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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 June 2017 (01.06.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/091282 A1

(51) International Patent Classification:
G06F 17/30 (2006.01) *G06F 13/16* (2006.01)

(21) International Application Number:
PCT/US2016/052454

(22) International Filing Date:
19 September 2016 (19.09.2016)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
14/948,892 23 November 2015 (23.11.2015) US

(71) Applicant: ADVANCED MICRO DEVICES, INC. [US/US]; One AMD Place, Sunnyvale, California 94085 (US).

(72) Inventor: ZHANG, Dong Ping; One AMD Place, Sunnyvale, California 94085 (US).

(74) Agent: MATTIOLI, Thomas A.; Volpe and Koenig, P.C., 30 S. 17th Street, United Plaza, Suite 1800, Philadelphia, Pennsylvania 19103 (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, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

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

Published:

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

(54) Title: METHOD AND APPARATUS FOR PERFORMING A PARALLEL SEARCH OPERATION

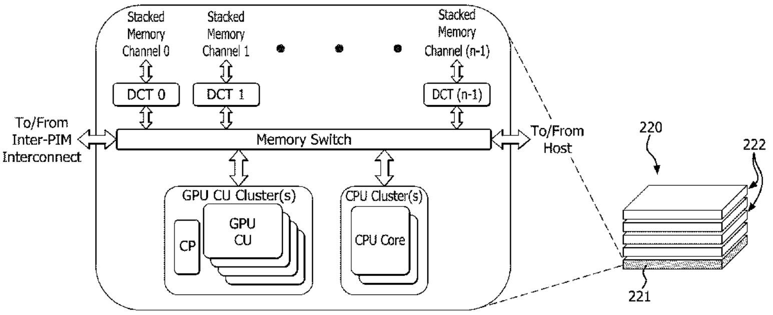

FIG. 4

(57) Abstract: A method and apparatus for performing a search in a processor-in-memory (PIM) system having a first processor and at least one memory module includes receiving one or more images by the first processor. The first processor sends a query for a search of memory for a matching image to the one or more images to at least one memory module, which searches memory in the memory module, in response to the received query. The at least one memory module sends the results of the search to the first processor, and the first processor performs a comparison of the received results from the at least one memory module to the received one or more images.

METHOD AND APPARATUS FOR PERFORMING A PARALLEL SEARCH OPERATION

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Nonprovisional Patent Application No. 14/948,892, filed November 23, 2015, which is incorporated by reference as if fully set forth herein.

FIELD OF THE INVENTION

[0002] The present invention is generally directed to performing a search operation, and more particularly to a method and apparatus for performing a parallel search operation.

BACKGROUND

[0003] The invention of the digital camera, and its incorporation into cellular phones, has ushered in an era of multitudes of pictures being taken and stored than ever before. People are taking hundreds, and even thousands, of pictures and uploading the images to memory, (e.g., on their computer) to a level that they may not even know what images they have. Accordingly, it can be a very daunting task to search memory to see if a particular image is stored in memory, particularly using conventional methods.

[0004] Conventional large scale web image searches are generally performed utilizing a textual "bag of words" scheme. That is, an image includes a textual description, (i.e., bag of words). A text search is initiated, which may optionally also include a geometric or image verification, and images are returned based upon the initiated text search. The "bag of words" approach heavily relies on user supplied meta-data or the accuracy of the object labeling and recognition. Image features are required to be tagged or translated to textual word descriptions. Oftentimes, when a user uploads pictures, for example, the images are uploaded with the image file being given a generic name by the camera during uploading. Users do not always rename all of the images stored in memory with accurate textual descriptions, making searching for particular types of images problematic.

[0005] Additionally, large scale multimedia searching is an issue that search engines and social networking sites are trying to tackle after the text search domain has become relatively mature and well established. With the growing amount of multimedia data, efficient, (e.g., both in terms of service quality and energy consumption), image searching in an image dataset or image searching in mixed unstructured data or videos is becoming important for the purpose of scene/object recognition/reconstruction, copyright attack detection, robotic vision, advertisement placement, and the like. Moreover, supporting web scale image searching may be expensive, as it may require significant hardware with large power consumption.

[0006] It would therefore be beneficial to provide a method and apparatus for performing an image search that searches for images based upon the properties of the images themselves.

SUMMARY OF EMBODIMENTS

[0007] An embodiment directed to a method of performing an image search is disclosed. The method includes performing a search in a processor-in-memory (PIM) system having a first processor and at least one memory module that receives a one or more images by the first processor. The host processor sends a query for a search of memory for a matching image to the one or more images to at least one memory module, which searches memory in the memory module, in response to the received query. The at least one memory module sends the results of the search to the first processor, and the first processor performs a comparison of the received results from the at least one memory module to the received one or more images.

[0008] An embodiment directed to a processor is disclosed. The processor includes circuitry configured to receive one or more images, and circuitry configured to send, to at least one memory module, a query for a search of memory for a matching image to the one or more images. The processor includes circuitry configured to receive, from the at least one memory module, results of a search of images stored in a memory in the memory module, in response to the received query, and circuitry configured to perform a comparison of the received results from the at least one memory module to the received one or more images.

[0009] An embodiment directed to a system is disclosed. The system includes a processor and at least one memory module. The processor comprises circuitry configured to receive one or more images, circuitry configured to send, to the at least one memory module, a query for a search of memory for a matching image to the one or more images, circuitry configured to receive, from the at least one memory module, results of a search of images stored in a memory in the memory module, and circuitry configured to perform a comparison of the received results from the at least one memory module to the received one or more images. The at least one memory module includes circuitry configured to perform a search of images stored in a memory in the memory module, in response to the received query, and circuitry configured to send the results of the search to the processor.

[0010] An embodiment directed to a method implemented in a processor-in-memory (PIM) system having a first processor and a plurality of memory modules is disclosed. The method includes uploading and classifying one or more images by their image data. The images are partitioned and stored on the plurality of memory modules based upon a classification of the image data. One or more query images is received by the first processor and the first processor sends, to the plurality of memory modules, a query for a search of memory for a matching image to the one or more query images. Each memory module of the plurality of memory modules performs a search of images stored in a memory in that memory module, in response to the received query. Each memory module of the plurality of memory modules sends results of the search to the first processor, and the first processor performs a comparison of the received results from the plurality of memory modules to the received one or more images.

[0011] An embodiment directed to a non-transitory computer-readable medium is disclosed. The non-transitory computer-readable medium has instructions recorded thereon that, when executed by a computing device, cause the computing device to perform operations including receiving one or more images, sending a query for a search of memory for a matching image to the one or more images, performing a search of images stored in a memory in a memory module, in response to the received query, sending results of the search, and performing a comparison of the received results from the memory module to the received one or more images.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] A more detailed understanding may be had from the following description, given by way of example in conjunction with the accompanying drawings wherein:

[0013] Figure 1 is a block diagram of an example device in which one or more disclosed embodiments may be implemented;

[0014] Figure 2 is a schematic representation of an example architecture for performing a parallel search operation according to an embodiment;

[0015] Figure 3 is a block diagram of an example structure for performing a parallel search operation according to an embodiment;

[0016] Figure 4 is an exploded schematic and block diagram of an example memory module of Figures 2 and 3 according to an embodiment;

[0017] Figure 5 is an example image comparison in accordance with an embodiment; and

[0018] Figure 6 is a flow diagram of an example method of performing a parallel search operation according to an embodiment.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[0019] Although a more detailed description of the embodiments is provided below, briefly a hybrid image search mechanism for a processor-in-memory (PIM) system is provided. There are various strategies that may be utilized for the compression of image features/descriptors. The image descriptors include several categories, such as, a global descriptor, (e.g., GIST), a local descriptor, (e.g., Scale Invariant Feature Transform (SIFT), Oriented FAST and Rotated Brief (ORB) and SURF), or a regional descriptor, (e.g, the basic properties of a region such as average greyscale or variance). While the global descriptor can represent the image relatively efficiently both in terms of storage space and its computational cost, global-descriptor-based image comparison can be less robust and less discriminant compared with local approaches. For example, the global descriptor may be sensitive to transformations, changes of viewpoint, illumination changes, cluttering, cropping, occlusion, and the like. However, the local descriptor approach can be very expensive to compute and its feature sets are often much larger than the global and regional ones for storage. But its invariance to rotation and scaling, and robustness towards other formats of image variations may make it useful in the image search domain, in particular as the main component of a "Bag of Features" (BoF) approach.

[0020] The method and apparatus described below relates to image searching with no additional metadata or tags being specified by users. That is, using image data alone to eliminate any dependency on user provided information. Many components of image searching may be performed in the local memory stack for each PIM, such as computing global descriptors for efficiency purposes with relatively low accuracy, classifications and dimensionality reduction of feature sets. Subsequent local descriptor computation and verification may also be performed in the local stack within a reduced search space, which may incur higher computational cost and higher accuracy levels. The query image, (i.e., image to be searched), is duplicated on each PIM and the search results from each PIM are presented to the host processer where they are merged to construct the final rank. This in-memory processing may reduce the aggregated data traffic across an entire system, as memory accesses are limited to predominantly the local stack or stacks. Accordingly, search efficiency per query may be increased, while energy consumption per query is reduced. Accordingly, below is described a hybrid method and apparatus that may utilize the merits that both global and local descriptors offer, while limiting their drawbacks, in a PIM-enabled system.

[0021] Figure 1 is a block diagram of an example device 100 in which one or more disclosed embodiments may be implemented. The device 100 may include a computer, for example, a desktop computer, a tablet computer, a gaming device, a handheld device, a set-top box, a television, or a mobile phone. The device 100 includes a processor 102, a memory 104, a storage 106, one or more input devices 108, and one or more output devices 110. The device 100 may also optionally include an input driver 112 and an output driver 114. It is understood that the device 100 may include additional components not shown in Figure 1.

[0022] The processor 102 may include a central processing unit (CPU), a graphics processing unit (GPU), a CPU and GPU located on the same die, or one or more processor cores, wherein each processor core may be a CPU or a GPU. The memory 104 may be located on the same die as the processor 102, or may be located separately from the processor 102. The memory 104 may include a volatile or non-volatile memory, for example, random access memory (RAM), dynamic RAM, or a cache.

[0023] The storage 106 may include a fixed or removable storage, for example, a hard disk drive, a solid state drive, an optical disk, or a flash drive. The input devices 108 may include a keyboard, a keypad, a touch screen, a touch pad, a detector, a microphone, an accelerometer, a gyroscope, a biometric scanner, or a network connection (e.g., a wireless local area network card for transmission and/or reception of wireless IEEE 802 signals). The output devices 110 may include a display, a speaker, a printer, a haptic feedback device, one or more lights, an antenna, or a network connection (e.g., a wireless local area network card for transmission and/or reception of wireless IEEE 802 signals).

[0024] The input driver 112 communicates with the processor 102 and the input devices 108, and permits the processor 102 to receive input from the input devices 108. The output driver 114 communicates with the processor 102 and the output devices 110, and permits the processor 102 to send output to the output devices 110. It is noted that the input driver 112 and the output driver 114 are optional components, and that the device 100 will operate in the same manner if the input driver 112 and the output driver 114 are not present. Although described embodiments include a main display, the invention may be practiced without a main display, and only include a source device of video. In this way, the control territory may be an office environment with a plurality of portable devices and no main display.

[0025] Figure 2 is a schematic representation of an example architecture 200 for performing a parallel search operation according to an embodiment. The architecture 200 may include a host processor 210, memory modules 220, (designated 220_1-220_8), in-package photonics units 230, and additional memory modules 240, (designated 240_1 and 240_2). The host processor 210 may be a heterogeneous high performance accelerated processing unit (APU). The memory modules 220 may include PIM modules having a logic die stacked beneath one or more dynamic random access memory (DRAM) dies. The additional memory modules 240 may also include PIM modules having logic dies stacked beneath one or more DRAM dies, and may be external to the chip to provide additional memory capacity. Memory modules 240 may be in communication via a link 250, (e.g., optical link), with the host processor 210, and may be considered second level memory, such as optical memory and the like.

[0026] The photonics units 230 may be used to send/receive data through an interconnect link, (e.g., the optical link 250). It should be noted that an optical link (250) is one example link, but additional types of links could be employed, (e.g., electrical). Additionally, logic die 221 in the memory module 220 may provide interface and RAS features.

[0027] Figure 3 is a block diagram of an example structure for performing a parallel search operation according to an embodiment. Figure 3 depicts, for example, the host processor 210 in communication with memory modules 220 via a host-memory interface. In addition, the memory modules 220 are in communication with one another via an inter-PIM interconnect. Further, in the example diagram of Figure 3, an example memory module 220 is depicted as including logic die with PIM 221 and one or more memory dies 222. The host processor 210 and memory modules 220 are disposed upon an interposer 260. The example structure depicted in Figure 3 may be based on the architecture 200 and may be referred to as a ring structure, where the memory modules 220 are situated around the host process 210 in a ring pattern. Figure 4 is an exploded schematic and block diagram of an example memory module 220 of Figures 2 and 3 according to an embodiment. More particularly, Figure 4 depicts an example schematic diagram of the logic die 221. The logic die 221 includes one or more graphics processing unit (GPU) computing unit (CU) cores in a GPU CU cluster, or clusters and one or more central processing unit (CPU) cores in a CPU cluster, or clusters. The GPU CU and CPU cluster(s) communicate with the memory dies 222 via a memory switch through the stacked memory channels 0 – (n-1), which may be in communication with memory areas, (e.g., DCT 0, DCT 1 ...(n-1)), located on one or more memory dies 222. Additionally, the logic die communicates to/from the host via the memory switch to the host-memory interface, and to other memory modules 220 via the memory switch via the inter-PIM interconnect.

[0028] Figure 5 is an example image comparison 500 in accordance with an embodiment. In comparison 500, a query image 501 is compared to an image stored in memory 502, which is an identical image of image 501 that has been rotated ninety (90) degrees clockwise.

[0029] Figure 6 is a flow diagram of an example method 600 of performing a parallel search operation according to an embodiment. The method 600 computes the global descriptor, (e.g., descriptors that describe attributes of the entire image such as color, etc.), and performs coarse level classification and image searching at a first level, to reduce the search space and cost for a local descriptor approach significantly. Secondly, a local descriptor approach, (e.g., localized image attributes of area within the images), is employed to perform a more accurate match and verification.

[0030] In step 610, images are loaded into memory and classified. Classification may include determining which images include similar image attributes and classifying those similar images as a class of images, (e.g., images of snowstorms may include similar attributes and categorized into a class). The image data may be partitioned and placed randomly among all the memory stacks, (e.g., memory modules 220 and/or 240). Once the image data is classified, a number of ways exist for subsequent placement methods to manage the image data and searching on the memory stacks.

[0031] For example, images of the same class may be scattered on all stacks instead of concentrating on one. The scattering algorithms may be any type of randomization, hashing, or other algorithmic mapping. Alternatively, there may be no grouping of any images after the classification. That is, each image still remains in its initial random assignment on the memory stack. In another alternative, images of the same categories may be placed on one memory stack, or multiple adjacent ones if there is not enough space in one. A hierarchical classification may be used to derive sub-categories to achieve this placement schematically. For example, if there are N images and two classes, a first level of classification divides the images into two categories, each containing N/2 images. However, each memory module may not be able to accommodate N/2 images, therefore a second level of classification may be performed to determine sub-categories within the first class, (i.e., within the N/2 images), and images belonging to an individual sub-category may be stored in the same memory module as much as possible. Further nesting, (e.g., additional sub-categories to the sub-categories), may be utilized to further classify images similar to one another and have those similar images stored as much as possible on the same module.

[0032] A number of classification algorithms may be utilized for the classification. For example, K-nearest neighbor, supporting vector machine, kernel methods, and other types of classification. In the case of a vast number of dimensions that feature vectors may include, linear or nonlinear dimensionality reduction algorithms, (e.g., manifold learning, Fisher kernel, Principal Component Analysis (PCA), and others) may be employed along with the classification. The linear or nonlinear reduction algorithms may also be used primarily for visualization purpose to inspect the quality of the classification.

[0033] In step 620, one or more query images are uploaded for comparison to the loaded images in memory. For example, the query images may be on a flashdrive that is inserted into a computer where images are stored in memory. At this point, for each query (q), global descriptors are initially computed and weak classification is performed, generating a vector P(q), here P is a vector with m elements, and each element represents how likely image q belongs to one category out of the m categories in total. That is, in the training phase, where images are first uploaded, global descriptors of all images in the database may be computed on each PIM that contains a subset of the images from the collection. For example, referring back to Figure 5, the global attributes of query image 501 are compared to the global attributes of image 502 stored in memory.

[0034] The query q is duplicated and sent to one or more of the PIMs in parallel for searching (step 630). That is, where different classifications of images are stored on more than one memory module, the query q is sent to multiple PIMs, (e.g., 220 and/or 240), for searching.

[0035] The PIMs then compute global descriptors (step 640). For example, the in-memory search at each PIM may begin with comparing each search entry with the local images categorized as being in the same class as the highest ranked element of the vector P(q), then proceed to subsequent ones in order. For each comparison, a similarity measurement is calculated (e.g., the similar attributes of a local image as compared to the searched query image), and a determination is made as to whether the similarity measurement exceeds a threshold.

[0036] For example, given one query image, the descriptor of this query image may be computed either on the host processor 210, with the query image and its descriptor then distributed to all PIMs for ranking and searching. Alternatively, the query image alone may be distributed to all the PIMs and its descriptor may be computed locally on each PIM. For a group query, (i.e., where multiple query images have been uploaded and are queried simultaneously), the aforementioned two alternatives may be applied to the group as a whole. That is, descriptors for multiple query images may be computed on the host processor 210 and the multiple query images and descriptors distributed to all PIMs for ranking and searching, or the multiple query images may be distributed to all the PIMs, which compute the descriptors locally on each PIM. Another alternative for a group query is to distribute a subset of the queries to each PIM and compute their descriptors locally, perform a search and forward the images and descriptors to another PIM to resume the ranking/search on another PIM module. The result on each module may then be aggregated either on the host or one of PIM modules.

[0037] If the similarity measurement exceeds the threshold, a local descriptor based comparison is then performed for further verification. That is, the PIMs compute the local descriptors (step 650). For example, localized points 503a corresponding to a part of the sun in images 501/502, 503b corresponding to a top of a tree in images 501/502, and 503c corresponding to a mountain peak in images 501/502, are compared to one another. Where a dimensionality reduction method is used, the number of comparisons and verifications may be reduced to increase the efficiency of this search.

[0038] In the case where images of the same categories were placed on one or adjacent memory stacks in step 610, the query q may be conducted in the order of the weak classification such that the PIM containing the images from the highest ranked class is performing the search first, followed by the second highest one and so on. In order to prevent a potential processing delay. A reordering of potential matches to the queries may be utilized to increase the average efficiency while degrading the response of a small number of queries. That is, potential matches to the query image may be re-ordered. The potential matches may be determined based on how similar the global descriptor of the query and the potential matches are. The local descriptors of the query and these potential matches are then compared to compute a pair-wise similarity of the query and each potential match. The potential matches may then be re-ordered based on this similarity metric.

[0039] As part of the local descriptor computation, a verification via image registration scheme may be employed, where a target image is one potential match in the data base, and the source image is the query image. Various local descriptor based verification schemes may be utilized. For example, verification via matching the SIFT descriptors of two images is one verification scheme that may be used, which compares the local features between a source and target image.

[0040] By having feature extraction and descriptor computation performed locally, each PIM processor, (e.g., logic die 221 of each memory module 220), is acting independently from each other and computing such descriptors of the images in its local memory stack. The computed features or descriptors can be compressed to a smaller fingerprint feature vector before inter-PIM communication or PIM-host communication for the purpose of classification and search on multiple PIMs. Additionally, they can be directly utilized. That is, the computed features or descriptors may be utilized to rank images prior to sending to host processor 210.

Once each local PIM has performed the global descriptor, local descriptor and verification phases, the search results are provided to the host processor, (e.g., 210) in step 660. The host processor then constructs the final rank (step 670) for which loaded images match the uploaded and searched images from all of the PIMs from which it receives images. Having the above locally oriented computation scheme with in-memory processing applying to both the global and local descriptor computations, as well as the verification phase may minimize aggregated data movement. This, in turn, may minimize the energy cost associated with such data transfer, and reduce the searching efficiency that the limitation of frequent memory accesses places on it.

[0041] It should be understood that many variations are possible based on the disclosure herein. Although features and elements are described above in particular combinations, each feature or element may be used alone without the other features and elements or in various combinations with or without other features and elements.

[0042] For example, the topology for the host processor and memory modules depicted in Figures 2 and 3 may be referred to as a ring topology. However, any topology may be utilized. Additionally, the host processor 210 may be a processor on one of the PIMs. Further, although an example single PIM node system has been described the methods and apparatus are not limited to any particular PIM system configuration. For example, multiple nodes may be connected to form a PIM network, or a secondary memory system, (e.g., memory modules 240), may be attached to the PIM node. The PIM node may be arranged as a ring or star structure with multiple memory stacks and PIMs. The PIM system can also be augmented with the two-level memory as shown in Figure 2, where each secondary level memory component, (e.g., memory modules 240), may be a PIM. The methods provided may be implemented in a general purpose computer, a processor, or a processor core. Suitable processors include, by way of example, a general purpose processor, a special purpose processor, a conventional processor, a digital signal processor (DSP), a plurality of microprocessors, one or more microprocessors in association with a DSP core, a controller, a microcontroller, Application Specific

Integrated Circuits (ASICs), Field Programmable Gate Arrays (FPGAs) circuits, any other type of integrated circuit (IC), and/or a state machine. Such processors may be manufactured by configuring a manufacturing process using the results of processed hardware description language (HDL) instructions and other intermediary data including netlists (such instructions capable of being stored on a computer readable media). The results of such processing may be maskworks that are then used in a semiconductor manufacturing process to manufacture a processor which implements aspects of the present invention.

[0043] The methods or flow charts provided herein may be implemented in a computer program, software, or firmware incorporated in a computer-readable storage medium for execution by a general purpose computer or a processor. Examples of computer-readable storage mediums include a read only memory (ROM), a random access memory (RAM), a register, cache memory, semiconductor memory devices, magnetic media such as internal hard disks and removable disks, magneto-optical media, and optical media such as CD-ROM disks, and digital versatile disks (DVDs).

[0044] A method implemented in a PIM system having a first processor and at least one memory module is disclosed herein. The method includes performing a search in the PIM system having the first processor and at least one memory module that receives a one or more images by the first processor. The host processor sends a query for a search of memory for a matching image to the one or more images to at least one memory module, which searches memory in the memory module, in response to the received query. The at least one memory module sends the results of the search to the first processor, and the first processor performs a comparison of the received results from the at least one memory module to the received one or more images.

[0045] In some examples, the method includes uploading and classifying the images stored in the memory of the at least one memory module. In some examples, uploaded images from a same class are stored on a plurality of memory modules. In some examples, uploaded images from a same class are stored on the same memory module.

[0046] In some examples, the search of images performed by the at least one memory module includes performing a global descriptor computation, a local descriptor computation, and a verification determination.

[0047] In some examples, the comparison of the received results by the first processor includes ranking the received results received from the at least one memory module. In some examples, the method includes constructing, by the first processor, a final rank of the results received from multiple memory modules.

[0048] A processor is disclosed herein. The processor includes circuitry configured to receive one or more images, and circuitry configured to send, to at least one memory module, a query for a search of memory for a matching image to the one or more images. The processor includes circuitry configured to receive, from the at least one memory module, results of a search of images stored in a memory in the memory module, in response to the received query, and circuitry configured to perform a comparison of the received results from the at least one memory module to the received one or more images.

[0049] In some examples, the processor includes circuitry configured to upload and classify the images stored in the memory of the at least one memory module. In some examples, the processor includes circuitry configured to rank the received results received from the at least one memory module. In some examples, the processor includes circuitry configured to construct a final rank of the results received from multiple memory modules.

[0050] A system is disclosed herein. The system includes a processor and at least one memory module. The processor comprises circuitry configured to receive one or more images, circuitry configured to send, to the at least one memory module, a query for a search of memory for a matching image to the one or more images, circuitry configured to receive, from the at least one memory module, results of a search of images stored in a memory in the memory module, and circuitry configured to perform a comparison of the received results from the at least one memory module to the received one or more images. The at least one memory module includes circuitry configured to perform a search of images stored in a memory in the memory module, in response to the received query, and circuitry configured to send the results of the search to the processor.

[0051] In some examples, the processor of the system includes circuitry configured to upload and classify the images stored in the memory of the at least one memory module. In some examples, uploaded images from a same class are stored on a plurality of memory modules. In some examples, uploaded images from a same class are stored on the same memory module.

[0052] In some examples, the search of images performed by the at least one memory module includes performing a global descriptor computation, a local descriptor computation, and a verification determination. In some examples, the processor of the system includes circuitry configured to rank the received results received from the at least one memory module. In some examples, the processor of the system includes circuitry configured to construct a final rank of the results received from multiple memory modules.

[0053] A method implemented in a processor-in-memory (PIM) system having a first processor and a plurality of memory modules is disclosed herein. The method includes uploading and classifying one or more images by their image data. The images are partitioned and stored on the plurality of memory modules based upon a classification of the image data. One or more query images is received by the first processor and the first processor sends, to the plurality of memory modules, a query for a search of memory for a matching image to the one or more query images. Each memory module of the plurality of memory modules performs a search of images stored in a memory in that memory module, in response to the received query. Each memory module of the plurality of memory modules sends results of the search to the first processor, and the first processor performs a comparison of the received results from the plurality of memory modules to the received one or more images.

[0054] In some examples, uploaded images from a same class are stored on the same memory module. In some examples, uploaded images from a same class are distributed among the plurality of memory modules. In some examples, uploaded images from the same class are distributed among adjacent memory modules.

[0055] In some examples, the search of images performed by the plurality of memory modules includes performing a global descriptor computation, a local descriptor computation, and a verification determination. In some examples, the computing of a global descriptor includes comparing the one or more query images to an image stored in the memory in that memory module categorized as being in a same class as the one or more query images. In some examples, a similarity measurement is computed and compared to a threshold. In some examples, if the similarity measurement exceeds the threshold, a local descriptor is computed.

[0056] A non-transitory computer-readable medium is disclosed herein. The non-transitory computer-readable medium has instructions recorded thereon that, when executed by a computing device, cause the computing device to perform operations including receiving one or more images, sending a query for a search of memory for a matching image to the one or more images, performing a search of images stored in a memory in a memory module, in response to the received query, sending results of the search, and performing a comparison of the received results from the memory module to the received one or more images.

[0057] In some examples, the images stored in the memory of the memory module are uploaded and classified. In some examples, uploaded images from a same class are stored on a plurality of memory modules. In some examples, uploaded images from a same class are stored on the same memory module.

[0058] In some examples, the search of images includes performing a global descriptor computation, a local descriptor computation, and a verification determination. In some examples, the comparison of the received results includes ranking the received results. In some examples, a final rank of the results received from multiple memory modules is constructed.

* * *

CLAIMS

What is claimed is:

1. A method implemented in a processor-in-memory (PIM) system having a first processor and at least one memory module, comprising:

receiving one or more images by the first processor;

sending, by the first processor, to at least one memory module, a query for a search of memory for a matching image to the one or more images;

performing, by the at least one memory module, a search of images stored in a memory in the memory module, in response to the received query;

sending, by the at least one memory module, results of the search to the first processor; and

performing, by the first processor, a comparison of the received results from the at least one memory module to the received one or more images.

2. The method of claim 1, further comprising uploading and classifying the images stored in the memory of the at least one memory module.

3. The method of claim 2 wherein the uploaded images from a same class are stored on a plurality of memory modules.

4. The method of claim 2 wherein the uploaded images from a same class are stored on the same memory module.

5. The method of claim 1 wherein the search of images performed by the at least one memory module includes performing a global descriptor computation, a local descriptor computation, and a verification determination.

6. The method of claim 1 wherein the comparison of the received results by the first processor includes ranking the received results received from the at least one memory module.

7. The method of claim 6, further comprising constructing, by the first processor, a final rank of the results received from multiple memory modules.

8. A processor, comprising:

circuitry configured to receive one or more images;

circuitry configured to send, to at least one memory module, a query for a search of memory for a matching image to the one or more images;

circuitry configured to receive, from the at least one memory module, results of a search of images stored in a memory in the memory module, in response to the received query; and

circuitry configured to perform a comparison of the received results from the at least one memory module to the received one or more images.

9. The processor of claim 8, further comprising circuitry configured to upload and classify the images stored in the memory of the at least one memory module.

10. The processor of claim 8, further comprising circuitry configured to rank the received results received from the at least one memory module.

11. The processor of claim 10, further comprising circuitry configured to construct a final rank of the results received from multiple memory modules.

12. A system, comprising:

a processor; and

at least one memory module;

wherein the processor comprises:

circuitry configured to receive one or more images;

circuitry configured to send, to the at least one memory module, a query for a search of memory for a matching image to the one or more images;

circuitry configured to receive, from the at least one memory module, results of a search of images stored in a memory in the memory module; and

circuitry configured to perform a comparison of the received results from the at least one memory module to the received one or more images; and

wherein the at least one memory module comprises:

circuitry configured to perform a search of images stored in a memory in the memory module, in response to the received query; and

circuitry configured to send the results of the search to the processor.

13. The system of claim 12, wherein the processor further comprises circuitry configured to upload and classify the images stored in the memory of the at least one memory module.

14. The system of claim 13 wherein the uploaded images from a same class are stored on a plurality of memory modules.

15. The system of claim 13 wherein the uploaded images from a same class are stored on the same memory module.

16. The system of claim 12 wherein the search of images performed by the at least one memory module includes performing a global descriptor computation, a local descriptor computation, and a verification determination.

17. The system of claim 12, wherein the processor further comprises circuitry configured to rank the received results received from the at least one memory module.

18. The system of claim 17, wherein the processor further comprises circuitry configured to construct a final rank of the results received from multiple memory modules.

19. A method implemented in a processor-in-memory (PIM) system having a first processor and a plurality of memory modules, comprising:

uploading and classifying one or more images by their image data;

partitioning and storing the images on the plurality of memory modules based upon a classification of the image data;

receiving one or more query images by the first processor;

sending, by the first processor, to the plurality of memory modules, a query for a search of memory for a matching image to the one or more query images;

performing, by each memory module of the plurality of memory modules, a search of images stored in a memory in that memory module, in response to the received query;

sending, by each memory module of the plurality of memory modules, results of the search to the first processor; and

performing, by the first processor, a comparison of the received results from the plurality of memory modules to the received one or more images.

20. The method of claim 19 wherein the uploaded images from a same class are stored on the same memory module.

21. The method of claim 19 wherein the uploaded images from a same class are distributed among the plurality of memory modules.

22. The method of claim 21 wherein the uploaded images from the same class are distributed among adjacent memory modules.

23. The method of claim 19 wherein the search of images performed by the plurality of memory modules includes performing a global descriptor computation, a local descriptor computation, and a verification determination.

24. The method of claim 23 wherein the computing of a global descriptor includes comparing the one or more query images to an image stored in the memory

in that memory module categorized as being in a same class as the one or more query images.

25. The method of claim 24, further comprising computing a similarity measurement and comparing the similarity measurement to a threshold.

26. The method of claim 25 wherein if the similarity measurement exceeds the threshold, a local descriptor is computed.

27. A non-transitory computer-readable medium having instructions recorded thereon that, when executed by a computing device, cause the computing device to perform operations comprising:

receiving one or more images;

sending a query for a search of memory for a matching image to the one or more images;

performing a search of images stored in a memory in a memory module, in response to the received query;

sending results of the search to the first processor; and

performing a comparison of the received results from the memory module to the received one or more images.

28. The non-transitory computer-readable medium of claim 27, further comprising uploading and classifying the images stored in the memory of the memory module.

29. The non-transitory computer-readable medium of claim 28 wherein the uploaded images from a same class are stored on a plurality of memory modules.

30. The non-transitory computer-readable medium of claim 28 wherein the uploaded images from a same class are stored on the same memory module.

31. The non-transitory computer-readable medium of claim 27 wherein the search of images includes performing a global descriptor computation, a local descriptor computation, and a verification determination.

32. The non-transitory computer-readable medium of claim 27 wherein the comparison of the received results includes ranking the received results.

33. The non-transitory computer-readable medium of claim 32, further comprising constructinga final rank of the results received from multiple memory modules.

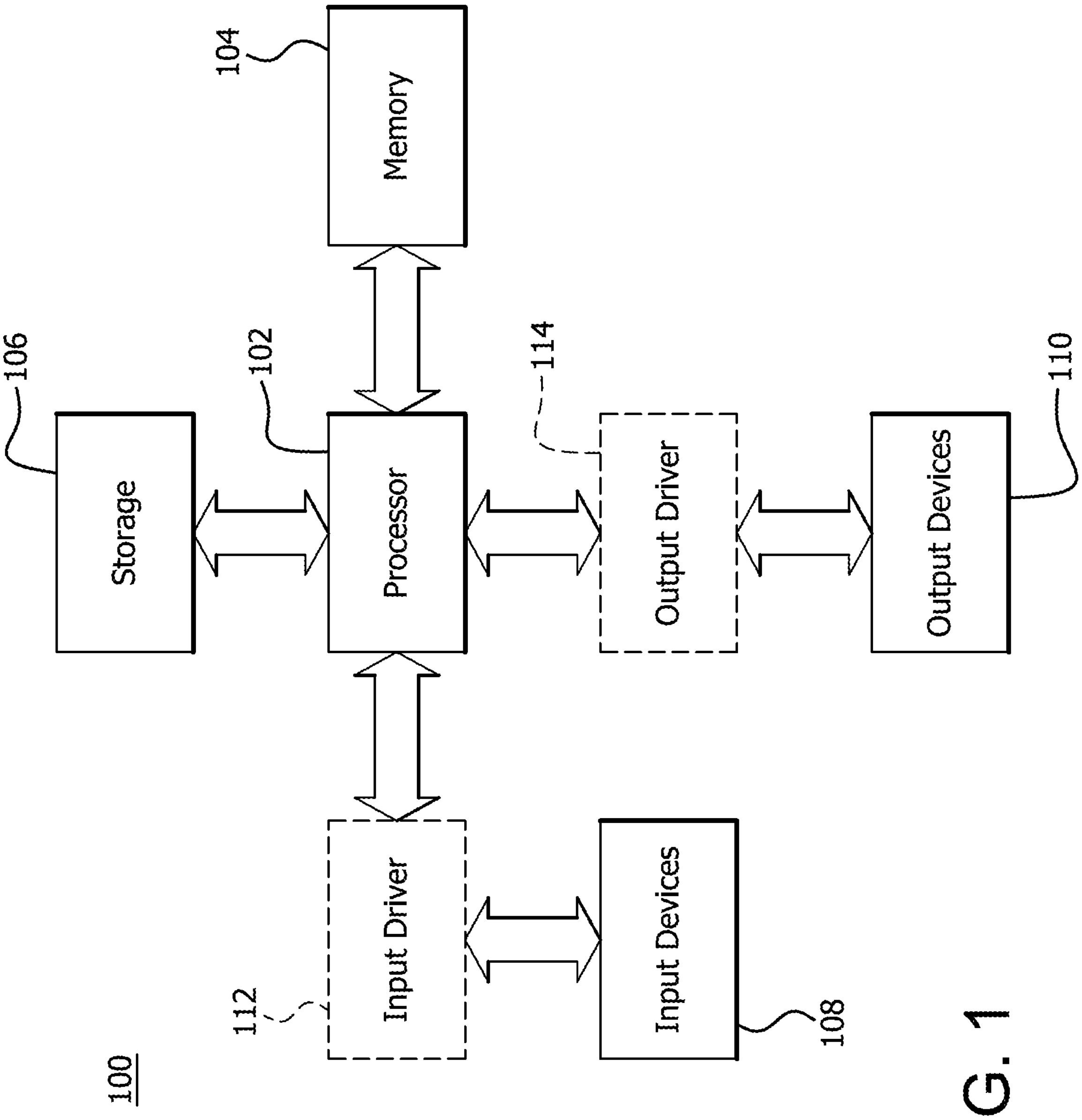
100
106
Storage
102
Processor
104
Memory
114
Output Driver
110
Output Devices
112
Input Driver
108
Input Devices

FIG. 1

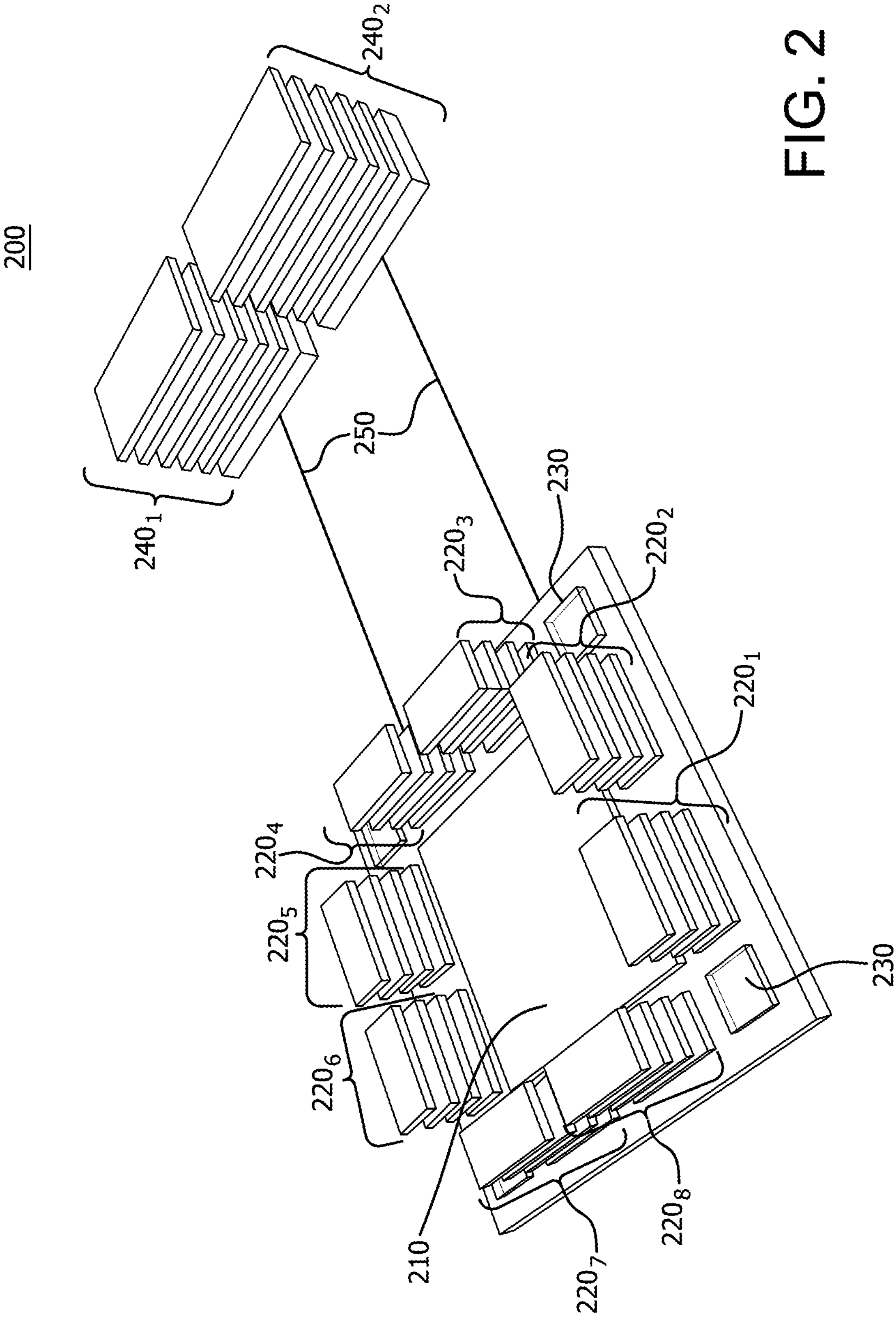
200
240$_2$
240$_1$
250
230
220$_3$
220$_2$
220$_1$
220$_4$
220$_5$
220$_6$
210
220$_7$
220$_8$
230

FIG. 2

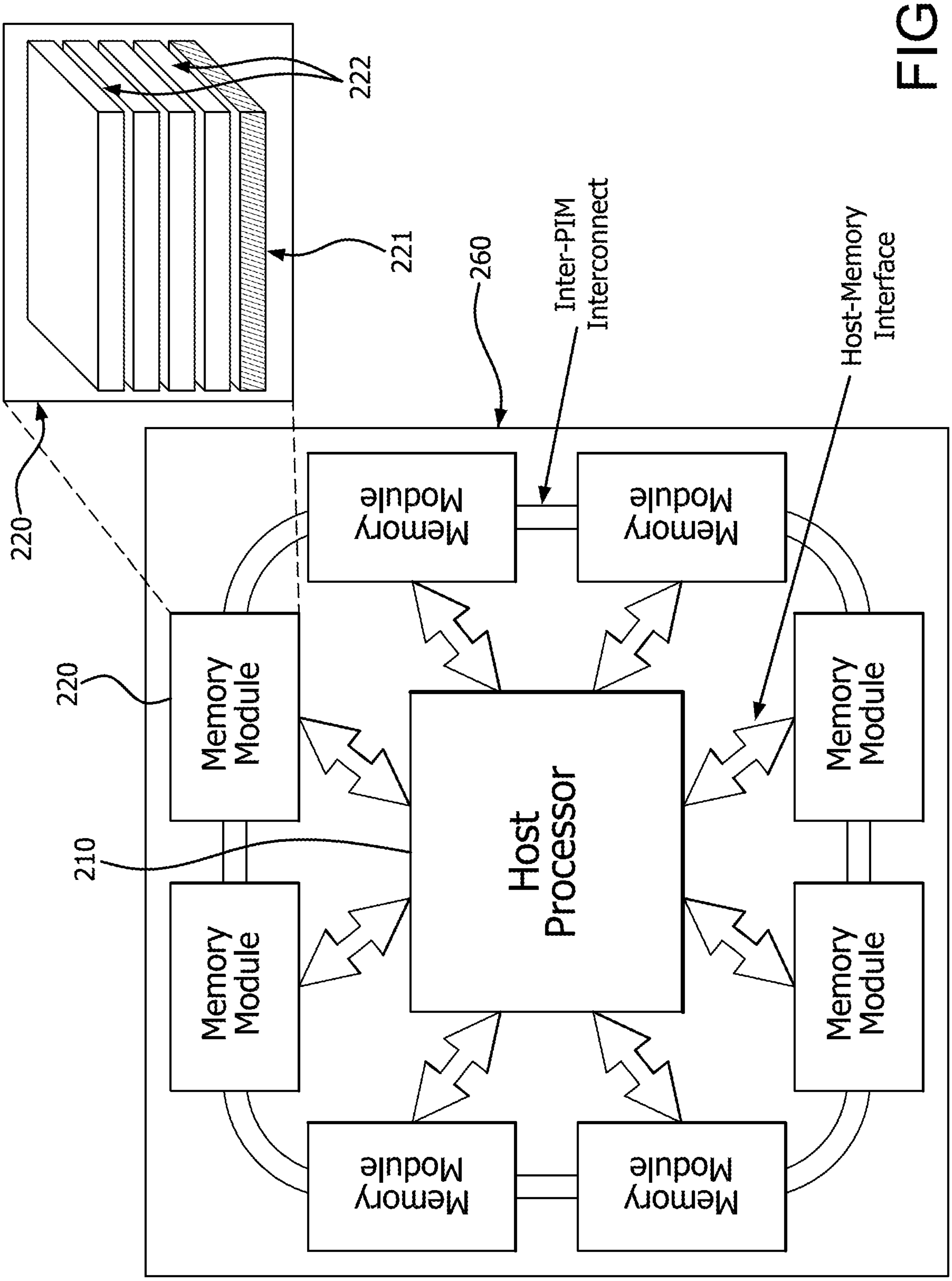
220
222
221
260
Inter-PIM Interconnect
Host-Memory Interface
220
210
Memory Module
Memory Module
Memory Module
Memory Module
Memory Module
Memory Module
Memory Module
Memory Module
Host Processor

FIG. 3

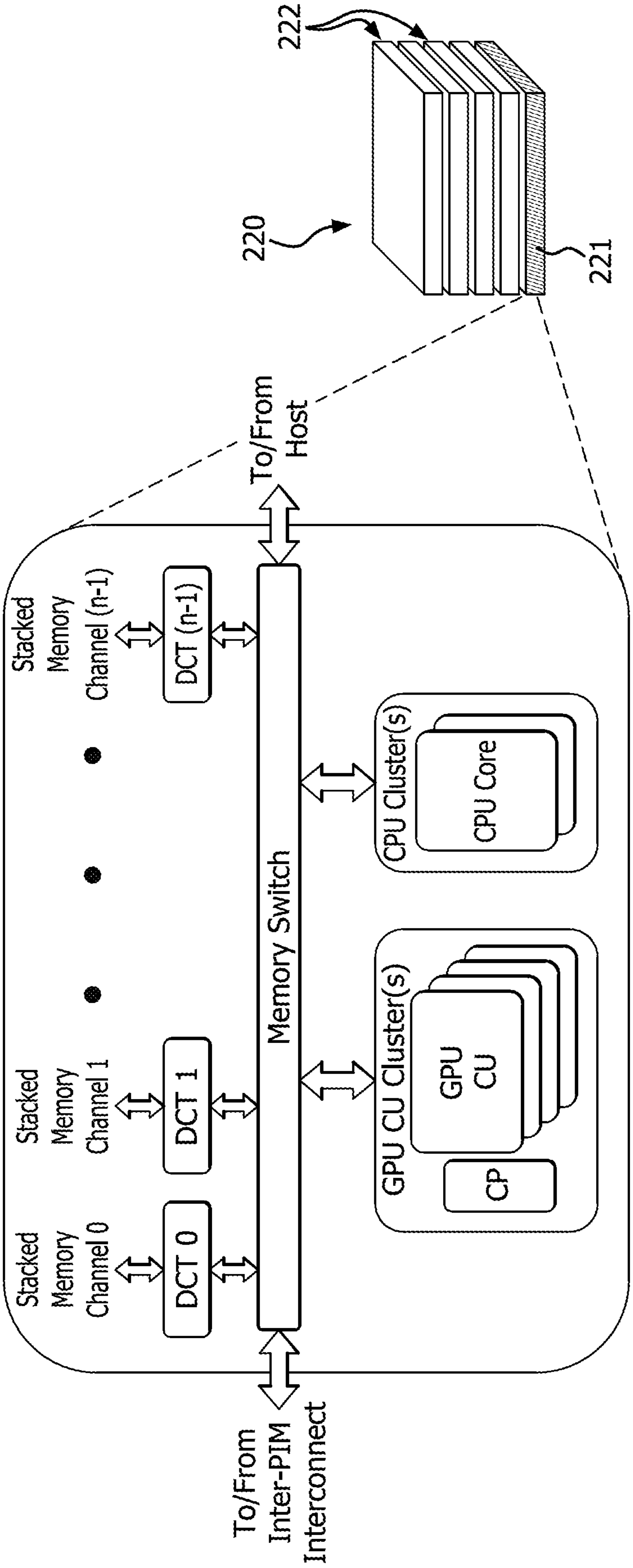
222
220
221
To/From
Host
Stacked
Memory
Channel 0
Stacked
Memory
Channel 1
Stacked
Memory
Channel (n-1)
DCT 0
DCT 1
DCT (n-1)
Memory Switch
GPU CU Cluster(s)
GPU
CU
CP
CPU Cluster(s)
CPU Core
To/From
Inter-PIM
Interconnect

FIG. 4

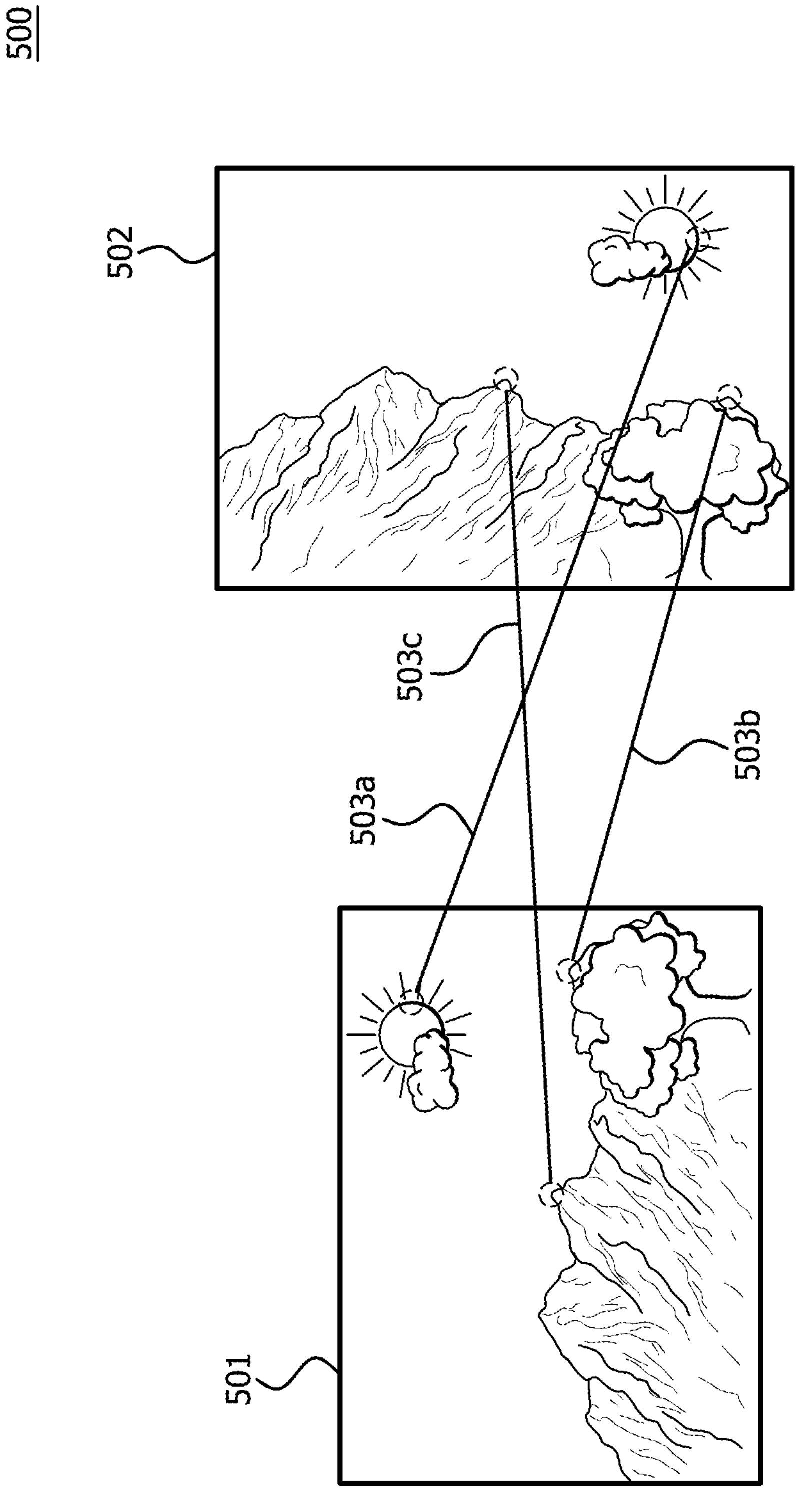

FIG. 5

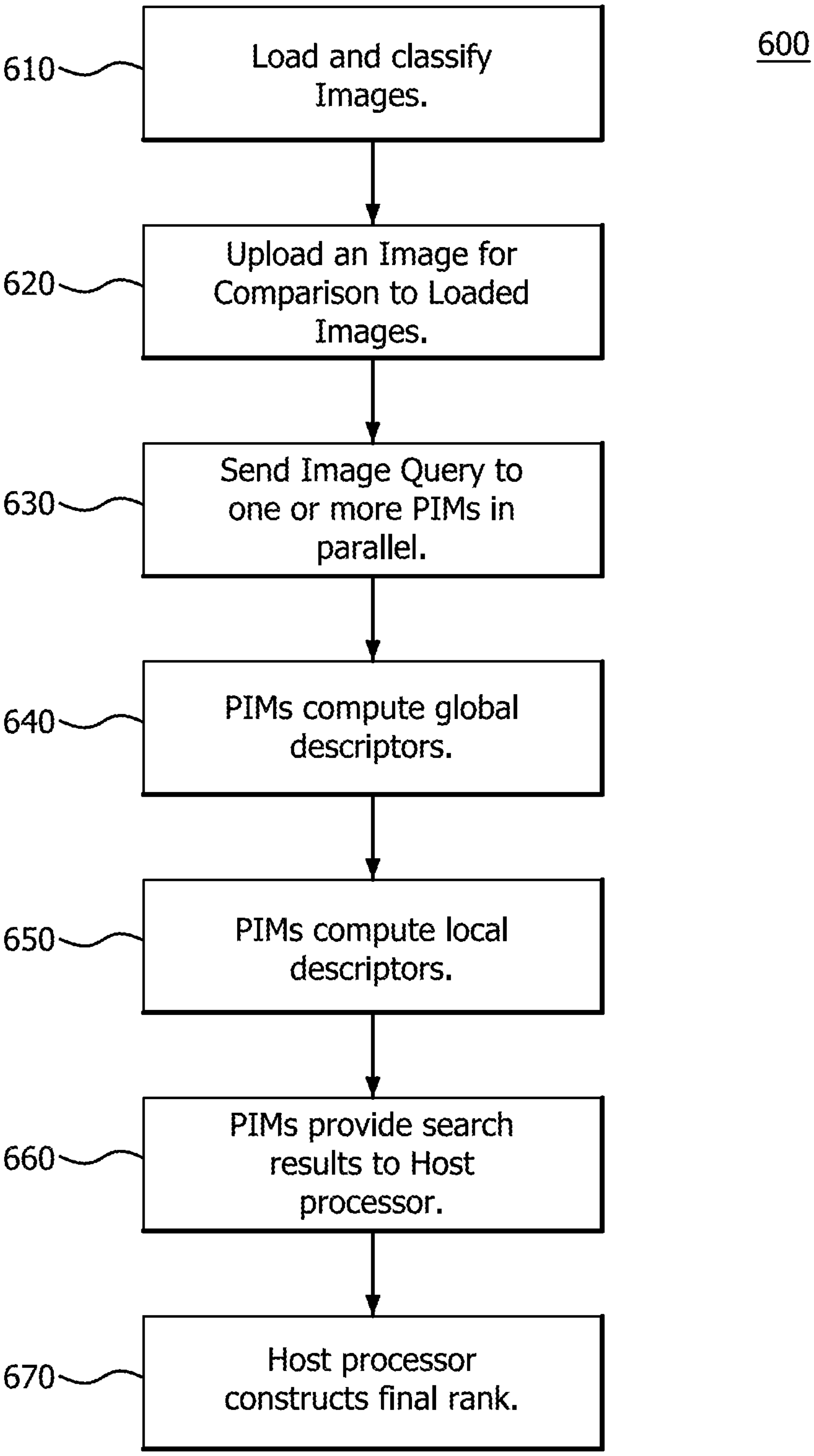
600
610
Load and classify Images.
620
Upload an Image for Comparison to Loaded Images.
630
Send Image Query to one or more PIMs in parallel.
640
PIMs compute global descriptors.
650
PIMs compute local descriptors.
660
PIMs provide search results to Host processor.
670
Host processor constructs final rank.

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2016/052454

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/30(2006.01)i, G06F 13/16(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; H01L 23/535; G06K 9/68; G06F 7/52; G06F 12/00; G06F 9/06; G06K 9/62; G06F 15/76; G06F 7/38; G06F 13/16

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: processor-in-memory (PIM), interposer, image, search, query, compare, and similar terms.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2014-0089609 A1 (ADVANCED MICRO DEVICES, INC.) 27 March 2014 See paragraphs [0019], [0034], and [0043]; and figures 1-3.	1-33
A	US 9,189,498 B1 (GOOGLE INC.) 17 November 2015 See column 14, line 51 - column 15, line 47; and figure 5.	1-33
A	US 2010-0318764 A1 (TERRY D. GREYZCK) 16 December 2010 See paragraphs [0015]-[0023] and figure 1.	1-33
A	US 2011-0085739 A1 (DONG-QING ZHANG et al.) 14 April 2011 See paragraphs [0031]-[0039] and figures 1-2.	1-33
A	WO 03-088033 A1 (UNIVERSITY OF ROCHESTER et al.) 23 October 2003 See page 6, line 1 - page 7, line 24; and figure 1.	1-33

□ 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 28 December 2016 (28.12.2016)	Date of mailing of the international search report **28 December 2016 (28.12.2016)**
Name and mailing address of the ISA/KR International Application Division Korean Intellectual Property Office 189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea Facsimile No. +82-42-481-8578	Authorized officer NHO, Ji Myong Telephone No. +82-42-481-8528

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2016/052454

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2014-0089609 A1	27/03/2014	WO 2014-052138 A1	03/04/2014
US 9189498 B1	17/11/2015	US 8949253 B1	03/02/2015
US 2010-0318764 A1	16/12/2010	US 8583898 B2	12/11/2013
US 2011-0085739 A1	14/04/2011	BR PI0822771 A2	30/06/2015
		CA 2726037 A1	10/12/2009
		CN 102057371 A	11/05/2011
		EP 2300941 A1	30/03/2011
		JP 2011-523137 A	04/08/2011
		JP 5774985 B2	09/09/2015
		KR 10-1582142 B1	05/01/2016
		KR 10-1622360 B1	19/05/2016
		KR 10-2011-0027666 A	16/03/2011
		WO 2009-148422 A1	10/12/2009
WO 03-088033 A1	23/10/2003	AU 2003-221680 A1	27/10/2003
		US 2003-0222879 A1	04/12/2003
		US 7167890 B2	23/01/2007