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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
10 December 2020 (10.12.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/246724 A1

(51) International Patent Classification:
G06N 3/063 (2006.01)

(21) International Application Number:
PCT/KR2020/006411

(22) International Filing Date:
15 May 2020 (15.05.2020)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10-2019-0066396 05 June 2019 (05.06.2019) KR

(71) Applicant: **SAMSUNG ELECTRONICS CO., LTD.** [KR/KR]; 129, Samsung-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do 16677 (KR).

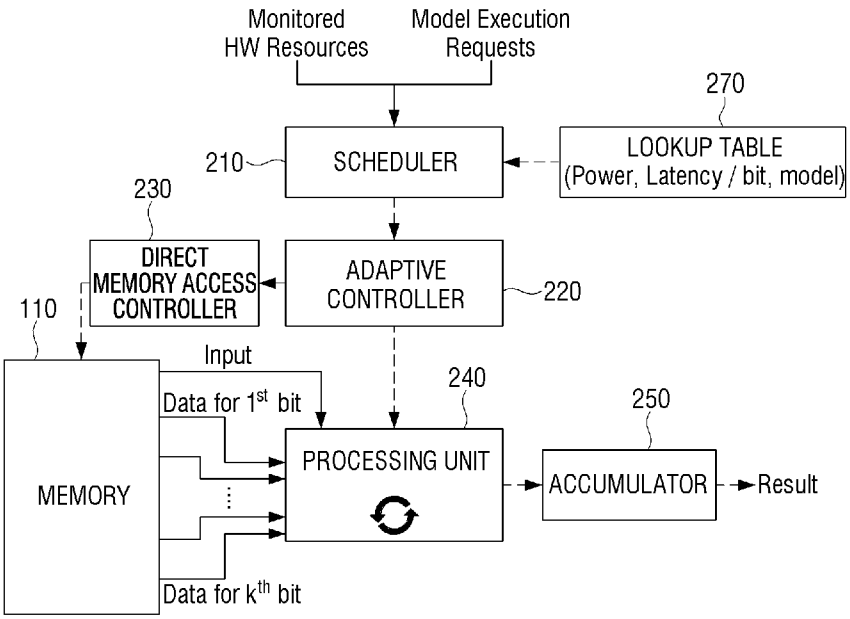
(72) Inventors: **KWON, Sejung**; 129, Samsung-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do 16677 (KR). **LEE, Dong-soo**; 129, Samsung-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do 16677 (KR).

(74) Agent: **KIM, Tae-hun** et al.; 9th Floor, Shinduk Bldg., 343, Gangnam-daero, Seocho-gu, Seoul 06626 (KR).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, 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, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, 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,

(54) Title: ELECTRONIC APPARATUS AND METHOD OF PERFORMING OPERATIONS THEREOF



(57) Abstract: A method for an electronic apparatus to perform an operation of an artificial intelligence model includes acquiring resource information for hardware of the electronic apparatus while a plurality of data used for an operation of a neural network model are stored in a memory, the plurality of data respectively having degrees of importance different from each other; obtaining data to be used for the operation of the neural network model among the plurality of data according to the degrees of importance of each of the plurality of data based on the acquired resource information; and performing the operation of the neural network model by using the obtained data.

WO 2020/246724 A1

[Continued on next page]

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))*

Description

Title of Invention: ELECTRONIC APPARATUS AND METHOD OF PERFORMING OPERATIONS THEREOF

Technical Field

[1] The disclosure relates to an electronic apparatus and a method of performing operations of the electronic apparatus, and more particularly, to a method of performing operations of an artificial neural network model.

Background Art

[2] Recently, a study of implementing an artificial intelligence model (e.g., a deep-learning model) by using hardware is being continued. In the case of implementing an artificial intelligence model by using hardware, the speed of operations of the artificial intelligence model can be greatly improved, and use of various deep-learning models that were previously difficult to use due to the memory sizes or limitation on the response time became possible.

[3] Algorithms for continuously improving the performance of an artificial intelligence model from the viewpoint of implementation of hardware are being suggested, as for example, a data quantization technology which reduces the amount of operation data for reducing operation delays and power consumption.

[4] Data quantization is, for example, a method of reducing the amount of information expressing parameters of a matrix, and may decompose real number data into binary data and scaling factors and express the data as an approximate value. As quantized data cannot reach the precision of the original data, the precision of inference by an artificial intelligence model using quantized data may be lower than the precision of the original inference by an artificial intelligence model. However, considering the limited circumstance of hardware, quantization to a certain extent can save the use amount of a memory or consumption of computing resources, and thus it is being actively studied in the field of on-device artificial intelligence.

Disclosure of Invention

Technical Problem

[5] Embodiments of the disclosure provide an electronic apparatus which reduces the data capacity of an artificial intelligence model, and at the same time, minimizes deterioration of the performance of the artificial intelligence model, and a method of performing an operation of the artificial intelligence model.

Solution to Problem

[6] In accordance with an aspect of the disclosure, a method for an electronic apparatus to perform an operation of an artificial intelligence model includes an operation of

acquiring resource information for hardware of the electronic apparatus while a plurality of data respectively having different degrees of importance from one another, which are used for an operation of a neural network model, are stored in a memory, an operation of obtaining some data to be used for an operation of the neural network model among the plurality of data according to the degrees of importance of each of the plurality of data based on the acquired resource information, and an operation of performing an operation of the neural network model by using the obtained some data.

[7] In accordance with an aspect of the disclosure, an electronic apparatus includes a memory storing a plurality of data respectively having different degrees of importance from one another, and a processor configured to obtain some data to be used for an operation of the neural network model among the plurality of data according to the degrees of importance of each of the plurality of data being stored in the memory based on resource information for the hardware of the electronic apparatus, and perform an operation of the neural network model by using the obtained some data.

Advantageous Effects of Invention

[8] According to embodiments, it is possible to flexibly adjust the amount of data used for a neural network model according to the requirements of hardware. For example, an improved effect can be expected in at least one of the aspect of latency, the aspect of power consumption, or the aspect of a user.

[9] In the aspect of latency, it may be possible to exclude binary data having a low degree of importance and selectively use only binary data having a high degree of importance in an operation of a neural network model in consideration of the time for requesting execution of the neural network model, and thereby satisfy the requirement with minimum reduction of accuracy.

[10] In the aspect of power consumption, in an example when the remaining amount of the battery of an electronic apparatus is determined to be low, the amount of data such that a neural network model operates with the minimum performance in consideration of the condition of hardware (e.g., low charge of the battery) may be controlled, and the operational time of the electronic apparatus thus may be extended.

[11] In the aspect of a user (or a developer), in the related art, it has been difficult for the user to judge the optimal amount of data to be used for a neural network model in consideration of the operation amounts and other restrictions of artificial intelligence applications mounted on an electronic apparatus. However, according to an embodiment, it is possible to automatically adjust the amount of data appropriately in consideration of latency and power consumption based on the condition of hardware, and thus it becomes possible to maintain the accuracy of inference of an artificial intelligence model above a certain degree, and at the same time, effectively execute a neural

network model.

[12] According to an embodiment, the problem that a neural network model is not executed or latency becomes great in a situation where the hardware resources are limited can be overcome. That is, it is possible to flexibly adjust the amount of data for an operation of the neural network model in consideration of the currently available resources of hardware, and, accordingly, even if the amount of data increases, the latency is maintained below a certain level, and a neural network model can operate without stoppage and above a certain threshold of accuracy.

Brief Description of Drawings

[13] FIG. 1 is a block diagram illustrating a configuration of an electronic apparatus according to an embodiment;

[14] FIG. 2 is a block diagram illustrating components for a neural network operation including a processing unit according to an embodiment;

[15] FIG. 3 illustrates an example of a scheduling syntax according to an embodiment;

[16] FIG. 4 is a block diagram illustrating components for a neural network operation including a plurality of processing units according to an embodiment;

[17] FIG. 5 is a diagram illustrating a process by which quantized parameter values are stored in a memory for each bit order according to an embodiment;

[18] FIG. 6 is a diagram illustrating a process by which quantized parameter values are stored in a memory for each bit order according to an embodiment;

[19] FIG. 7 is a flowchart illustrating a method for an electronic apparatus to perform an operation according to an embodiment; and

[20] FIG. 8 is a block diagram illustrating a detailed configuration of an electronic apparatus according to an embodiment.

Best Mode for Carrying out the Invention

[21] -

Mode for the Invention

[22] The embodiments of the present disclosure may be diversely modified. Accordingly, specific embodiments are illustrated in the drawings and are described in detail in the detailed description. However, it is to be understood that the present disclosure is not limited to a specific embodiment, but includes all modifications, equivalents, and substitutions without departing from the scope and spirit of the present disclosure. Also, well-known functions or constructions are not described in detail since they would obscure the disclosure with unnecessary detail.

[23] Hereinafter, various embodiments of the disclosure will be described in detail with reference to the accompanying drawings.

[24] FIG. 1 is a block diagram illustrating a configuration of an electronic apparatus 100

according to an embodiment. As illustrated in FIG. 1, the electronic apparatus 100 includes a memory 110 and a processor 120.

[25] The electronic apparatus 100 may be a server, a desktop PC, a laptop computer, a smartphone, a tablet PC, etc. Alternatively, the electronic apparatus 100 is an apparatus using an artificial intelligence model, and it may be a cleaning robot, a wearable apparatus, a home appliance, a medical apparatus, an Internet of Things (IoT) apparatus, or an autonomous vehicle.

[26] The memory 110 in FIG. 1 may store a plurality of data respectively having different degrees of importance from one another. The plurality of data respectively having different degrees of importance from one another may include, for example, parameter values of a quantized matrix used for an operation of a neural network model. In this case, if there are a plurality of matrices used for an operation of a neural network model, the electronic apparatus 100 may include parameter values respectively having different degrees of importance from one another for each of the plurality of quantized matrices.

[27] As another example, the plurality of data respectively having different degrees of importance from one another may be a plurality of neural network layers respectively having different degrees of importance from one another used for an operation of a neural network model.

[28] As still another example, the plurality of data having different degrees of importance from one another may be parameter values of a matrix before being quantized. In this case, the parameter values of the matrix may consist of binary data respectively having different degrees of importance from one another, and for example, the degrees of importance may increase according to the bit orders.

[29] In the case the plurality of data are parameter values of a quantized matrix, the quantization process of the matrix performed for acquiring the parameter values may be performed by the electronic apparatus 100. Alternatively, the quantization process may be performed at an external apparatus and the parameter values of the quantized matrix may be stored in the memory 110 in advance.

[30] In the case the parameter values of a matrix are quantized, the parameter values of the matrix of a full-precision value may be converted into binary data (or, quantized bits) in k numbers (e.g., +1 and -1) b_i and a scaling coefficient factor a_i values. In the case of performing an operation of a neural network model by using the parameter values of a quantized matrix, the use amount of the memory and the use amount of the computer during inference between neural network layers may decrease, but the accuracy of inference may deteriorate.

[31] Accordingly, various quantization algorithms for enhancing the accuracy of inference may be used.

[32] For example, in the case of quantizing parameter values of a w matrix to have bit numbers in k numbers, various algorithms satisfying the condition of [Formula 1] may be used.

[33]

[34] [Formula 1]

[35]

$$\min_{\{a_i, b_i\}_{i=1}^{k}} \left\| w - \sum_{i=1}^{k} a_i b_i \right\|^2, \quad \text{with} \quad b_i \quad \{-1, +1\}^n$$

[36]

[37] For satisfying the condition of [Formula 1], as an example, an alternating algorithm (e.g., an alternating multi-bit algorithm), etc. may be used. An alternating algorithm is an algorithm which repetitively updates binary data and a coefficient factor and finds a value by which the [Formula 1] becomes minimal. For example, in an alternating algorithm, binary data may be calculated again based on an updated coefficient factor and updated, and a coefficient factor may be calculated again based on updated binary data and updated. This process may be repeated until the error value becomes smaller than or equal to a specific value.

[38] An alternating algorithm may guarantee high accuracy, but may require a large amount of computing resources and operation time for updating binary data and a coefficient factor. In particular, in an alternating algorithm, in the case parameter values are quantized into bits in k numbers, all of the k bits have similar degrees of importance, and thus accurate inference can be possible only when an operation of a neural network model is performed by using all of the bits in k numbers.

[39] In other words, in the case of performing an operation of a neural network model while omitting some bits, the accuracy of the neural network operation may deteriorate. As an example, in an environment where the resources of the electronic apparatus 100 are limited (e.g., an on-device artificial intelligence chip environment), in the case an operation of a neural network model is performed by using only some bits in consideration of the resources of the electronic apparatus 100, the accuracy of the neural network operation may deteriorate.

[40] Accordingly, an algorithm that can flexibly respond toward limited hardware resources may be required. As an example, for quantization of parameter values of a matrix, a greedy algorithm quantizing the parameter values such that each bit of binary data has a different degree of importance from one another may be used.

[41] In the case of quantizing the parameter values of a matrix into bits in k numbers by using a greedy algorithm, the first binary data and the coefficient factor of the bits in k

numbers in [Formula 1] may be calculated by using [Formula 2].

[42] [Formula 2]

[43] $$b^* = \text{sign}(w),\ a^* = \frac{w^T b^*}{n}$$

[44]

[45] Next, the ith bit ($1 < i \leq k$) may repeat the same calculation as in [Formula 3] toward r which is the difference between the original parameter value and the first quantized value. That is, by calculating the ith bit by using the residues that remained after calculating the (i - 1) bit, the parameter values of a quantized matrix having bits in k numbers may be acquired.

[46] [Formula 3]

[47] $$\min_{a_i,\, b_i} \left\| r_{i-1} - a_i b_i \right\|^2,\ \text{where } r_{i-1} = w - \sum_{j=1}^{i-1} a_j b_j,\ 1 < i \leq k$$

[48]

[49] Accordingly, the parameter values of a quantized matrix having bits in k numbers may be acquired.

[50] In addition to the above, for further minimalizing an error between the original parameter value and a quantized parameter value of a matrix, a refined greedy algorithm based on a greedy algorithm may be used. A refined greedy algorithm may update a coefficient factor as in [Formula 4] by using a vector b determined through a greedy algorithm.

[51] [Formula 4]

[52] $$[a_1, \ldots, a_j] = ((B_j^T B_j)^{-1} B_j^T w)^T,\quad \text{with}\quad B_j = [b_1, \ldots, b_j]$$

[53]

[54] In the case of using a greedy algorithm (or, a refined greedy algorithm), as the order of a bit becomes higher, the value of a coefficient factor becomes smaller, and accordingly, the degree of importance of the bit decreases. Thus, even if an operation is omitted for a bit having a high bit order, there may be little influence on inference of a neural network. In a neural network model, a method of applying noises intentionally to about 10% of parameter values may be used for improving the accuracy of inference. In this case, even if an operation is performed while omitting some bits of binary data quantized by a greedy algorithm, it is difficult to be deemed that the accuracy of inference of a neural network deteriorates in general. Rather, a circumstance where the accuracy of inference of a neural network is rather improved may occur.

[55] In the case of quantizing parameter values of a matrix to include bits having different degrees of importance, an adaptive operation of a neural network model in consideration of the requirements (e.g., power consumption, operation time) of given computing resources becomes possible. That is, it becomes possible to adjust the performance of a neural network model according to the degree of importance for each of various neural network models.

[56] Also, without having to make an effort to consider the optimal number of quantized bits for each neural network model, it becomes possible to flexibly adjust the number of quantized bits to be applied to a neural network model according to the condition or limited condition after mounting a matrix quantized to a certain degree. In addition, the cost of development required for finding an optimal condition for execution for each neural network model can be saved.

[57] The processor 120 in FIG. 1 may control the overall operations of the electronic apparatus 100. The processor 120 may be a generic-purpose processor (e.g., a central processing unit (CPU) or an application processor), a graphic-dedicated processor (e.g., a GPU), or a system on chip (SoC) where processing is performed (e.g., an on-device artificial intelligence (AI) chip), a large scale integration (LSI), or a field programmable gate array (FPGA). The processor 120 may include one or more of a CPU, a micro controller unit (MCU), a micro processing unit (MPU), a controller, an application processor (AP) or a communication processor (CP), and an ARM processor, or may be defined by the terms.

[58] While the processor 120 stores a plurality of data having different degrees of importance from one another in the memory 110, the processor 120 may obtain some data to be used for an operation of a neural network model among the plurality of data according to the degrees of importance of each of the plurality of data stored in the memory 110, based on resource information for the hardware of the electronic apparatus 100. As an example, in the case the plurality of data include binary data as parameter values of a quantized matrix, the processor 120 may obtain the number of binary data to be used for an operation of a neural network model among the plurality of data. For example, as the bit orders of binary data increase, the degrees of importance may decrease.

[59] When some data to be used for an operation of a neural network model are obtained, the processor 120 may perform an operation of a neural network model by using the obtained some data. As an example, the processor 120 may perform matrix operations for an input value and each bit of the binary data, and sum up the operation results for each bit and acquire an output value. In the case where a plurality of neural network processing units exist, the processor 120 may perform a matrix parallel operation using a plurality of neural network processing units based on the order of each bit of binary

data.

[60] FIG. 2 is a block diagram illustrating components for a neural network operation including a processing unit according to an embodiment.

[61] The electronic apparatus 100 into which the block diagram of FIG. 2 is included may include, for example, an on-device artificial intelligence chip performing inference of a neural network by using hardware.

[62] The parameter values of a matrix used for inference of a neural network by using hardware may be, for example, in a state of being quantized by using a greedy algorithm, so that important binary data can be selectively used. The degree of importance of the binary data which are the quantized parameter values may decrease as the bit orders increase.

[63] In FIG. 2, the electronic apparatus 100 may include a scheduler 210, an adaptive controller 220, a direct memory access controller (DMAC) 230, a processing unit 240, and an accumulator 250. At least one of the scheduler 210, the adaptive controller 220, the direct memory access controller 230, the processing unit 240, or the accumulator 250 may be implemented as software and/or hardware. For example, the scheduler 210, the adaptive controller 220, the direct memory access controller 230, the processing unit 240, and the accumulator 250 may be the function blocks of the processor 120. As another example, the scheduler 210, the adaptive controller 220, and the direct memory access controller 230 are the function blocks of the first processor which is the sub-processor of the processor 120, and the processing unit 240 and the accumulator 250 may be the function blocks of the second processor which is another sub-processor of the processor 120. The first processor is a processor which is in charge of control of the second processor, and the second processor is a processor optimized for operations, and for example, it may be an artificial intelligence processor or a graphics processor.

[64] The scheduler 210 may receive resource information related to hardware, and instructions requesting execution of a neural network model.

[65] In response to an instruction requesting execution of a neural network model, the scheduler 210 may determine the number of quantized bits to be used for operations of each neural network model with reference to a lookup table 270. A lookup table may be stored in a read only memory (ROM) or a random access memory (RAM) area of the processor 120, or stored in the memory 110 outside the processor 120.

[66] A lookup table may store, for example, scheduling modes in k numbers. In this case, the number of quantized bits to be used for operations of neural network models may be defined in advance for each scheduling mode. For example, the number of quantized bits may be defined differently according to the degrees of importance of neural network models.

[67] The scheduler 210 may first determine hardware conditions for performing an inference job according to a request instruction with a degree of accuracy above a certain level. For example, as hardware conditions, the scheduler 210 may determine the total number of operations of the processing unit 240 for performing an inference job, power consumption which is power for an operation process of a neural network model, and latency which is the time until acquiring an output value which is the operation time of a neural network model. The scheduler 210 may compare the hardware resource information (e.g., power consumption per time, latency) which is currently available and hardware conditions for performing the inference job, and determine the number of quantized bits for each neural network model.

[68] The process where the scheduler 210 determines the number of quantized bits for each neural network model will be described in more detail through the scheduling syntax in FIG. 3.

[69] The adaptive controller 220 may control the operation orders of neural network models, or may perform control such that operations are performed while each neural network model has a different amount of bits from one another.

[70] For example, the adaptive controller 220 may control the processing unit 240 and the direct memory access controller 230 in consideration of the number of quantized bits for each neural network model acquired from the scheduler 210. Alternatively, the adaptive controller 220 may acquire resource information related to hardware resources from the scheduler 210, and determine the number of quantized bits to be used for operations of neural network models in consideration of this.

[71] The adaptive controller 220 may control the processing unit 240 and the direct memory access controller 230, and thereby perform control such that quantized bits after a specific number are not used for operations of neural network models.

[72] The direct memory access controller 230 may perform control such that an input value and quantized parameter values stored in the memory 110 are provided to the processing unit 240 by control of the adaptive controller 220. In the memory 110, quantized parameter values may be aligned according to the orders of bits and stored. For example, quantized parameter values of the first bit may be aligned as one data format and stored, and quantized parameter values of the second bit may be aligned as one data format and stored, and in succession, quantized parameter values of the Nth bit may be aligned as one data format and stored. In this case, the direct memory access controller 230 may perform control such that the quantized parameter values of the first bit to the quantized parameter values of the Nth bit stored in the memory 110 are sequentially provided to the processing unit 240 by control of the adaptive controller 220. Alternatively, the direct memory access controller 230 may perform control such that the quantized parameter values of the first bit to the Kth bit ($N < K$)

are sequentially provided to the processing unit 240.

[73] The processing unit 240 may perform matrix operations by using the input value and the quantized parameter values received from the memory 110 and acquire operation results for each bit order, and the accumulator 250 may sum up the operation results for each bit order and acquire an output result (or, an output value).

[74] As an example, in the case parameter values are quantized into the N bit, the processing unit 240 may be called N times for multiplication operations of the matrix, and operations may be performed sequentially for the first bit to the Nth bit.

[75] FIG. 3 illustrates an example of a scheduling syntax performed at the scheduler 210 according to an embodiment.

[76] In FIG. 3, the definition part 310 of the scheduling syntax may define in advance an 'Execution Model' having neural network models (e.g., a voice recognition model, an image recognition model, etc.) which are the subjects of execution as values, 'Constraints' having information related to hardware resources (e.g., power consumption, latency) as a value, a 'mode' having scheduling modes as values, a 'max_cost' which is the maximum operation cost in consideration of hardware resources acquired from a lookup table, and a 'cost' which is an operation cost acquired from a lookup table with respect to neural network models.

[77] The scheduling modes may be, for example, included in a lookup table and define the optimal number of quantized bits to be used for operations of each neural network model. As an example, in FIG. 3, 16 scheduling modes are defined, and it may be defined that, in the case of the 0 mode, all neural network models are executed with full-precision, and in the case of the 15th mode, all neural network models are calculated by using only 1 bit, and in the 2nd mode, for example, a voice recognition model uses 3 bits as quantized bit data, and an image recognition model uses 2 bits as quantized bit data.

[78] In FIG. 3, the while conditional sentence 320 may compare the operation costs of neural network models according to the current scheduling mode and the maximum operation cost in consideration of hardware resources.

[79] As a result of comparison, when optimal scheduling modes for each neural network model are determined in consideration of hardware resources, the scheduler 210 may acquire the optimal number of quantized bits to be used for each neural network model under the current hardware condition as the return value 330 of the scheduling syntax from the lookup table and provide it to the adaptive controller 220.

[80] FIG. 4 is a block diagram illustrating components for a neural network operation including a plurality of processing units according to an embodiment.

[81] In FIG. 4, in the case there is a spare space in the operation area of the processor 120, a plurality of processing units 241 to 244, e.g., a first processing unit 241, a second

processing unit 242, a third processing unit 243, and a fourth processing unit 244, may be provided in the processor 120. In this case, it may be possible to perform parallel operations by using the plurality of processing units 241 to 244.

[82] The scheduler 210, the adaptive controller 220, the direct memory access controller 230, and the accumulator 250 are described above with reference to FIG. 2, and thus overlapping descriptions will be omitted.

[83] In FIG. 4, the plurality of processing units 241 to 244 may perform matrix parallel operations based on bit orders. For example, the first processing unit 241 may perform an operation with respect to an input value and the first quantized bit, the second processing unit 242 may perform an operation with respect to an input value and the second quantized bit, the third processing unit 243 may perform an operation with respect to an input value and the third quantized bit, and the fourth processing unit 244 may perform an operation with respect to an input value and the fourth quantized bit. The adder 260 may collect operation results of the plurality of processing units 241 to 244 and transmit them to the accumulator 250.

[84] In the case of using the plurality of processing units 241 to 244, the adaptive controller 220 may control each of the plurality of processing units 241 to 244. The direct memory access controller 230 may control the memory 110 such that the quantized parameter values are input while being distinguished according to the bit orders in consideration of the bit orders that each of the plurality of processing units 241 to 244 processes. In particular, in a circumstance where the plurality of processing units 241 to 244 are used, the scheduler 210 may determine a scheduling mode in further consideration of the power consumption of the processor according to operation of the plurality of processing units 241 to 244 and the latency of a neural network operation.

[85] In the case of performing an operation of a neural network model by using the plurality of processing units 241 to 244, in the memory 110, quantized parameter values may be realigned for each bit order and stored.

[86] FIG. 5 is a diagram illustrating a process where quantized parameter values are stored in the memory 110 for each bit order according to an embodiment.

[87] For example, in FIG. 5, 32 bit parameter values in a real number type may exist as parameter values of a neural network model. In this case, if the parameter values are quantized into 3 bits, the parameter values in a real number type may be expressed as coefficient factors 521 and binary data of 3 bits 522, 523, 524 (numeral 520 in FIG. 5).

[88] For effective operation of the plurality of processing units 241 to 244, the quantized parameter values may be realigned. As shown by reference numeral 530 in FIG. 5, the quantized parameter values may be realigned according to the bit orders. For example, with respect to each of the first processing unit 241, the second processing unit 242,

and the third processing unit 243 in FIG. 4, the quantized parameter values may be realigned as shown by reference numerals 531, 532, and 533 in FIG. 5 and stored in the memory 110.

[89] FIG. 6 illustrates a state where quantized parameter values are stored in the memory 110. In an embodiment, the memory 110 may include a DRAM 600. The quantized parameter values may be aligned according to the bit orders and stored in the DRAM 600. In this case, 32 binary data may be included in 32 bit-word values.

[90] As described above, quantized parameter values are stored in the memory 110 in word units to correspond to the operations of each of the plurality of processing units 241 to 244, and accordingly, quantized parameter values for a neural network operation may be effectively read from the memory 110 and transmitted to each of the plurality of processing units 241 to 244.

[91] FIG. 7 is a flowchart illustrating a method for the electronic apparatus 100 to perform an operation according to an embodiment.

[92] A plurality of data having different degrees of importance from one another which are used for an operation of a neural network model may have been stored in the memory. The plurality of data having different degrees of importance from one another may include parameter values of a quantized matrix used for an operation of a neural network model. The parameter values of the quantized matrix may include binary data having different degrees of importance from one another. For example, as the bit orders of the binary data increase, the degrees of importance of the binary data may decrease. The parameter values of the quantized matrix may include parameter values of the matrix quantized by using a greedy algorithm.

[93] In operation 701 of FIG. 7, while the plurality of data having different degrees of importance from one another are stored in the memory, the electronic apparatus 100 may acquire resource information related to hardware. The resource information related to hardware may include, for example, at least one of the power consumption of the electronic apparatus 100, the number, type, and/or specifications of the processing units performing an operation of a neural network model, or the latency of a neural network model.

[94] In operation 703 of FIG. 7, based on the acquired resource information, the electronic apparatus 100 may obtain some data to be used for an operation of a neural network model among the plurality of data according to the degrees of importance of each of the plurality of data. For example, the electronic apparatus 100 may obtain some data to be used for an operation of a neural network model with reference to a lookup table where a plurality of scheduling modes are defined. The electronic apparatus 100 may obtain the number of binary data used for a neural network model among the plurality of data.

[95] In operation 705 of FIG. 7, the electronic apparatus 100 may perform an operation of a neural network model by using the obtained some data. For example, the electronic apparatus 100 may perform matrix operations for an input value and each bit of binary data and sum up the operation results for each bit and acquire an output value. Alternatively, in the case a plurality of neural network processing units exist, the electronic apparatus 100 may perform a matrix parallel operation using the plurality of neural network processing units.

[96] According to embodiments of the disclosure, a plurality of neural network models may be provided on the electronic apparatus 100. The plurality of neural network models may be, for example, implemented as at least one on-device chip and provided on the electronic apparatus 100, or may be stored in the memory 110 of the electronic apparatus 100 as software. For adaptive operation in consideration of limited hardware resources, the electronic apparatus 100 may acquire resource information related to hardware, and determine at least one neural network model to be operated among the plurality of neural network models based on the acquired resource information. For example, the electronic apparatus 100 may determine at least one neural network model according to the priorities in consideration of the accuracy of an inference or the operation speed of a neural network model.

[97] According to embodiments of the disclosure, the electronic apparatus 100 may download at least one neural network model to be operated among the plurality of neural network models from an external apparatus. For example, for adaptive operation in consideration of limited hardware resources, the electronic apparatus 100 may acquire resource information related to hardware of the electronic apparatus 100, and transmit the acquired resource information to an external apparatus. When the external apparatus transmits at least one neural network model based on the acquired resource information to the electronic apparatus 100, the electronic apparatus 100 may store the received neural network model in the memory 110 and use it when performing an inference function. In this case, the electronic apparatus 100 may be provided with minimum neural network models for inference, and thus it is possible to reduce consumption of internal resources of the electronic apparatus 100 or consumption of network resources for communication with a server, and provide a fast result for a request for inference.

[98] FIG. 8 is a block diagram illustrating a detailed configuration of the electronic apparatus 100 according to an embodiment.

[99] According to FIG. 8, the electronic apparatus 100 includes a memory 110, a processor 120, a communicator 130, a user interface 140, a display 150, an audio processor 160, and a video processor 170. Among the components illustrated in FIG. 8, regarding the parts overlapping with the components illustrated in FIG. 1, detailed de-

scriptions will be omitted.

[100] The processor 120 controls the overall operations of the electronic apparatus 100 by using various kinds of programs stored in the memory 110.

[101] Specifically, the processor 120 includes a RAM 121, a ROM 122, a main CPU 123, a graphics processor 124, first to nth interfaces 125-1 to 125-n, and a bus 126.

[102] The RAM 121, the ROM 122, the main CPU 123, the graphics processor 124, and the first to nth interfaces 125-1 to 125-n may be connected with one another through the bus 126.

[103] The first to nth interfaces 125-1 to 125-n are connected with the various kinds of components described above. One of the interfaces may be a network interface connected with an external apparatus through a network.

[104] The main CPU 123 accesses the memory 110, and performs booting by using the operating system (OS) stored in the memory 110. Then, the main CPU 123 performs various operations by using various kinds of programs, etc. stored in the memory 110.

[105] The ROM 122 stores a set of instructions, etc. for system booting. When a turn-on instruction is input and power is supplied, the main CPU 123 copies the OS stored in the memory 110 in the RAM 121 according to the instructions stored in the ROM 122, and boots the system by executing the OS. When booting is completed, the main CPU 123 copies the various kinds of application programs stored in the memory 110 in the RAM 121, and performs various kinds of operations by executing the application programs copied in the RAM 121.

[106] The graphics processor 124 generates a screen including various objects such as icons, images, and texts by using an operation part and a rendering part. The operation part operates attribute values such as coordinate values, shapes, sizes, and colors by which each object will be displayed according to the layout of the screen based on received control instructions. The rendering part generates screens in various layouts including objects, based on the attribute values operated at the operation part. The screens generated at the rending part are displayed in a display area of the display 150.

[107] The above-described operations of the processor 120 may be performed by the programs stored in the memory 110.

[108] The memory 110 is provided separately from the processor 120, and may be implemented as a hard disk, a non-volatile memory, a volatile memory, etc.

[109] The memory 110 may store a plurality of data used for operations of neural network models. The plurality of data may include, for example, parameter values of a quantized matrix.

[110] According to an embodiment, the memory 110 may include at least one of an OS software module for operating the electronic apparatus 100, an artificial intelligence model, a quantized artificial intelligence model, or a quantization module for

quantizing an artificial intelligence model (e.g., a greedy algorithm module).

[111] The communicator 130 is a component performing communication with various types of external apparatuses according to various types of communication methods. The communicator 130 includes a Wi-Fi chip 131, a Bluetooth chip 132, a wireless communication chip 133, a near field communication (NFC) chip 134, etc. The processor 120 performs communication with various kinds of external apparatuses by using the communicator 130.

[112] The Wi-Fi chip 131 and the Bluetooth chip 132 perform communication by using a Wi-Fi method and a Bluetooth method, respectively. In the case of using the Wi-Fi chip 131 or the Bluetooth chip 132, various types of connection information such as a service set identifier (SSID) or a session key is transmitted and received first, and connection of communication is performed by using the information, and various types of information can be transmitted and received thereafter. The wireless communication chip 133 refers to a chip performing communication according to various communication standards such as IEEE, ZigBee, 3rd generation (3G), 3rd generation partnership project (3GPP), and long term evolution (LTE). The NFC chip 134 refers to a chip that operates in an NFC method using a 13.56 MHz band among various RF-ID frequency bands such as 135 kHz, 13.56 MHz, 433 MHz, 860~960 MHz, and 2.45 GHz.

[113] The processor 120 may receive parameter values of at least one of an artificial intelligence module, a matrix included in an artificial intelligence model, or a quantized matrix from an external apparatus through the communicator 130, and store the received data in the memory 110. Alternatively, the processor 120 may directly train an artificial intelligence model through an artificial intelligence algorithm, and store the trained artificial intelligence model in the memory 110. The artificial intelligence model may include at least one matrix.

[114] The user interface 140 receives various user interactions. The user interface 140 may be implemented in various forms according to implementation examples of the electronic apparatus 100. For example, the user interface 140 may be a button provided on the electronic apparatus 100, a microphone receiving user voices, a camera detecting user motions, etc. When the electronic apparatus 100 is implemented as a touch-based electronic apparatus, the user interface 140 may be implemented as a touch screen constituting an inter-layer structure with a touch pad. In this case, the user interface 140 may be used as the display 150.

[115] The audio processor 160 is a component performing processing of audio data. At the audio processor 160, various types of processing such as decoding or amplification, noise filtering, etc. of audio data may be performed.

[116] The video processor 170 is a component performing processing of video data. At the

video processor 170, various types of image processing such as decoding, scaling, noise filtering, frame rate conversion, and resolution conversion of video data may be performed.

[117] Through the method as described above, the processor 120 may quantize a matrix included in an artificial intelligence model.

[118] Embodiments of the disclosure may be implemented as software (e.g., the program) including one or more instructions stored in a machine-readable (e.g., a computer-readable) storage medium (e.g., an internal memory) or an external memory, that can be read by machines (e.g., computers). In an embodiment, the machine (e.g., the processor of the electronic apparatus 100) may load one or more instructions stored in a storage medium, and can operate according to the instructions. When an instruction is executed by a processor, the processor may perform a function corresponding to the instruction itself, or may use other components under its control. An instruction may include a code generated or executed by a compiler or an interpreter. A storage medium that is readable by machines may be a non-transitory storage medium. The term 'non-transitory' means that a storage medium does not include signals, and is tangible, but does not indicate whether data is stored in the storage medium semi-permanently or temporarily.

[119] The method according to embodiments may be provided while being stored as a computer program product. A computer program product refers to a product, and it can be traded between a seller and a buyer. A computer program product can be distributed on-line as a storage medium that is readable by machines (e.g., a compact disc ROM (CD-ROM)), or through an application store (e.g., play store TM). In the case of on-line distribution, at least a portion of a computer program product may be stored in a storage medium such as the server of the manufacturer, the server of the application store, and the memory of the relay server at least temporarily, or may be generated temporarily.

[120] Embodiments described above may be implemented in a recording medium that can be read by a computer or an apparatus similar to a computer, by using software, hardware, or a combination thereof. In some cases, the embodiments described above may be implemented as a processor itself. According to implementation by software, the embodiments such as processes and functions described above may be implemented as separate software modules. Each of the software modules can perform one or more functions and operations described in this specification.

[121] Computer instructions for performing processing operations of machines according to embodiments may be stored in a non-transitory computer-readable medium. Computer instructions stored in such a non-transitory computer-readable medium make the processing operations at machines according to embodiments performed by a

specific machine, when the instructions are executed by the processor of the specific machine. A non-transitory computer-readable medium refers to a medium that stores data semi-permanently, and is readable by machines, but not a medium that stores data for a short moment such as a register, a cache, and a memory. As specific examples of a non-transitory computer-readable medium, there may be a CD, a DVD, a hard disk, a blue-ray disk, a USB, a memory card, a ROM, and the like.

[122] In addition, each of the components according to embodiments (e.g., a module or a program) may consist of a singular object or a plurality of objects. Also, among the above-described components, some components may be omitted, or other components may be further included in the embodiments. Some components (e.g., a module or a program) may be integrated as an object, and perform the functions that were performed by each of the components before integration identically or in a similar manner. Operations performed by a module, a program, or other components according to embodiments may be executed sequentially, in parallel, repetitively, or heuristically. At least some of the operations may be executed in a different order or omitted, or other operations may be added.

[123] While embodiments of the disclosure have been particularly shown and described with reference to the drawings, the embodiments are provided for the purposes of illustration and it will be understood by one of ordinary skill in the art that various modifications and equivalent other embodiments may be made from the disclosure. Accordingly, the true technical scope of the disclosure is defined by the technical spirit of the appended claims.

Claims

[Claim 1] A method for an electronic apparatus to perform an operation of an artificial intelligence model, the method comprising:
obtaining resource information for hardware of the electronic apparatus while a plurality of data used for an operation of a neural network model are stored in a memory, the plurality of data respectively having degrees of importance different from each other;
obtaining data to be used for the operation of the neural network model among the plurality of data according to the degrees of importance of each of the plurality of data based on the acquired resource information; and
performing the operation of the neural network model by using the obtained data.

[Claim 2] The method of claim 1, wherein the plurality of data include parameter values of a quantized matrix used for the operation of the neural network model.

[Claim 3] The method of claim 2, wherein the parameter values of the quantized matrix include binary data respectively having degrees of importance different from each other.

[Claim 4] The method of claim 3, wherein, in the binary data, the degrees of importance of the binary data decrease as bit orders of the binary data increase, respectively.

[Claim 5] The method of claim 3, wherein the performing the operation of the neural network model further comprises:
performing matrix operations for an input value and each bit of the binary data;
summing up results of the operation for each bit; and
acquiring an output value.

[Claim 6] The method of claim 3, wherein the performing the operation of the neural network model further comprises performing a matrix parallel operation using a plurality of neural network processing units based on an order of each bit of the binary data, respectively.

[Claim 7] The method of claim 2, wherein the parameter values of the quantized matrix include the parameter values of the matrix quantized by using a greedy algorithm.

[Claim 8] The method of claim 1, wherein the obtaining the data further comprises obtaining the data to be used for the operation of the neural

network model by referring to a lookup table where a plurality of scheduling modes are defined.

[Claim 9] The method of claim 1, wherein the obtaining the data further comprises obtaining a number of binary data to be used for the operation of the neural network model among the plurality of data that is less than a number of the plurality of data.

[Claim 10] The method of claim 1, wherein the resource information for the hardware includes at least one of a power consumption of the electronic apparatus, a number of processing units performing the operation of the neural network model, or a predetermined latency of the neural network model.

[Claim 11] An electronic apparatus comprising:
a memory storing a plurality of data respectively having degrees of importance different from each other; and
a processor configured to:
obtain data to be used for an operation of a neural network model among the plurality of data according to the degrees of importance of each of the plurality of data stored in the memory based on resource information for hardware of the electronic apparatus, and
perform the operation of the neural network model by using the obtained data.

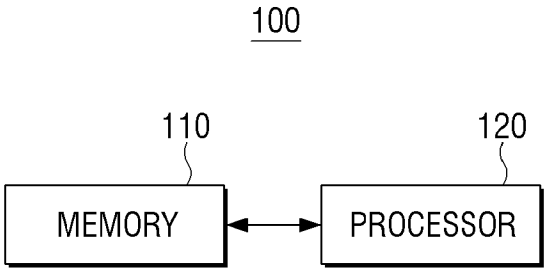
[Claim 12] The electronic apparatus of claim 11, wherein the plurality of data include parameter values of a quantized matrix used for the operation of the neural network model.

[Claim 13] The electronic apparatus of claim 12, wherein the parameter values of the quantized matrix include binary data respectively having degrees of importance different from each other.

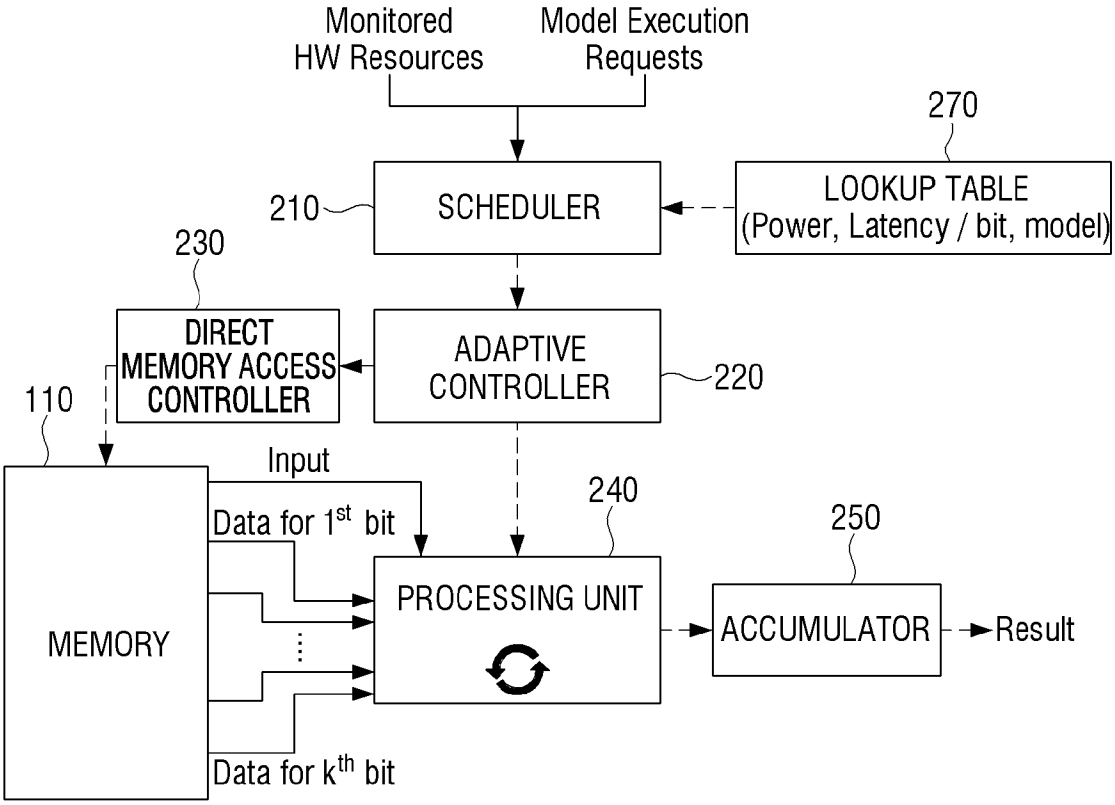
[Claim 14] The electronic apparatus of claim 13, wherein, in the binary data, the degrees of importance of the binary data decrease as bit orders of the binary data increase, respectively.

[Claim 15] The electronic apparatus of claim 13, wherein the processor is further configured to:
perform matrix operations for an input value and each bit of the binary data,
sum up results of the operation for each bit, and
acquire an output value.

[Fig. 1]


100
110
120
MEMORY
PROCESSOR

[Fig. 2]


Monitored HW Resources
Model Execution Requests
270
210
SCHEDULER
LOOKUP TABLE (Power, Latency / bit, model)
230
DIRECT MEMORY ACCESS CONTROLLER
ADAPTIVE CONTROLLER
220
110
Input
Data for 1st bit
240
250
PROCESSING UNIT
ACCUMULATOR
Result
MEMORY
Data for kth bit

[Fig. 3]

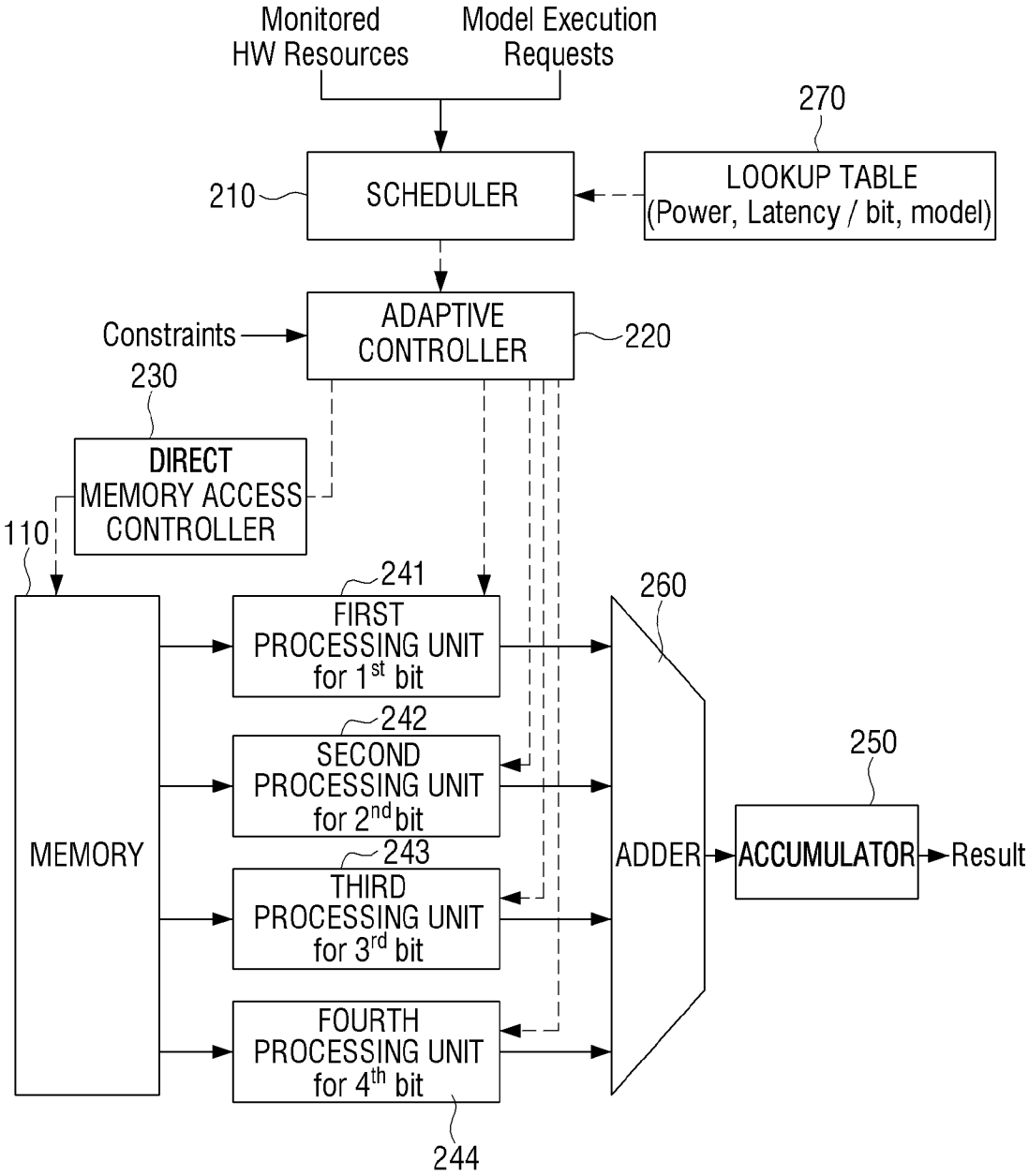
```
ExecutionModel = {Model1, Model2, ···, Modeli}
Constraints = {(Power, Ph), (Latency, Lh), ···}
mode = current_mode #Assume that 'mode' is 0~15
max_cost = LUT.GetMaxCost (Constraints
costs = LUT.GetCost(ExecutionModel)

while True:
        if cost[mode] > max_cost:
                mode += 1
        else if cost[mode - 1] < max_cost:
                mode -= 1
        else:
                break

return LUT.GetOrder(ExecutionModel, mode)
```

310

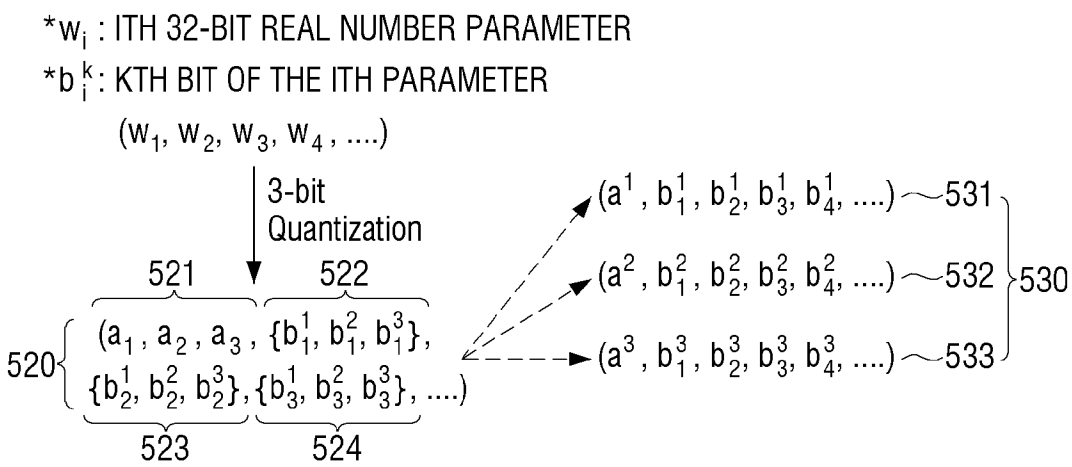
320

330

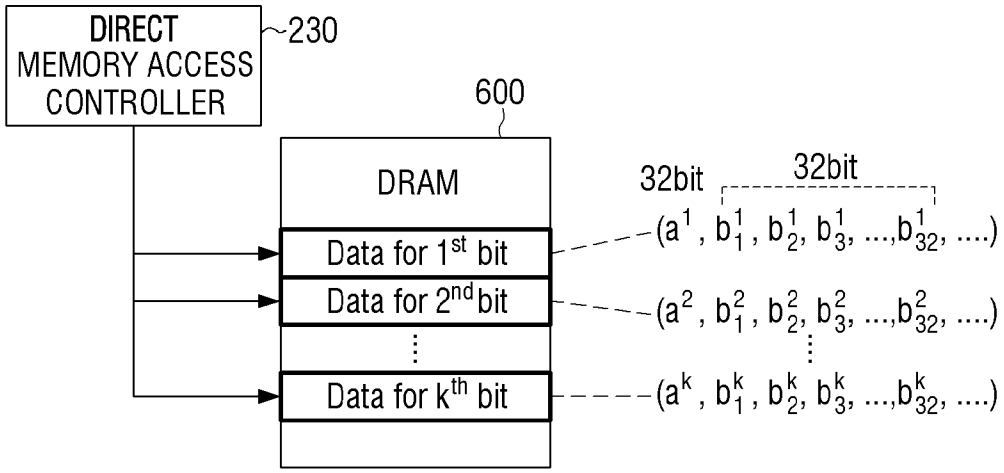
[Fig. 4]


Monitored HW Resources
Model Execution Requests
270
210
SCHEDULER
LOOKUP TABLE (Power, Latency / bit, model)
Constraints
ADAPTIVE CONTROLLER
220
230
DIRECT MEMORY ACCESS CONTROLLER
110
241
260
FIRST PROCESSING UNIT for 1st bit
242
SECOND PROCESSING UNIT for 2nd bit
250
MEMORY
243
THIRD PROCESSING UNIT for 3rd bit
ADDER
ACCUMULATOR
Result
FOURTH PROCESSING UNIT for 4th bit
244

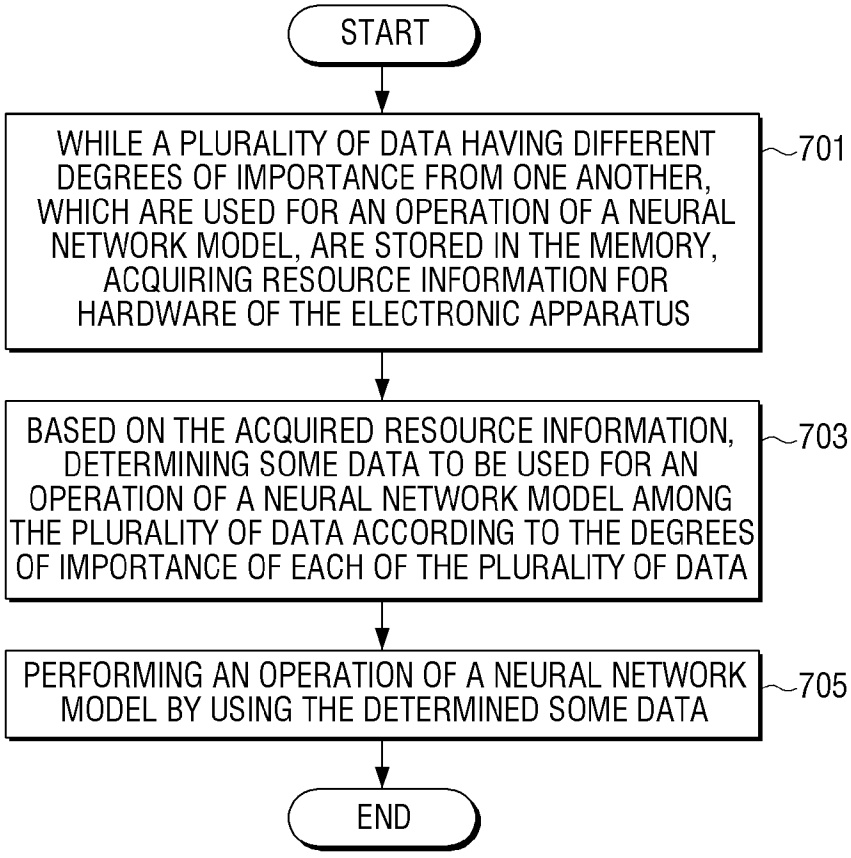
[Fig. 5]


*w_i : ITH 32-BIT REAL NUMBER PARAMETER
*b_i^k : KTH BIT OF THE ITH PARAMETER
$(w_1, w_2, w_3, w_4, ...)$
3-bit Quantization
521
522
520
$(a_1, a_2, a_3, \{b_1^1, b_1^2, b_1^3\},$
$\{b_2^1, b_2^2, b_2^3\}, \{b_3^1, b_3^2, b_3^3\}, ...)$
523
524
$(a^1, b_1^1, b_2^1, b_3^1, b_4^1, ...)$ 531
$(a^2, b_1^2, b_2^2, b_3^2, b_4^2, ...)$ 532
$(a^3, b_1^3, b_2^3, b_3^3, b_4^3, ...)$ 533
530

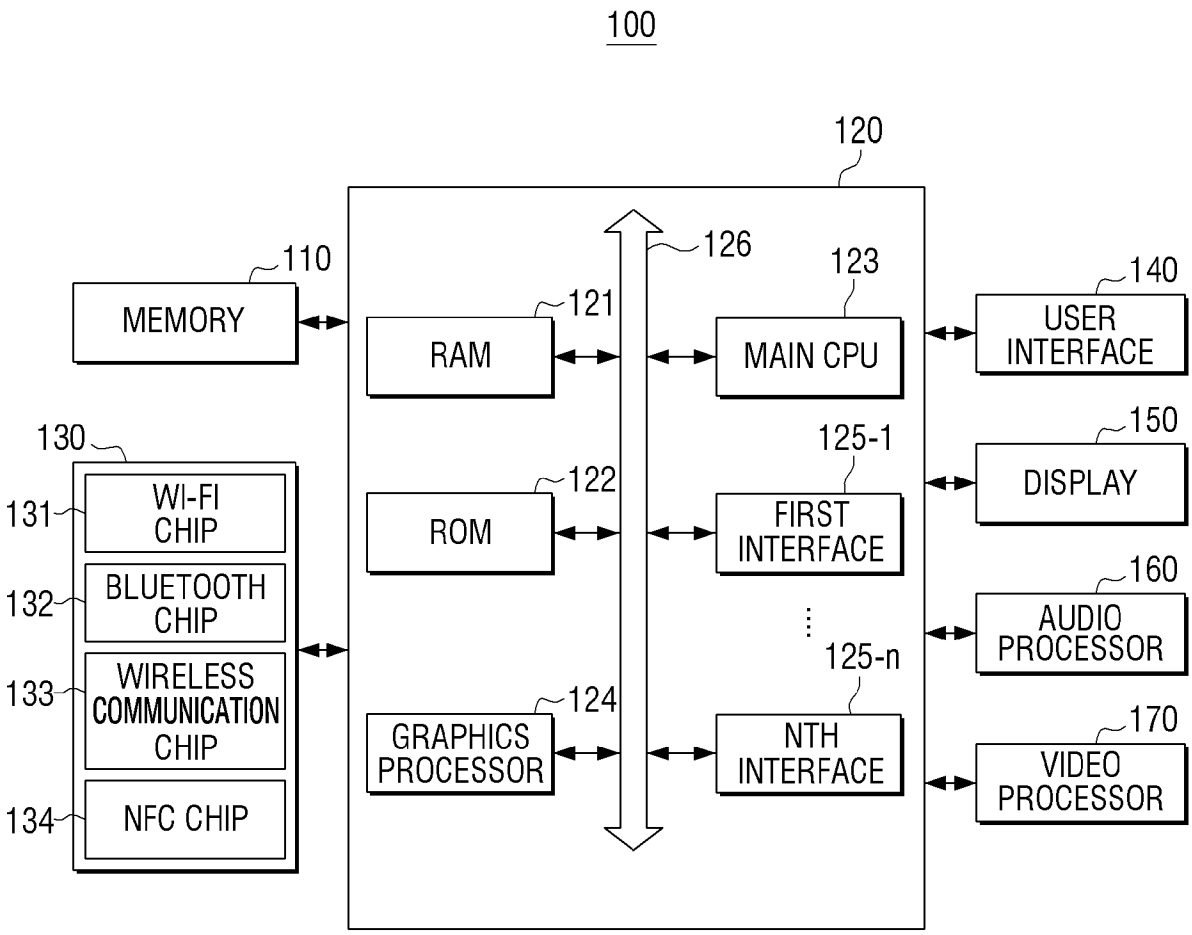
[Fig. 6]


DIRECT MEMORY ACCESS CONTROLLER
230
600
DRAM
32bit
32bit
Data for 1st bit
Data for 2nd bit
Data for kth bit
(a1, b11, b21, b31, ...,b321, ...)
(a2, b12, b22, b32, ...,b322, ...)
(ak, b1k, b2k, b3k, ...,b32k, ...)

[Fig. 7]


START
WHILE A PLURALITY OF DATA HAVING DIFFERENT DEGREES OF IMPORTANCE FROM ONE ANOTHER, WHICH ARE USED FOR AN OPERATION OF A NEURAL NETWORK MODEL, ARE STORED IN THE MEMORY, ACQUIRING RESOURCE INFORMATION FOR HARDWARE OF THE ELECTRONIC APPARATUS
701
BASED ON THE ACQUIRED RESOURCE INFORMATION, DETERMINING SOME DATA TO BE USED FOR AN OPERATION OF A NEURAL NETWORK MODEL AMONG THE PLURALITY OF DATA ACCORDING TO THE DEGREES OF IMPORTANCE OF EACH OF THE PLURALITY OF DATA
703
PERFORMING AN OPERATION OF A NEURAL NETWORK MODEL BY USING THE DETERMINED SOME DATA
705
END

[Fig. 8]


100
120
110
MEMORY
126
123
140
121
RAM
MAIN CPU
USER INTERFACE
130
125-1
150
131
WI-FI CHIP
122
ROM
FIRST INTERFACE
DISPLAY
132
BLUETOOTH CHIP
160
AUDIO PROCESSOR
125-n
133
WIRELESS COMMUNICATION CHIP
124
GRAPHICS PROCESSOR
NTH INTERFACE
170
VIDEO PROCESSOR
134
NFC CHIP

INTERNATIONAL SEARCH REPORT

International application No.
PCT/KR2020/006411

A. CLASSIFICATION OF SUBJECT MATTER

G06N 3/063(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)
G06N 3/063; G06F 13/28; G06F 15/18; G06N 3/04; G06N 3/08; G06N 99/00

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: neural network, resource information, battery, latency, quantized matrix, binary data, importance

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	CN 107103113 B (INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES) 11 January 2019 claims 1, 3, 5	1-15
Y	US 5809490 A (JOHN P. GUIVER et al.) 15 September 1998 column 5, lines 16-19; and column 10, line 61 - column 12, line 40	1-15
A	US 2019-0087744 A1 (SAP SE) 21 March 2019 paragraphs [0043]; claim 1; and figure 2	1-15
A	US 2018-0300614 A1 (MICROSOFT TECHNOLOGY LICENSING, LLC) 18 October 2018 paragraphs [0089]-[0090]; and claim 1	1-15
A	KR 10-2019-0054449 A (KOREA ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY) 22 May 2019 paragraphs [0028]-[0030]; and claim 1	1-15

☐ 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
"D" document cited by the applicant in the international application
"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
14 August 2020 (14.08.2020)	**14 August 2020 (14.08.2020)**

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
YANG JEONG ROK
Telephone No. +82-42-481-5709

Form PCT/ISA/210 (second sheet) (July 2019)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/KR2020/006411

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
CN 107103113 B	11/01/2019	CN 107103113 A	29/08/2017
		WO 2018-171717 A1	27/09/2018
US 5809490 A	15/09/1998	None	
US 2019-0087744 A1	21/03/2019	None	
US 2018-0300614 A1	18/10/2018	AU 2018-256212 A1	19/09/2019
		CN 110520853 A	29/11/2019
		CN 110520857 A	29/11/2019
		CN 110537194 A	03/12/2019
		CN 110546611 A	06/12/2019
		EP 3612990 A1	26/02/2020
		EP 3612991 A1	26/02/2020
		JP 2020-517014 A	11/06/2020
		KR 10-2019-0141694 A	24/12/2019
		US 10540584 B2	21/01/2020
		US 10628345 B2	21/04/2020
		US 2018-0300615 A1	18/10/2018
		US 2018-0300616 A1	18/10/2018
		US 2018-0300617 A1	18/10/2018
		US 2018-0300633 A1	18/10/2018
		US 2018-0300634 A1	18/10/2018
		WO 2018-194988 A1	25/10/2018
		WO 2018-194993 A1	25/10/2018
		WO 2018-194994 A2	25/10/2018
		WO 2018-194995 A1	25/10/2018
		WO 2018-194996 A1	25/10/2018
KR 10-2019-0054449 A	22/05/2019	None	