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

(19) **World Intellectual Property Organization**
International Bureau

(43) **International Publication Date**
10 January 2019 (10.01.2019)

WIPO | PCT

(10) **International Publication Number**
WO 2019/008661 A1

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

(21) **International Application Number:** PCT/JP2017/024459

(22) **International Filing Date:** 04 July 2017 (04.07.2017)

(25) **Filing Language:** English

(26) **Publication Language:** English

(71) **Applicant: NEC CORPORATION** [JP/JP]; 7-1, Shiba 5-chome, Minato-ku, Tokyo, 1088001 (JP).

(72) **Inventor: SOMBATSIRI, Salita**; c/o NEC Corporation, 7-1, Shiba 5-chome,Minato-ku, Tokyo, 1088001 (JP).

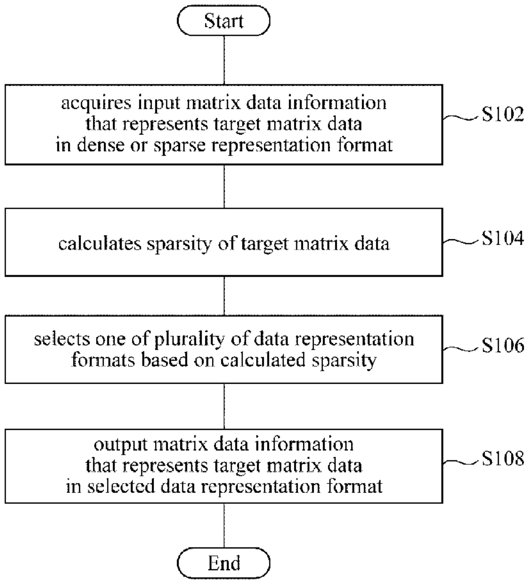
(74) **Agent: HAYAMI, Shinji**; KDX Gotanda Bldg. 9F, 9-2, Nishi-Gotanda 7-chome, Shinagawa-ku, Tokyo, 1410031 (JP).

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

(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:** INFORMATION PROCESSING APPARATUS, CONTROL METHOD, AND PROGRAM

[Fig. 7]


Start
acquires input matrix data information that represents target matrix data in dense or sparse representation format
S102
calculates sparsity of target matrix data
S104
selects one of plurality of data representation formats based on calculated sparsity
S106
output matrix data information that represents target matrix data in selected data representation format
S108
End

(57) **Abstract:** The information processing apparatus (2000) includes an acquisition unit (2020), a sparsity calculation unit (2040), a selection unit (2060), and an output unit (2080). The acquisition unit (2020) acquires the input matrix data information. The sparsity calculation unit (2040) calculates the sparsity of the target matrix data represented by the input matrix data information. The selection unit (2060) selects a representation format to be applied to the output matrix data information from a plurality of representation formats, based on the sparsity calculated by the sparsity calculation unit (2040). The plurality of representation formats include the dense representation format and at least two sparse representation formats. The output unit (2080) outputs the output matrix data information that represents the target matrix data in the representation format selected by the selection unit (2060).

[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: INFORMATION PROCESSING APPARATUS, CONTROL METHOD, AND PROGRAM

Technical Field

[0001] The present invention generally relates to data representation.

Background Art

[0002] Array data has been used to describe many kinds of information. For example, output data from Deep Neural Networks (DNN) can be described with array data.

[0003] Matrix data is a type of array data, which is composed of rows and columns. Matrix data is able to be represented in many types of data formats. The data formats used to represent matrix data falls into two main classes: dense representation format and sparse representation format. The dense representation format represents matrix data with all data elements. On the other hand, the sparse representation format represents matrix data with non-zero data elements (data elements the value of which is not zero) and their locations in the matrix. NPL 1 discloses various types of sparse representation format such as compressed sparse row (CSR), compressed sparse column (CSC), a Coordinate list (COO), block sparse row (BSR), list of list (LOL), and etc.

[0004] There is no representation format best in common for all matrix data, and which representation format is suitable depends on matrix data to be represented. PTL1 discloses a way of selecting a representation format to be used to represent matrix data. In this document, the sparsity of the matrix data is compared with the threshold in order to choose dense or sparse data representation. The sparsity of matrix data is a value indicating how sparse the matrix is. For example, the sparsity of matrix data is defined by the ratio of the number of zero-valued data elements to the total number of data elements in the matrix data. When it is determined to use sparse representation based on the sparsity of the matrix data, either of CSC and CSR are further chosen based on the number of rows and columns of the matrix data.

Citation List

Patent Literature

[0005] [PTL1] US Patent Application Publication No. US 2016/0364327 A1

Non-Patent Literature

[0006] [NPL1] Reginald P. Tewarson, "Sparse Matrices", ACADEMIC PRESS INC, May 1, 1973

[NPL2] Alex Krizhevsky et al., "ImageNet Classification with Deep Convolutional Neural Networks", THE NEURAL INFORMATION PROCESSING SYSTEMS

CONFERENCE, pp. 1097-1105, Decempber, 2012

[NPL3] Geoffrey Hinton et al., "Deep Neural Networks for Acoustic Modeling in Speech Recognition", IEEE SIGNAL PROCESSING MAGAZINE, VOL 29, ISSUE 6, pp. 82-97, October 18, 2012

[NPL4] Alex Graves et al., "Speech Recognition with Deep Recurrent Neural Networks," IEEE International Conference on Acoustics, Speech and Signal Processing 2013, pp. 26-31, May 26-31, 2013

Summary of Invention

Technical Problem

[0007] The technique disclosed by PTL1 uses the sparsity of the matrix data in order to merely distinguish high sparsity and low sparsity of matrix data, and determine whether to use dense or sparse representation format. Therefore, this technique is not effective for matrix data with moderate sparsity. An object of the present invention is to provide a technique capable of effectively determining suitable representation format even for matrix data with moderate sparsity.

Solution to Problem

[0008] The present invention provides an information processing apparatus comprising: an acquiring unit acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format; a sparsity calculation unit calculating sparsity of the target matrix data; a selection unit selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and an output unit outputting output matrix data information in which the target matrix data is represented by in the selected representation format.

[0009] The present invention provides a control method to be performed by a computer. The control method comprises: acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format; calculating sparsity of the target matrix data; selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and outputting output matrix data information in which the target matrix data is represented by in the selected representation format.

[0010] The present invention provides a program causing a computer executes: acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format; calculating sparsity of the target matrix data; selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and outputting output matrix data information in which the target matrix data is represented by in the selected representation format.

Advantageous Effects of Invention

[0011] According to the present invention, it is provided that a technique capable of determining suitable representation format even for matrix data with moderate sparsity.

Brief Description of Drawings

[0012] The above-mentioned objects, other objects, features and advantages will be made clearer from the preferred example embodiments described below, and the following accompanying drawings.

[0013] [fig.1]Fig. 1 illustrates an information processing apparatus of the first example embodiment.

[fig.2]Fig. 2 illustrates examples of the matrix data information representing the target matrix in the dense representation format.

[fig.3]Fig. 3 illustrates the matrix data information representing the target matrix data in the CSR sparse representation format.

[fig.4]Fig. 4 illustrates the matrix data information representing the target matrix data in the CSC sparse representation format.

[fig.5]Fig. 5 illustrates the matrix data information representing the target matrix data in the row-major COO sparse representation format.

[fig.6]Fig. 6 illustrates a block diagram of a hardware configuration of the information processing apparatus in which the information processing apparatus is implemented by a combination of hardware elements and software elements.

[fig.7]Fig. 7 illustrates a flowchart showing a flow of processes performed by the information processing apparatus of the first example embodiment.

[fig.8]Fig. 8 illustrates three examples of matrix data.

[fig.9]Fig. 9 illustrates an example flow of selecting a representation format.

[fig.10]Fig. 10 illustrates an example of the matrix data information that represents the target matrix data in the row-major order element-wise flag sparse representation format.

[fig.11]Fig. 11 illustrates an example of the matrix data information that represents matrix data in the column-major order element-wise flag sparse representation format.

[fig.12]Fig. 12 illustrates another example of flow of selecting a representation format of the output matrix data information where there are three options of sparsity representation format.

[fig.13]Fig. 13 illustrates a flowchart in which the output unit 2080 operates in parallel with the sparsity calculation unit 2040 and the selection unit 2060.

[fig.14]Fig.14 illustrates the information processing apparatus of the second example embodiment.

[fig.15]Fig. 15 illustrates how the conversion unit 2100 works when 1D array data is input.

Description of Embodiments

[0014] Example embodiments of the present invention are described in detail below referring to the accompanying drawings. Note that, in block diagrams, each block represents a function-based configuration instead of a hardware-based configuration.

[0015] <First Example Embodiment>

Fig. 1 illustrates an information processing apparatus 2000 of the first example embodiment. The information processing apparatus 2000 handles matrix data information, which represents matrix data in one of a plurality of representation formats. Hereinafter, the matrix data represented by the matrix data information is called "target matrix data".

[0016] The plurality of representation formats include the dense representation format and at least two sparse representation formats. When representing the target matrix data in the dense representation format, the matrix data information may include all data elements of the target matrix data in either row-major order or column-major order. Fig. 2 illustrates examples of the matrix data information representing the target matrix in the dense representation format. In Fig. 2, the matrix data information 10-1 includes a data sequence 12-1 that indicates all data elements of the target matrix data in row-major order, and a format flag 14-1 that indicates the representation format used in the matrix data information 10. The format flag 14 indicates that the row-major dense representation format is used. On the other hand, the matrix data information 10-2 includes the data sequence 12-2 that indicates all data elements of the target matrix data in column-major order. The format flag 14-2 indicates that the column-major dense representation format is used.

[0017] When representing the matrix data in a sparse representation format, the matrix data information does not include at least one of all data elements. For example, the matrix data information in a sparse representation format includes non-zero data elements and location information. The location information is information that can be used to determine locations of each non-zero data elements. For example, the location information includes indices of each non-zero data elements or those of each zero-valued data elements.

[0018] CSR, CSC, and COO are examples of sparse representation format. Fig. 3 illustrates the matrix data information representing the target matrix data in the CSR sparse representation format. The matrix data information 10-3 includes the data sequence 12-3, the format flag 14-3, and the location information 16-3. It is assumed that x5 and x6 are zero in Fig. 3. The data sequence 12-3 includes only non-zero data elements in row-major order, and does not include x5 and x6. The format flag 14-3 shows that the CSR sparse representation format is used. The location information 16-3 includes column indices corresponding to the non-zero data elements, and row pointer.

[0019] Fig. 4 illustrates the matrix data information representing the target matrix data in the CSC sparse representation format. The matrix data information 10-4 includes the data sequence 12-4, the format flag 14-4, and the location information 16-4. Fig. 4 also assumes that x5 and x6 are zero. The data sequence 12-4 includes only non-zero data elements in column-major order, and does not include x5 and x6. The format flag 14-4 shows that the CSC sparse representation format is used. The location information 16-4 includes row indices corresponding to the non-zero data elements, and column pointer.

[0020] Fig. 5 illustrates the matrix data information representing the target matrix data in the row-major COO sparse representation format. The matrix data information 10-5 includes the data sequence 12-5, the format flag 14-5, and the location information 16-5. Fig. 6 also assumes that x5 and x6 are zero. The data sequence 12-5 includes only non-zero data elements in row-major order, and does not include x5 and x6. The format flag 14-5 shows that the row-major COO sparse representation format is used. The location information 16-5 includes row indices and column indices corresponding to the non-zero data elements. Note that, the column-major COO can also be used.

[0021] The information processing apparatus 2000 acquires input matrix data information that represents the target matrix data in the dense representation format or a sparse representation format, calculates the sparsity of the target matrix data, selects a representation format to be used in output matrix data information, and outputs the output matrix data information that represents the target matrix data in the selected representation format. The representation format to be used in the output matrix data information is selected based on the sparsity of the target matrix data from one of the above-mentioned plurality of the representation formats, i.e. the dense representation format and at least two sparse representation formats.

[0022] In order to realize the above-mentioned operations, the information processing apparatus 2000 includes an acquisition unit 2020, a sparsity calculation unit 2040, a selection unit 2060, and an output unit 2080. The acquisition unit 2020 acquires the input matrix data information. The sparsity calculation unit 2040 calculates the sparsity of the target matrix data represented by the input matrix data information. The selection unit 2060 selects a representation format to be applied to the output matrix data information from the above-mentioned plurality of representation formats, based on the sparsity calculated by the sparsity calculation unit 2040. The output unit 2080 outputs the output matrix data information that represents the target matrix data in the representation format selected by the selection unit 2060.

[0023] <Advantageous Effects>

According to the present example embodiment of the information processing apparatus 2000, the representation format of the target matrix data is determined among the dense representation format and at least two sparse representation formats based on the sparsity of the matrix data. Thus, the sparsity of the matrix data is used not only for determining whether to use dense or sparse representation format, but also for which one of multiple sparse representation formats is to be used to represent the target matrix data. Therefore, unlike the technique disclosed by PTL1 that use the sparsity of matrix data merely for determining whether to use dense or sparse representation format, the information processing apparatus 2000 is able to effectively determine suitable representation format even for the target matrix data with moderate sparsity.

[0024] An example use of matrix data is to describe data in DNN. A typical DNN architecture for image recognition is a Deep Convolutional Neural Network (DCNN) as described in NPL2, and a DNN architecture for speech recognition is either a Deep Feed Forward Neural Network (DFF) or a Deep Recurrent Neural Network (DRNN) as described in NPL3 and NPL4, respectively. Generally, the output data from DNN is called feature or activation data, and are described with 1D vector, matrix, or n-dimensional array. If the activation data is the output from DCNN, then it is usually called feature map and is a matrix or multi-dimensional array. On the other hand, if the activation data is from DFF or DRNN, it is called feature and is a vector.

[0025] A DCNN consists of a stack of convolutional layers, activation layers, pooling layers, and fully-connected layers, which convolute the input feature maps with their kernels to extract features, transform the input feature with a non-linear function, downsample the input feature, and perform matrix multiplication in order to classify the input into a class, respectively. A DFF consists of a stack of fully-connected layers and activation layers. A DRNN consists of a stack of recurrent layers, which performs matrix multiplication of the past and present context, and activation layers. An activation layer generates non-uniform sparse matrix data by applying non-linear functions to the input. The non-linear functions may be sigmoid or Rectified Linear Unit (ReLU) function.

[0026] Since a large amount of matrix data is input and output in DNN, efficient representation of matrix data is quite important in terms of storage space, network bandwidth, and so on. Suitable selection of representation format of matrix data with the information processing apparatus of the present example embodiment is therefore useful in DNN.

[0027] Note that, DNN is merely an example of applications of the information processing apparatus 2000, and the information processing apparatus 2000 is applicable to a lot of other domains in which matrix data is used.

[0028] Hereinafter, the information processing apparatus 2000 of the present invention will be described in more detail.

[0029] <Example of Hardware Configuration>

Each functional component of the information processing apparatus 2000 may be implemented only by hardware elements that implement each functional component shown in Fig. 1 (for example, an hard-wired electronic circuit), or may be implemented by a combination of hardware elements and software elements (for example, a combination of electronic circuits and a program controlling the electronic circuits).

[0030] Fig. 6 illustrates a block diagram of a hardware configuration of the information processing apparatus 2000 in which the information processing apparatus 2000 is implemented by a combination of hardware elements and software elements. The information processing apparatus 2000 includes a bus 1020, a processor 1040, a memory 1060, a storage device 1080, an input-output interface 1100, and a network interface 1120. The bus 1020 is a data transmission channel through which the processor 1040, the memory 1060, the storage device 1080, the input-output interface 1100, and the network interface 1120 exchange data. Note that, a way of connecting the hardware elements with each other is not limited to bus connection.

[0031] The processor 1040 is an electronic circuitry that carries out instructions of a computer program, such as central processing unit (CPU) or graphics processing unit (GPU). In another example, the processor 1040 may be a specific circuit such as Application-Specific Integrated Circuit (ASIC), Application-Specific Instruction set

Processor (ASIP), or reconfigurable devices such as Field Programmable Gate Array (FPGA). The memory 1060 is a primary storage device such as random access memory (RAM) or read only memory (ROM). The storage 1080 is a secondary storage device such as hard disk, solid state drive (SSD), or memory card. The input-output interface 1100 is an interface through which the information processing apparatus 2000 connects to peripheral devices, such as keyboard, display, and so on. The network interface 1120 is an interface through which the information processing apparatus connect to network, such as local area network (LAN), wide area network (WAN), and so on.

[0032] The storage device 1080 stores program modules with which the above-mentioned functional components of the information processing apparatus 2000 are realized. The processor 1040 loads the program modules into the memory 1060, and executes the loaded program modules.

[0033] <Flow of Processing>

Fig. 7 illustrates a flowchart showing a flow of processes performed by the information processing apparatus 2000 of the first example embodiment. The acquiring unit 2020 acquires the input matrix data information that represents the matrix data in the dense or a sparse representation format (S102). The sparsity calculation unit 2040 calculates the sparsity of the target matrix data (S104). The representation determination unit 2060 selects one of the plurality of representation formats based on the calculated sparsity (S106). The output unit 2080 outputs output matrix data information that represents the target matrix data in the selected representation format (S108).

[0034] <Acquisition of Matrix Data Information: S102>

The acquiring unit 2020 acquires the input matrix data information (S102). The input matrix data information can be acquired with various ways. For example, the input matrix data information may be stored in the storage device 1080 in advance. In this case, the acquiring unit 2020 acquires the input matrix data information from the storage device 1080. In another example, the input matrix data information may be input by a user of the information processing apparatus 2000 using an input device such as a keyboard or a touch panel. In another example, the acquiring unit 2020 may access to an external device, such as a server machine or network attached storage (NAS), that stores the input matrix data information, and acquire the input matrix data information from the external device. In another example, the acquiring unit 2020 may receive the input matrix data information transmitted by the external devices.

[0035] <Calculation of Sparsity of Matrix Data: S104>

The sparsity calculation unit 2040 calculates the sparsity of the target matrix data (S104). The sparsity of matrix data may be defined by the following equation.

<Equation 1>

$$S = \frac{n_{zero}}{n_{total}}$$

S represents the sparsity of the matrix data. n_zero represents the number of zero-valued data elements of the matrix data. n_total represents the total number of data elements of the matrix data. By this definition, the larger the value of S is, the sparser the matrix is.

[0036] Fig. 8 illustrates three examples of matrix data: matrix data A, B, and C. As for the matrix data A, the number of zero-valued data elements and the number of total data elements are 2 and 25, respectively. Thus, the sparsity of the matrix data A is 0.08 (2/25). As for the matrix data B, the number of zero-valued data elements and the number of total data elements are 6 and 25, respectively. Thus, the sparsity of the matrix data B is 0.24 (6/25). As for the matrix data C, the number of zero-valued data elements and the number of total data elements are 48 and 49, respectively. Thus, the sparsity of the matrix data C is 0.98 (48/49).

[0037] The sparsity calculation unit 2040 calculates the sparsity of the target matrix data using the input matrix data information acquired by the acquisition unit 2020. For example, the sparsity calculation unit 2040 counts the number of zero-valued data elements and the number of non-zero data elements of the target matrix data, respectively. Next, the sparsity calculation unit 2040 calculates the total number of data elements of the matrix data by summing the number of zero-valued data elements and the number of non-zero data elements of the target matrix data. Then, the sparsity calculation unit 2040 calculates the sparsity S of the target matrix data by applying to Equation 1 the number of zero-valued data elements and the total number of data elements of the target matrix. Note that, how to recognize the number of non-zero data elements and the total number of data elements of matrix data is not limited to the above exemplified way, and various well-known techniques can be applied.

[0038] When the input matrix data information represents the target matrix data in a sparse representation format, the data sequence 12 indicated by the input matrix data information does not include zero-valued data elements. Thus, the sparsity calculation unit 2040 cannot count the zero-valued data elements only with the data sequence 12. In this case, for example, the sparsity calculation unit 2040 generates data sequence of the target matrix data represented in the dense representation format using the data sequence 12 and the location information 14 indicated by the input matrix data information. Then, the sparsity calculation unit 2040 counts the number of zero-valued data elements of the target matrix data using the generated data sequence.

[0039] In another example, the input matrix data information may indicate the number of non-zero data elements of the target matrix data or the total number of data elements of the target matrix data. With this configuration, the sparsity calculation unit 2040 can calculate the sparsity of the target matrix data without counting zero-valued data elements of the target matrix data.

[0040] <Selection of Representation format: S106>

The format selection unit 2060 selects one of the plurality of representation formats based on the calculated sparsity (S106). Specifically, the format selection unit 2060 compares the calculated sparsity of the target matrix data with pre-determined thresholds, and selects the representation format based on the result of the comparison.

[0041] Fig. 9 illustrates an example flow of selecting a representation format. In this example, there are two pre-determined thresholds: High-sparsity threshold and Low-sparsity threshold. High-sparsity threshold is greater than Low-sparsity threshold.

[0042] In the step 202, the format selection unit 2060 compares the calculated sparsity of the target matrix data with Low-sparsity threshold and determines whether or not the calculated sparsity of the target matrix data is smaller than Low-sparsity threshold. If it is determined the calculated sparsity of the target matrix data being smaller than Low-sparsity threshold (S202: YES), the format selection unit 2060 selects the dense representation format (S204).

[0043] On the other hand, if it is determined the calculated sparsity of the target matrix data not being smaller than Low-sparsity threshold (S202: NO), the format selection unit 2060 compares the calculated sparsity of the matrix data with High-sparsity threshold and determines whether or not the calculated sparsity of the target matrix data is smaller than High-sparsity threshold (S206). If it is determined the calculated sparsity of the target matrix data being smaller than High-sparsity threshold (S206: YES), the format selection unit 2060 selects the first sparsity representation format (S208). On the other hand, if it is determined the calculated sparsity of the target matrix data being not smaller than High-sparsity threshold (S206: NO), the format selection unit 2060 selects the second sparsity representation format (S210).

[0044] The first and second sparsity representation formats are different from each other in that the first sparsity representation format is suitable for matrix data with moderate sparsity, whereas the second sparsity representation format is suitable for that with high sparsity. Thus, the format selection unit 2060 selects the second representation format when the sparsity of the target matrix data is greater than High-sparsity threshold, whereas it selects the first representation format when the sparsity of the target matrix data is less than or equal to High-sparsity threshold.

[0045] An example of the first sparsity representation format is element-wise flag sparse representation format. The element-wise flag sparse representation format represents matrix data with non-zero data elements of the matrix data and Non-zero-element flags for each element of the matrix data, in either row-major or column-major order. Non-zero-element flags show whether or not the value of data elements is zero for each data elements of the matrix data. Hereinafter, the element-wise flag sparse representation format in which the non-zero data elements and Non-zero-element flags are described in row-major order is called "the row-major element-wise flag sparse representation format", whereas the element-wise flag sparse representation format in which the non-zero data elements and Non-zero-element flags are described in column-major order is called "the column-major element-wise flag sparse representation format".

[0046] Fig. 10 illustrates an example of the matrix data information that represents the target matrix data in the row-major order element-wise flag sparse representation format. The matrix data information 10-6 represents the matrix data A in the row-major order element-wise flag sparse representation format. In this example, it is assumed that x5 and x6 are zero.

[0047] The matrix data information 10-6 includes the data sequence 12-6, the format flag 14-6, and the location information 16-6. Since x5 and x6 are assumed to be zero, the data sequence 12-6 does not include x5 and x6, but include x0 to x4, x7, and x8 in row-major order. The format flag 14-6 indicates that the row-major order element-wise flag sparse representation format is used. The location information 16-6 includes Non-zero-element flags. Non-zero-element flags corresponding to x0 to x4, x7, and x8 indicate 1, whereas those corresponding to x5 and x6 indicate 0, in row-major order.

[0048] Fig. 11 illustrates an example of the matrix data information that represents matrix data in the column-major order element-wise flag sparse representation format. Matrix data information 10-7 represents the matrix data A in the column-major order element-wise flag sparse representation format. Also in this example, it is assumed that x5 and x6 are zero.

[0049] As illustrated by Fig. 11, the data sequence 12-7 does not include x5 and x6, but include x0 to x4, x7, and x8 in column-major order. As for Non-zero-element flags (location information 14-7), those corresponding to x0 to x4, x7, and x8 indicate 1 whereas those corresponding to x5 and x6 indicate 0, in column-major order. The format flag 14-7 indicates that the column-major order element-wise flag sparse representation format is used.

[0050] The matrix data information representing the target matrix data in the element-wise flag sparse representation format can be generated from the input matrix data information by, for example, sequentially scanning each data element of the data sequence of the input matrix data information when the input matrix data information represents the target matrix data in the dense representation format. When the scanned data element is zero, the corresponding Non-zero-element flag is set as zero. On the other hand, when the scanned data element is not zero, the corresponding Non-zero-element flag is set as 1, and the scanned data element is added into the data sequence of the matrix data information representing the target matrix data in the element-wise flag sparse representation format. Note that, the data sequence of the input matrix data information is scanned in column-major order when using the column-major order element-wise flag sparse representation format, whereas the data sequence of the input matrix data information is scanned in row-major order when using the row-major order element-wise flag sparse representation format.

[0051] Note that, it is preferable that the sparsity calculation unit 2040 is configured to generate Non-zero-element flags when counting the number of zero-valued data elements and the number of non-zero data elements of the target matrix data since the sparsity calculation unit 2040 necessarily determines whether each data element of the target matrix data is zero or not. With this configuration, the output unit 2080 can use Non-zero-element flags generated by the sparsity calculation unit 2040 and does not need to generate them when it is determined that the element-wise flag sparse representation is used.

[0052] The second sparse representation format may be, for example, CSR, CSC, COO, BSR, or LOL. Note that, the first sparse representation format may also be one of the above four sparse representation format. For example, CSR and COO may be used as the first and the second sparse representation format, respectively.

[0053] Definition of High-sparsity threshold depends on which representation formats are used. For example, if the element-wise flag sparse representation format is used as the first sparse representation format and CSC is used as the second sparse representation format, High-sparsity threshold may be defined by the following Equation 2.

<Equation 2>

$$Th_1 = 1 - \frac{R - \log_2(C)}{R \times \log_2(R)}$$

Th1 represents High-sparsity threshold. R represents the number of rows of the target matrix data. C represents the number of columns of the target matrix data.

[0054] As a concept, threshold values are derived from the comparison between representation formats in terms of the amount of data used to represent the target matrix data. In terms of the above exemplified case, the comparison between the number of bits used to represent the target matrix data in the element-wise flag sparse representation format and that in CSR is described as the following Equation 3. Th1 in Equation 2 is obtained by solving Equation 3 for S (Th1 = S).

<Equation 3>

#bits used to represent matrix in Element-wise flag ≤ #bits used in CSR

$$R \times C + (1-S) \times B \times R \times C \leq (1-S) \times B \times R \times C + R \times \log(R) + (1-S) \times R \times C \times \log(C)$$

B represents the number of bits used to represent each data element of the target matrix data. S represents the sparsity of the target matrix data.

[0055] In another example, if the element-wise flag sparse representation format is used as the first sparse representation format and CSR is used as the second sparse representation format, High-sparsity threshold may be defined by the following Equation 4.

<Equation 4>

$$Th_1 = 1 - \frac{C - log_2(R)}{C \times log_2(C)}$$

[0056] On the other hand, if the element-wise flag sparse representation format is used as the first sparse representation format, Low-sparsity threshold may be defined by the following Equation 5.

<Equation 5>

$$Th_2 = 1 / B$$

Th2 represents Low-sparsity threshold. B represents the number of bits used to represent each data element of the target matrix data.

[0057] The information processing apparatus 2000 may have three or more options of sparsity representation formats. Fig. 12 illustrates another example flow of selecting a representation format of the output matrix data information where there are three options of sparsity representation format. In this example, there are three predetermined thresholds: High-sparsity threshold, Mid-sparsity threshold, and Low-sparsity threshold. Mid-sparsity threshold is lower than High-sparsity threshold and greater than Low-sparsity threshold.

[0058] In the step 302, the format selection unit 2060 determines whether or not the calculated sparsity of the target matrix data is smaller than Low-sparsity threshold. If it is determined the calculated sparsity of the target matrix data being smaller than Low-sparsity threshold (S302: YES), the format selection unit 2060 selects the dense representation format (S304).

[0059] On the other hand, if it is determined the calculated sparsity of the target matrix data not being smaller than Low-sparsity threshold (S302: NO), the format selection unit 2060 compares the calculated sparsity of the target matrix data with Mid-sparsity threshold and determines whether or not the calculated sparsity of the target matrix data is smaller than Mid-sparsity threshold (S306). If it is determined the calculated sparsity of the target matrix data being smaller than Mid-sparsity threshold (S306: YES), the format selection unit 2060 selects the 1st sparsity representation format (S308). If it is determined the calculated sparsity of the target matrix data not being smaller than Mid-sparsity threshold (S306: NO), the format selection unit 2060 determines whether or not the calculated sparsity of the target matrix data is smaller than High-sparsity threshold (S310). If it is determined the calculated sparsity of the target matrix data being smaller than High-sparsity threshold (S310: YES), the format selection unit 2060 selects the second sparsity representation format (S312). On the other hand, if it is determined the calculated sparsity of the target matrix data being not smaller than High-sparsity threshold (S310: NO), the format selection unit 2060 selects the third sparsity representation format (S314).

[0060] The first, second, and third sparsity representation formats may be, for example, element-wise flag sparse representation format, CSR, and COO, respectively. In this case, Mid-sparsity threshold and Low-sparsity threshold may be defined as Th1 in Equation 4 and Th2 in the Equation 5, respectively. In addition, High-sparsity threshold may be defined by the following equation.

<Equation 6>

$$Th_3 = \frac{C-1}{C}$$

C represents the number of columns of the target matrix data.

[0061] <Output of Matrix Data Information: S108>

The output unit 2080 outputs the output matrix data information (S108). The output matrix data information is generated by the output unit 2080. For example, the output unit 2080 acquires the result of the selection performed by the selection unit 2060, and then generates the output matrix data information that represents the target matrix data in the representation format selected by the selection unit 2060.

[0062] In another example, the output unit 2080 may prepare the output matrix data information in parallel with the selection unit 2060 selecting the representation format of the output matrix data information. Specifically, the output unit 2080 may prepare all candidates of the output matrix data information each of which represents the target matrix data in different representation formats from each other. Suppose that element-wise flag sparse representation format, CSR, and COO are the options of sparse representation formats. In this case, the output unit 2080 prepares the output matrix data by generating three candidates of output matrix data information that represent the target matrix data in the element-wise flag sparse representation format, CSR, and COO respectively in parallel with the selection unit 2060 selecting the representation format of the output matrix data information.

[0063] After preparing the candidates of the output matrix data information, the output unit 2080 acquires the information that indicates the selected representation format from the selection unit 2060. The output unit 2080 then output one of the candidates of the output matrix data information the representation format of which matches the selected representation format, as the output matrix data information. Note that, if the representation format selected by the selection unit 2060 is the same as that used in the input matrix data information, the output unit 2080 may output the input matrix data information as the output matrix data information.

[0064] In another example, the preparation of candidates of the output matrix data information may be performed in parallel with the calculation of the sparsity of the target matrix data.

[0065] Fig. 13 illustrates a flowchart in which the output unit 2080 operates in parallel with the sparsity calculation unit 2040 and the selection unit 2060. Note that, the steps S102, S104, S106, and S108 are the same as those in Fig. 6, and they are performed by the acquisition unit 2020, the sparsity calculation unit 2040, the selection unit 2060, and the output unit 2080, respectively.

[0066] The input matrix data information is used to generate the output matrix data information. How to generate the output matrix data information from the input matrix data information depends on the representation format of the input matrix data information. When the target matrix data is represented in the dense representation format in the input matrix data information, the output unit 2080 generates the output matrix data information using the data sequence 12 that includes all data elements of the target matrix data. Note that, well-known techniques can be used to convert the format of the target matrix data from the dense representation format to a sparse representation format.

[0067] On the other hand, when the target matrix data is represented in a sparse representation format in the input matrix data information, the output unit 2080 generates the output matrix data information using the data sequence 12 that includes non-zero data elements and the location information 14. For example, the output unit 2080 retrieves all data elements of the target matrix data (converts the input matrix data information into the dense representation format) based on non-zero data elements and the location information, and converts the target matrix data (the retrieved data elements) from the dense representation format to the representation format selected by the selection unit 2060.

[0068] In another example, the output unit generates the output matrix data information by directly converting the input matrix data information into the representation format selected by the selection unit 2060. In this case, the output unit 2080 may include algorithms of converting the format of the target matrix data for each combination of the input format and the output format. Suppose that there are three choices of data representation formats that the selection unit 2060 can select: their names are f1, f2 and f3, respectively. In this case, the output unit 2080 may include algorithms of converting the format of the target matrix data from f1 to f2, from f1 to f3, from f2 to f1, from f2 to f3, from f3 to f1, and from f3 to f2.

[0069] The output matrix data information can be output to the inside or the outside of the information processing apparatus 2000 in various ways. For example, the output unit 2080 writes the output matrix data information into the memory 1060 or the storage device 1080. In another example, the output unit 2080 displays the output matrix data information on a display device that is connected with the information processing apparatus 2000 through the input-output interface 1100. In another example, the output 2080 transmits the output matrix data information to a server machine or a NAS through the network interface 1120.

Second Example Embodiment

[0070] Fig.14 illustrates the information processing apparatus 2000 of the second example embodiment. Except for functions described below, the information processing apparatus 2000 of the present example embodiment has similar functions to those of the first example embodiment.

[0071] The information processing apparatus 2000 of the present invention accepts input data that is not described as matrix (2-dimensional array data), but as one-dimensional (1D) array or three-or-more-dimensional array. This input data is handled as one or more of matrix data, and each matrix data is processed as described in the first example embodiment.

[0072] In order to do so, the information processing apparatus 2000 includes a conversion unit 2100. The conversion unit 2100 acquires input data and converts it into one or more pieces of the input matrix data information.

[0073] When the input data is described as 1D array, the conversion unit 2100 equally divides the input data into a plurality of rows or a plurality of columns, and thereby generating the input matrix data information. The length of each row or the length of each column may be defined in advance.

[0074] Fig. 15 illustrates how the conversion unit 2100 works when 1D array data is input. Fig. 15 supposes that the input data is described as 1D array. The input data includes 15 data elements (x0 to x14). The length of each row is defined as 5. In this case, the conversion unit 2100 equally divides the input data into three parts.

Specifically, the sequence of x0 to x4 is converted into the first row, the sequence of x5 to x9 is converted into the second row, and the sequence of x10 to x14 is converted into the third row, respectively.

[0075] When the input data is described as 3-or-more dimensional array data, the conversion unit 2100 handles the input data as the collection of multiple matrix data. For example, 3D array data can be handled as a sequence of multiple matrix data. Thus, the conversion unit 2100 retrieves each matrix data included in the 3-or-more dimensional array data, and generates a plurality of input matrix data information each of which includes respective ones of the retrieved matrix data.

[0076] The format flag of the generated input matrix data indicates the representation format of the input data that the conversion unit 2100 acquires. For example, when the representation format of the input data is the dense representation format, the conversion unit 2100 generates one or more pieces of the input matrix data information each of which has the representation flag indicating the dense representation format.

[0077] <Advantageous Effect>

According to the present example embodiment of the information processing apparatus 2000, not only matrix data but also 1D array and 3-or-more dimensional array can be handled to convert their representation format into more effective one based on their sparsity.

[0078] <Appendices>

Hereinafter, examples of reference configurations will be described.

(Appendix 1)

An information processing apparatus comprising:

an acquiring unit acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format;

a sparsity calculation unit calculating sparsity of the target matrix data;

a selection unit selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and

an output unit outputting output matrix data information in which the target matrix data is represented in the selected representation format.

(Appendix 2)

The information processing apparatus according to appendix 1, wherein the selection unit performs:

determining whether or not the calculated sparsity is greater than low-sparsity threshold;

selecting the dense representation format when it is determined that the calculated sparsity is smaller than the low-sparsity threshold;

determining whether or not the calculated sparsity is smaller than high-sparsity threshold when it is determined that the calculated sparsity is not smaller than the low-sparsity threshold, the high-sparsity threshold being greater than the low-sparsity threshold;

selecting a first sparse representation format when it is determined that the calculated sparsity is smaller than the high-sparsity threshold; and

selecting a second sparse representation format when it is determined that the calculated sparsity is not smaller than the high-sparsity threshold.

(Appendix 3)

The information processing apparatus according to appendix 2,

wherein the high-sparsity threshold is defined by comparing the number of bits used to represent the target matrix data in the first sparse representation format with the number of bits used to represent the target matrix data in the second sparse representation format, and

wherein the low-sparsity threshold is defined by comparing the number of bits used to represent the target matrix data in the dense representation format with the number of bits used to represent the target matrix data in the first sparse representation format.

(Appendix 4)

The information processing apparatus according to appendix 3, wherein the high-sparsity threshold is defined by Equation 7 when the first sparse representation format is element-wise flag sparse representation format and the second sparse representation format is compressed sparse row, wherein Th1 represents the high-sparsity threshold, R represents a number of rows of the target matrix data, and C represents a number of columns of the target matrix data.

<Equation 7>

$$Th_1 = 1 - \frac{C - \log_2(R)}{C \times \log_2(C)}$$

(Appendix 5)

The information processing apparatus according to appendix 3 or 4, wherein the low-sparsity threshold is defined by Equation 8 when the first sparse representation format is element-wise flag sparse representation format, wherein Th2 represents the low-sparsity threshold, B represents a number of bits used to represent each data element of the target matrix data.

<Equation 8>

$$Th_2 = 1 / B$$

(Appendix 6)

The information processing apparatus according to any one of appendices 1 to 5, further comprising a conversion unit that acquires one-dimensional array data, divides the one-dimensional array into a plurality of rows or columns, and generates the input matrix data information includes the plurality of rows or columns,

wherein the acquisition unit acquires the input matrix data generated by the conversion unit.

(Appendix 7)

The information processing apparatus according to any one of appendices 1 to 6, further comprising a conversion unit that acquires three-or-more-dimensional array data, retrieves a plurality of matrix data from the three-or-more-dimensional array data, and generates a plurality pieces of the input matrix data information each of which includes respective ones of the retrieved matrix data,

wherein the acquisition unit acquires the plurality pieces of the input matrix data information generated by the conversion unit.

(Appendix 8)

A control method to be performed by a computer, the method comprising:

acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format;

calculating sparsity of the target matrix data;

selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and

outputting output matrix data information in which the target matrix data is represented by in the selected representation format.

(Appendix 9)

The control method according to appendix 7, wherein the selecting of the representation format includes:

determining whether or not the calculated sparsity is greater than low-sparsity threshold;

selecting the dense representation format when it is determined that the calculated sparsity is smaller than the low-sparsity threshold;

determining whether or not the calculated sparsity is smaller than high-sparsity threshold when it is determined that the calculated sparsity is not smaller than the low-sparsity threshold, the high-sparsity threshold being greater than the low-sparsity threshold;

selecting a first sparse representation format when it is determined that the calculated sparsity is smaller than the high-sparsity threshold; and

selecting a second sparse representation format when it is determined that the calculated sparsity is not smaller than the high-sparsity threshold.

(Appendix 10)

The control method according to appendix 9,

wherein the high-sparsity threshold is defined by comparing the number of bits used to represent the target matrix data in the first sparse representation format with the number of bits used to represent the target matrix data in the second sparse representation format, and

wherein the low-sparsity threshold is defined by comparing the number of bits used to represent the target matrix data in the dense representation format with the number of bits used to represent the target matrix data in the first sparse representation format.

(Appendix 11)

The control method according to appendix 10, wherein the high-sparsity threshold is defined by Equation 9 when the first sparse representation format is element-wise flag sparse representation format and the second representation format is compressed sparse row, wherein Th1 represents the high-sparsity threshold, R represents a number of rows of the target matrix data, and C represents a number of columns of the target matrix data.

<Equation 9>

$$Th_1 = 1 - \frac{C - \log_2(R)}{C \times \log_2(C)}$$

(Appendix 12)

The control method according to appendix 10 or 11, wherein the low-sparsity threshold is defined by Equation 10 when the first sparse representation format is element-wise flag sparse representation format, wherein Th2 represents the low-sparsity threshold, B represents a number of bits used to represent each data element of the target matrix data.

<Equation 10>

$$Th_2 = 1 / B$$

(Appendix 13)

The control method according to any one of appendices 8 to 12, further comprising: acquiring one-dimensional array data; dividing the one-dimensional array into a plurality of rows or columns; and generating the input matrix data information includes the plurality of rows or columns,

wherein the acquiring of the input matrix data information acquires the input matrix data information generated from the one-dimensional array data.

(Appendix 14)

The control method according to any one of appendices 8 to 13, further comprising: acquiring three-or-more-dimensional array data, retrieving a plurality of matrix data from the three-or-more-dimensional array data, and generating a plurality pieces of the input matrix data information each of which includes respective ones of the retrieved matrix data,

wherein the acquiring of the input matrix data information acquires the input matrix data information generated from the three-or-more-dimensional array data.

(Appendix 15)

A program that causes a computer to execute:

acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format;

calculating sparsity of the target matrix data;

selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and

outputting output matrix data information in which the target matrix data is represented by in the selected representation format.

(Appendix 16)

The program according to appendix 15, wherein the selecting of the representation format includes:

determining whether or not the calculated sparsity is smaller than low-sparsity threshold;

selecting the dense representation format when it is determined that the calculated sparsity is smaller than the low-sparsity threshold;

determining whether or not the calculated sparsity is smaller than high-sparsity threshold when it is determined that the calculated sparsity is not smaller than the low-sparsity threshold, the high-sparsity threshold being smaller than the low-sparsity threshold;

selecting a first sparse representation format when it is determined that the calculated sparsity is smaller than the high-sparsity threshold; and

selecting a second sparse representation format when it is determined that the calculated sparsity is not smaller than the high-sparsity threshold.

(Appendix 17)

The program according to appendix 16,

wherein the high-sparsity threshold is defined by comparing the number of bits used to represent the target matrix data in the first sparse representation format with the number of bits used to represent the target matrix data in the second sparse representation format, and

wherein the low-sparsity threshold is defined by comparing the number of bits used to represent the target matrix data in the dense representation format with the number of bits used to represent the target matrix data in the first sparse representation format.

(Appendix 18)

The program according to appendix 17, wherein the high-sparsity threshold is defined by Equation 11 when the first sparse representation format is element-wise flag sparse representation format and the second sparse representation format is compressed sparse row, wherein Th1 represents the high-sparsity threshold, R represents a number of rows of the target matrix data, and C represents a number of columns of the target matrix data.

<Equation 11>

$$Th_1 = 1 - \frac{C - \log_2(R)}{C \times \log_2(C)}$$

(Appendix 19)

The program according to appendix 17 or 18, wherein the low-sparsity threshold is defined by Equation 12 when the first sparse representation format is element-wise flag sparse representation format, wherein Th2 represents the low-sparsity threshold, B represents a number of bits used to represent each data element of the target matrix data.

<Equation 12>

$$Th_2 = 1 / B$$

(Appendix 20)

The program according to any one of appendices 15 to 19, causing the computer to further execute: acquiring one-dimensional array data; dividing the one-dimensional array into a plurality of rows or columns; and generating the input matrix data information includes the plurality of rows or columns,

wherein the acquiring of the input matrix data information acquires the input matrix data information generated from the one-dimensional array data.

(Appendix 21)

The program according to any one of appendices 15 to 20, causing the computer to further execute: acquiring three-or-more-dimensional array data, retrieving a plurality of matrix data from the three-or-more-dimensional array data, and generating a plurality pieces of the input matrix data information each of which includes respective ones of the retrieved matrix data,

wherein the acquiring of the input matrix data information acquires the input matrix data information generated from the three-or-more-dimensional array data.

Claims

[Claim 1] An information processing apparatus comprising:

an acquiring unit acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format;

a sparsity calculation unit calculating sparsity of the target matrix data;

a selection unit selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and

an output unit outputting output matrix data information in which the target matrix data is represented in the selected representation format.

[Claim 2] The information processing apparatus according to claim 1, wherein the selection unit performs:

determining whether or not the calculated sparsity is greater than low-sparsity threshold;

selecting the dense representation format when it is determined that the calculated sparsity is smaller than the low-sparsity threshold;

determining whether or not the calculated sparsity is smaller than high-sparsity threshold when it is determined that the calculated sparsity is not smaller than the low-sparsity threshold, the high-sparsity threshold being greater than the low-sparsity threshold;

selecting a first sparse representation format when it is determined that the calculated sparsity is smaller than the high-sparsity threshold; and

selecting a second sparse representation format when it is determined that the calculated sparsity is not smaller than the high-sparsity threshold.

[Claim 3] The information processing apparatus according to claim 2,

wherein the high-sparsity threshold is defined by comparing the number of bits used to represent the target matrix data in the first sparse representation format with the number of bits used to represent the target matrix data in the second sparse representation format, and

wherein the low-sparsity threshold is defined by comparing the number of bits used to represent the target matrix data in the dense representation format with the number of bits used to represent the target matrix data in the first sparse representation format.

[Claim 4] The information processing apparatus according to claim 3, wherein the high-sparsity threshold is defined by Equation 1 when the first sparse representation format is element-wise flag sparse representation format and the second sparse representation format is compressed sparse row, wherein Th1 represents the high-sparsity threshold, R represents a number of rows of the target matrix data, and C represents a number of columns of the target matrix data.

<Equation 1>

$$Th_1 = 1 - \frac{C - log_2(R)}{C \times log_2(C)}$$

[Claim 5] The information processing apparatus according to claim 3 or 4, wherein the low-sparsity threshold is defined by Equation 2 when the first sparse representation format is element-wise flag sparse representation format, wherein Th2 represents the low-sparsity threshold, B represents a number of bits used to represent each data element of the target matrix data.

<Equation 2>

$$Th_2 = 1 / B$$

[Claim 6] The information processing apparatus according to any one of claims 1 to 5, further comprising a conversion unit that acquires one-dimensional array data, divides the one-dimensional array into a plurality of rows or columns, and generates the input matrix data information includes the plurality of rows or columns,

wherein the acquisition unit acquires the input matrix data generated by the conversion unit.

[Claim 7] The information processing apparatus according to any one of claims 1 to 6, further comprising a conversion unit that acquires three-or-more-dimensional array data, retrieves a plurality of matrix data from the three-or-more-dimensional array data, and generates a plurality pieces of the input matrix data information each of which includes respective ones of the retrieved matrix data,

wherein the acquisition unit acquires the plurality pieces of the input matrix data information generated by the conversion unit.

[Claim 8] A control method to be performed by a computer, the method comprising:

acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format;

calculating sparsity of the target matrix data;

selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and

outputting output matrix data information in which the target matrix data is represented by in the selected representation format.

[Claim 9] A program that causes a computer to execute:

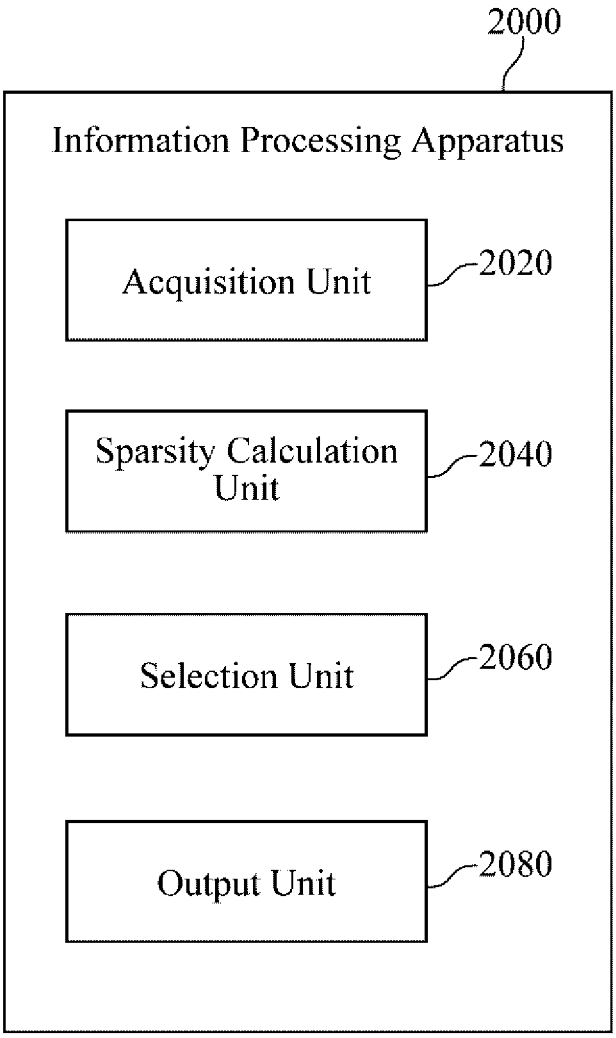
acquiring input matrix data information that represents target matrix data in a dense representation format or a sparse representation format, the target matrix data being represented with all data elements of the target matrix data when the target matrix data is represented in the dense representation formats, the target matrix data being represented with non-zero data elements of the target matrix data when the target matrix data is represented in a sparse representation format;

calculating sparsity of the target matrix data;

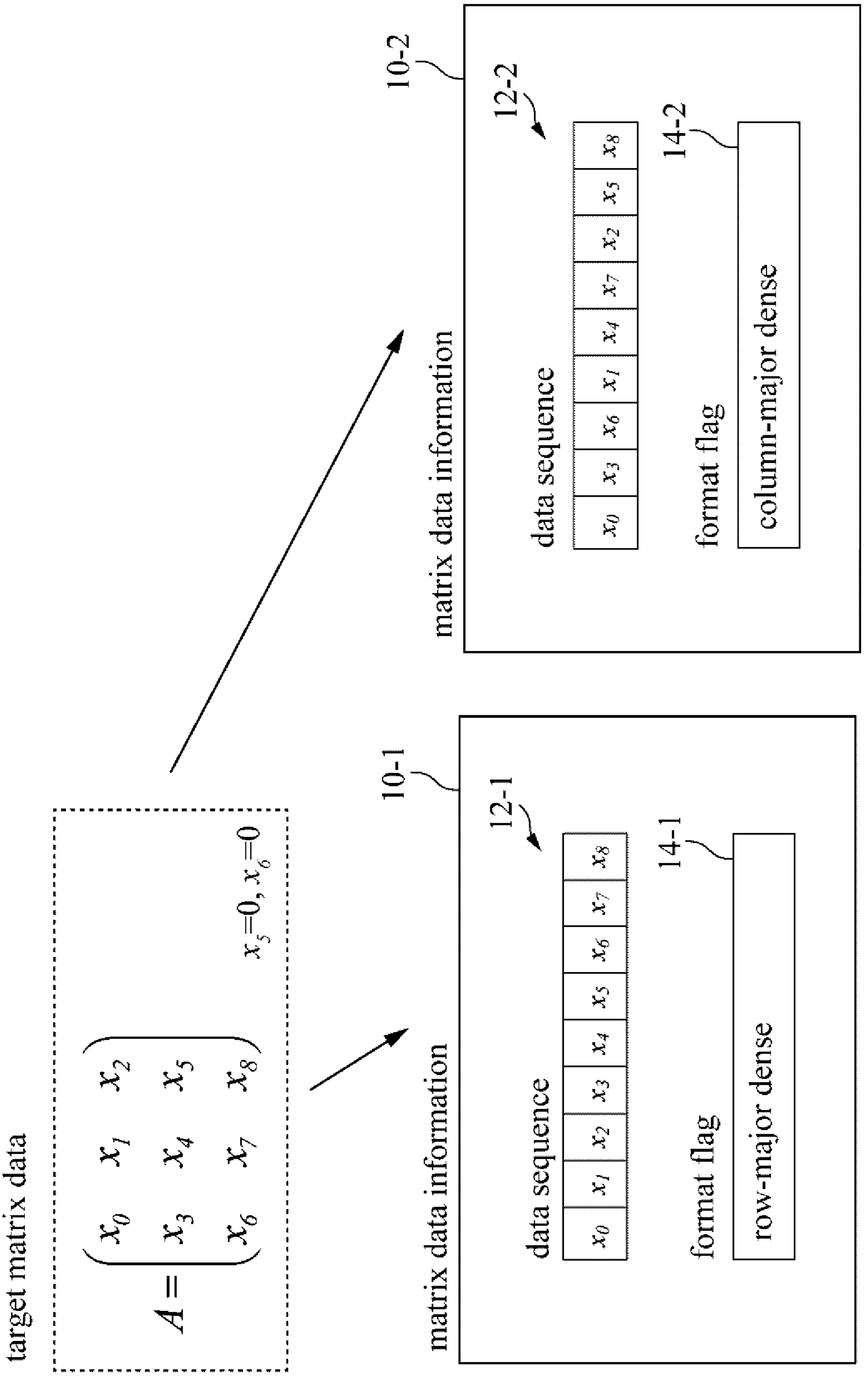
selecting one of a plurality of representation formats based on the calculated sparsity, the plurality of representation formats including the dense representation format and at least two types of sparse representation formats; and

outputting output matrix data information in which the target matrix data is represented by in the selected representation format.

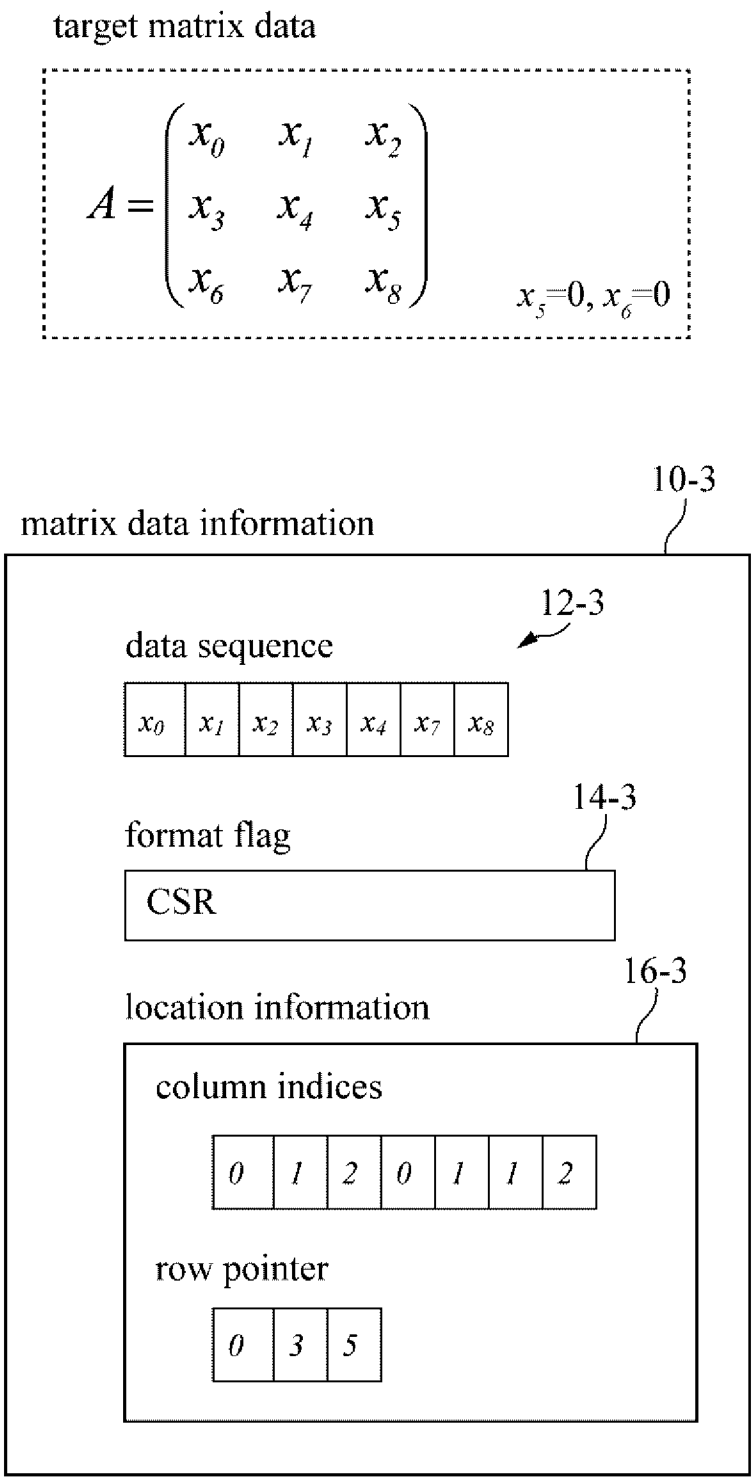
[Fig. 1]


2000
Information Processing Apparatus
Acquisition Unit
2020
Sparsity Calculation Unit
2040
Selection Unit
2060
Output Unit
2080

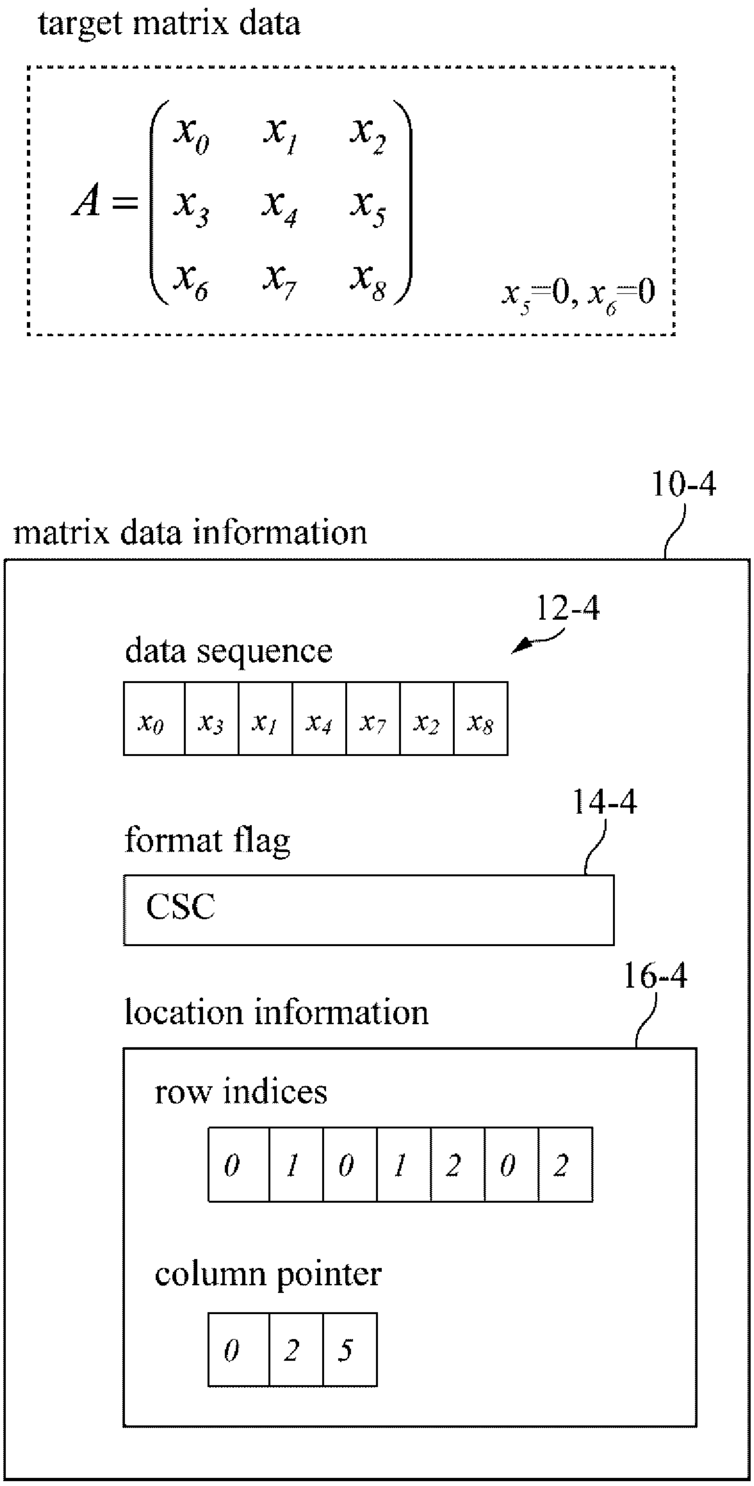
[Fig. 2]


target matrix data
$A = \begin{pmatrix} x_0 & x_1 & x_2 \\ x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 \end{pmatrix}$
$x_5=0, x_6=0$
10-1
matrix data information
12-1
data sequence
x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8
14-1
format flag
row-major dense
10-2
matrix data information
12-2
data sequence
x_0 x_3 x_6 x_1 x_4 x_7 x_2 x_5 x_8
14-2
format flag
column-major dense

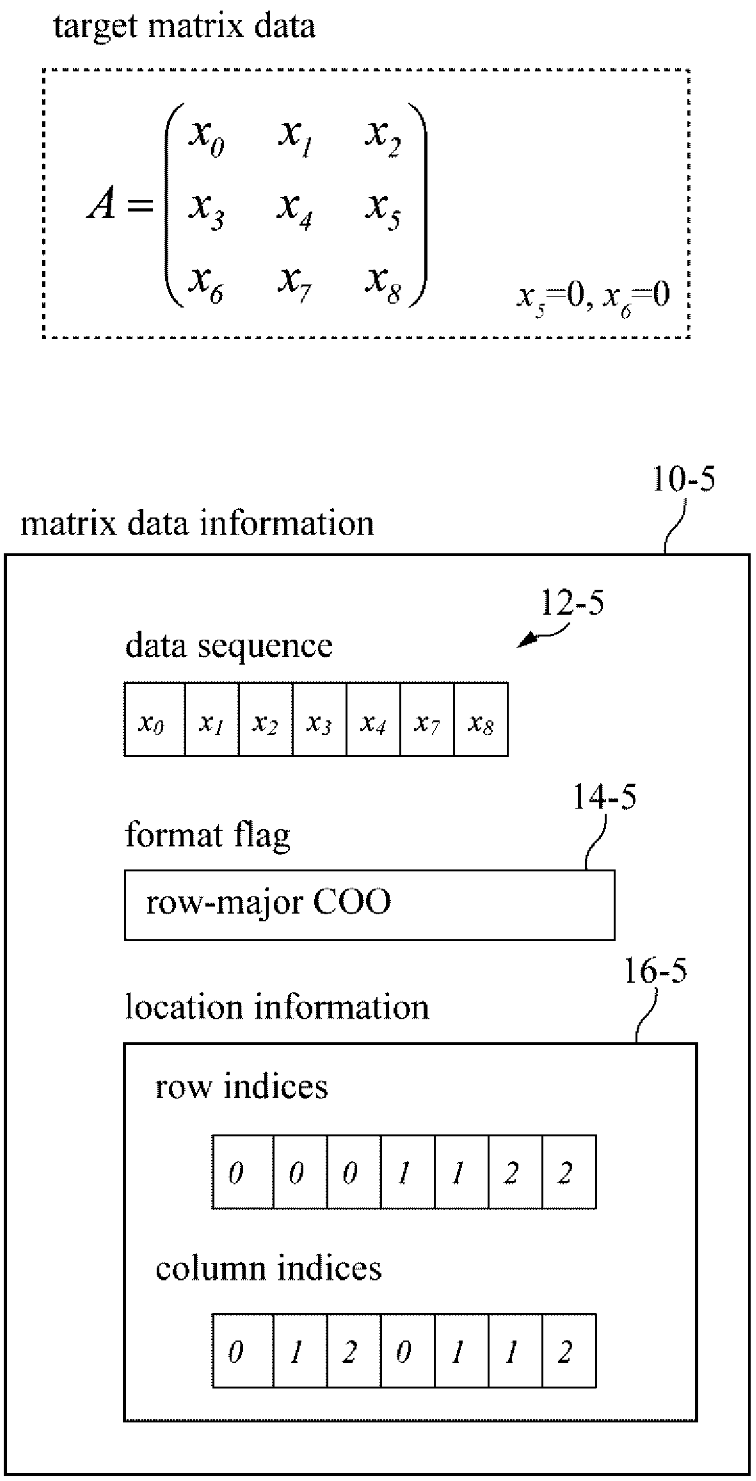
[Fig. 3]


target matrix data
$A = \begin{pmatrix} x_0 & x_1 & x_2 \\ x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 \end{pmatrix}$
$x_5=0, x_6=0$
10-3
matrix data information
12-3
data sequence
x_0 x_1 x_2 x_3 x_4 x_7 x_8
14-3
format flag
CSR
16-3
location information
column indices
0 1 2 0 1 1 2
row pointer
0 3 5

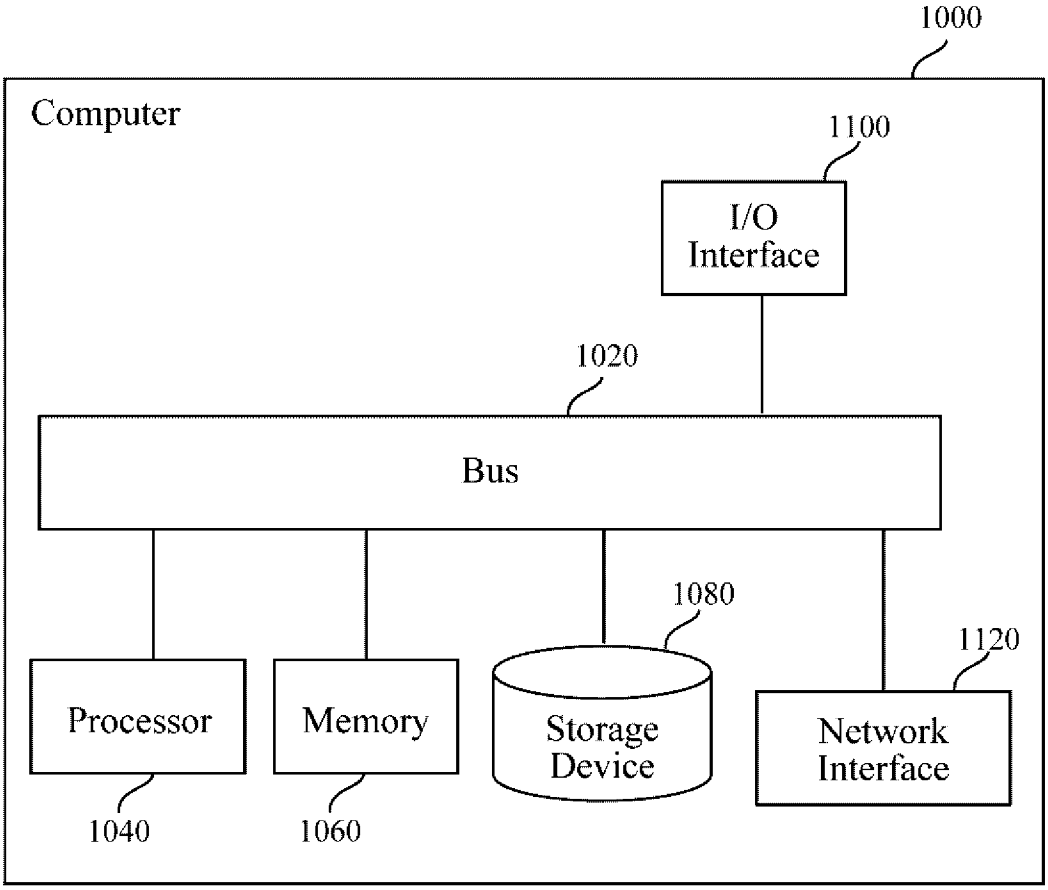
[Fig. 4]


target matrix data
$A = \begin{pmatrix} x_0 & x_1 & x_2 \\ x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 \end{pmatrix}$
$x_5=0, x_6=0$
10-4
matrix data information
12-4
data sequence
x_0 x_3 x_1 x_4 x_7 x_2 x_8
14-4
format flag
CSC
16-4
location information
row indices
0 1 0 1 2 0 2
column pointer
0 2 5

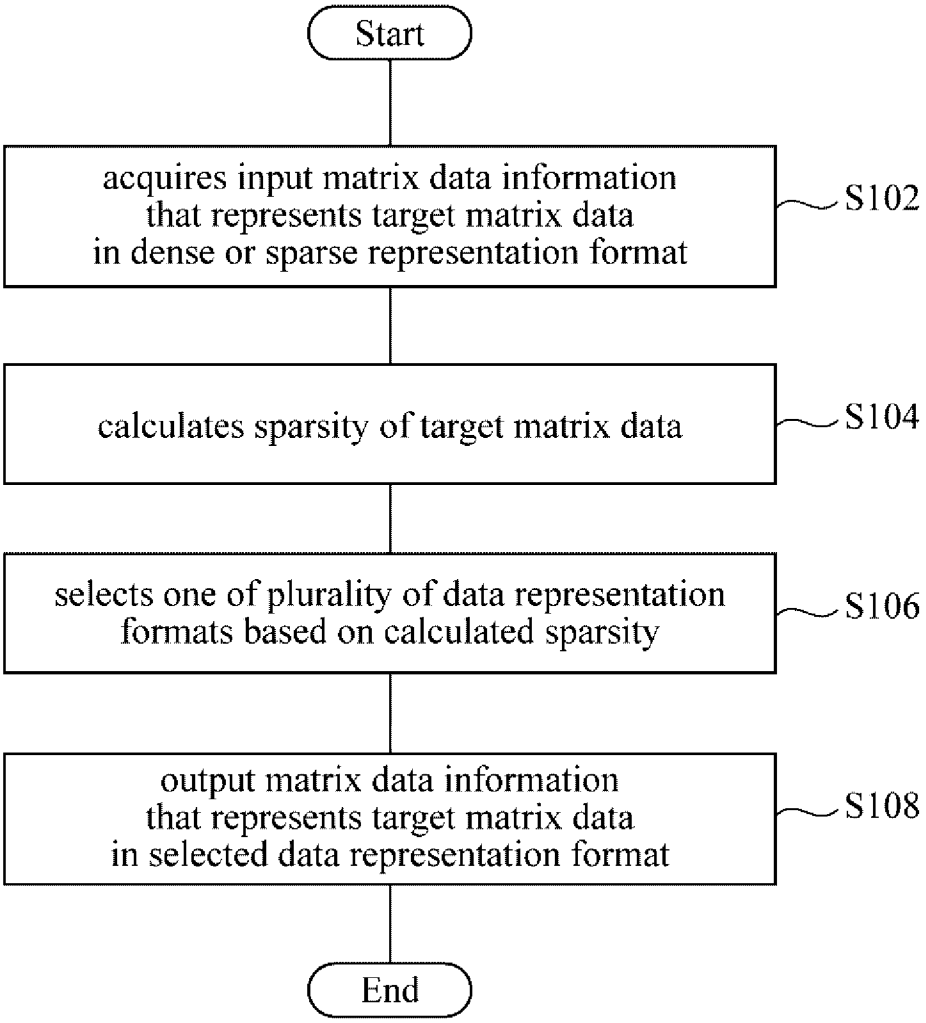
[Fig. 5]


target matrix data
$A = \begin{pmatrix} x_0 & x_1 & x_2 \\ x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 \end{pmatrix}$
$x_5=0, x_6=0$
10-5
matrix data information
12-5
data sequence
x_0 x_1 x_2 x_3 x_4 x_7 x_8
14-5
format flag
row-major COO
16-5
location information
row indices
0 0 0 1 1 2 2
column indices
0 1 2 0 1 1 2

[Fig. 6]


1000
Computer
1100
I/O Interface
1020
Bus
1080
1120
Processor
Memory
Storage Device
Network Interface
1040
1060

[Fig. 7]


Start
acquires input matrix data information that represents target matrix data in dense or sparse representation format
S102
calculates sparsity of target matrix data
S104
selects one of plurality of data representation formats based on calculated sparsity
S106
output matrix data information that represents target matrix data in selected data representation format
S108
End

[Fig. 8]

$$A = \begin{pmatrix} 3 & 4 & 9 & 1 & 1 \\ 4 & 5 & 2 & 3 & 4 \\ 5 & 3 & 8 & 1 & 8 \\ 0 & 5 & 5 & 4 & 2 \\ 8 & 7 & 0 & 1 & 1 \end{pmatrix}$$

No. of Zero = 2
Total No. = 25

Sparsity = 2/25 = 0.08

$$B = \begin{pmatrix} 0 & 4 & 9 & 1 & 1 \\ 4 & 5 & 2 & 0 & 4 \\ 5 & 0 & 8 & 1 & 8 \\ 0 & 5 & 5 & 4 & 0 \\ 8 & 7 & 0 & 1 & 1 \end{pmatrix}$$

No. of Zero = 6
Total No. = 25

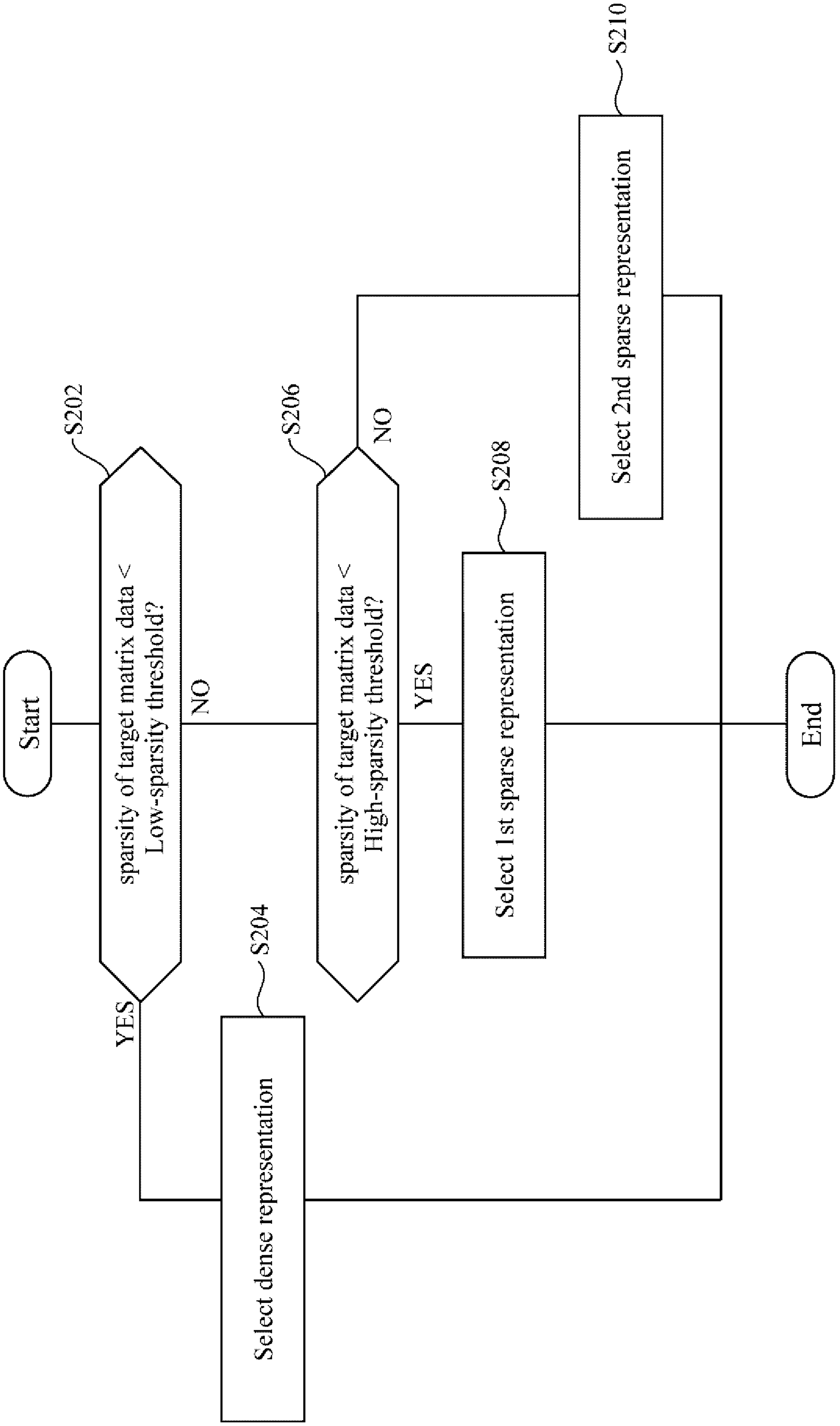
Sparsity = 6/25 = 0.24

$$C = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

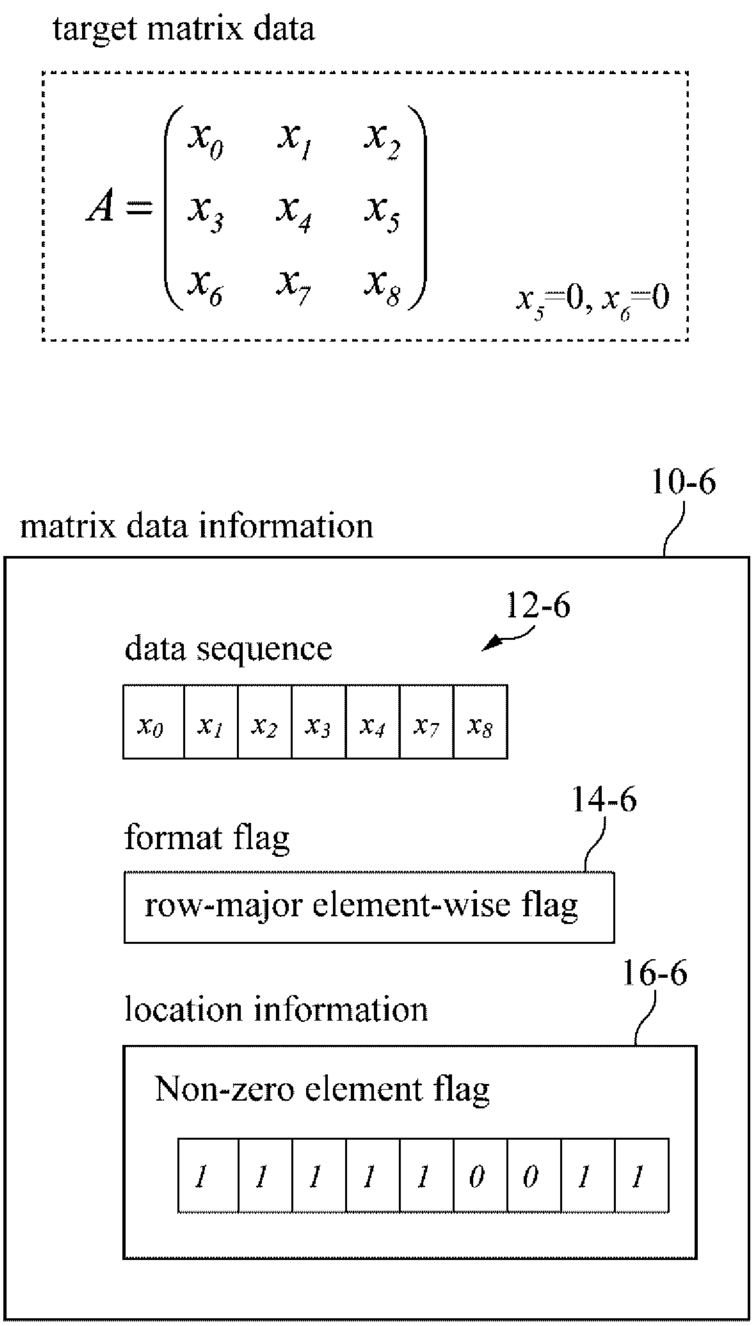
No. of Zero = 48
Total No. = 49

Sparsity = 48/49 = 0.98

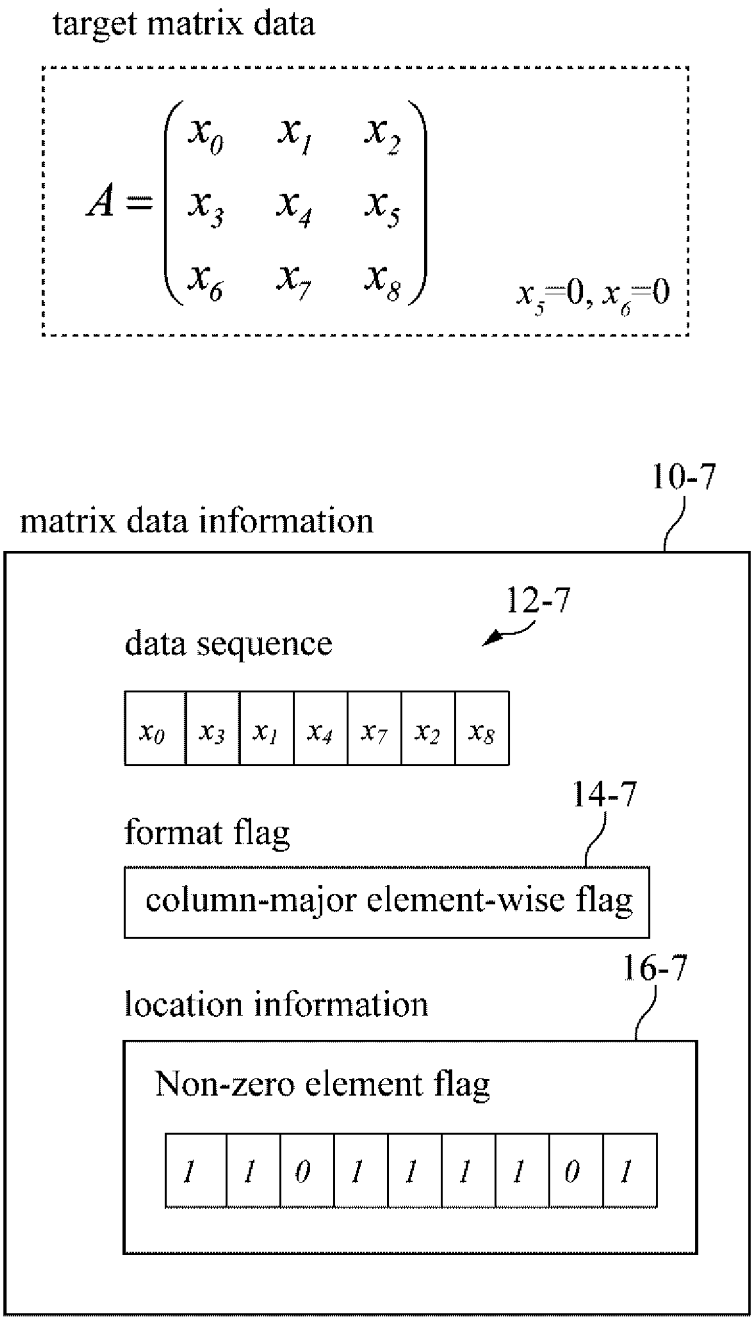
[Fig. 9]


Start
sparsity of target matrix data < Low-sparsity threshold?
S202
NO
YES
Select dense representation
S204
sparsity of target matrix data < High-sparsity threshold?
S206
YES
NO
Select 1st sparse representation
S208
Select 2nd sparse representation
S210
End

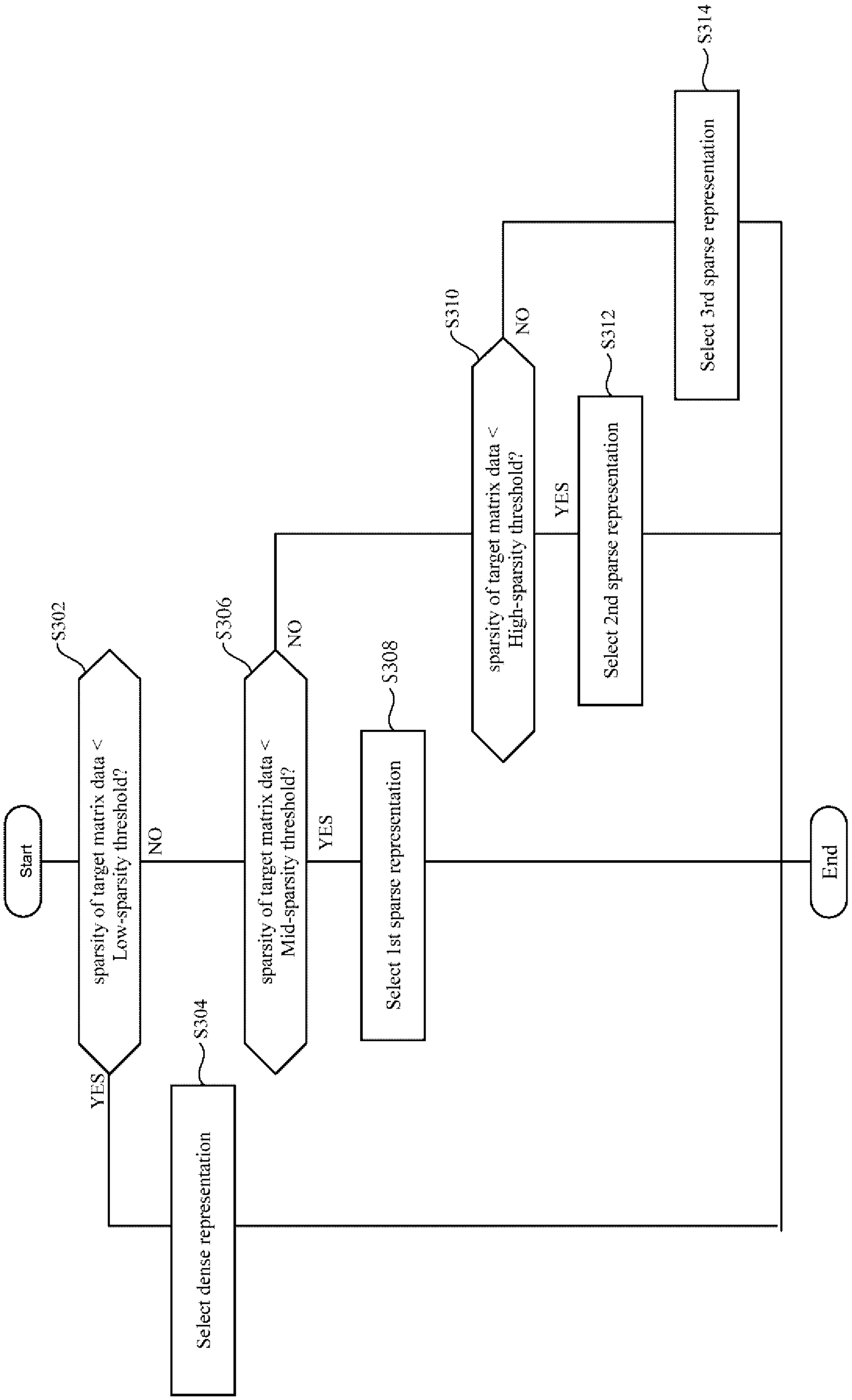
[Fig. 10]


target matrix data
$A = \begin{pmatrix} x_0 & x_1 & x_2 \\ x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 \end{pmatrix}$
$x_5=0, x_6=0$
matrix data information
10-6
data sequence
12-6
x_0 x_1 x_2 x_3 x_4 x_7 x_8
format flag
14-6
row-major element-wise flag
location information
16-6
Non-zero element flag
1 1 1 1 1 0 0 1 1

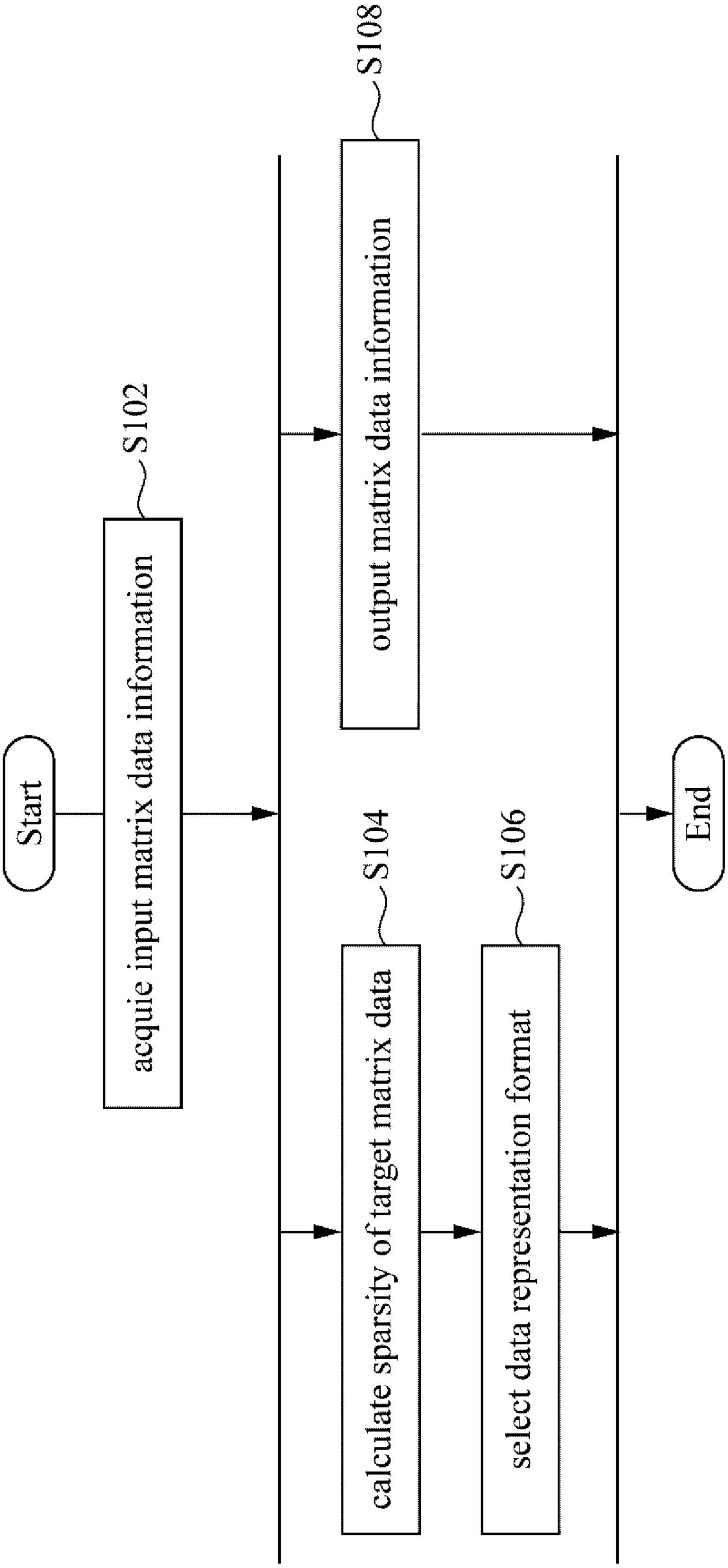
[Fig. 11]


target matrix data
$A = \begin{pmatrix} x_0 & x_1 & x_2 \\ x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 \end{pmatrix}$
$x_5=0, x_6=0$
matrix data information
10-7
12-7
data sequence
x_0 x_3 x_1 x_4 x_7 x_2 x_8
14-7
format flag
column-major element-wise flag
16-7
location information
Non-zero element flag
1 1 0 1 1 1 1 0 1

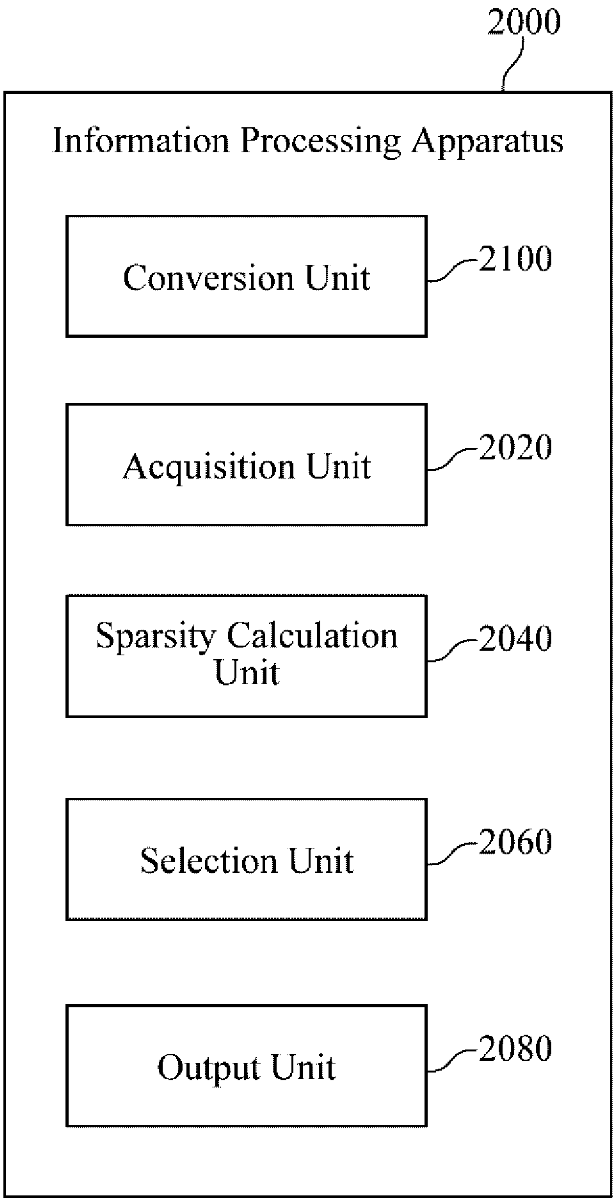
[Fig. 12]


Start
S302
sparsity of target matrix data <
Low-sparsity threshold?
YES
NO
S304
Select dense representation
S306
sparsity of target matrix data <
Mid-sparsity threshold?
YES
NO
S308
Select 1st sparse representation
S310
sparsity of target matrix data <
High-sparsity threshold?
YES
NO
S312
Select 2nd sparse representation
S314
Select 3rd sparse representation
End

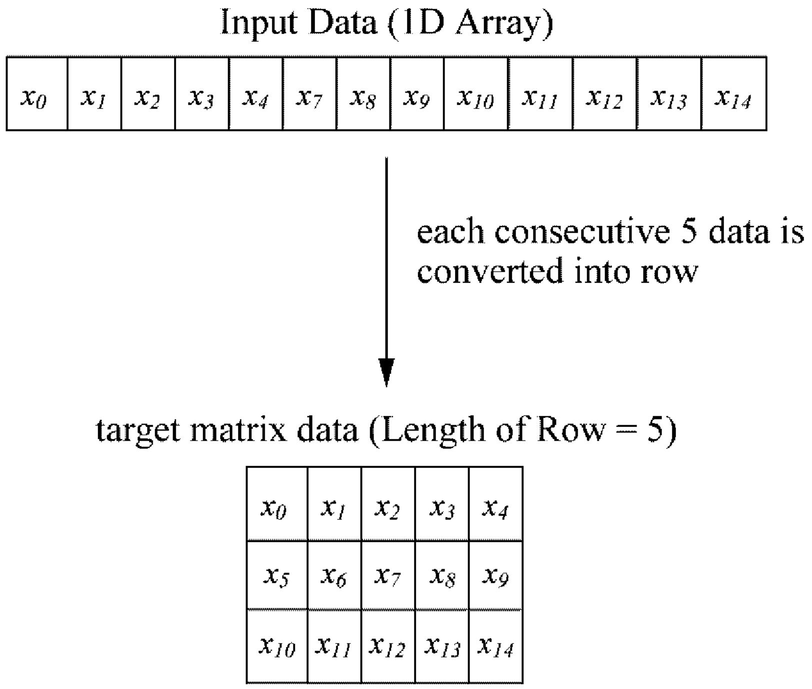
[Fig. 13]


Start
acquie input matrix data information
S102
calculate sparsity of target matrix data
S104
select data representation format
S106
output matrix data information
S108
End

[Fig. 14]


2000
Information Processing Apparatus
Conversion Unit
2100
Acquisition Unit
2020
Sparsity Calculation Unit
2040
Selection Unit
2060
Output Unit
2080

[Fig. 15]



Input Data (1D Array)

x_0	x_1	x_2	x_3	x_4	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}

each consecutive 5 data is converted into row

target matrix data (Length of Row = 5)

x_0	x_1	x_2	x_3	x_4
x_5	x_6	x_7	x_8	x_9
x_{10}	x_{11}	x_{12}	x_{13}	x_{14}

INTERNATIONAL SEARCH REPORT

International application No
PCT/JP2017/024459

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F17/16
ADD.

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X Y	US 2015/113031 A1 (REINWALD BERTHOLD [US] ET AL) 23 April 2015 (2015-04-23) paragraphs [0006], [0022] - paragraph [0024]; figures 1,6 -----	1,8,9 2-7
Y	US 2016/259826 A1 (ACAR EMRAH [US] ET AL) 8 September 2016 (2016-09-08) paragraph [0103] - paragraph [0105]; figures 5, 6 -----	2-7

☐ 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 8 March 2018	Date of mailing of the international search report 15/03/2018
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Virnik, Elena

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/JP2017/024459

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2015113031 A1	23-04-2015	US 2015113031 A1 US 2016217066 A1 US 2016364327 A1	23-04-2015 28-07-2016 15-12-2016
US 2016259826 A1	08-09-2016	NONE	