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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
9 October 2014 (09.10.2014)

WIPO | PCT

(10) International Publication Number
WO 2014/163979 A1

(51) International Patent Classification:
G06F 17/30 (2006.01) *G06F 3/048* (2013.01)
G06F 3/14 (2006.01)

(21) International Application Number:
PCT/US2014/019563

(22) International Filing Date:
28 February 2014 (28.02.2014)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
13/793,890 11 March 2013 (11.03.2013) US

(71) Applicant: **YAHOO! INC.** [US/US]; 701 First Avenue, Sunnyvale, California 94089 (US).

(72) Inventors: **CULPEPPER, Benjamin Jackson**; 2121 Amherst Street, Palo Alto, California 94306 (US). **GARRIGUES, Pierre**; 40 Ramona Avenue, San Francisco, California 94103 (US). **JANOCH, Allison**; 1308B Delaware Street, Berkeley, California 94702 (US). **NGUYEN, Huy X.**; 2322 Shattuck Avenue, Apt. 304, Berkeley, California 94704 (US).

(74) Agents: **HEILBRUNN, Elise R.** et al.; Weaver Austin Villeneuve & Sampson LLP, P.O. Box 70250, Oakland, California 94612-0250 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, 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, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(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,

[Continued on next page]

(54) Title: AUTOMATIC IMAGE PILING

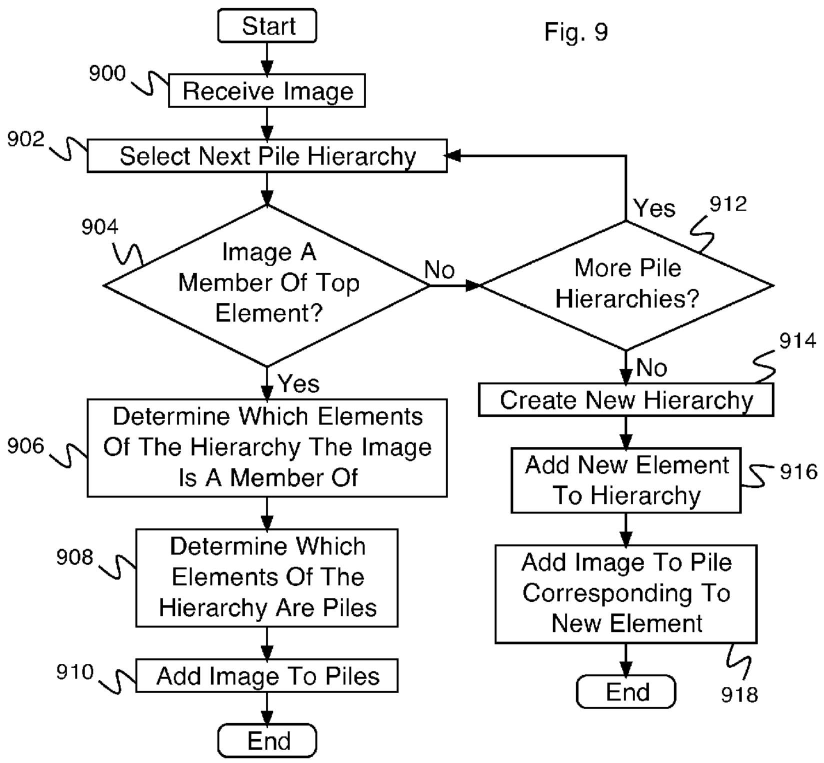

Fig. 9

[Continued on next page]

WO 2014/163979 A1

UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

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

(57) Abstract: A system for determining piles comprises an interface and a processor. The interface is configured to receive an image. The processor is configured to determine one or more attributes of the image; to determine whether the image is a member of a top of a hierarchy based at least in part on the attributes. In the event it is determined that the image is a member of the top of the hierarchy,: determine a set of elements of the hierarchy the image is a member of, based at least in part on the attributes and determine which of the set of entities are piles.

AUTOMATIC IMAGE PILING

RELATED APPLICATIONS

[0001] This application claims priority from U.S. Patent Application No. 13/793,890, entitled "Automatic Image Piling," filed on March 11, 2013, by Culpepper et al, which is incorporated herein by reference in its entirety for all purposes.

BACKGROUND OF THE INVENTION

[0002] A common mode of interaction with printed images is to sort a large collection into sub-collections by placing them into piles. Many different sorts of categories can be used, for instance, separating piles by who is in them, by the day they were taken, by the location they were taken, by their predominant color, by the kind of object that features most prominently, or by any other appropriate scheme. A person might desire to organize digital images in the same way. This can be a daunting task for a digital photographer, as it is typical to have a very large number of images in a given collection and a limited size display.

BRIEF DESCRIPTION OF THE DRAWINGS

[0003] Various embodiments of the invention are disclosed in the following detailed description and the accompanying drawings.

[0004] Figure 1 is a diagram illustrating an embodiment of a system for capturing and classifying images.

[0005] Figure 2A is a diagram illustrating an embodiment of a set of image piles.

[0006] Figure 2B is a diagram illustrating an embodiment of a pile, a stack, and a condensed stack.

[0007] Figure 2C is a diagram illustrating an embodiment of a pile, a stack, and a condensed stack.

[0008] Figure 3A is a diagram illustrating an embodiment of an image piling engine.

[0009] Figure 3B is a diagram illustrating an embodiment of a display for photos.

[0010] Figure 3C is a diagram illustrating an embodiment of a time grouping of photos.

[0011] Figure 4 is a diagram illustrating an embodiment of the relationship between objects and attributes.

[0012] Figure 5 is a flow diagram illustrating an embodiment of a process for determining piles.

[0013] Figure 6 is a flow diagram illustrating an embodiment of a process for detecting features.

[0014] Figure 7 is a block diagram illustrating an embodiment of attribute detection.

[0015] Figure 8 is a block diagram illustrating an embodiment of a hierarchy for determining a pile.

[0016] Figure 9 is a flow diagram illustrating an embodiment of a process for automatic image piling.

[0017] Figure 10 is a diagram illustrating an embodiment of a display for a smart album.

DETAILED DESCRIPTION

[0018] The invention can be implemented in numerous ways, including as a process; an apparatus; a system; a composition of matter; a computer program product embodied on a computer readable storage medium; and/or a processor, such as a processor configured to execute instructions stored on and/or provided by a memory coupled to the processor. In this specification, these implementations, or any other form that the invention may take, may be referred to as techniques. In general, the order of the steps of disclosed processes may be altered within the scope of the invention. Unless stated otherwise, a component such as a processor or a memory described as being configured to perform a task may be implemented as a general component that is temporarily configured to perform the task at a given time or a specific component that is manufactured to perform the task. As used herein, the term 'processor' refers to one or more devices, circuits, and/or processing cores configured to process data, such as computer program instructions.

[0019] A detailed description of one or more embodiments of the invention is provided below along with accompanying figures that illustrate the principles of the invention. The invention is described in connection with such embodiments, but the invention is not limited to any embodiment. The scope of the invention is limited only by the claims and the invention encompasses numerous alternatives, modifications and equivalents. Numerous specific details are set forth in the following description in order to provide a thorough understanding of the invention. These details are provided for the purpose of example and the invention may be practiced according to the claims without some or all of these specific details. For the purpose of clarity, technical material that is known in the technical fields related to the invention has not been described in detail so that the invention is not unnecessarily obscured.

[0020] Automatic image piling is disclosed. A system for determining piles comprises an interface and a processor. The interface is configured to receive an image. The processor is configured to determine one or more attributes of the image; to determine whether the image is a member of a top of a hierarchy based at least in part on the attributes. In the event it is determined that the image is a member of the top of the hierarchy: determine a set of elements of the hierarchy the image is a member of, based at least in part on the attributes; and determine which of the set of entities are piles.

[0021] In some embodiments, a system for determining piles comprises an interface configured to receive an image. The system comprises a memory for storing an ontology. The system comprises a processor configured to determine one or more attributes of an image; determine whether the image is a member of an entity in the ontology based at least in part on the attributes; in the event it is determined that the image is a member of a parent entity in the ontology: determine the set of child entities of the ontology the image is a member of, based at least in part on the attributes; and determine which of the set of entities are piles; and a memory coupled to the processor and configured to provide the processor with instructions. A memory also stores the image.

[0022] In some embodiments, an ontology is made up of "instances", "classes", "attributes", and "relations". Instances and classes are nodes in a directed, acyclic graph (DAG) with a root node "object". The edges in this graph are "is a" relations; that is, if node "car" is a child of "object" (and “object” is a parent of “car”), and "1993 White Nissan Pathfinder" is a child of "car" (and “car” is a parent of “1993 White Nissan Pathfinder”), then we can infer that both "car" and 1993 White Nissan Pathfinder" are objects and "1993 White Nissan Pathfinder" is a car. Since all nodes are children of "object", an equivalent designation of the same structure omits the root "object" node and is a list of directed acyclic graphs with more specific classes at their root, and this is the notion that is used here. In this case, "car" could be the top of one hierarchy, and more specific cars would be described as children of the "car" root. The leaves of these DAGs are "instances" and the internal nodes are "classes". Attributes are properties or characteristics of objects and classes. Relations are ways that classes and instances can be related to one another. For example, a "car" might have a "has a" relation to "steering wheel", and a "horse" might have a "can be ridden by" relation to "person". The "is a" relations that form the edges of a DAG are a subset of all relations. Other relations may be specified in the same way, but the resulting graph may no longer be acyclic.

[0023] A system for automatic image piling is disclosed. The system receives and processes an image and determines which pile or piles the image is a member of. In some embodiments, the system receives an image and determines low-level features and high-level attributes of the image. The high-level attributes are then tested against a set of image type hierarchies to determine the image type. Any hierarchy the image is determined to be a member of is further analyzed to determine which elements the image is a member of. One or more of the elements which the image is a member of is then selected as a pile for the image, and the pile is updated to include the image. In various embodiments, determining the set of hierarchies to further analyze comprises a best-first search or a beam search, or any other appropriate search. In various embodiments, determining the set of elements of the hierarchy the image is a member of comprises a best-first search or a beam search, or any other appropriate search.

[0024] Figure 1 is a diagram illustrating an embodiment of a system for capturing and classifying images. In the example shown, the system for capturing and classifying images comprises camera 100 and computing device 102. Camera 100 communicates with computing device 102 to transfer digital images – for example, digital images captured by camera 100 for classification by computing device 102. In various embodiments, camera 100 and computing device 102 communicate via a wire, via a wireless connection, via the transfer of a data storage device, or by any other appropriate means. In some embodiments, camera 100 and computing device 102 comprise a hybrid device (e.g., a camera-enabled mobile phone, a camera-enabled desktop computer, a smart phone, or any other appropriate device) and images are communicated internally to the hybrid device. In various embodiments, camera 100 and computing device 102 are remote to one another, and images are received by the computing device via one or more of a network (e.g., a local area network, a wide area network, the Internet, etc.), an external storage device (e.g., an external hard drive, a flash hard drive, etc.), or any other appropriate means. Camera 100 provides a collection of images to computing device 102. In some embodiments, a collection of images stored on computing device 102 comprises images received from more than one camera. Computing device 102 comprises a system for automatic image piling. In some embodiments, computing device 102 comprises a system for determining piles. In some embodiments, the image source (e.g., camera 100, an external storage device, the Internet, etc.) does not affect the image piling process. In some embodiments, the image source is used as an input to the image piling process. Computing device 102 additionally comprises a user interface. In various embodiments, the user interface of computing device 102 comprises a user interface for displaying image piles, for manually creating image piles, for interacting with automatically created image piles, for displaying individual images, or for any other appropriate user interface purpose.

[0025] In various embodiments, computing device 102 comprises a server, a desktop, a cloud computation processor, or any other appropriate processor. In some embodiments, images are captured from an imaging device (e.g., camera 100) and then transferred to a processor (e.g., computing device 102) to be automatically put into piles and/or efficiently displayed for a user (e.g., using a specialized index or reference number to identify images).

[0026] Figure 2A is a diagram illustrating an embodiment of a set of image piles. In some embodiments, the image piles of Figure 2 are determined automatically by a computing device or processor (e.g., computing device 102 of Figure 1). In some embodiments, the image piles of Figure 2 are determined manually by a human (e.g., by arranging images into image piles to train an automatic piler). In the example shown, pile 200 comprises images of food, pile 202 comprises images of Alice, pile 204 comprises images of Bob, pile 206 comprises images of Alcatraz, and pile 208 comprises images of the Golden Gate Bridge.

[0027] Figure 2B is a diagram illustrating an embodiment of a pile, a stack, and a condensed stack. In the example shown, image 210, image 214, and image 218 are processed to identify important or memorable regions of the images. For example, region 212 of image 210, region 216 of image 214, and region 218 of image 220. A pile is made with image 232, image 226, and image 222 with important or memorable region 230, region 228, and region 224, respectively. The pile is constructed so that region 230, region 228, and region 224 are visible and not occluded. A stack is made with the images with maximal overlap and only one image is visible (e.g., image 234 and important or memorable region 236). A condensed stack is made where the important or memorable region 238 is displayed only. In some embodiments, a rectangle that just fits region 238 is shown.

[0028] Figure 2C is a diagram illustrating an embodiment of a pile, a stack, and a condensed stack. In the example shown, image 240 and image 246 are processed to identify important or memorable regions of the images, where image 240 and image 246 have different aspect ratios. For example, region 242 of image 240 and region 244 of image 246. A pile is made with image 252 and image 251 with important or memorable region 248 and region 250, respectively. The pile is constructed so that region 248 and region 250 are visible and not occluded. A stack is made with the images with maximal overlap and only one image is visible (e.g., image 256 and important or memorable region 254 and image 258 and important and memorable region 260 that is occluded but partially visible due to different aspect ratios of the two images). A condensed stack is made visible where the important or memorable region 262 is displayed only. In some embodiments, a rectangle that just fits region 262 is shown.

[0029] Figure 3A is a diagram illustrating an embodiment of an image piling engine. In some embodiments, image piling engine 300 of Figure 3A comprises an image piling engine running on a computing device (e.g., computing device 102 of Figure 1). In the example shown, image piling engine 300 comprises a system for receiving a set of images and producing a set of one or more image piles. Image piling engine 300 comprises communications 302, user interface 304, feature detection 306, attribute computation 308, and image piling logic 312. Communications 302 comprises communications for receiving a set of images. In various embodiments, communications 302 comprises one or more of: a wireless data transfer interface, a wired data transfer interface, a network interface, or any other appropriate type of data transfer interface.

[0030] User interface 304 comprises a user interface for interacting with image piling engine 300. In various embodiments, user interface 304 receives commands to download images, to initiate automatic image piling, to view image piles, to modify image piles, to store image piles, to terminate a program, or any other appropriate commands. In various embodiments, user interface 304 displays images, image data, image piles, image pile data, system status information, or any other appropriate information. In some embodiments, user interface 304 comprises a graphical interface on a mobile device (e.g., a smart phone) that enables submission of images, display of images in piles that are automatically generated, display of images within piles, editing which piles the images are in, or any other appropriate graphical interface functions. For example, when a user initially begins browsing, piles of photos are shown as stacks. The stack is expanded when attended to -- for example, by pointing or clicking; this expansion occurs by reducing the amount of overlap between photos in the pile, effectively spreading them out and increasing the amount of area required for their display. Expansion can occur horizontally, or vertically, or in both directions, and by any distance, either partially or wholly revealing photos in the stack. An expansion of a stack may reveal new stacks in addition to single photos. In some embodiments, on a touchscreen, a stack is expanded via the "expand" touch gesture, and, in some embodiments, an image is expanded by a "click". The fundamental actions for navigating the piles are:

1. expanding a stack
2. contracting a stack
3. enlarging a photo
4. contracting a photo
5. changing the size of the "table" (e.g., looking at a larger or smaller area of the table)
6. selecting which pile to view (e.g., photos of hiking? photos of mom?)
7. “cleaning up” the piles, to take better advantage of all available space on the table

In some embodiments, the gestures available on a touch device comprise:

1. "pinch" expand
2. "pinch" contract
3. "tap"
4. double "tap"
5. swipe up
6. long press

In some embodiments, the gestures available on a workstation comprise:

1. mouse hover
2. a left, middle, or right mouse click
3. bump
4. scroll up/down

[0031] Feature detection 306 comprises software for detecting features in images. In some embodiments, features comprise recognizable local image regions. In various embodiments, a feature comprises an edge, a texture, a face, a logo, a symbol, a barcode, text or any other appropriate image feature. Feature detection 306 receives an image and produces a set of features for further processing. In some embodiments, features produced by feature detection 306 additionally comprise feature metadata (e.g., feature location, feature confidence, number of similar features found, or any other appropriate feature metadata).

[0032] Attribute detection 308 comprises software for detecting attributes in images. In some embodiments, image attributes comprise detectors for "has an X" relationships between an image and some X. In various embodiments, X comprises a face, an identified face, a person, text, a fruit, a product, a landmark, or any other appropriate X, including more specific forms such that indicate spatial relationships between objects, such as "has an apple next to an orange," or actions taking place between objects in the image, such as "has a man riding a horse," or hierarchical properties of the image such as "has a sign that reads Stop." Attribute detection 308 receives a set of features and produces a set of attributes. In some embodiments, attribute detection 308 additionally receives an image (e.g., the image associated with the received set of figures). In some embodiments, attributes produced by attribute detection 308 additionally comprise attribute metadata (e.g., attribute location, attribute confidence, attribute frequency, or any other attribute metadata). Attribute detection 308 utilizes object information stored in object databases 310 when detecting attributes. In various embodiments, object information stored in object databases includes features comprising an attribute, features that are present in an attribute, relative feature locations for an attribute, attribute detection algorithms, or any other appropriate object information. In some embodiments, attributes are detected in an image that are not yet stored in object databases 310 and object databases 310 are updated with the new attributes. For example, in certain cases it is possible to identify where many photos of the same "thing" have been taken (e.g., a landmark). When many photos of the same unlabeled "thing" have been taken, it can be detected, and the photos of that "thing" can be grouped without even knowing what the "thing" is. Specifically, for faces, when many faces of the same person are received, with some confidence, the faces of the same person are grouped without knowing the name of the person.

[0033] Image piling logic 312 comprises logic for determining image piles. Image piling logic 312 receives attributes from attribute detection 308 and determines one or more piles associated with the image. Piles are determined by applying a function mapping the attributes of a photo, a set of existing piles, and an ontology to a set of piles. This function may be learned from labeled data. In some embodiments, a geo-location and/or time information associated with the image is used to aid in automatically determining piles.

[0034] In various embodiments, the emphasis of grouping is changeable from grouping by time, by place, by people, by activity, or any other appropriate grouping emphasis.

[0035] Figure 3B is a diagram illustrating an embodiment of a display for photos. In the example shown, a layout for images that a user sees on a display is shown. Row 1 includes a display of one large photo and 4 smaller photos (e.g., photo display space 330, 332, 334, 336, and 338). Photo display space 330 includes pile indicator 370 (e.g., 4). Photo display space 332 includes a pile indicator 370 (e.g., 2). Photo display space 334 includes a pile indicator 370 (e.g., 3). Row 2 includes a display of one large photo and 4 smaller photos (e.g., 348, 340, 342, 344, and 346). Photo display space 346 includes a pile indicator 372 (e.g., N). Photo display space 348 includes a pile indicator 372 (e.g., M). Row 3 shows two large photos (e.g., photo display space 350 and 352). Photo display space 350 includes a pile indicator 372 (e.g., 2). Row 4 shows 6 smaller photos and 1 tall photo (e.g., photo display space 354, 356, 358, 360, 364, 368, and 362). Each row is representative of a pile (e.g., a cluster of photos based for example on an attribute).

[0036] In some embodiments, the way the photos occlude each other is manipulated by dragging the edges that designate occlusion boundaries. For example, the edge shared by image 330, 332 and 336 can be dragged to the left, revealing more of images 332 and 336 and less of 330. In some embodiments, a photo is subdivided when it is selected. For example, on a touchscreen, a double tap on a photo display space with a pile indicator greater than zero splits the pile of photos into four display spaces revealing four piles of photos that lie below it in the hierarchy.

[0037] In the example shown in figure 3B, local label 376 is shown above the photos. In some embodiments, local label 376 comprises a date. Local label 376 shows some text that gives context relating to all the photos that are currently seen.

[0038] In some embodiments, changing the size of the table results in newly available display space. A button or gesture for "clean up", when activated, reorganizes the piles to take advantage of existing or newly available space. In some embodiments, empty space is automatically filled with piles or photos as the size of the table is changed and new display space becomes available.

[0039] Figure 3C is a dendrogram diagram illustrating an embodiment of a hierarchical time grouping of photos. The example shows a grouping of photos taken during a trip to hike on Muir Beach. The trip is made up of a hierarchy of piles of photos taken in bursts that occur at way-points or during sub-events along the way. In this simple example, there is no ontology, and the function designating pile memberships is learned from the set of times that the photos were taken. These times are clustered using bottom up, single link agglomerative clustering, where the similarity function is the inverse absolute difference between a pair of times. The times at which photos were taken is shown as vertical lines touching the x-axis. The value of the similarity function is indicated on the y-axis. Pairs of photos that are more similar than the value shown on the y-axis are grouped together. At any point on the y-axis, a horizontal line (a cut) can be drawn to designate a set of top-level clusters and their children. For example, at 0.1 there are 22 distinct clusters and at 0.9 there are 6. The cutting point may be determined by a number of criteria: 1) the size of the table and the display size of each pile on the table determines a number of clusters that can be visualized; 2) a prespecified level of similarity; 3) where the gap between two successive combination of similarities is largest (such gaps arguably indicate “natural” clusterings); or 4) any other appropriate criteria. Other similarity functions, besides distances in time, can be used to determine clusters – for example, a path length along the edges in an ontological graph of "is a" relationships. In some embodiments, a "place" sense is used for clustering – for example, one photo is labeled "beach", another "Muir beach", and another "Golden Gate Bridge". In this case, it can be computed from the ontology that beach and Muir beach are more closely related than beach and Golden Gate Bridge. In some embodiments, a relationship other than "is a" is used such as "brother", "sister", "mother", "friend", etc. In various embodiments, ways to organize are "time", "person", "place", "activity", or any other appropriate manner.

[0040] In some embodiments, switching organizational senses is like moving from one "table" to another (e.g., switching is activated using a swipe with four fingers and it looks like "the page is turning" to a different set of piles). In some embodiments, moving around the table is achieved by dragging with one finger around on a touch sensitive display. In some embodiments, zooming in and out by closing two fingers towards each other (e.g., pinching) and opening two fingers away from each other.

[0041] In some embodiments, changing how photos are occluding each other is done by dragging a photo with two fingers out from under a photo. In some embodiments, a border of a photo is held down until it is high-lighted and the photo is brought to the top of a pile. In some embodiments, tables are switch is flipped by moving three fingers on the touch sensitive display. In some embodiments, tapping twice on any photo with a number greater than one in the upper right hand corner subdivides it into four piles and zooms in a little (perhaps as much as one column width).

[0042] Figure 4 is a diagram illustrating an embodiment of the relationship between objects and attributes. In the example shown, image 400 is analyzed to determine objects and attributes associated with image 400. For example, the scene within image 400 is determined to have a 0.9 probability of having the attribute sunset, 0.0 probability of having the attribute beach, 0 probability of having attribute snow, and 0.2 probability of having attribute mountain. Object 402 is determined to have 0.9 probability of being a person and a location within image 400. Object 404 is determined to have 0.95 probability of being a face and a location within image 400. Object 406 is determined to have 0.8 probability of being a person and a location within image 400. Object 408 is determined to have 0.95 probability of being a face, 0.9 probability of having an identity name of 'Jessica Smith' and a location within image 400. Object 410 is determined to have 1.0 probability of being a landmark with a name of 'Golden Gate Bridge.' Object 412 is determined to be text with a location within image 400. Object 414 is determined to be text with a location within image 400 and 0.9 probability of having transcription 'Golden Gate Bridge.'

[0043] Figure 5 is a flow diagram illustrating an embodiment of a process for determining piles. In some embodiments, the process of Figure 5 is executed by image piling engine 300 of Figure 3A. In the example shown, the process of Figure 5 is used to update or determine one or more image piles in response to received image(s). In 500, an image is received. In 502, image features are detected. In 504, image attributes are detected. In 506, one or more image piles (e.g., one or more image piles to be associated with the received image) are determined. In 508, attribute databases are updated. In some embodiments, attribute databases are updated only if necessary. In 510, image piles are updated. In some embodiments, updating image piles comprises adding the received image to the appropriate image piles. In some embodiments, updating image piles comprises determining a new image pile ordering. In some embodiments, updating image piles comprises determining a new top image for the image piles. In 512, updated image piles are displayed.

[0044] In some embodiments, important/memorable areas of images in a pile are viewable and not covered up. In some embodiments, an attribute of an image is a ranking that is used to determine relative importance. In some embodiments, the most important images (e.g., the highest ranking) are displayed in a viewable manner. In some embodiments, the important areas of the most important photos are displayed in a viewable manner.

[0045] In some embodiments, features are computed over different sized windows. In some embodiments, features are detected using scanning windows.

[0046] Figure 6 is a flow diagram illustrating an embodiment of a process for detecting features. In the example shown, in 600 a feature set is set to empty. A feature level is set to zero. An input to pixel is set. In 602, image features are detected from input using detectors for feature set. Detected image features are added to feature set. In 604, image features are pooled that were previously detected. Pooled features are added to feature set. In 606, it is determined whether there are enough pooled features to pool again. In the event that there are enough pooled features to pool again, then in 608 the feature level is incremented, the input is set to previously pooled features, and control is passed to 602. In the event that there are not enough pooled features to pool again, then the process ends.

[0047] Figure 7 is a block diagram illustrating an embodiment of attribute detection. In some embodiments, the block diagram of Figure 7 is used by attribute detection 308 of Figure 3A to detect attributes from features. In the example shown, attribute detection begins with feature set 700. Feature set 700 is provided to face detector 702, text detector 704, object class detector 706, feature match detector 708, and scene detector 710. Face detector 702 processes the feature set and determines whether a face or faces are present. If face detector 702 determines that a face or faces are present, face attribute(s) is/are stored along with confidence factors, locations, and any other appropriate attribute metadata. Any determined face attributes are provided to face identity detector 712. Face identity detector 712 determines any appropriate face identity attributes and stores any determined face identity attributes associated with the face attributes. Text detector 704 processes feature set 700 and determines if text is present. If text detector 704 determines that text is present, a text attribute is stored, along with a confidence factor, a location, and any other appropriate attribute metadata. Any determined text attributes are provided to text transcription 714. Text transcription 714 determines transcribed text values wherever possible and stores any determined transcribed text values associated with the text attributes.

[0048] Object class detector 706 detects whether an instance or instances of a local object class are present. In some embodiments, instances of an object class comprise a set of image features that have correspondence and/or a statistical correlation to a set of image features in a set of labeled database images corresponding to the object class. For instance, an object class could be a chair, a car, a cup, a pair of sunglasses, or a table. If object class detector 706 determines that an instance or instances of an object class are present, object class attributes are stored, along with confidence factors, locations, and any other appropriate attribute metadata. Feature match detector 708 detects whether features match features associated with a known attribute. If feature match detector 708 determines that features match features associated with a known attribute, feature match attributes are stored, along with confidence factors, locations, and any other appropriate attribute metadata. Any determined feature match attributes are provided to constrained feature match detector 716. Constrained feature match detector 716 provides constraints to feature matches found by feature match detector 708. In some embodiments, constrained feature match detector 716 provides homographic constraints to feature matches. In some embodiments, a homographic constraint on a feature match comprises a geometric transformation from the set of local features to the matching set of features. If constrained feature match detector 716 determines that features match when subject to constraints, constrained feature match attributes are stored, along with confidence factors, locations, and any other appropriate attribute metadata. Any constrained feature matches found are provided to object instance detector 718. Object instance detector 718 detects whether constrained feature matches found by constrained feature match detector 716 comprise object instances. For instance, object instances comprise the cover art for a particular book, a movie poster, a page from a magazine, the packaging of a product, or a currency bill. If object instance detector 718 determines that object instances are detected, object instances are stored, along with confidence factors, locations, and any other appropriate metadata.

[0049] Constrained feature matches found by constrained feature match detector 716 and feature set700 are provided to landmark detector 720. Landmark detector 720 detects landmark attributes from constrained feature match detector 716 and feature set 700. Landmark attributes comprise descriptions of well-known physical objects or locations. If landmark detector 720 determines that a landmark is detected, landmark attributes are stored, along with confidence factors, locations, and any other appropriate metadata. Feature set 700 is additionally provided to scene detector 710. Scene detector 710 determines the overall scene attribute of an image. For instance, scene attributes comprise night, day, indoors, outdoors, ocean, desert, or any other appropriate scene attribute. Scene detector 710 stores any detected scene attribute, along with confidence factor, fraction of image occupied by the scene, or any other appropriate metadata. In some embodiments, any other appropriate attribute detectors are included in the attribute detection system of Figure 7.

[0050] In the example shown, the flow of the image attribute computation algorithm is as follows: First, a set of features is computed from the image. Such a computation may occur at one frequency band (e.g., for a grayscale image), or at multiple frequency bands (for a color image). In conjunction with labeled training data, these features are used to form models of objects and image categories. The labels may describe the visual representation of an object or category by a collection of 2-dimensional views and contours demarcating a sub-region of each image. It is useful to distinguish between training images, which are a limited set of images that have been annotated with noisy labels by humans, and input data, for which attributes are computed entirely by a machine in conjunction with the models. In both cases, the same sets of features are used. The models are computed from the features in the training images, then applied to the input images to compute attributes. Some attribute models may utilize other attributes in addition to the sets of features. For example, the "portrait" attribute may use the face detector attribute.

[0051] Figure 8 is a block diagram illustrating an embodiment of a hierarchy for determining a pile. In some embodiments, a hierarchy for determining a pile is used by image piling logic 312 for determining a pile from a set of attributes. The hierarchy of Figure 8 receives an image with an associated set of attributes and determines a pile or piles. In the example shown, the hierarchy of Figure 8 comprises document element 800, book cover element 806, menu element 812, receipt element 818, and Bette’s Diner menu 824. Each element comprises an associated member function and images. When an image and associated image attributes are received by the hierarchy, it is determined whether the image is a member of the top element (e.g., document element 800), using the member function of the top element (e.g., document member function 802). In some embodiments, the member function determines whether the image is a member of the element using the image attributes. In some embodiments, the member function determines whether the image is a member of the element using a combination of the image and the image attributes. If the member function determines that the image is a member of the element, the image is added to the element images set (e.g., document images 804). It is then determined whether the image is a member of any of the children of the top element (e.g., book cover element 806, menu element 812, or receipt element 818), using the appropriate member functions (e.g., book cover member function 808, menu member function 814, or receipt member function 820). An image proceeds down the hierarchy in this way until the set of elements in the hierarchy that the image is a member of is determined. Once the set of elements in the hierarchy that the image is a member of is determined, it is determined which of these elements are piles for the image. In various embodiments, all of the elements are piles for the image, the elements that are piles are determined from the number of other images that are members of each element (e.g., the process tries to create a small number of piles, the process tries to create a large number of piles, etc.), the elements that are piles are determined from the element position in the hierarchy (e.g., only the lowest element in the hierarchy is a pile, the lowest element of each branch is a pile, etc.), or the elements that are piles are determined in any other appropriate way. In some embodiments, if the image is determined to be a member of a hierarchy element using a member function, and is associated with an image already stored in the image set of that element, a new child element is created. The new child element represents the object found in both the new image and the associated image. In some embodiments, the hierarchy of Figure 8 comprises one of a set of hierarchies, each for assigning an image in a different image recognition category (e.g., documents, food, landmarks, furniture, etc.).

[0052] Figure 9 is a flow diagram illustrating an embodiment of a process for automatic image piling. The process of Figure 9 comprises a process for determining piles. In some embodiments, the process of Figure 9 is executed by image piling engine 300 of Figure 3A. In the example shown, in 900, an image is received. In various embodiments, an image is received from a camera, a storage device, a network, a wired connection, a wireless connection, a software image synthesizer, or any other appropriate image source. In 902, the next pile hierarchy is selected. In some embodiments, the next pile hierarchy comprises the first pile hierarchy. In some embodiments, the next pile hierarchy comprises the hierarchy of Figure 8. In 904, it is determined whether the image is a member of the top element (e.g., a member of the top element of the hierarchy). In some embodiments, it is determined whether the image is a member of the top element using the member function of the top element. If it is determined in 904 that the image is a member of the top element, control passes to 906. In 906, it is determined which elements of the hierarchy the image is a member of. In some embodiments, determining which elements of the hierarchy the image is a member of comprises testing the image using the member function of each element of the hierarchy. In some embodiments, determining which elements of the hierarchy the image is a member of comprises proceeding down the hierarchy from the top element and testing the image using the member function of an element of the hierarchy if the image is determined to be a member of the parent of the element. In 908, it is determined which elements of the hierarchy are piles. In various embodiments, all elements of the hierarchy the image is a member of are determined to be piles, a subset of the elements of the hierarchy the image is a member of are determined to be piles, a single element of the hierarchy the image is a member of is determined to be a pile, or any other appropriate number of elements of the hierarchy are determined to be piles. In some embodiments, an element or elements of the hierarchy are determined to be piles based at least in part on the position of the element or elements in the hierarchy. In 910, the image is added to the piles.

[0053] If it is determined in 904 that the image is not a member of the top element, control passes to 912. In 912, it is determined whether there are more pile hierarchies. If it is determined that there are more pile hierarchies, control passes to 902. If it is determined that there are not more pile hierarchies, control passes to 914. In 914, a new hierarchy is created. In 916, a new element is added to the hierarchy (e.g., to the hierarchy created in 914). In 918, the image is added to the pile corresponding to the new element.

[0054] Figure 10 is a diagram illustrating an embodiment of a display for a smart album. In the example shown, a display of a smart album includes a list of filters (e.g., text filter 1000) and a display of photos (e.g., image 1010, image 1012, image 1014, image 1020, image 1022, image 1024, image 1030, image 1032, image 1034, image 1040, image 1042, image 1044, image 1050, image 1052, and image 1054). The photos can be filtered by selecting a tag to filter by and pushing submit. The photos have been automatically tagged and these tags are listed for filtering. Similar photos are piled together to save display space. Indicators (e.g., indicator 1002, indicator 1004, and indicator 1006) indicate similar photos exist in the display pile. For example, image 1020 is one of 5 photos in the pile. In some embodiments, clicking on the indicator expands the display of the photos in the pile. For another example, indicator 1004 indicates that image 1042 is one of 3 photos in a pile. For another example, indicator 1006 indicates that image 1050 is one of 2 photos in a pile.

[0055] Although the foregoing embodiments have been described in some detail for purposes of clarity of understanding, the invention is not limited to the details provided. There are many alternative ways of implementing the invention. The disclosed embodiments are illustrative and not restrictive.

CLAIMS

1. A system for determining piles, comprising:

an interface configured to receive an image;

a processor configured to:

determine one or more attributes of the image;

determine whether the image is a member of a top of a hierarchy based at least in part on the attributes;

in the event it is determined that the image is a member of the top of the hierarchy:

determine a set of elements of the hierarchy the image is a member of, based at least in part on the attributes; and

determine which of the set of entities are piles.

2. The system of claim 1, wherein the processor is further configured to determine features of the image.

3. The system of claim 2, wherein the attributes are based at least in part on the features.

4. The system of claim 1, wherein the hierarchy is one of a plurality of hierarchies.

5. The system of claim 4, wherein the plurality of hierarchies comprises an initial set of hierarchies.

6. The system of claim 1, wherein elements of the set of elements of the hierarchy comprise element member functions.

7. The system of claim 1, wherein elements of the set of elements of the hierarchy comprise element images.

8. The system of claim 1, wherein the processor is further configured to create a new element of the set of elements of the hierarchy.

9. The system of claim 8, wherein the processor is further configured to create the new element of the set of elements of the hierarchy in the event that the image is associated with an element image of an existing member of the hierarchy.

10. The system of claim 1, wherein the processor is further configured to create a new hierarchy in the event that the image is determined to not be a member of the top element of the hierarchy.

11. The system of claim 1, wherein the set of elements of the hierarchy the image is a member of is determined by checking each element of the hierarchy.

12. The system of claim 1, wherein the set of elements of the hierarchy the image is a member of is determined by checking elements of the hierarchy in top down order.

13. The system of claim 12, wherein an element of the set of elements of the hierarchy is only checked in the event that the image is determined to be a member of its parent.

14. The system of claim 1, wherein it is determined which of the elements of the set of elements of the hierarchy are piles based at least in part on the number of images associated with each element.

15. The system of claim 1, wherein it is determined which of the elements of the set of elements the image is a member of are piles based at least in part on an element position in the hierarchy.

16. The system of claim 15, wherein an element the image is a member of furthest down in the hierarchy is determined to be a pile.

17. A method for determining piles, comprising:

determining, using a processor, attributes of an image;

determining whether the image is a member of a top of a hierarchy based at least in part on the attributes;

in the event it is determined that the image is a member of the top of the hierarchy:

determining a set of elements of the hierarchy the image is a member of, based at least in part on the attributes; and

determining which of the set of elements are piles.

18. A computer program product for tagging an object, the computer program product being embodied in a tangible computer readable storage medium and comprising computer instructions for:

determining attributes of an image;

determining whether the image is a member of a top of a hierarchy based at least in part on the attributes;

in the event it is determined that the image is a member of the top of the hierarchy:

determining a set of elements of the hierarchy the image is a member of, based at least in part on the attributes; and

determining which of the set of elements are piles.

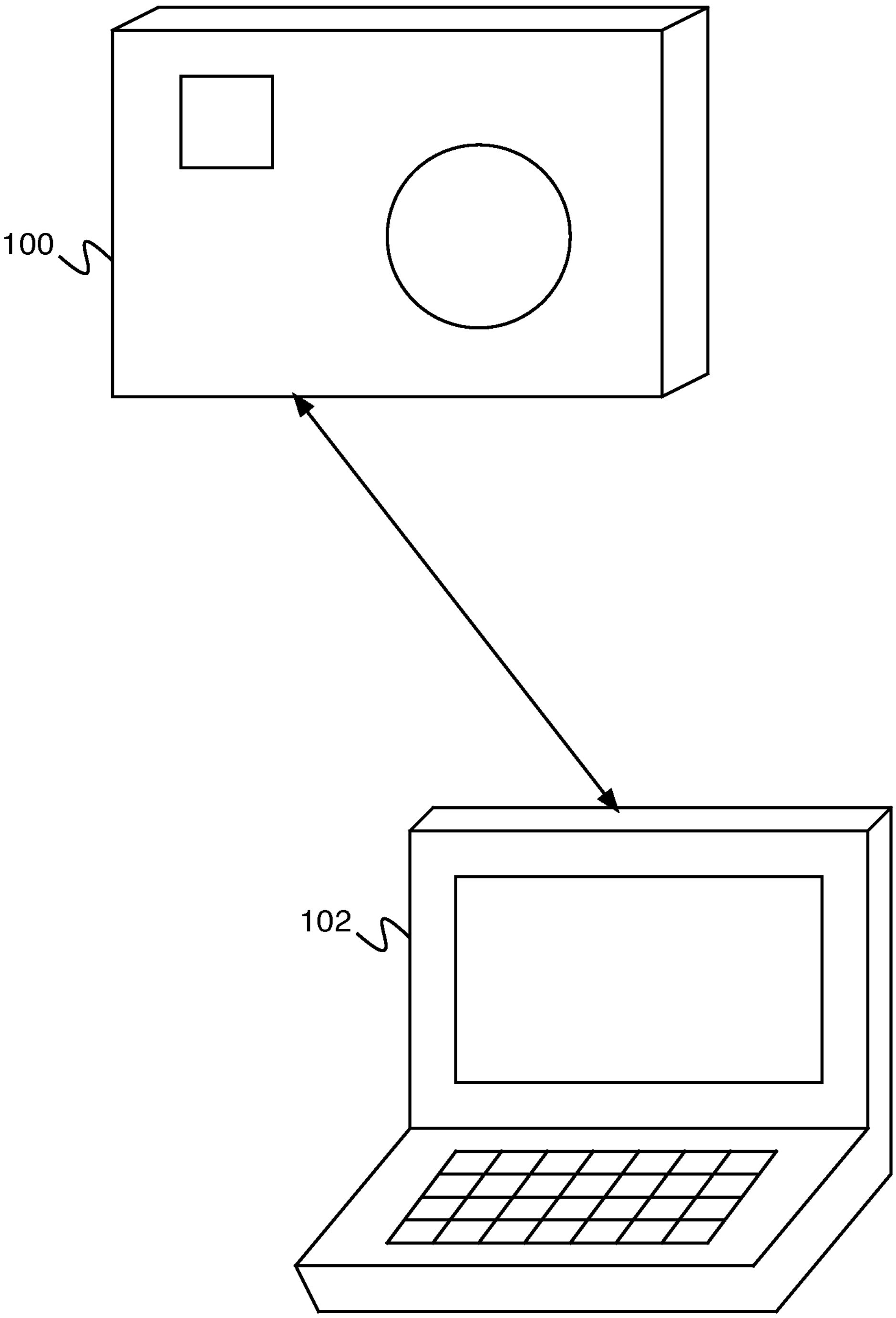

Fig. 1

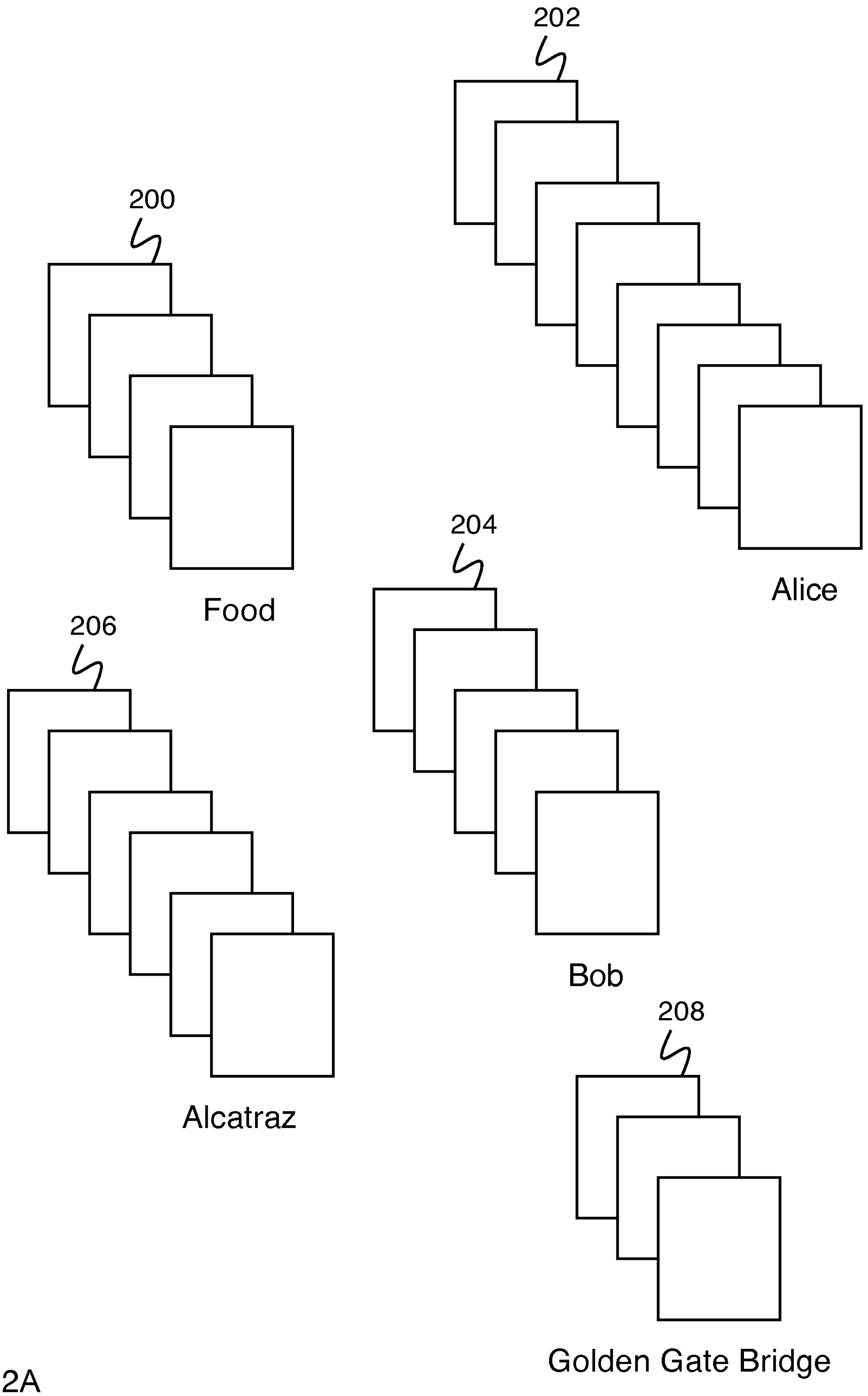
202
200
Alice
204
Food
206
Bob
208
Alcatraz
Golden Gate Bridge

Fig. 2A

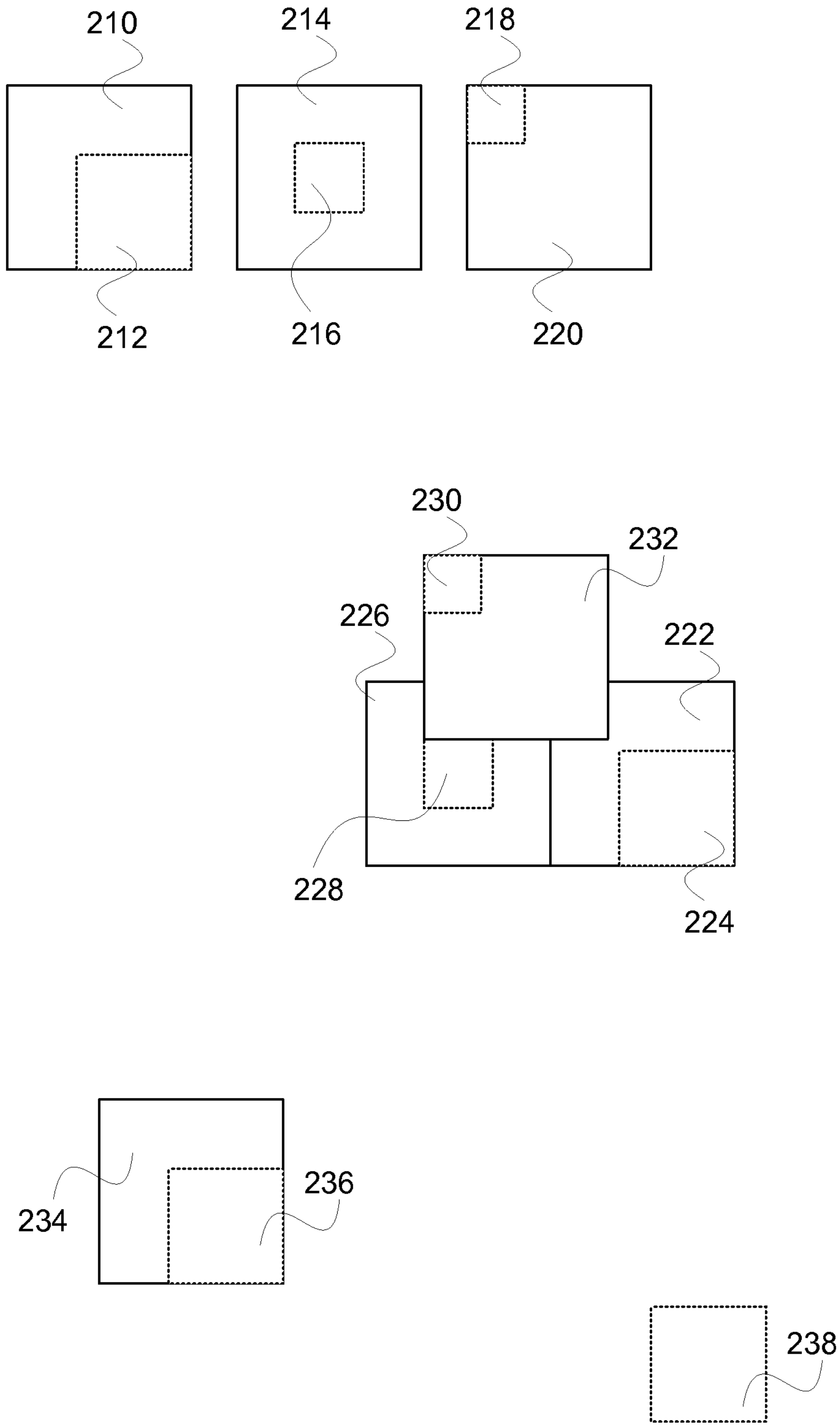

FIG. 2B

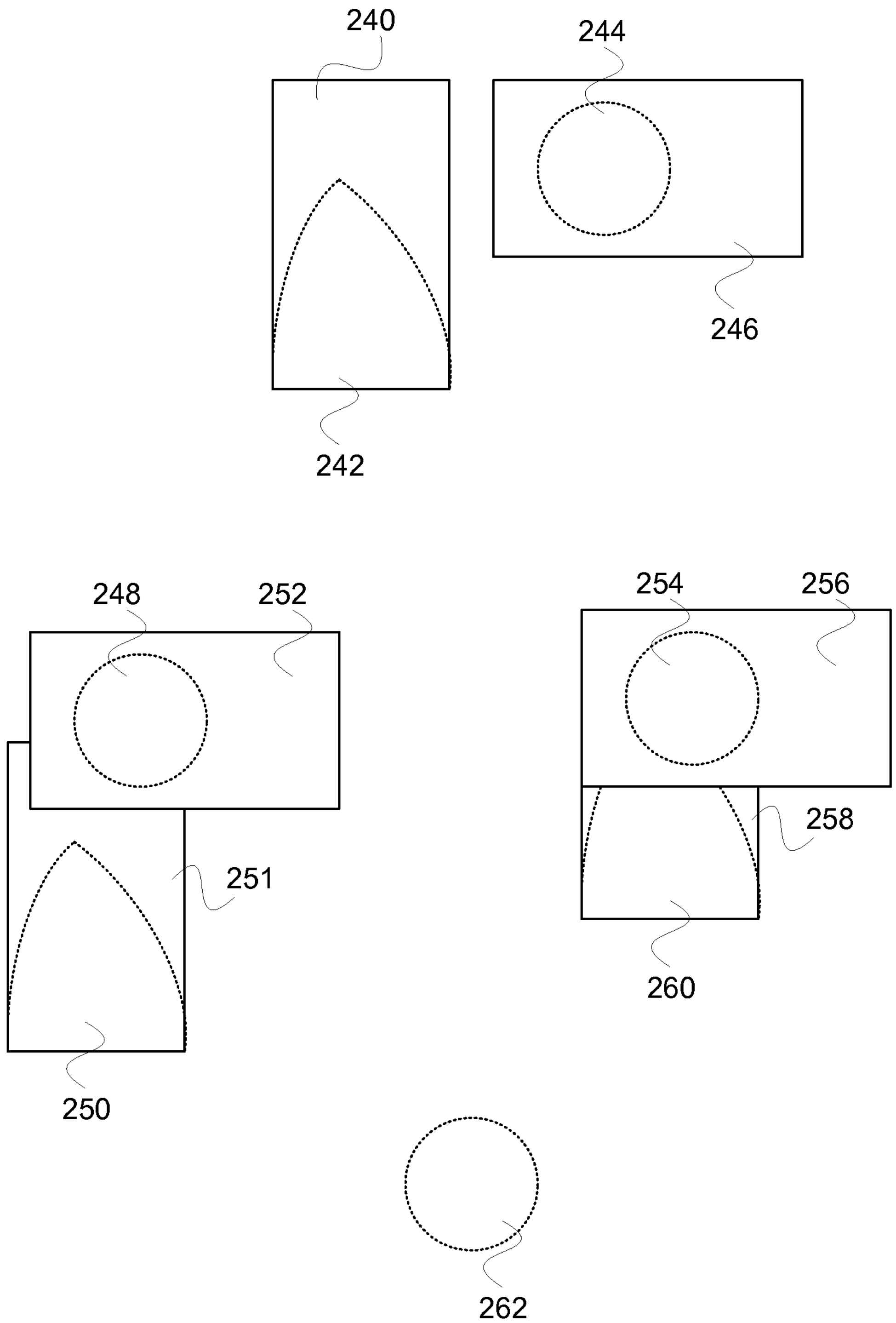

FIG. 2C

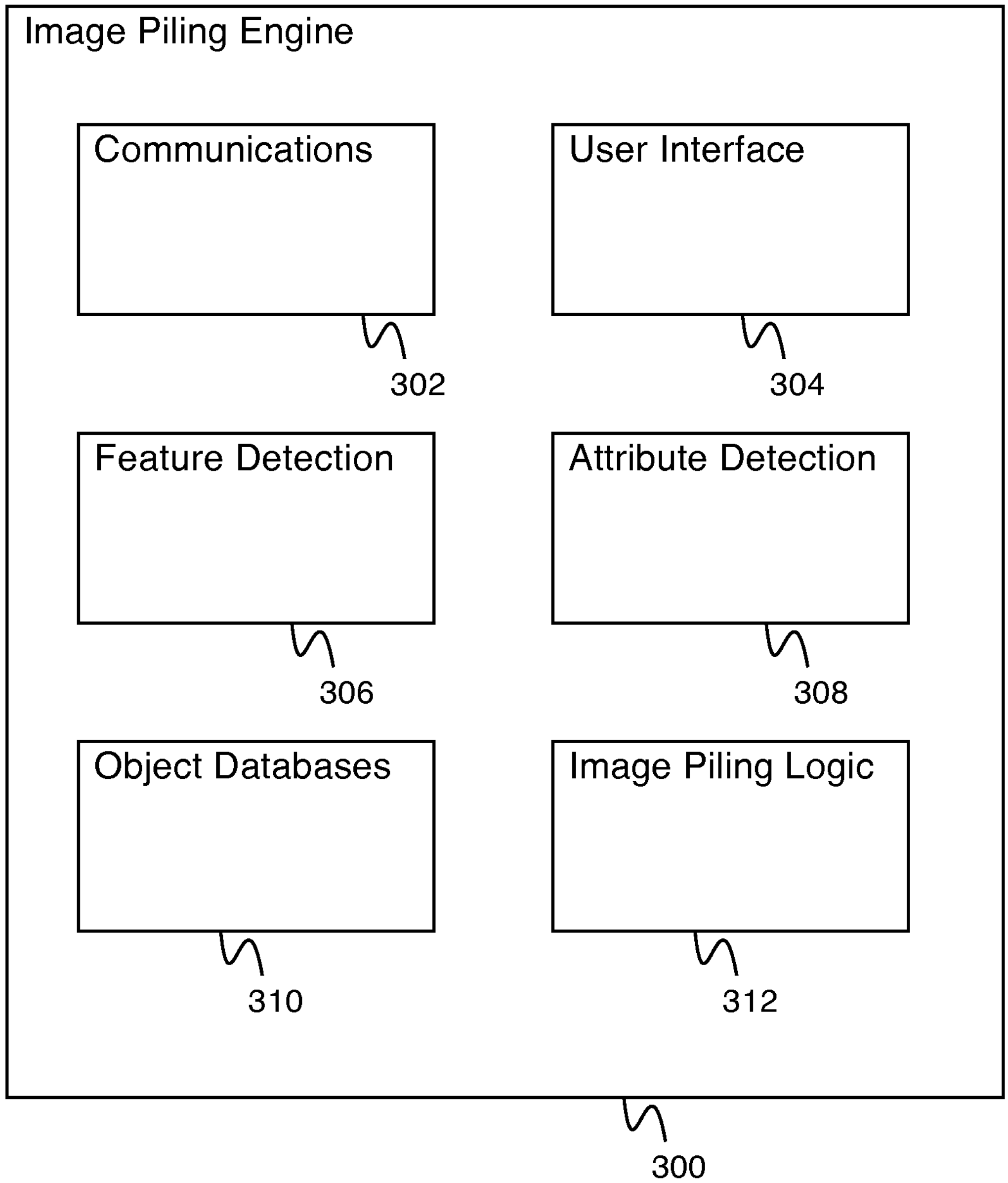
Image Piling Engine
Communications
302
User Interface
304
Feature Detection
306
Attribute Detection
308
Object Databases
310
Image Piling Logic
312
300

Fig. 3A

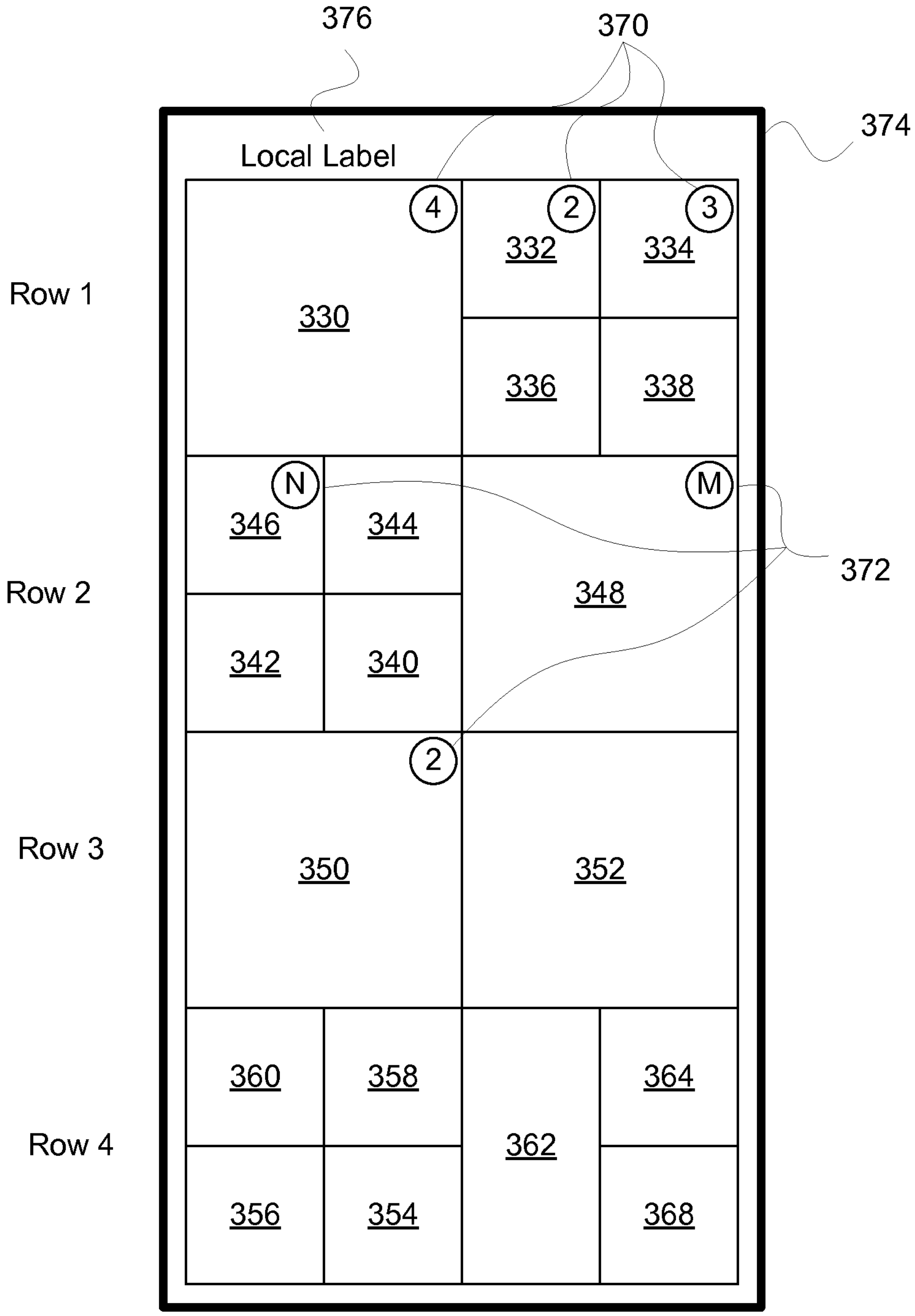
376
370
374
Local Label
4
2
3
332
334
Row 1
330
336
338
N
M
346
344
372
Row 2
348
342
340
2
Row 3
350
352
360
358
364
Row 4
362
356
354
368

FIG. 3B

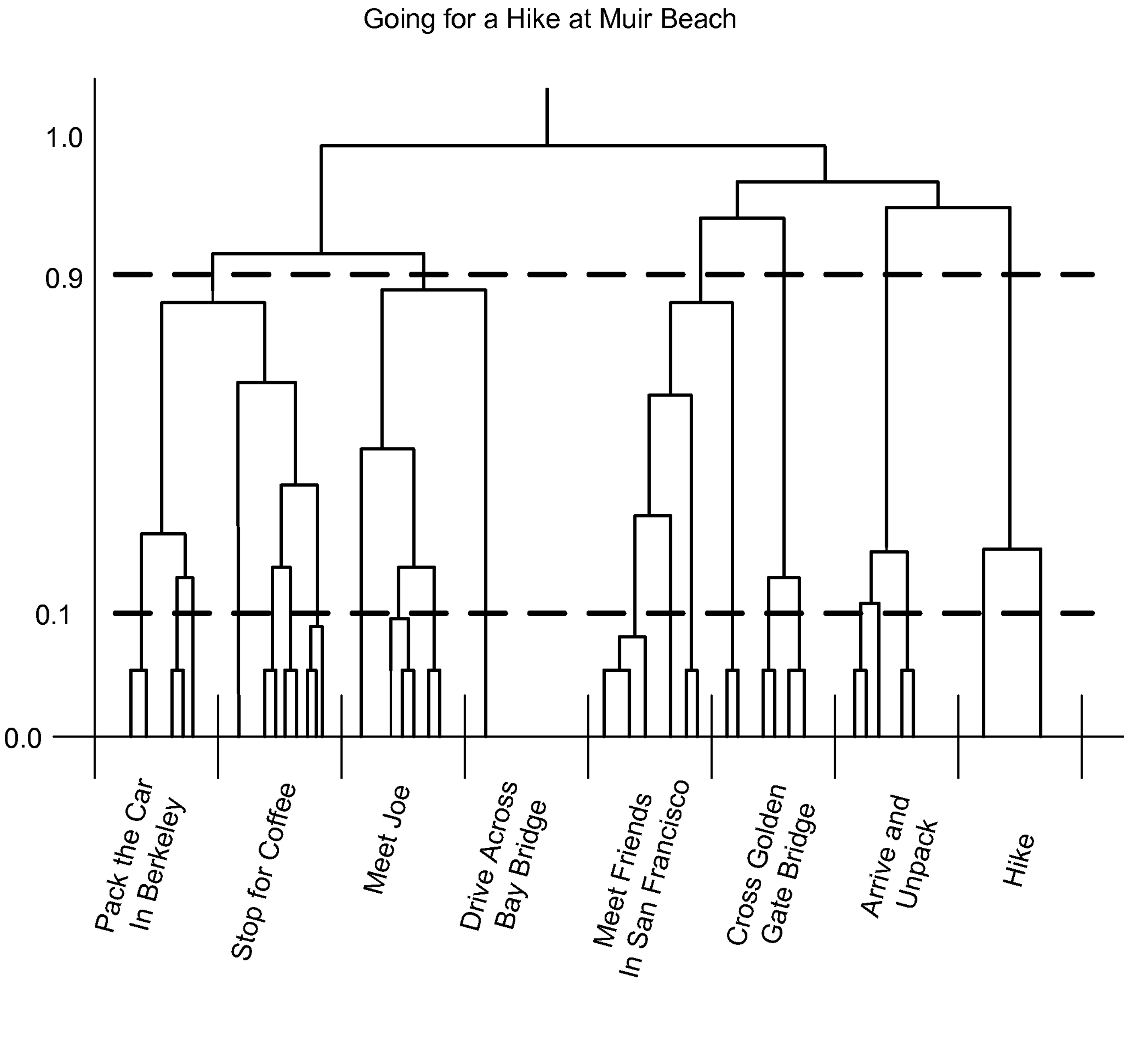
Going for a Hike at Muir Beach
1.0
0.9
0.1
0.0
Pack the Car In Berkeley
Stop for Coffee
Meet Joe
Drive Across Bay Bridge
Meet Friends In San Francisco
Cross Golden Gate Bridge
Arrive and Unpack
Hike

FIG. 3C

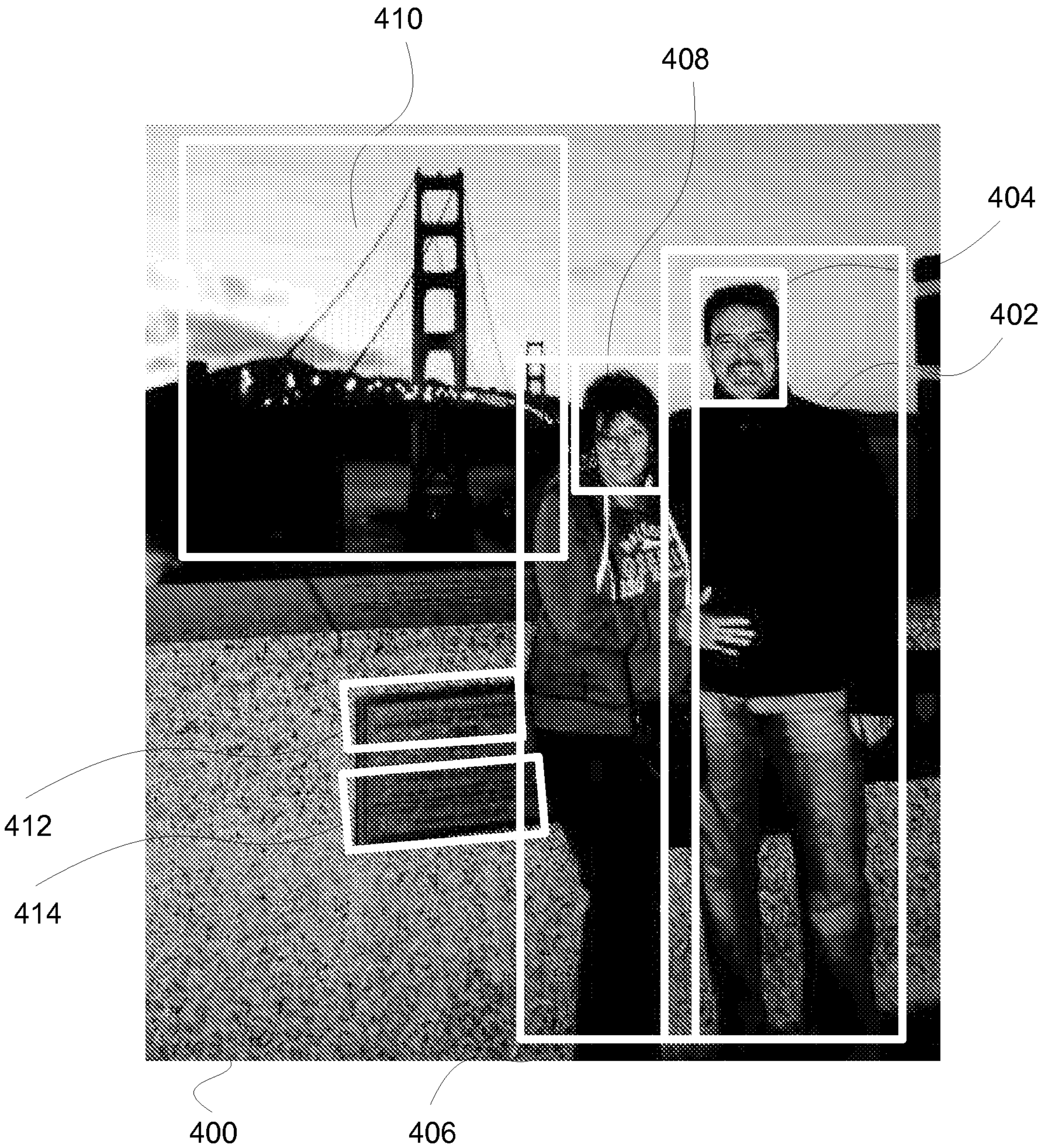

FIG. 4

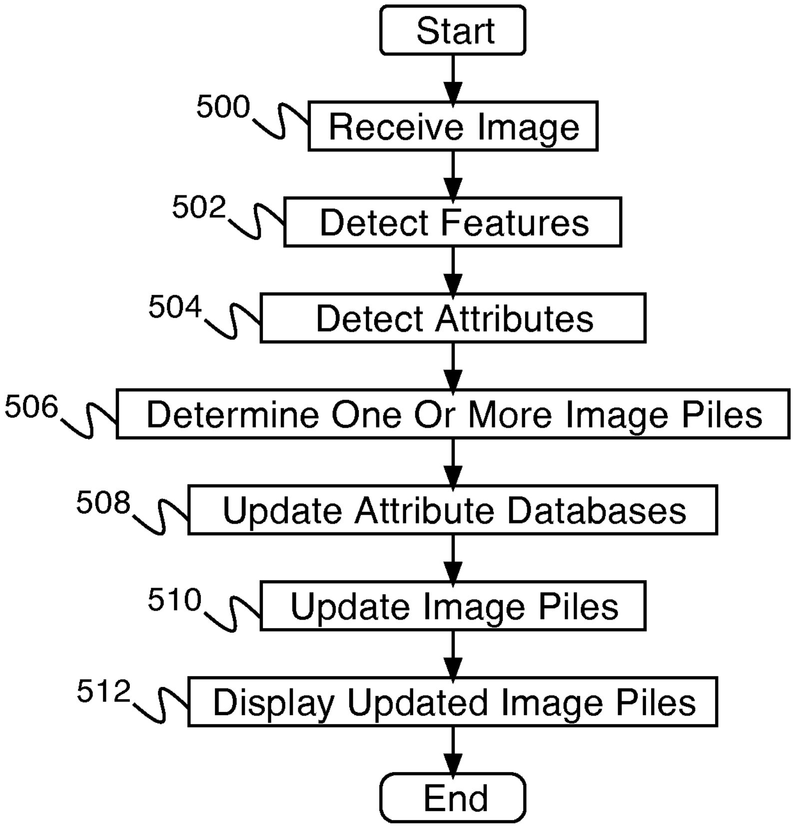
Start
500
Receive Image
502
Detect Features
504
Detect Attributes
506
Determine One Or More Image Piles
508
Update Attribute Databases
510
Update Image Piles
512
Display Updated Image Piles
End

Fig. 5

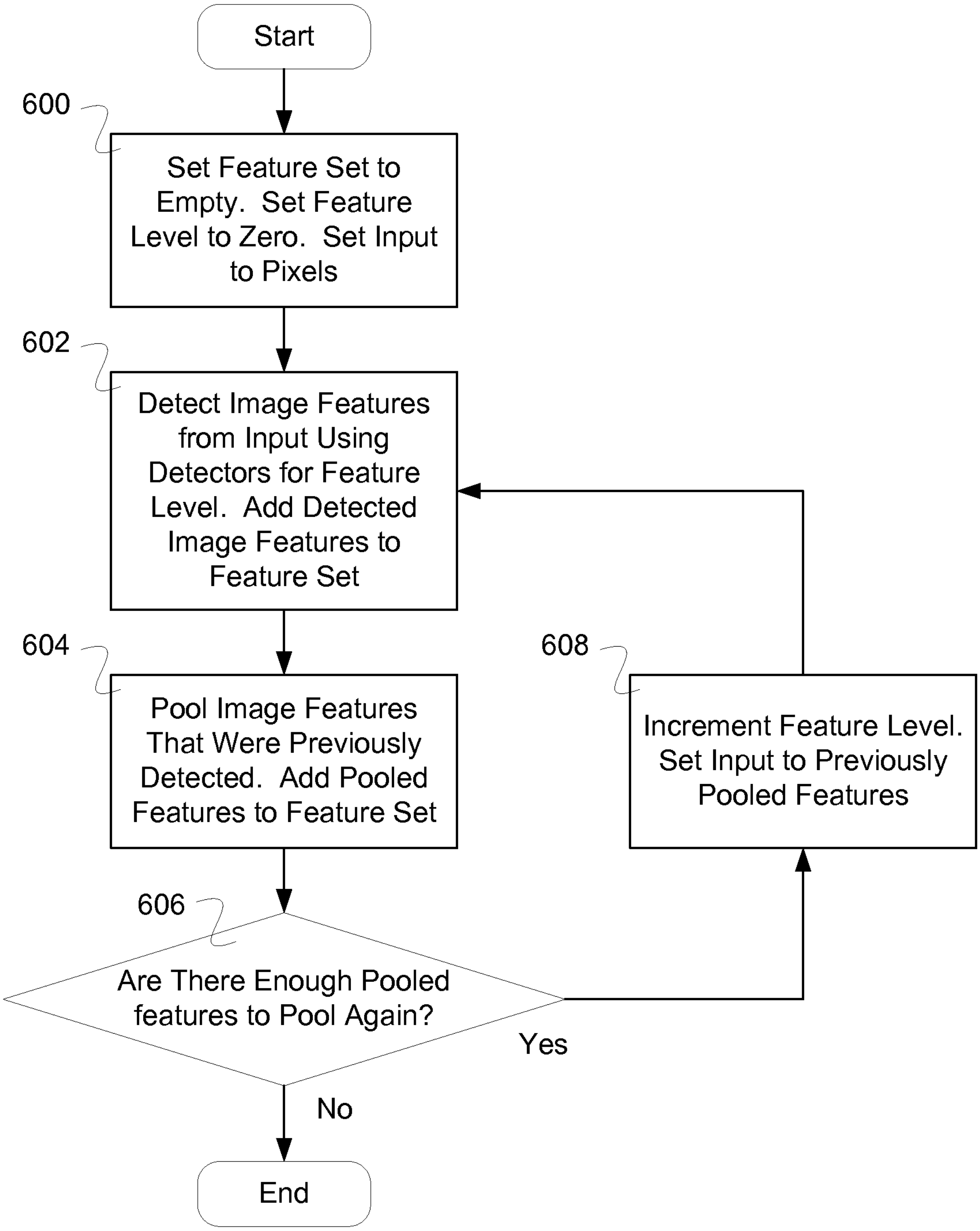
Start
600
Set Feature Set to Empty. Set Feature Level to Zero. Set Input to Pixels
602
Detect Image Features from Input Using Detectors for Feature Level. Add Detected Image Features to Feature Set
604
Pool Image Features That Were Previously Detected. Add Pooled Features to Feature Set
606
Are There Enough Pooled features to Pool Again?
Yes
No
End
608
Increment Feature Level. Set Input to Previously Pooled Features

FIG. 6

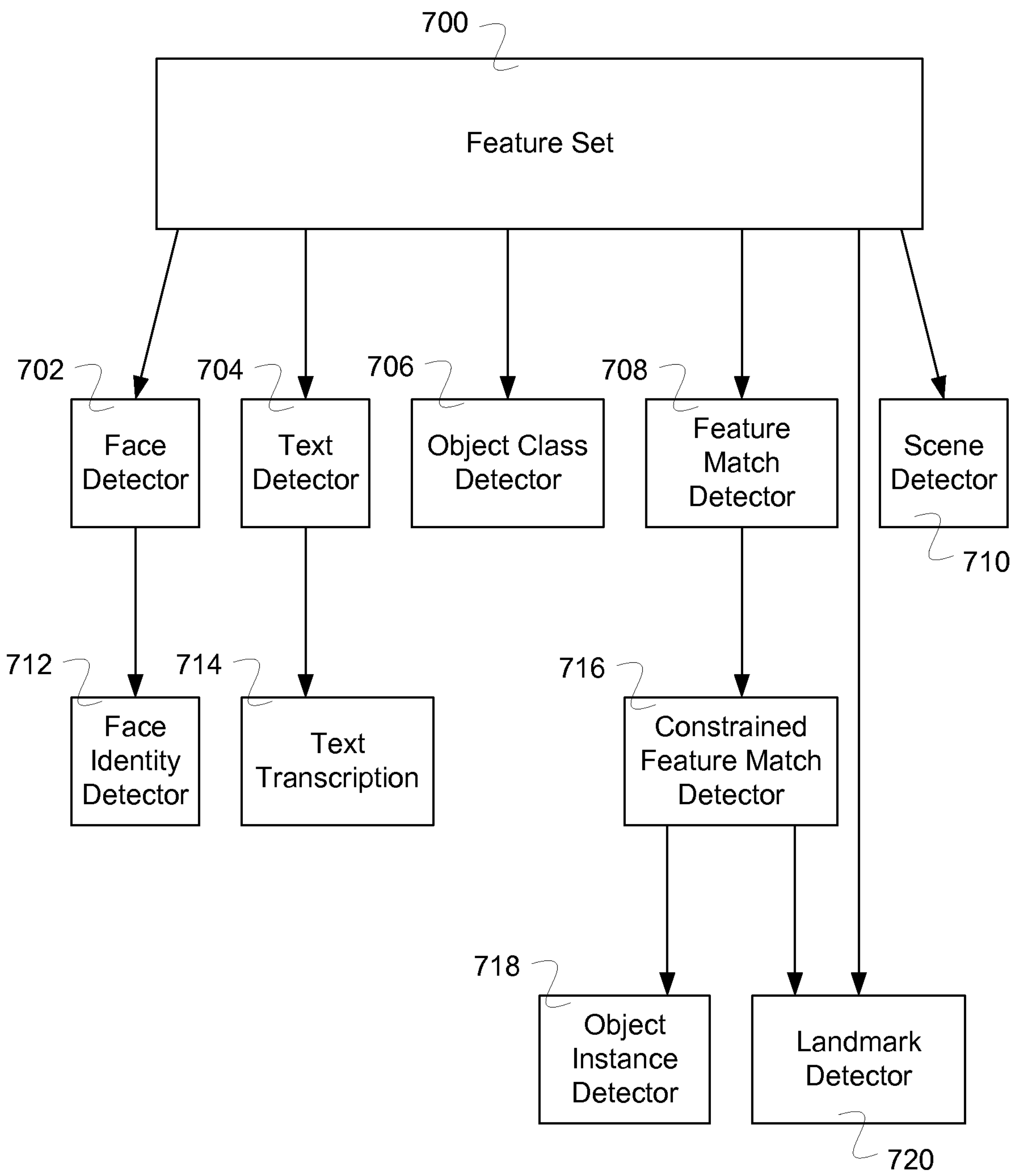
700
Feature Set
702
Face Detector
704
Text Detector
706
Object Class Detector
708
Feature Match Detector
Scene Detector
710
712
Face Identity Detector
714
Text Transcription
716
Constrained Feature Match Detector
718
Object Instance Detector
Landmark Detector
720

FIG. 7

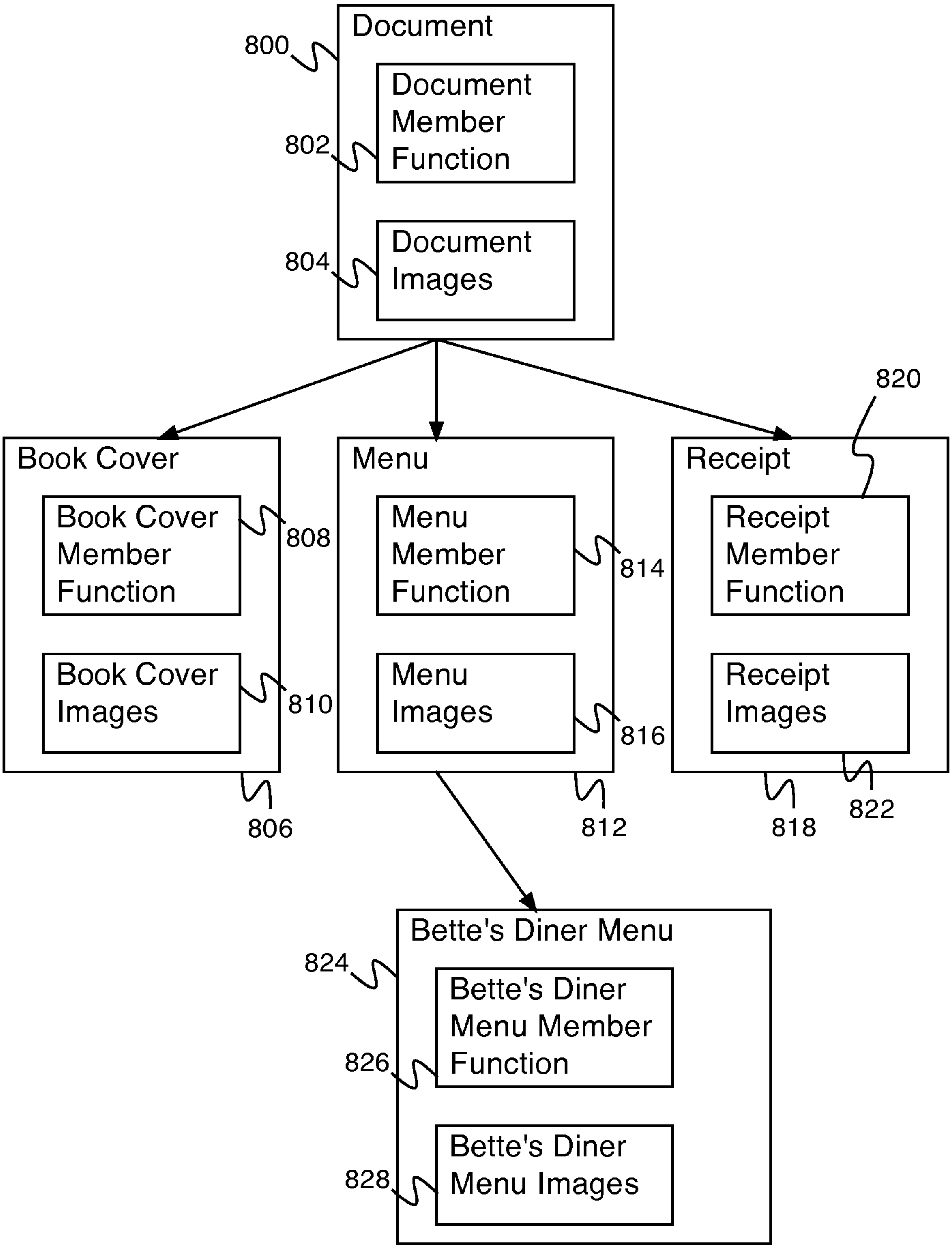
Document
800
Document Member Function
802
Document Images
804
Book Cover
Book Cover Member Function
808
Book Cover Images
810
806
Menu
Menu Member Function
814
Menu Images
816
812
Receipt
820
Receipt Member Function
Receipt Images
818
822
Bette's Diner Menu
824
Bette's Diner Menu Member Function
826
Bette's Diner Menu Images
828

Fig. 8

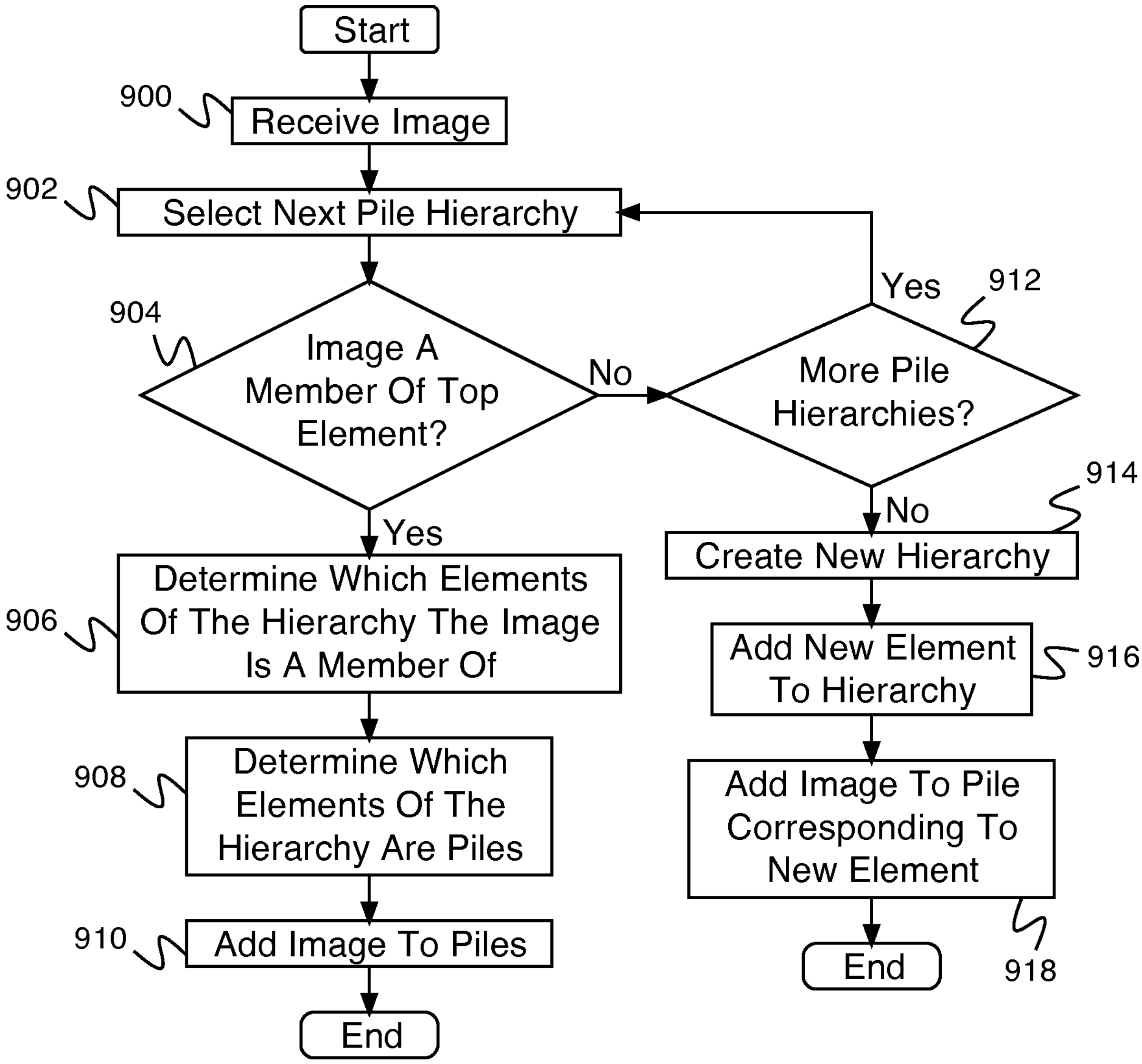
Start
900
Receive Image
902
Select Next Pile Hierarchy
904
Image A Member Of Top Element?
No
Yes
912
More Pile Hierarchies?
Yes
No
914
Create New Hierarchy
906
Determine Which Elements Of The Hierarchy The Image Is A Member Of
916
Add New Element To Hierarchy
908
Determine Which Elements Of The Hierarchy Are Piles
Add Image To Pile Corresponding To New Element
918
910
Add Image To Piles
End
End

Fig. 9

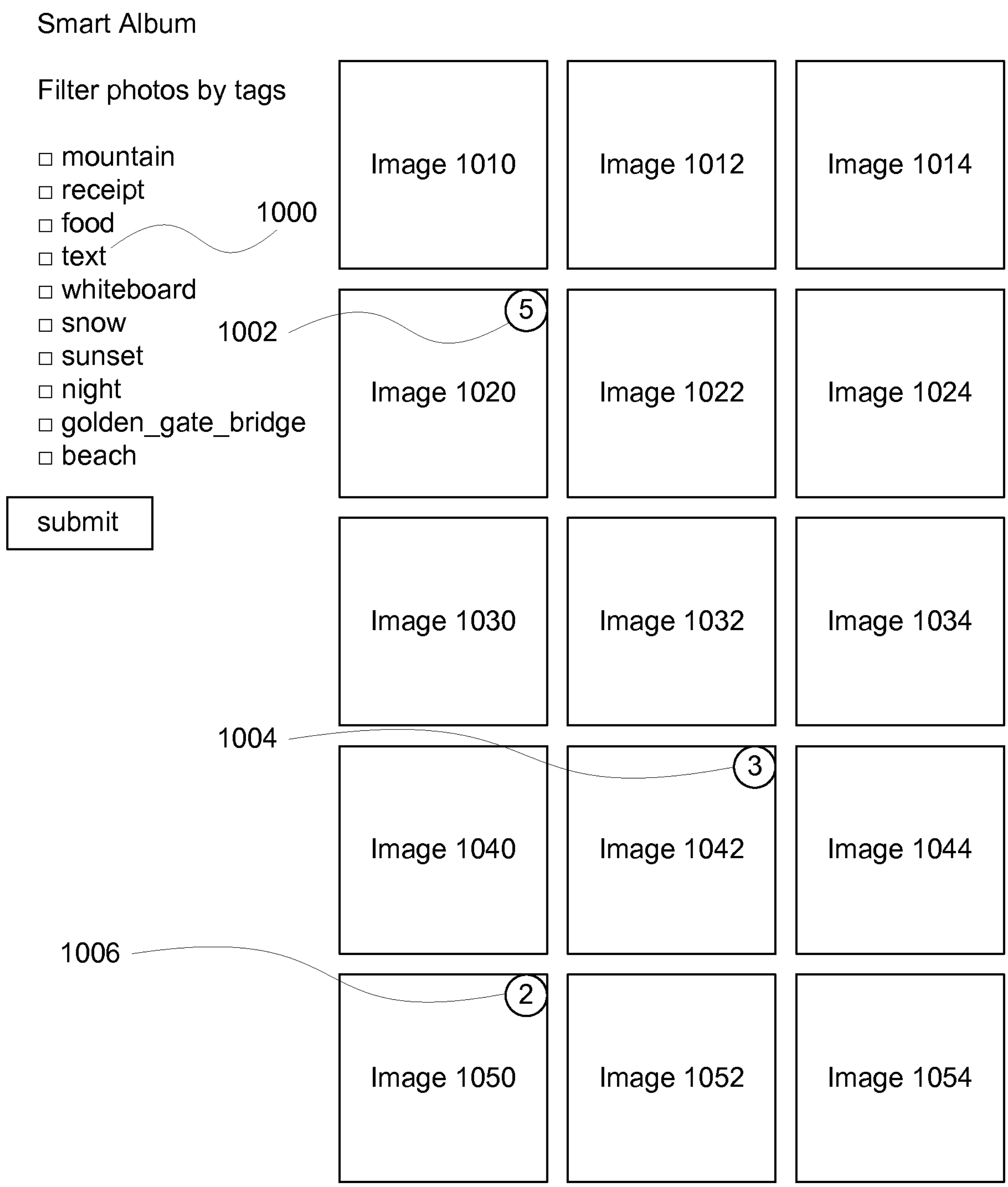
Smart Album
Filter photos by tags
□ mountain
□ receipt
□ food
□ text
□ whiteboard
□ snow
□ sunset
□ night
□ golden_gate_bridge
□ beach
submit
1000
1002
1004
1006
5
3
2
Image 1010
Image 1012
Image 1014
Image 1020
Image 1022
Image 1024
Image 1030
Image 1032
Image 1034
Image 1040
Image 1042
Image 1044
Image 1050
Image 1052
Image 1054

FIG. 10

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/019563

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/30(2006.01)i, G06F 3/14(2006.01)i, G06F 3/048(2006.01)i

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F 17/30; G06K 9/46; G06Q 30/00; G09G 5/00; G06F 3/14; G06F 3/048

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & keywords: image, attribute, sort, hierarchy

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2012-0054208 A1 (YOSHINOBU KITAMARU) 1 March 2012 See paragraphs [0016], [0080]-[0089]; claim 1; and figure 2.	1-18
A	US 2008-0150892 A1 (JONATHAN ANTHONY DUHIG et al.) 26 June 2008 See paragraphs [0068]-[0072]; and figure 6.	1-18
A	US 2011-0161174 A1 (ANDREW SIMMS et al.) 30 June 2011 See paragraphs [0131]-[0134]; and figures 11-12.	1-18
A	US 2011-0170782 A1 (KOTA IWAMOTO et al.) 14 July 2011 See paragraphs [0068]-[0075]; and figures 5-6.	1-18
A	KR 10-2006-0076311 A (NOKIA CORPORATION) 4 July 2006 See abstract, pages 5-7; and figure 2.	1-18

☐ 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
26 June 2014 (26.06.2014)	**27 June 2014 (27.06.2014)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 302-701, Republic of Korea
Facsimile No. +82-42-472-7140

Authorized officer
JI, Jeong Hoon
Telephone No. +82-42-481-5688

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2012-0054208 A1	01/03/2012	CN 101165689 A	23/04/2008
		CN 101165689 B	02/06/2010
		JP 05149570 B2	20/02/2013
		JP 2008-123495 A	29/05/2008
		RU 2007138224 A	20/04/2009
		RU 2378685 C2	10/01/2010
		US 2008-0091749 A1	17/04/2008
		US 8078627 B2	13/12/2011
US 2008-0150892 A1	26/06/2008	AU 2006-252190 A1	10/07/2008
		AU 2006-252190 B2	25/03/2010
US 2011-0161174 A1	30/06/2011	AU 2007-306939 A1	17/04/2008
		AU 2007-306939 B2	07/06/2012
		WO 2008-043160 A1	17/04/2008
US 2011-0170782 A1	14/07/2011	CN 102165490 A	24/08/2011
		CN 102165490 B	22/01/2014
		WO 2010-023808 A1	04/03/2010
KR 10-2006-0076311 A	04/07/2006	EP 1668540 A1	14/06/2006
		JP 2007-507775 A	29/03/2007
		JP 2010-191997 A	02/09/2010
		US 2005-0076056 A1	07/04/2005
		US 7313574 B2	25/12/2007
		WO 2005-031601 A1	07/04/2005