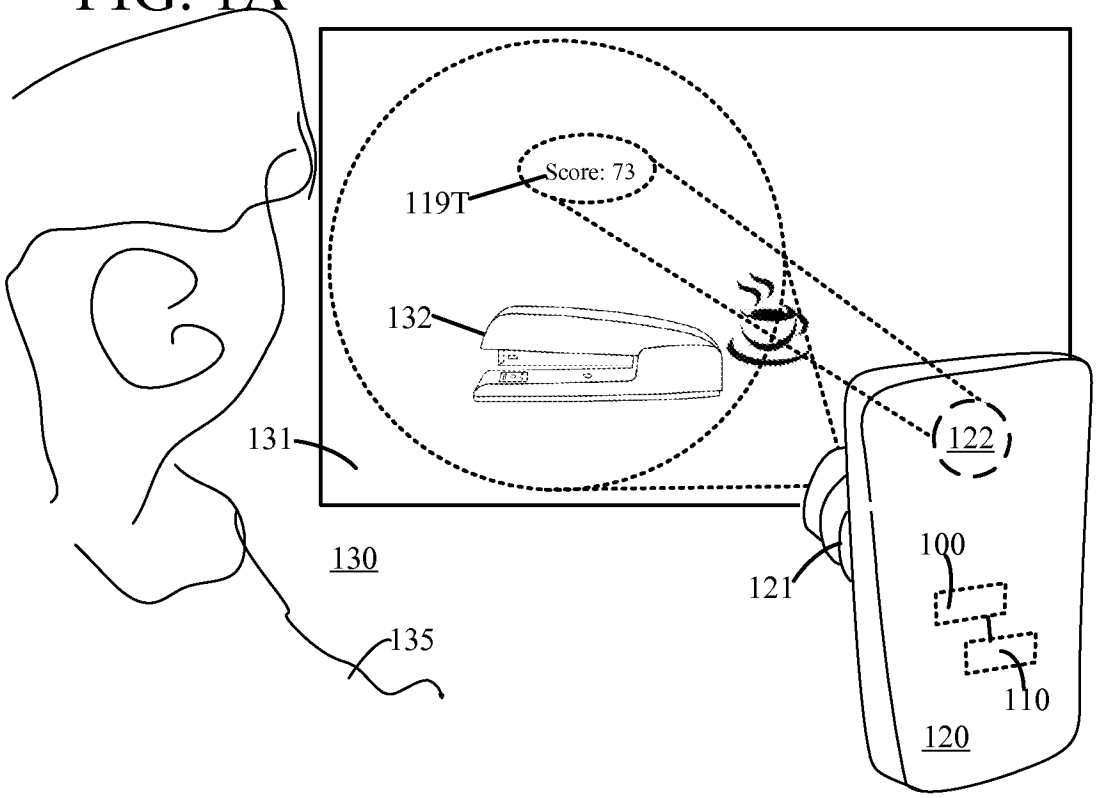

Score: 73
119T
132
131
122
100
130
121
135
110
120

FIG. 1B

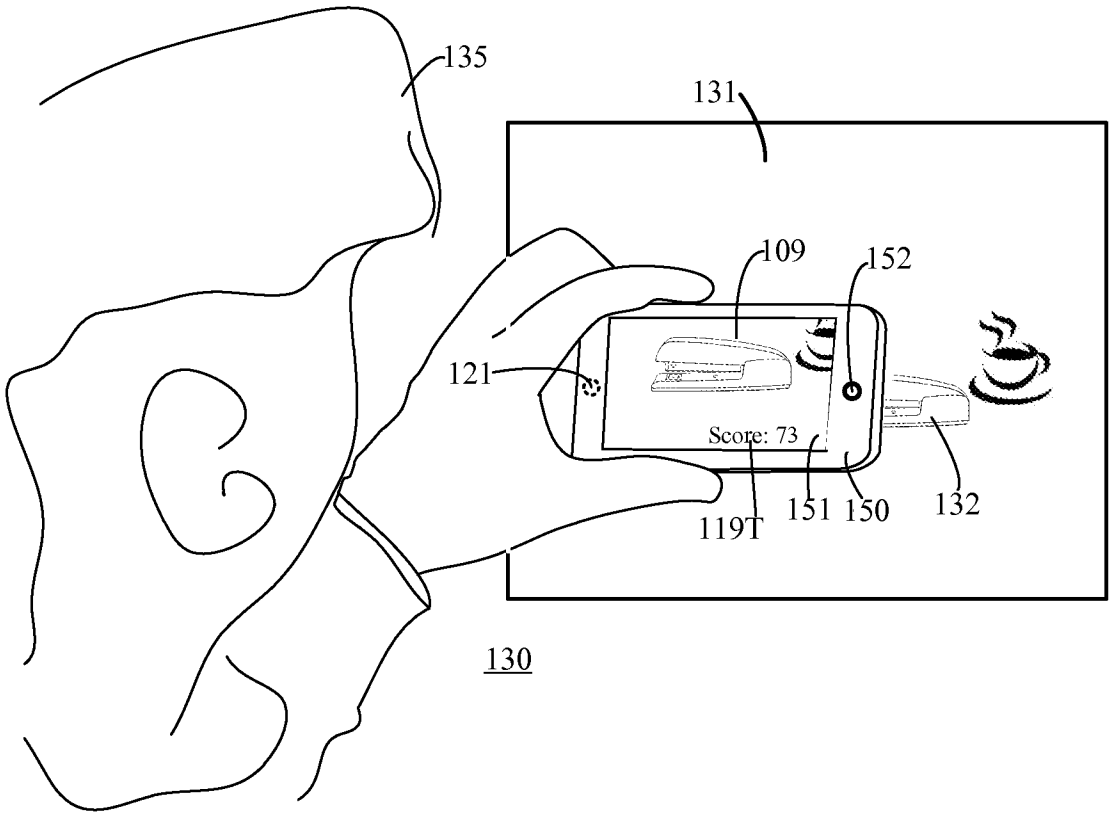

135
131
109
152
121
Score: 73
119T
151
150
132
130

FIG. 1C

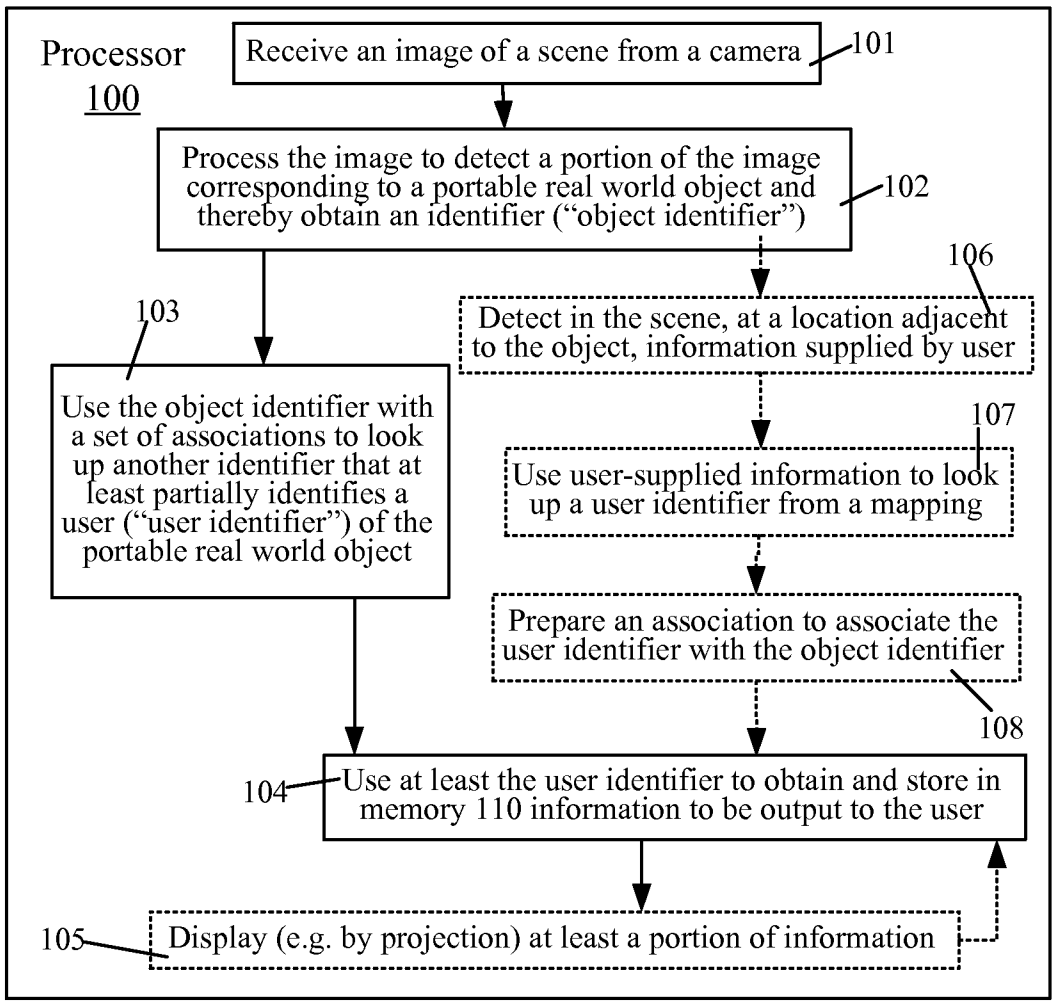

Processor
100
Receive an image of a scene from a camera
101
Process the image to detect a portion of the image corresponding to a portable real world object and thereby obtain an identifier ("object identifier")
102
106
Detect in the scene, at a location adjacent to the object, information supplied by user
103
Use the object identifier with a set of associations to look up another identifier that at least partially identifies a user ("user identifier") of the portable real world object
107
Use user-supplied information to look up a user identifier from a mapping
Prepare an association to associate the user identifier with the object identifier
108
104
Use at least the user identifier to obtain and store in memory 110 information to be output to the user
105
Display (e.g. by projection) at least a portion of information

FIG. 1D

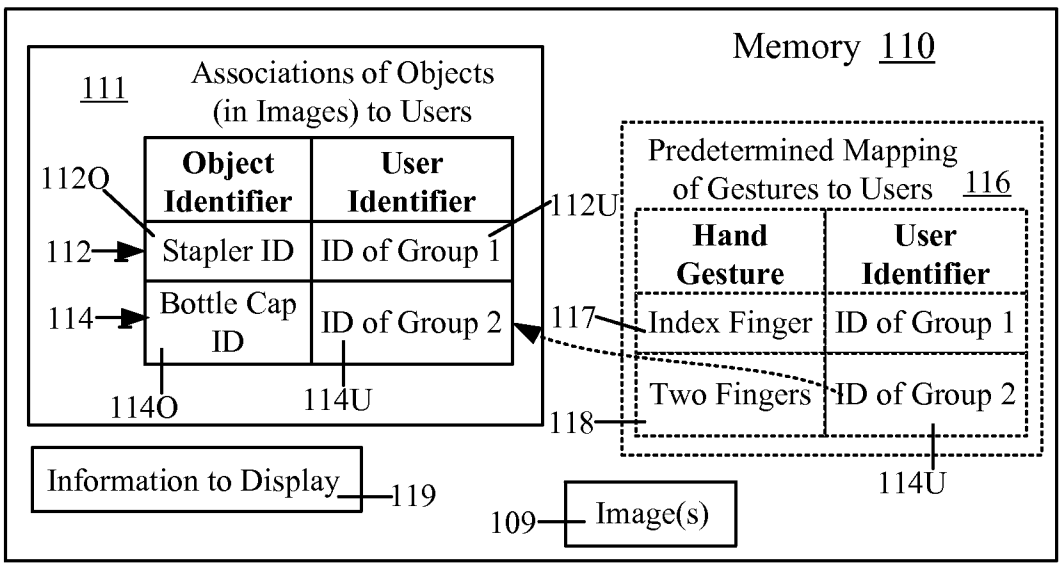

Memory 110
111
Associations of Objects (in Images) to Users
Object Identifier
User Identifier
112O
112U
112
Stapler ID
ID of Group 1
114
Bottle Cap ID
ID of Group 2
114O
114U
Predetermined Mapping of Gestures to Users 116
Hand Gesture
User Identifier
117
Index Finger
ID of Group 1
Two Fingers
ID of Group 2
118
114U
Information to Display
119
109
Image(s)

FIG. 1E

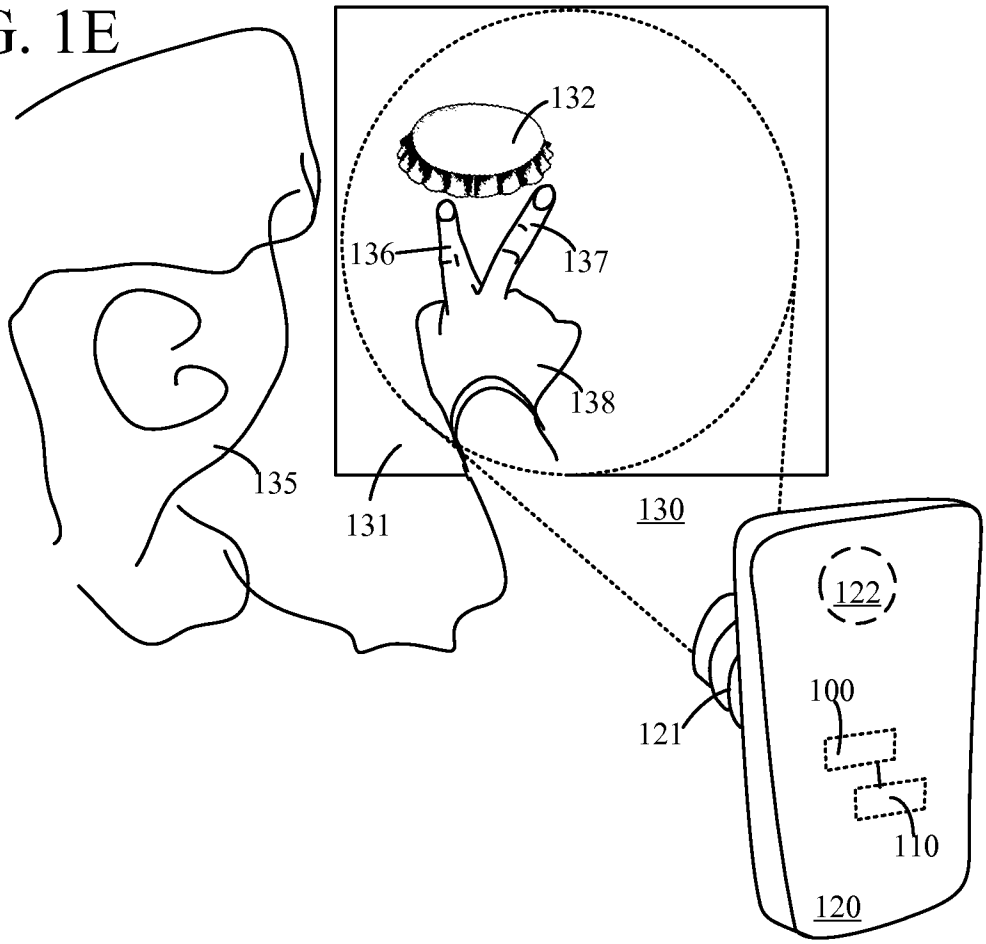

132
136
137
138
135
131
130
122
100
121
110
120

FIG. 1F

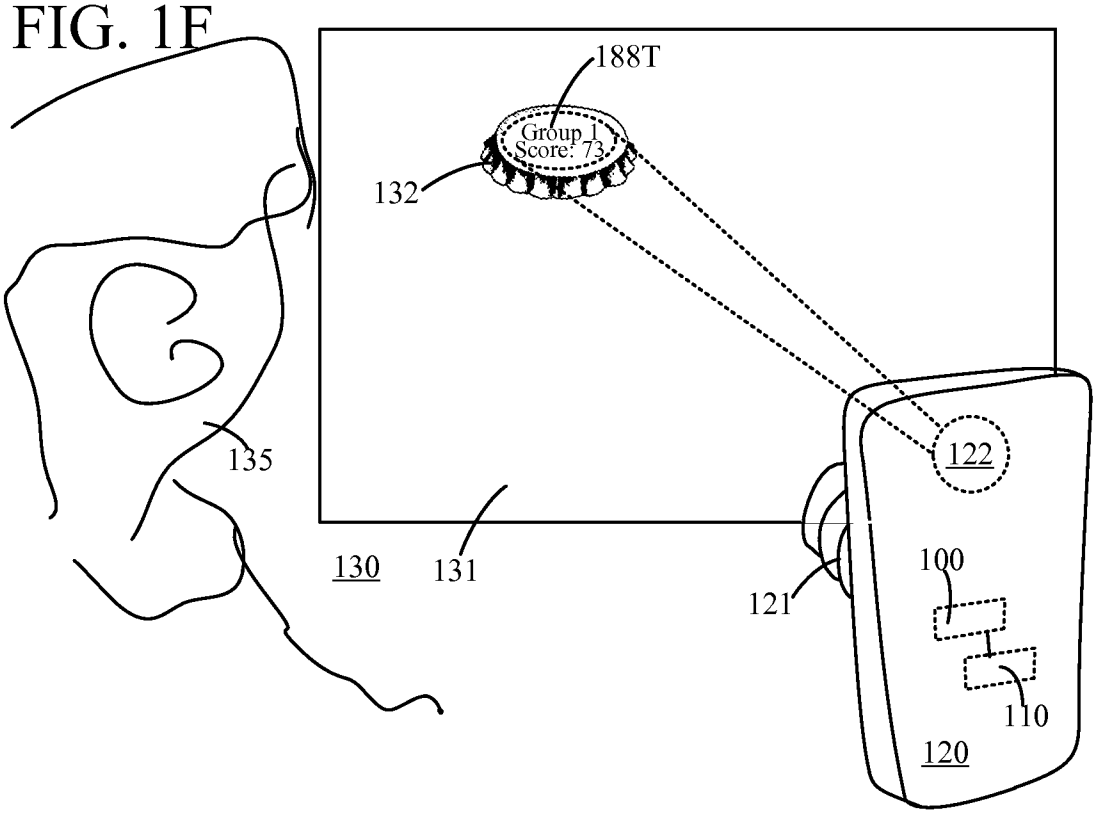

188T
Group 1
Score: 73
132
135
130
131
122
100
121
110
120

FIG. 1G

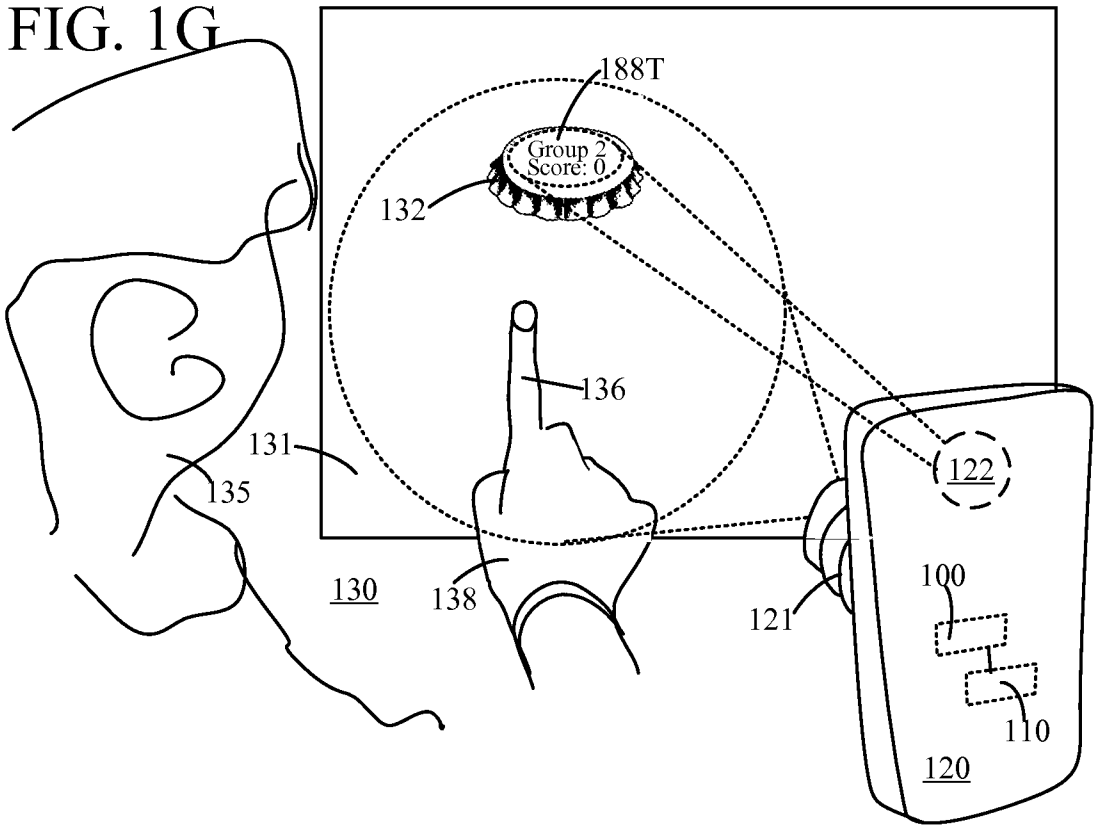

188T
Group 2
Score: 0
132
136
131
135
122
100
130
138
121
110
120

FIG. 1H

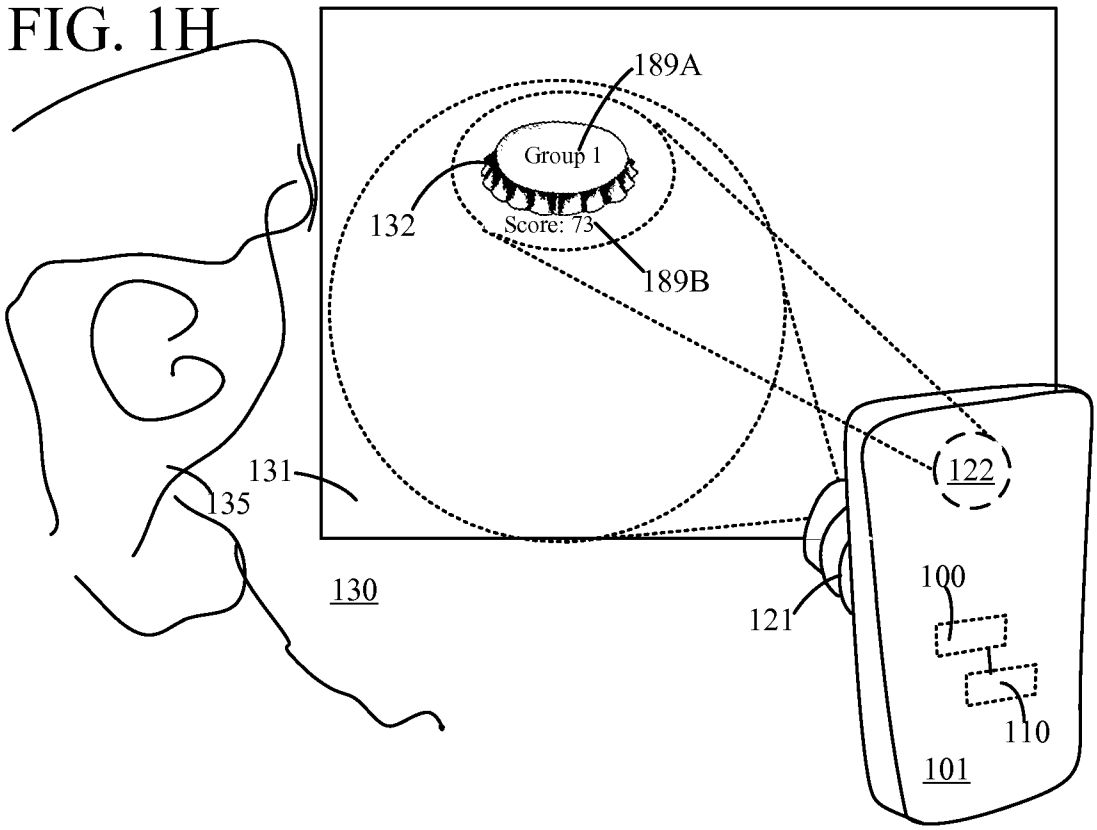

189A
Group 1
132
Score: 73
189B
131
135
122
100
130
121
110
101

FIG. 1I

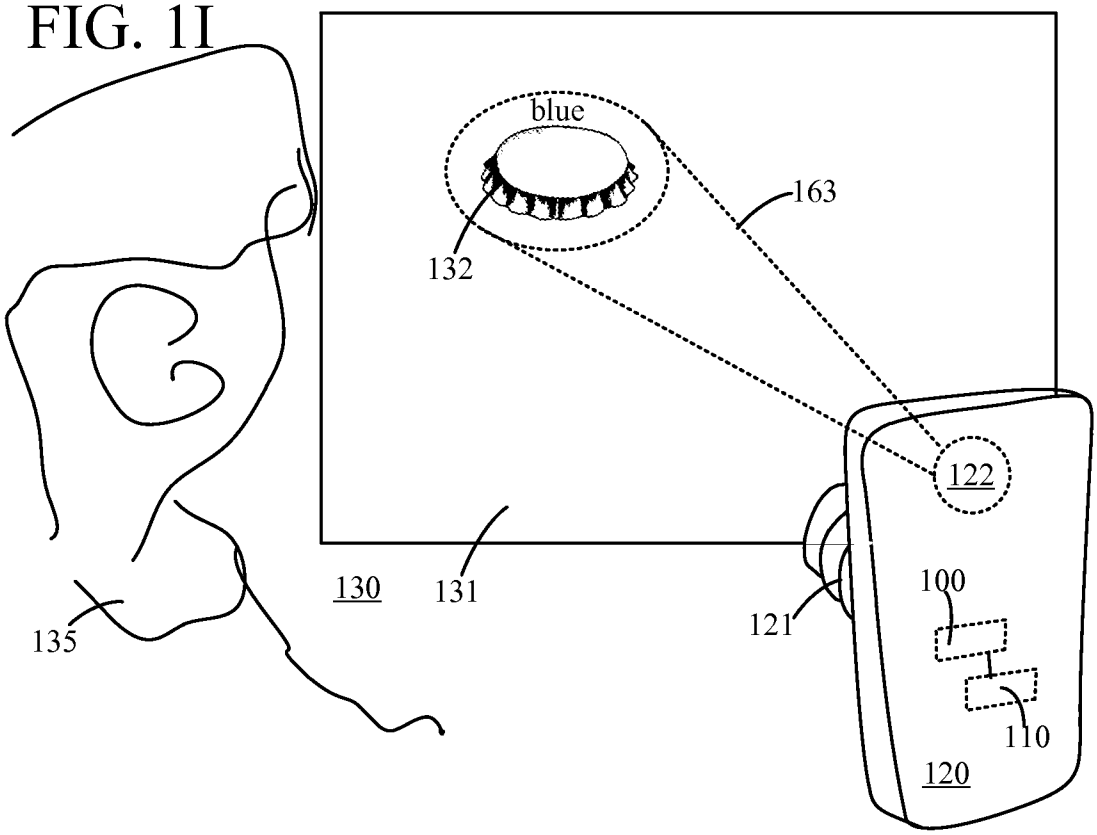

blue
163
132
122
130
131
121
100
110
120
135

FIG. 1J

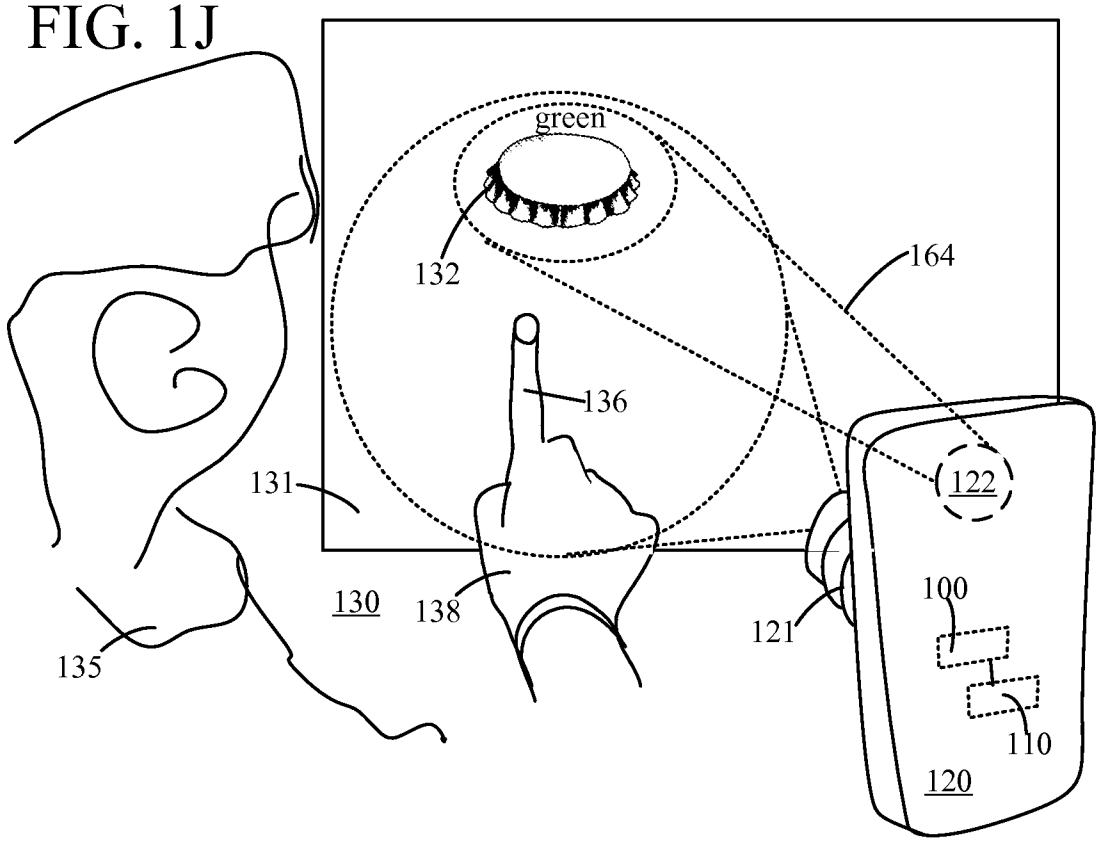

green
164
132
136
131
122
130
138
121
100
110
120
135

FIG. 2A

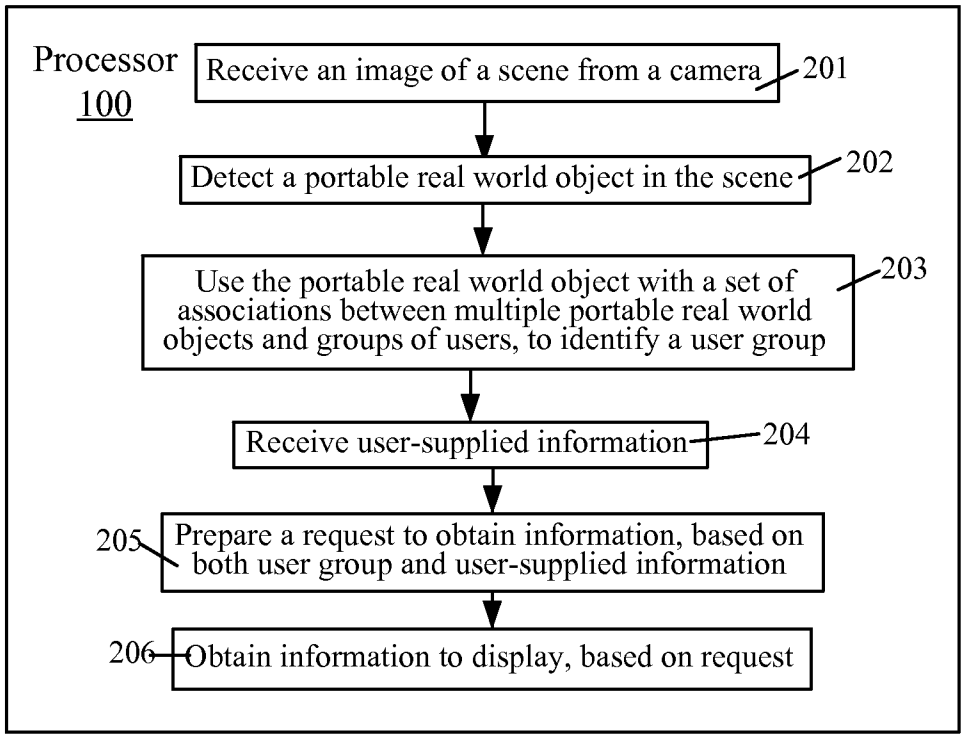

Processor
100
Receive an image of a scene from a camera
201
Detect a portable real world object in the scene
202
Use the portable real world object with a set of associations between multiple portable real world objects and groups of users, to identify a user group
203
Receive user-supplied information
204
205
Prepare a request to obtain information, based on both user group and user-supplied information
206
Obtain information to display, based on request

FIG. 2B

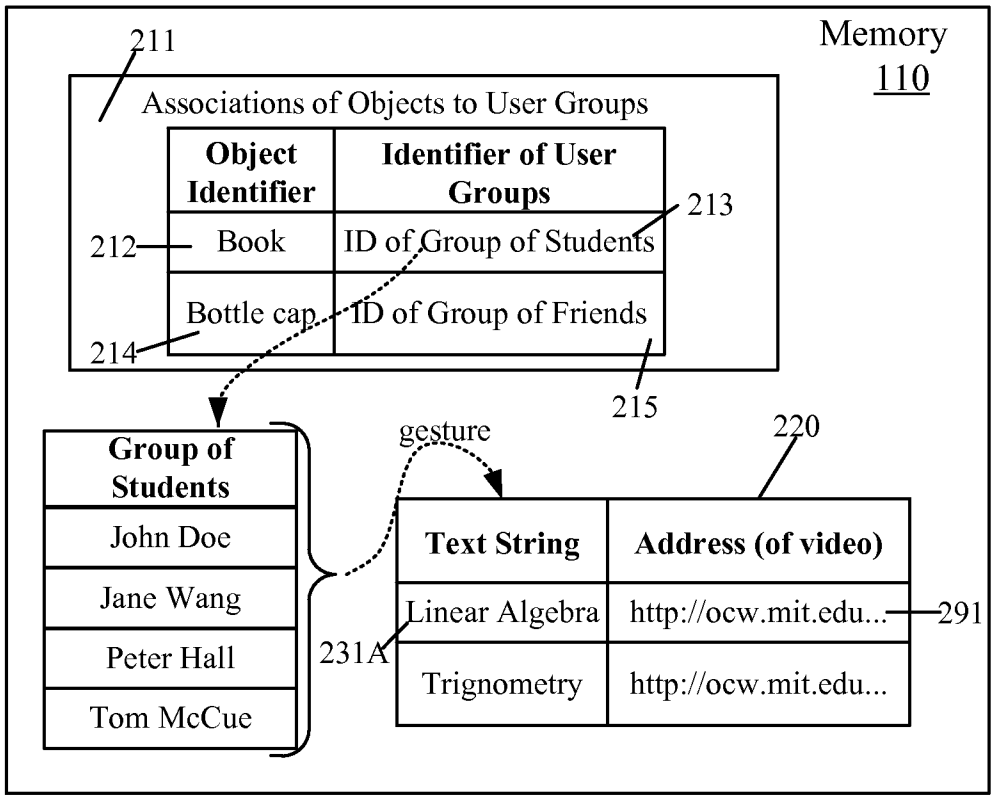

Memory
110
211
Associations of Objects to User Groups
Object Identifier
Identifier of User Groups
212
Book
ID of Group of Students
213
Bottle cap
ID of Group of Friends
214
215
gesture
220
Group of Students
John Doe
Jane Wang
Peter Hall
Tom McCue
Text String
Address (of video)
Linear Algebra
http://ocw.mit.edu...
291
231A
Trignometry
http://ocw.mit.edu...

FIG. 2C

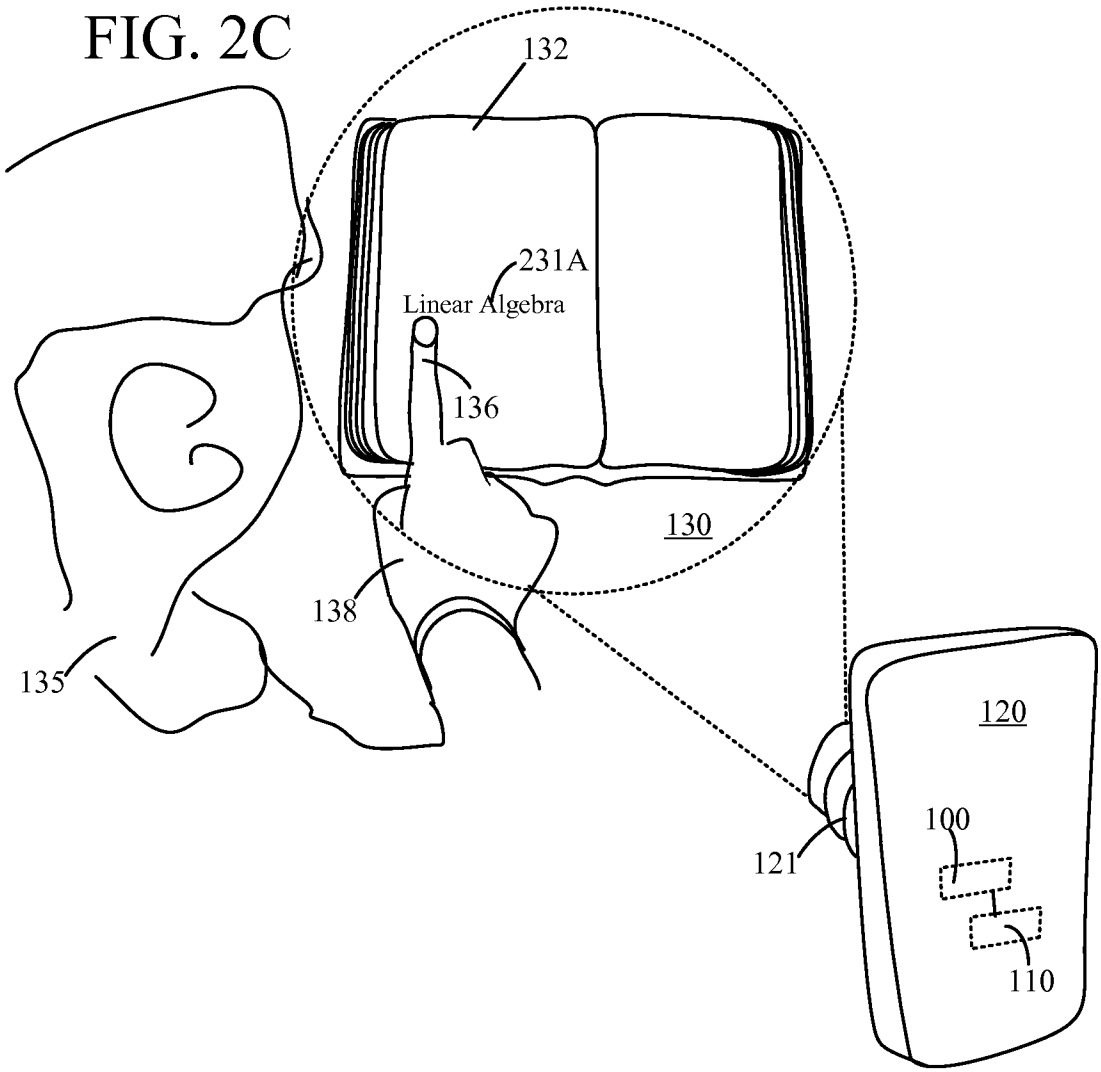

132
231A
Linear Algebra
136
130
138
135
120
100
121
110

FIG. 2D

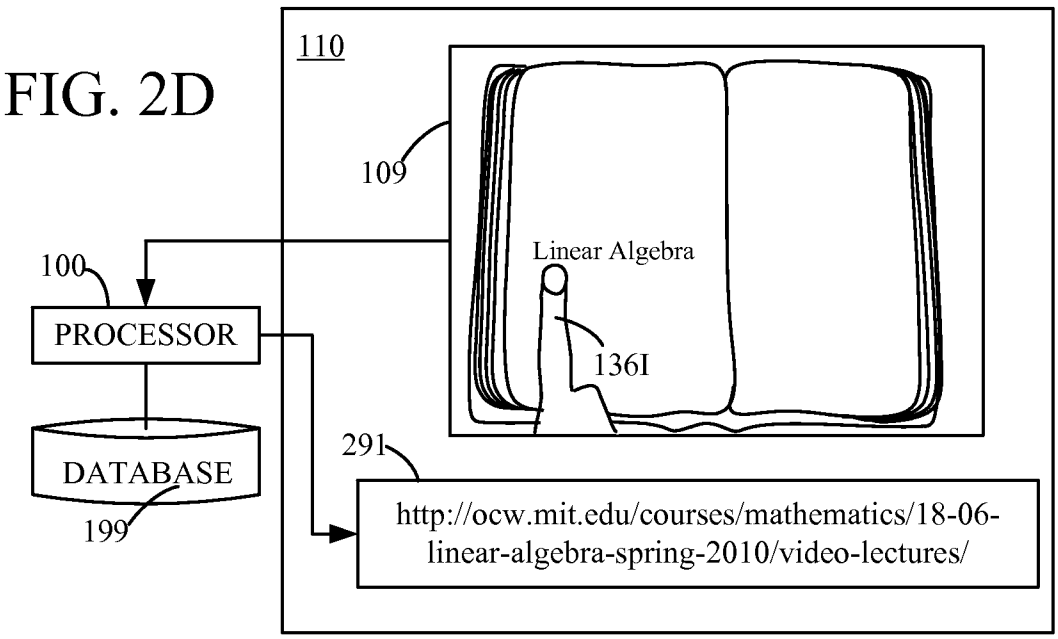

110
109
Linear Algebra
136I
100
PROCESSOR
DATABASE
199
291
http://ocw.mit.edu/courses/mathematics/18-06-linear-algebra-spring-2010/video-lectures/

FIG. 2E

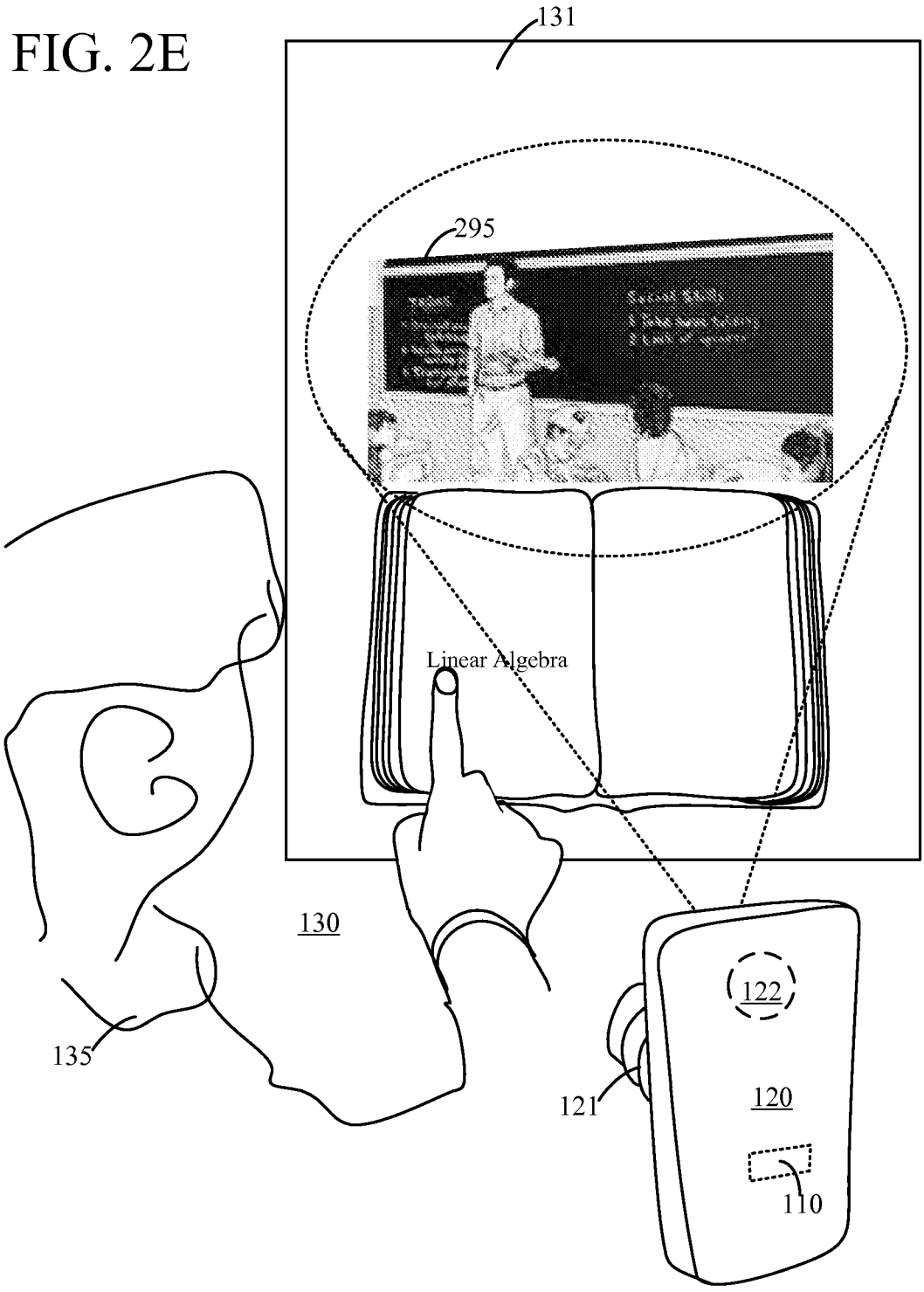


FIG. 3A

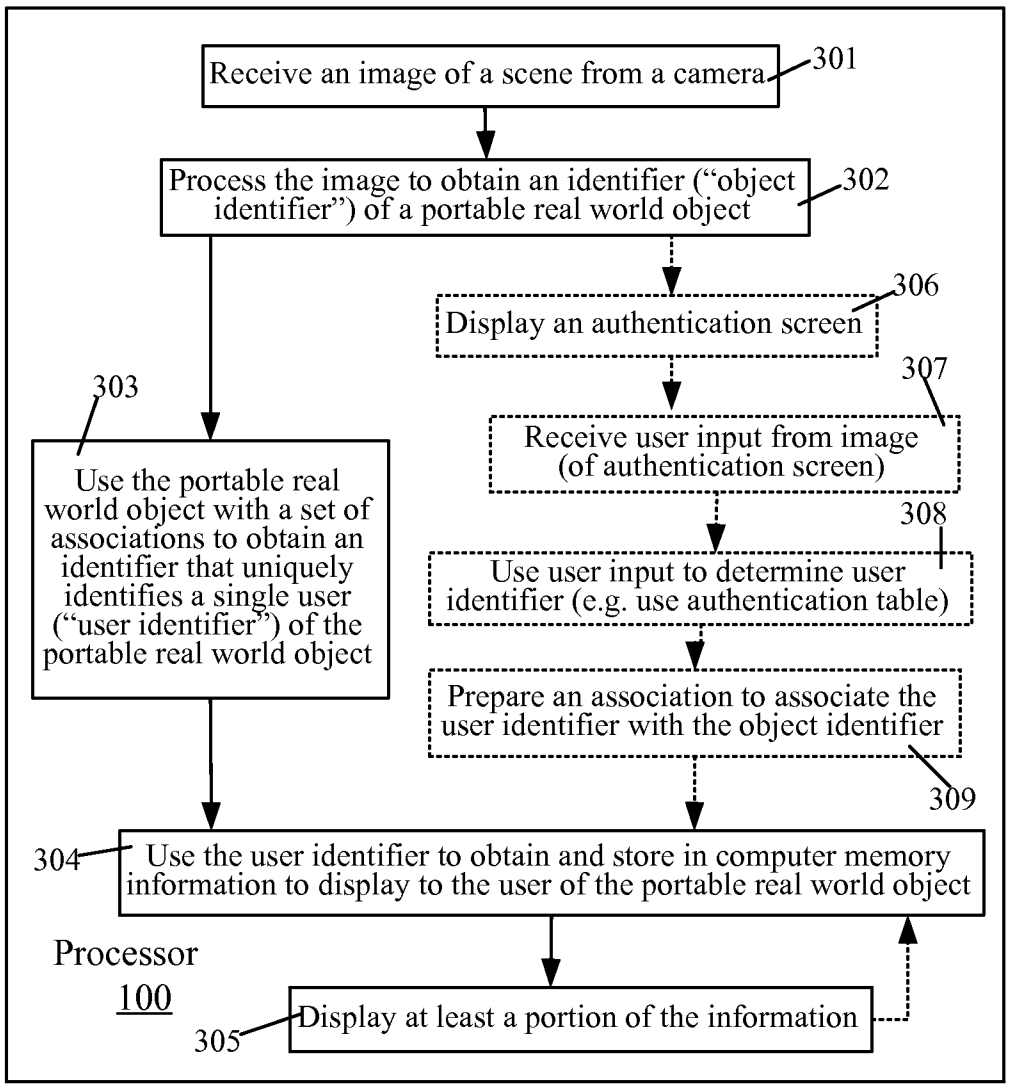

Receive an image of a scene from a camera
301
Process the image to obtain an identifier ("object identifier") of a portable real world object
302
Display an authentication screen
306
303
307
Receive user input from image (of authentication screen)
Use the portable real world object with a set of associations to obtain an identifier that uniquely identifies a single user ("user identifier") of the portable real world object
308
Use user input to determine user identifier (e.g. use authentication table)
Prepare an association to associate the user identifier with the object identifier
309
304
Use the user identifier to obtain and store in computer memory information to display to the user of the portable real world object
Processor 100
Display at least a portion of the information
305

FIG. 3B

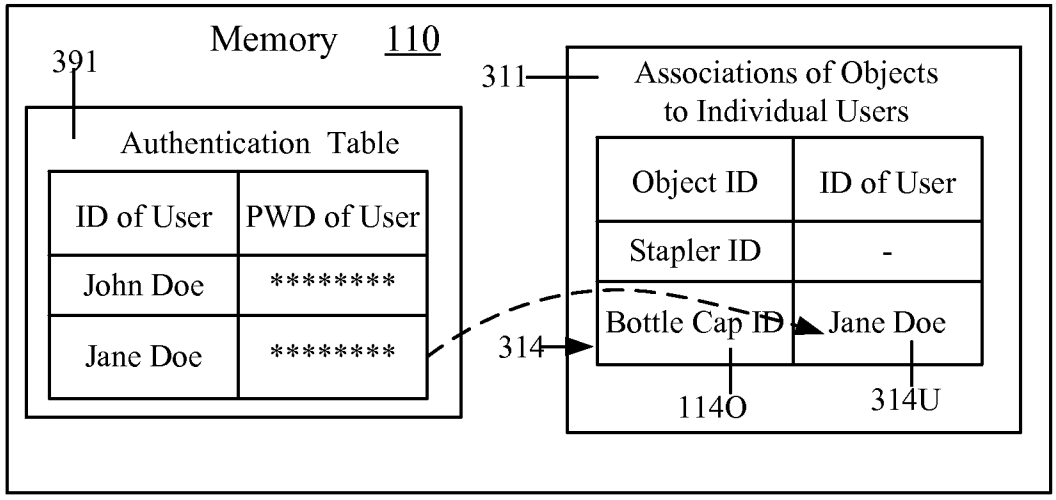

Memory 110
391
Authentication Table
ID of User
PWD of User
John Doe
****
Jane Doe
****
311
Associations of Objects to Individual Users
Object ID
ID of User
Stapler ID
-
Bottle Cap ID
Jane Doe
314
114O
314U

FIG. 3C

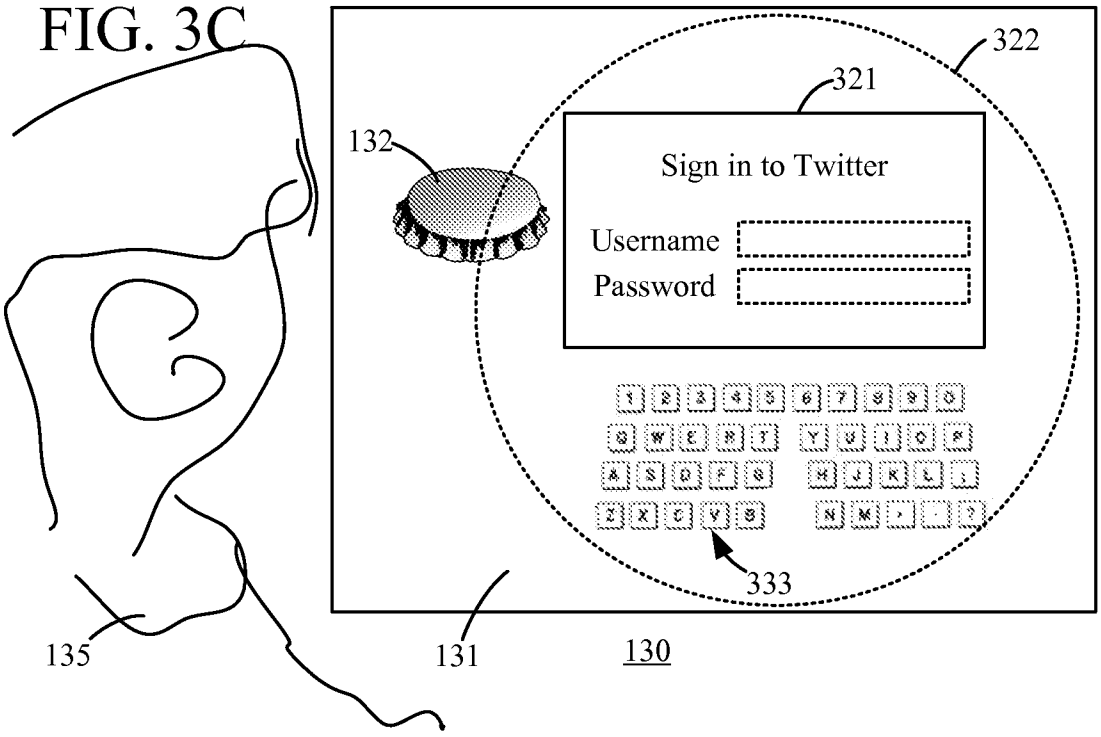

322
321
132
Sign in to Twitter
Username
Password
333
135
131
130

FIG. 3D

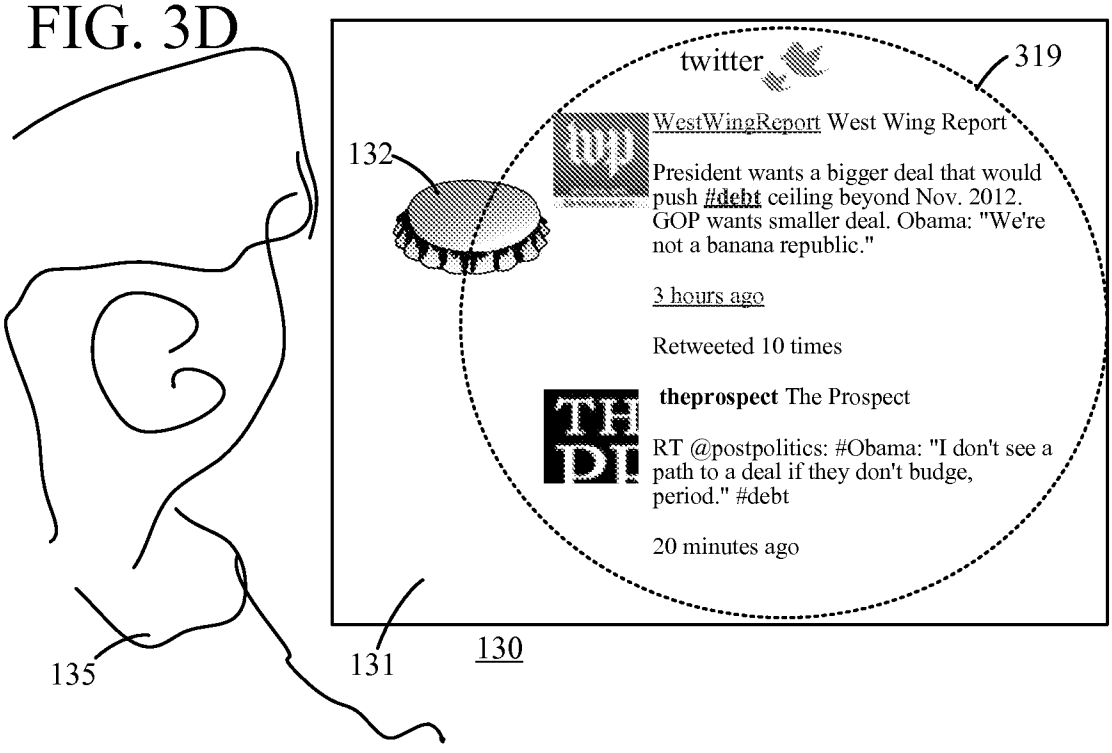

twitter
319
132
WestWingReport West Wing Report
President wants a bigger deal that would push #debt ceiling beyond Nov. 2012. GOP wants smaller deal. Obama: "We're not a banana republic."
3 hours ago
Retweeted 10 times
theprospect The Prospect
RT @postpolitics: #Obama: "I don't see a path to a deal if they don't budge, period." #debt
20 minutes ago
135
131
130

FIG. 3E

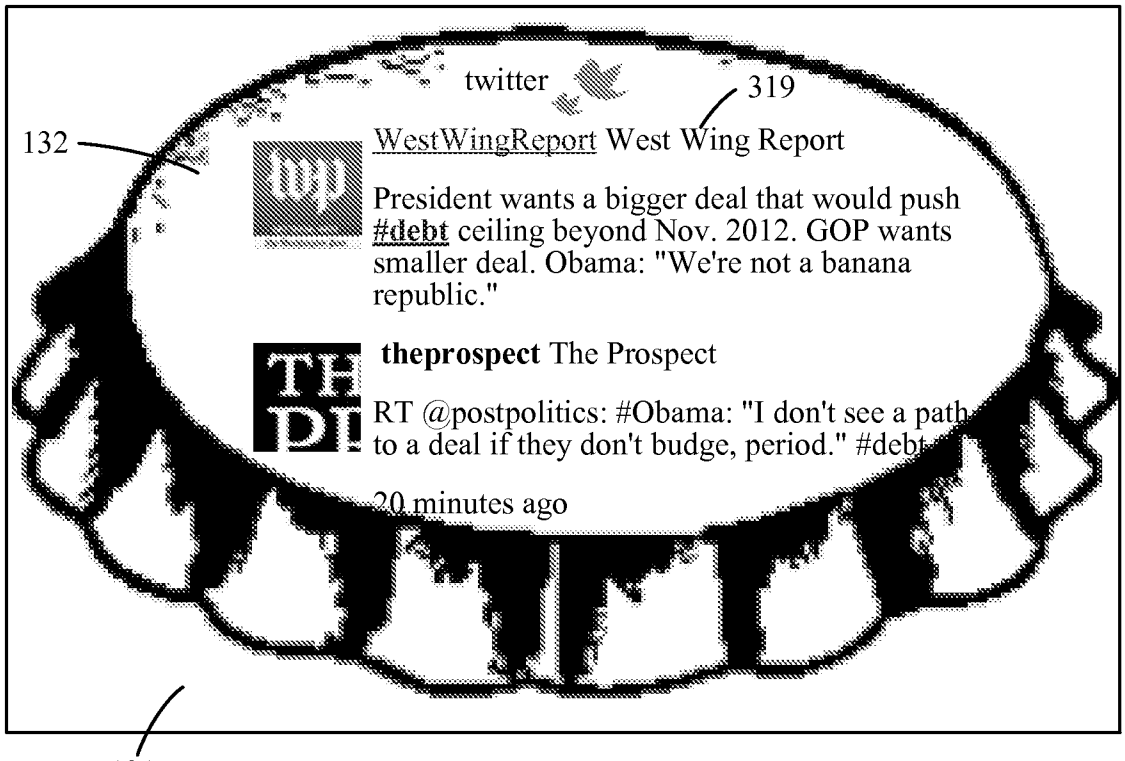
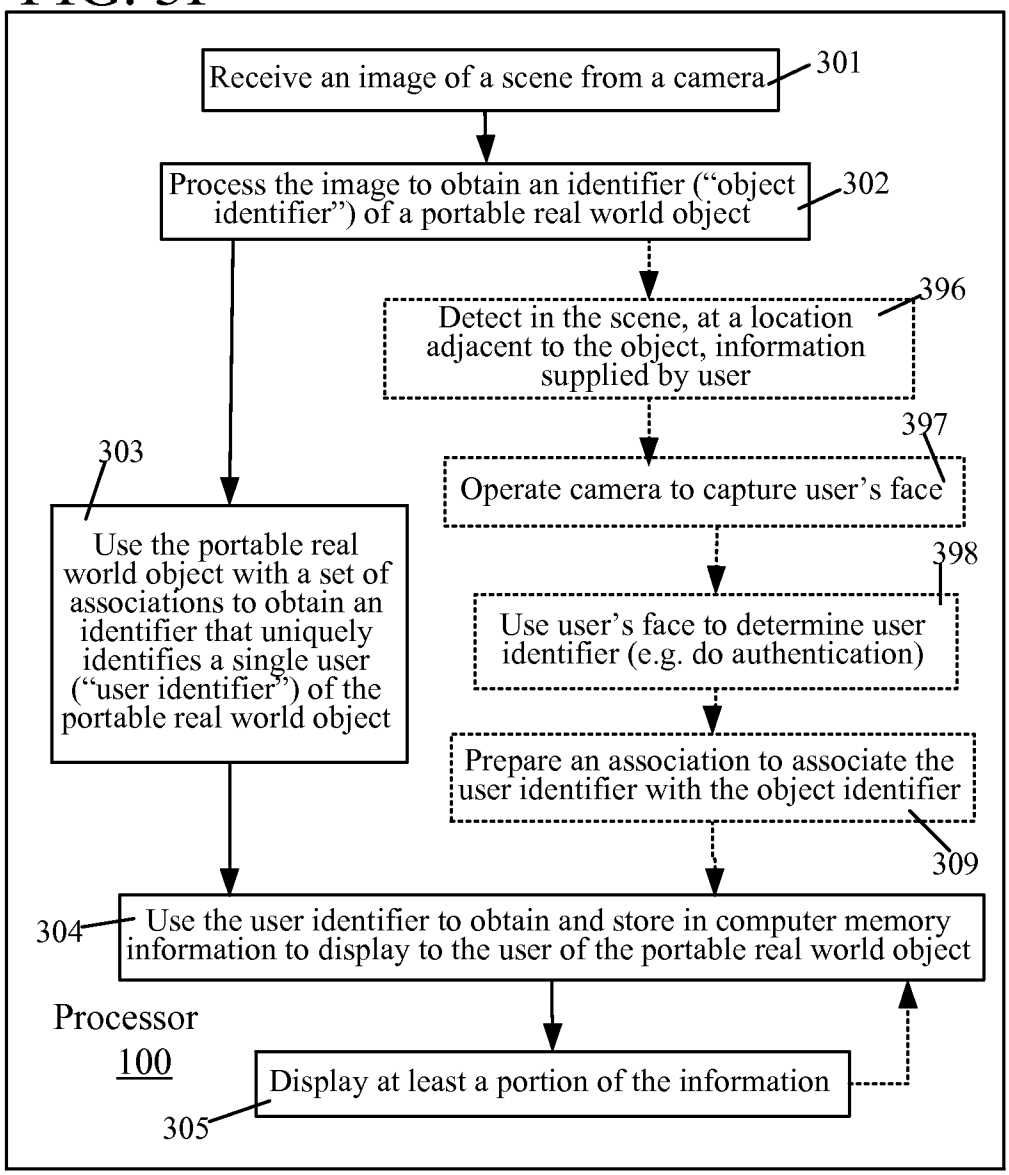

twitter
319
132
WestWingReport West Wing Report
President wants a bigger deal that would push #debt ceiling beyond Nov. 2012. GOP wants smaller deal. Obama: "We're not a banana republic."
theprospect The Prospect
RT @postpolitics: #Obama: "I don't see a path to a deal if they don't budge, period." #debt
20 minutes ago
131
130

FIG. 3F


Receive an image of a scene from a camera
301
Process the image to obtain an identifier ("object identifier") of a portable real world object
302
Detect in the scene, at a location adjacent to the object, information supplied by user
396
Operate camera to capture user's face
397
303
Use the portable real world object with a set of associations to obtain an identifier that uniquely identifies a single user ("user identifier") of the portable real world object
Use user's face to determine user identifier (e.g. do authentication)
398
Prepare an association to associate the user identifier with the object identifier
309
304
Use the user identifier to obtain and store in computer memory information to display to the user of the portable real world object
Processor 100
Display at least a portion of the information
305

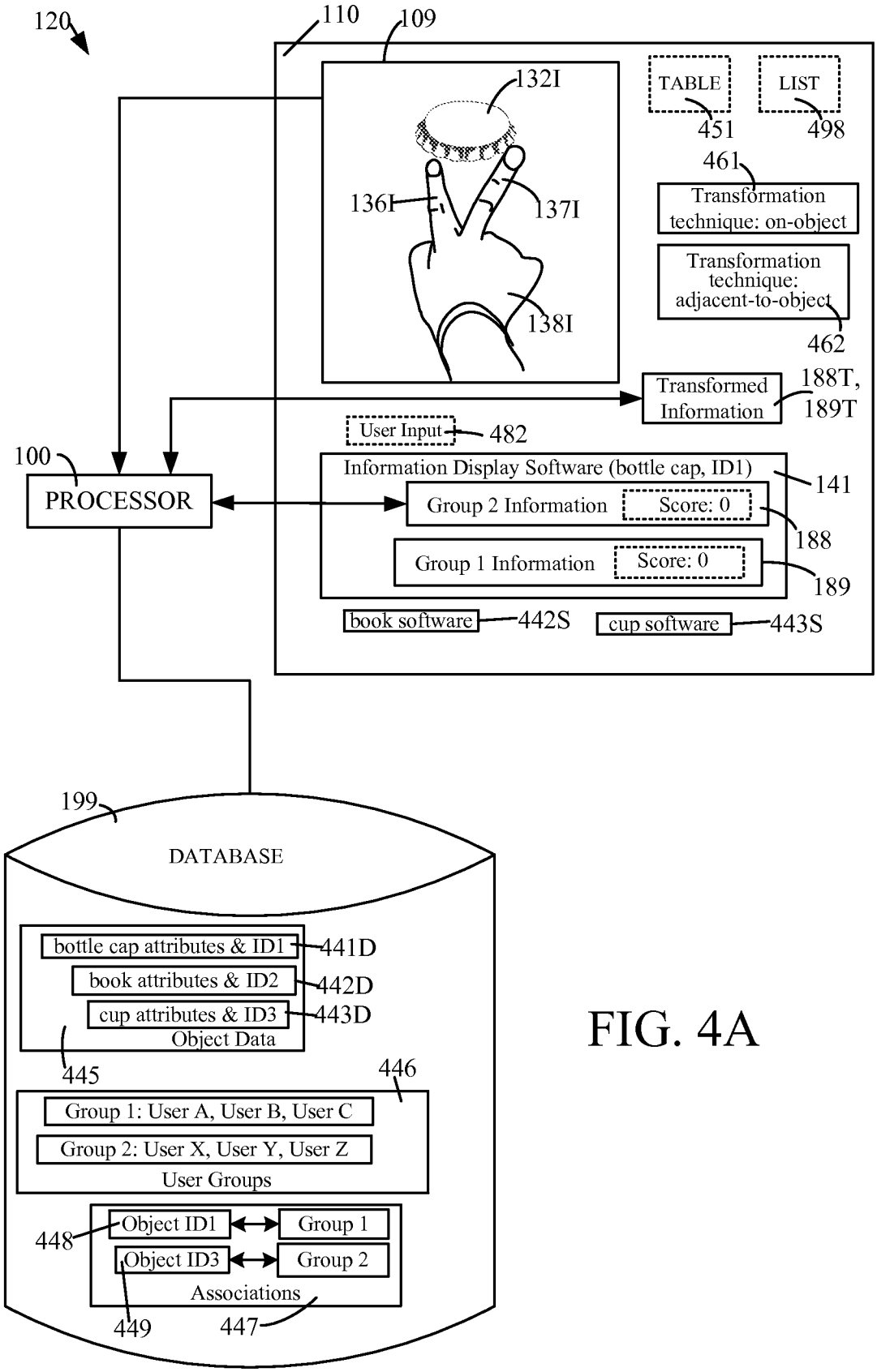
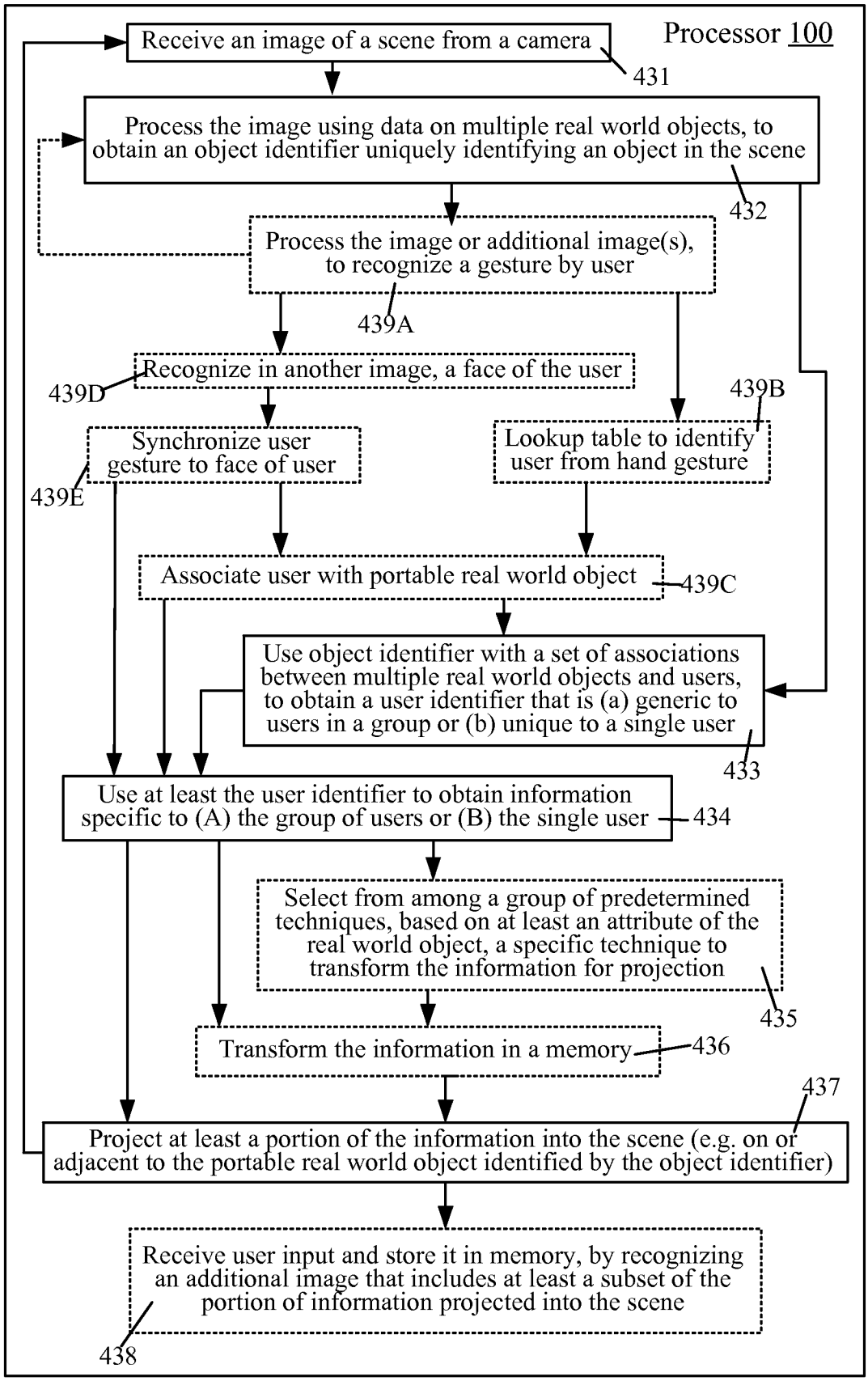

120
110
109
132I
136I
137I
138I
TABLE
LIST
451
498
461
Transformation technique: on-object
Transformation technique: adjacent-to-object
462
Transformed Information
188T, 189T
User Input
482
100
PROCESSOR
Information Display Software (bottle cap, ID1)
141
Group 2 Information
Score: 0
188
Group 1 Information
Score: 0
189
book software
442S
cup software
443S
199
DATABASE
bottle cap attributes & ID1
441D
book attributes & ID2
442D
cup attributes & ID3
443D
Object Data
445
446
Group 1: User A, User B, User C
Group 2: User X, User Y, User Z
User Groups
448
Object ID1
Group 1
Object ID3
Group 2
Associations
449
447

FIG. 4A

FIG. 4B


Processor 100
Receive an image of a scene from a camera
431
Process the image using data on multiple real world objects, to obtain an object identifier uniquely identifying an object in the scene
432
Process the image or additional image(s), to recognize a gesture by user
439A
Recognize in another image, a face of the user
439D
439B
Synchronize user gesture to face of user
439E
Lookup table to identify user from hand gesture
Associate user with portable real world object
439C
Use object identifier with a set of associations between multiple real world objects and users, to obtain a user identifier that is (a) generic to users in a group or (b) unique to a single user
433
Use at least the user identifier to obtain information specific to (A) the group of users or (B) the single user
434
Select from among a group of predetermined techniques, based on at least an attribute of the real world object, a specific technique to transform the information for projection
435
Transform the information in a memory
436
437
Project at least a portion of the information into the scene (e.g. on or adjacent to the portable real world object identified by the object identifier)
Receive user input and store it in memory, by recognizing an additional image that includes at least a subset of the portion of information projected into the scene
438

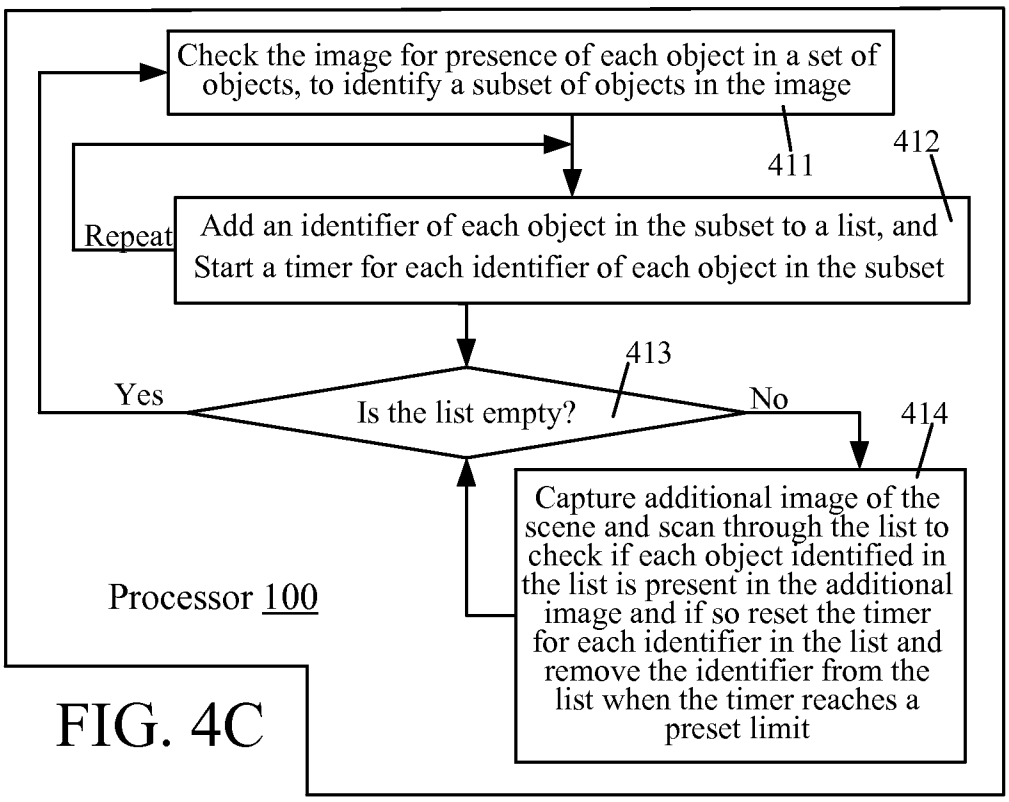

Check the image for presence of each object in a set of objects, to identify a subset of objects in the image
411
412
Repeat
Add an identifier of each object in the subset to a list, and Start a timer for each identifier of each object in the subset
413
Yes
Is the list empty?
No
414
Capture additional image of the scene and scan through the list to check if each object identified in the list is present in the additional image and if so reset the timer for each identifier in the list and remove the identifier from the list when the timer reaches a preset limit
Processor 100

FIG. 4C

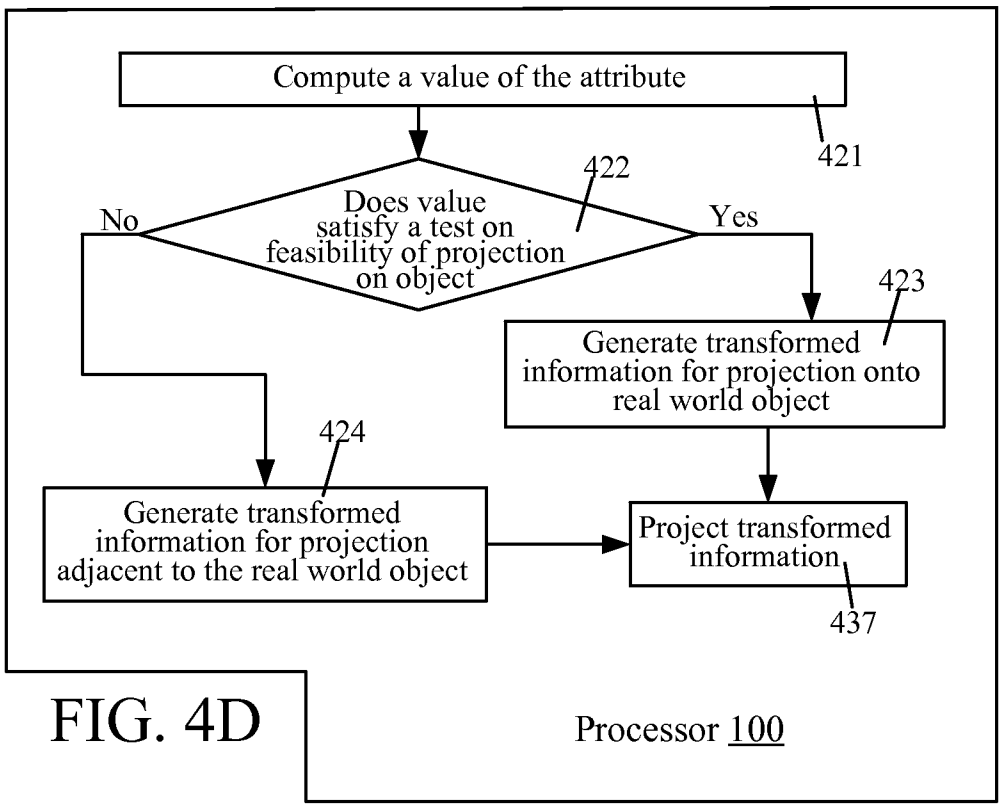

Compute a value of the attribute
421
422
No
Does value satisfy a test on feasibility of projection on object
Yes
423
Generate transformed information for projection onto real world object
424
Generate transformed information for projection adjacent to the real world object
Project transformed information
437
Processor 100

FIG. 4D

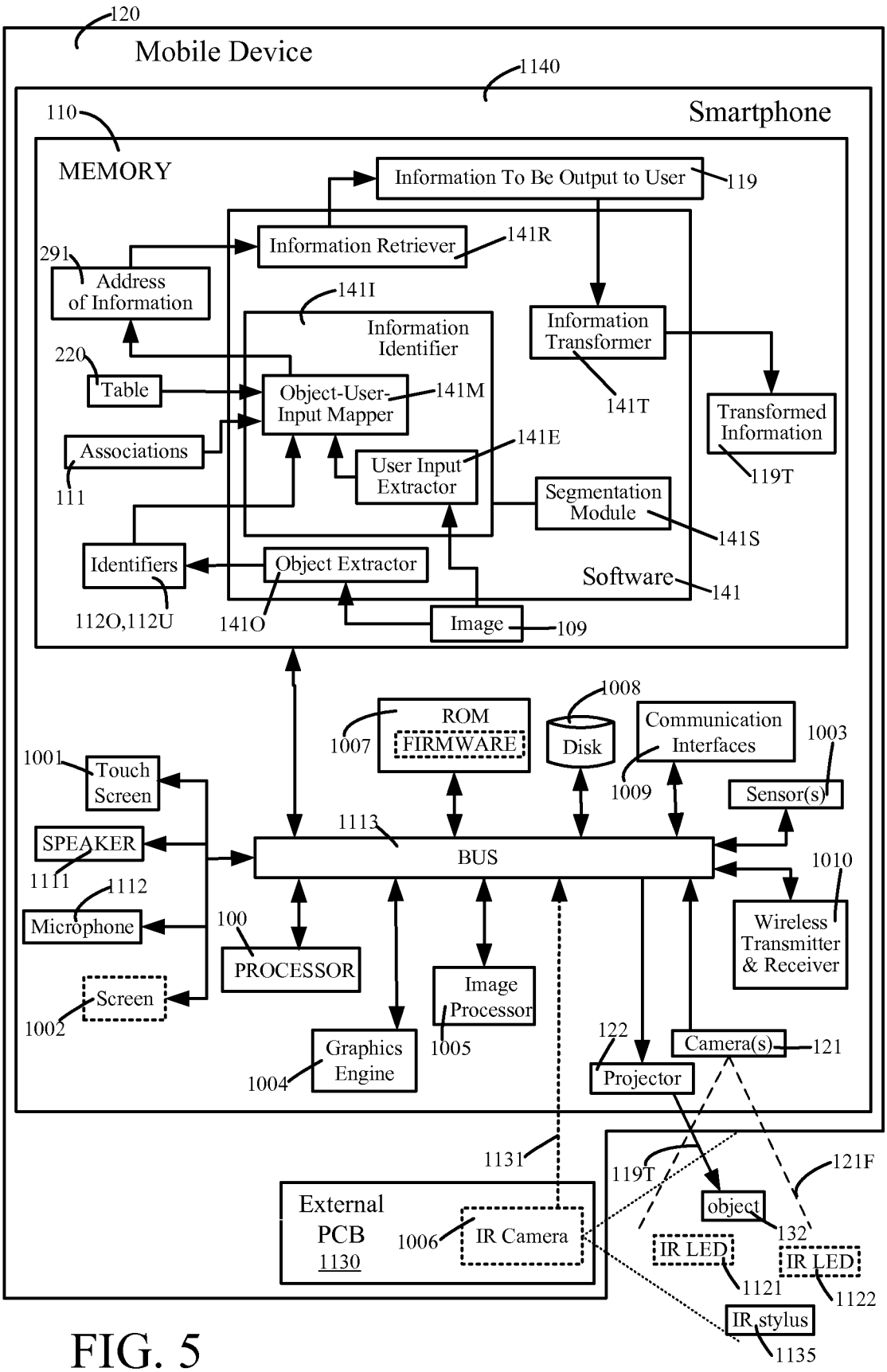

120
Mobile Device
1140
Smartphone
110
MEMORY
Information To Be Output to User
119
Information Retriever
141R
291
Address of Information
141I
Information Identifier
Information Transformer
141T
Transformed Information
119T
220
Table
Object-User-Input Mapper
141M
Associations
111
User Input Extractor
141E
Segmentation Module
141S
Identifiers
112O,112U
Object Extractor
141O
Software
141
Image
109
1008
ROM
FIRMWARE
1007
Disk
Communication Interfaces
1009
1003
Sensor(s)
1001
Touch Screen
SPEAKER
1111
1113
BUS
1010
Wireless Transmitter & Receiver
1112
Microphone
100
PROCESSOR
Image Processor
1005
Screen
1002
Graphics Engine
1004
122
Camera(s)
121
Projector
1131
119T
121F
External PCB
1130
1006
IR Camera
object
132
IR LED
1121
IR LED
1122
IR stylus
1135

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2012/046816

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06K9/22 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)
G06F 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, PAJ, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2009/179734 A1 (DO LYDIA M [US] ET AL) 16 July 2009 (2009-07-16)	1,2,4,9, 10,12, 17,22
Y	claims 17,22	7,15

Y	US 2008/069404 A1 (LEE YONG [KR] ET AL) 20 March 2008 (2008-03-20) figures 1, 7b	7,15

A	US 2007/050468 A1 (GAZIT AMIR [IL]) 1 March 2007 (2007-03-01) abstract; figure 1	1-24

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

☒ See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
30 January 2013	08/02/2013

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	Chateau, Jean-Pierre

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2012/046816

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2009179734 A1	16-07-2009	CN 101482879 A	15-07-2009
		JP 2009169951 A	30-07-2009
		US 2009179734 A1	16-07-2009
US 2008069404 A1	20-03-2008	NONE	
US 2007050468 A1	01-03-2007	NONE	