

SPECIFICATION

NEURAL NETWORK OPERATION METHOD AND APPARATUS, ELECTRONIC DEVICE AND STORAGE MEDIUM

5

TECHNICAL FIELD

[0001] The present disclosure relates to the technical field of deep learning, and particularly to a neural network operation method and apparatus, an electronic device and a storage medium.

10

BACKGROUND

[0002] With the development of technologies, large-scale neural networks have been applied to various scenarios such as scenarios of automatic driving, image identifying, etc. After being constructed, the large-scale neural network may be operated by a computing device.

15

SUMMARY

[0003] In view of this, the present disclosure at least provides a neural network operation method and apparatus, an electronic device and a storage medium.

20

[0004] In a first aspect, there is provided a neural network operation method, comprising: determining a network layer to be processed in a target neural network; determining a target operator and a target chunking strategy corresponding to the network layer to be processed in the target neural network from a plurality of determined operators and a plurality of determined chunking strategies, wherein the plurality of operators are configured to implement corresponding functions of the network layer to be processed, and the plurality of chunking strategies match with operational requirements of a computing device for operating the target neural network; and operating the target neural network containing the target operator based on the target chunking strategy corresponding to the network layer to be processed.

25

[0005] In the above method, after a network layer to be processed in a target neural network is determined, a target operator and a target chunking strategy corresponding to the network layer to be processed may be determined from the plurality of determined operators and the plurality of chunking strategies; and because the chunking strategy meets the operational requirement of the computing device, when the target neural network containing the target operator is operated based on the target chunking strategy corresponding to the

30

35

network layer to be processed, the operational requirement of the computing device can be met. At the same time, because the target chunking strategy may chunk the parameter data for the target operator corresponding to the matched network layer to be processed, the resource consumption situation for operating the network layer to be processed based on the chunked parameter data is minimal, for example, the resource consumption situation may be represented by a total computational overhead, that is, when the operational requirement of the computing device is met, the efficiency of operating the target neural network containing the target operator based on the respective target chunking strategy corresponding to at least one network layer to be processed is high.

[0006] In a possible implementation, the chunking strategy is configured to perform chunking on parameter data of the target operator corresponding to the network layer to be processed; and among the plurality of chunking strategies, resource consumption for operating the network layer to be processed based on parameter data obtained by chunking the parameter data of the target operator with the target chunking strategy is minimal.

[0007] In a possible implementation, based on a plurality of network layers to be processed, said determining the target operator and the target chunking strategy corresponding to the network layer to be processed in the target neural network from a plurality of determined operators and a plurality of determined chunking strategies includes: for any network layer to be processed in the target neural network, determining one or more target candidate operators corresponding to the network layer to be processed from the plurality of operators, and determining one or more target candidate chunking strategies matched with the target candidate operator from the plurality of chunking strategies; and in response to a plurality of target candidate operators and/or a plurality of target candidate chunking strategies corresponding to any network layer to be processed, determining the target operator and the target chunking strategy corresponding to the network layer to be processed based on the target candidate operators and the target candidate chunking strategies respectively corresponding to each network layer to be processed.

[0008] In the above implementation, firstly, the target candidate operator corresponding to each network layer to be processed and the target candidate chunking strategy matched with the target candidate operator may be determined respectively, so that the local optimization of the target candidate operator and the target candidate chunking strategy of each network layer to be processed is implemented. Further, in a case where there are a plurality of target candidate operators and/or a plurality of target candidate chunking strategies corresponding to any network layer to be processed, the target operator and the target chunking strategy corresponding to each network layer to be processed are determined based on the target

candidate operators and the target candidate chunking strategies corresponding to various network layers to be processed, so that the global optimization of the target candidate operators and the target candidate chunking strategies of various network layers to be processed is implemented.

5 **[0009]** In a possible implementation, determining the target operator and the target chunking strategy corresponding to each network layer to be processed based on the target candidate operators and the target candidate chunking strategies respectively corresponding to each network layer to be processed includes: determining a plurality of test networks corresponding to the target neural network based on the target candidate operator
10 corresponding to each network layer to be processed and the target candidate chunking strategy corresponding to the target candidate operator, wherein each test network in the plurality of test networks includes one target candidate operator corresponding to the network layer to be processed and one target candidate chunking strategy matched with the target candidate operator; operating the plurality of test networks respectively to obtain a plurality of test results,
15 wherein any test network in the plurality of test networks corresponds to one test result; selecting a target test network from the plurality of test networks based on the plurality of test results; and determining the target candidate operator and the target candidate chunking strategy of the network layer to be processed in the target test network as the target operator and the target chunking strategy corresponding to the network layer to be processed in the
20 target neural network.

[0010] In the above implementation, a plurality of test networks corresponding to the target neural network are determined based on the target candidate operators corresponding to the respective network layers to be processed and the target candidate chunking strategies corresponding to the target candidate operators; then a plurality of test networks are operated
25 by a computing device to determine the test result of each test network; and the target test network is determined based on the test result, for example, when the test result is the computational overhead, the test network with minimal computational overhead may be selected as the target test network, and the target candidate operators and the target candidate chunking strategies for each network layer to be processed in the target test network are
30 determined as the target operators and the target chunking strategies corresponding to the respective network layers to be processed respectively in the target neural network, so that the global optimization of the target operators and the target chunking strategies is implemented.

[0011] In a possible implementation, determining one or more target candidate operators corresponding to the network layer to be processed from the plurality of operators, and
35 determining one or more target candidate chunking strategies matched with the target candidate

operator from the plurality of chunking strategies for any network layer to be processed in the target neural network includes: determining, for the network layer to be processed, one or more first candidate operators from the plurality of operators; and selecting one or more target candidate operators corresponding to the network layer to be processed from the first candidate operators and the target candidate chunking strategies corresponding to the target candidate operators from the plurality of chunking strategies, based on a resource consumption situation for the first candidate operator in each chunking strategy of the plurality of chunking strategies.

[0012] Here, after one or more first candidate operators corresponding to the network layer to be processed are determined for each network layer to be processed, one or more target candidate operators corresponding to the network layer to be processed and one or more target candidate chunking strategies corresponding to the target candidate operators may be selected from the first candidate operators and the plurality of chunking strategies based on the resource consumption situation of the first candidate operator in each chunking strategy in plurality of chunking strategies, for example, the first candidate operator and chunking strategy with minimal resource consumption may be selected as the target candidate operator and the target candidate chunking strategy to implement a local optimization of the target candidate operator and the target candidate chunking strategy corresponding to each network layer to be processed.

[0013] In a possible implementation, the resource consumption situation is represented by a computational overhead value, and the computational overhead value for the first candidate operator in each chunking strategy is determined by: determining a restricted scenario for the first candidate operator with a preset size, wherein the restricted scenario is determined based on a computing time cost and a transmission time cost for a data capacity corresponding to the first candidate operator with the preset size; in response to the restricted scenario being a bandwidth restricted scenario, determining a total direct memory access (DMA) data transmission amount, a number of DMA tasks, and data conversion overhead corresponding to the first candidate operator in the chunking strategy based on a result of chunking by the chunking strategy; and determining the computational overhead value for the first candidate operator in the chunking strategy based on the total DMA data transmission amount, the number of the DMA tasks, the data conversion overhead, a DMA rate corresponding to the computing device, and DMA task overhead, wherein the data conversion overhead is time consumed in performing a data arrangement conversion on input data corresponding to the first candidate operator based on to a target data arrangement corresponding to the first candidate operator; in response to the restricted scenario being a computation restricted scenario, determining a computing time cost for the parameter data corresponding to the first candidate

operator in the chunking strategy, number of operator calls for the first candidate operator, a total amount of initial data transmission, the number of DMA tasks, and the data conversion overhead based on a result of chunking by the chunking strategy, and determining the computational overhead value for the first candidate operator in the chunking strategy based on the computing time cost, the number of operator calls, the total amount of initial data transmission, the data conversion overhead, the DMA task overhead, the number of the DMA tasks, and the DMA rate corresponding to the computing device.

[0014] In the above implementation, a restricted scenario corresponding to the first candidate operator with a preset size may be determined, and different restricted scenarios correspond to different determining ways of the computational overhead value. For example, in a bandwidth restricted scenario, the computational overhead value may be determined based on a total DMA data transmission amount, the number of DMA tasks, data conversion overhead, a DMA rate, and DMA task overhead; and in a computation restricted scenario, the computational overhead value may be determined based on a computing time cost, the number of operator calls, a total transmission amount of initial data, data conversion overhead, DMA task overhead, the number of DMA tasks, and the DMA rate.

[0015] In a possible implementation, the selecting one or more target candidate operators corresponding to the network layer to be processed and one or more target candidate strategies corresponding to the target candidate operators from the first candidate operators and the plurality of chunking strategies based on the resource consumption situation for the first candidate operator in each chunking strategy in the plurality of chunking strategies includes: selecting a target resource consumption situation meeting a predetermined condition from a plurality of resource consumption situations corresponding to the first candidate operator; wherein one first candidate operator corresponds to one resource consumption situation in one chunking strategy; determining the chunking strategy corresponding to the target resource consumption situation as the candidate chunking strategy, operating the network layer to be processed containing a second candidate operator corresponding to the target resource consumption situation based on the candidate chunking strategy, and determining a test result corresponding to the candidate chunking strategy and the second candidate operator; and determining one or more target candidate operators corresponding to the network layer to be processed and the one or more target candidate chunking strategies corresponding to the target candidate operators based on the test result.

[0016] By the above method, the resource consumption situation may be used first to select the second candidate operator from the first candidate operators and the candidate chunking strategy matched with the second candidate operator from the chunking strategies;

and testing the second candidate operator and the candidate chunking strategy, and then utilizing the test result to determine at least one target candidate operator and target candidate chunking strategy corresponding to the network layer to be processed, so that the at least one determined target candidate operator and target candidate chunking strategy corresponding to the network layer to be processed are preferred options.

[0017] In a possible implementation, prior to selecting one or more target candidate operators corresponding to the network layer to be processed from the first candidate operators and the target candidate chunking strategies corresponding to the target candidate operators from the plurality of chunking strategies, the method further includes: aligning the parameter data corresponding to the first candidate operator based on determined minimal granularity information corresponding to the target neural network to obtain the aligned parameter data corresponding to the first candidate operator; wherein the minimal granularity information includes minimal granularities corresponding to the parameter data in different dimensions; and in different dimensions the size of the aligned parameter data is of an integer multiple of the minimal granularity in the respective dimension indicated by the minimal granularity information.

[0018] Here, aligning may be performed on parameter data corresponding to each first candidate operator based on minimal granularity information corresponding to the target neural network to obtain the aligned parameter data corresponding to the first candidate operator, and the size of the aligned parameter data in different dimensions is of an integer multiple of the minimum granularity in the corresponding dimension indicated by the minimum granularity information, so that the probability of parameter data loss when the target neural network is operated subsequently based on the target chunking strategy may be reduced.

[0019] In a possible implementation, in response to the parameter data including input data and constant data, the plurality of chunking strategies include at least one of: by taking all input data as initial data, chunking the constant data in a specified dimension based on a determined dimension parameter of the constant data to obtain a result of chunking, wherein the initial data is data written into an initial data region allocated by the direct memory access (DMA) task in response to the computing device operating the target neural network; by taking all constant data as the initial data, chunking the input data in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking; by taking part of the input data as the initial data, chunking the constant data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking, wherein a target size of the part of the input data is determined based on the minimal granularity of the first dimension of the input data; and by taking part of the constant

data as the initial data, chunking the input data in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking; wherein the target size of the part of the constant data is determined based on the minimal granularity of the first dimension of the constant data.

5 **[0020]** In a possible implementation, chunking, by taking part of the input data as the initial data, the constant data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking includes: determining the target size of the part of the input data based on i times of the minimal granularity of the first dimension of the input data; by taking respectively part of the input data of the target size as
10 the initial data, chunking the constant data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking; where i is a positive integer that makes the data capacity of the part of the input data and the data capacity of a constant data block determined based on the minimum granularity of the dimension parameter of the constant data meet a memory requirement of the computing device after the target size of
15 the part of the input data is determined.

[0021] In a possible implementation, chunking, by taking part of the input data as the initial data, the constant data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking includes: determining the target size of the part of the constant data based on j times of the minimal granularity of the first
20 dimension of the constant data; and taking respectively the part of the constant data of the target size as the initial data, and chunking the input data in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking; where j is a positive integer that makes the data capacity of the part of the constant data and the data capacity of the input data block determined based on the minimum granularity of the input data
25 for the dimension parameters meet the memory requirement of the computing device after the target size of the part of the constant data is determined.

[0022] In a possible implementation, a plurality of chunking strategies are set here, so that each network layer to be processed can select a better target operator and a better target chunking strategy matched with the target operator.

30 **[0023]** In a possible implementation, in response to the specified dimension being one dimension and the dimension parameter including the first dimension, by taking the constant data and the input data respectively as the target data, performing a one-dimensional chunking on the target data based on the determined first dimension of the target data to obtain the result of chunking includes: determining k times of the minimal granularity corresponding to the first
35 dimension of the target data as a target chunking size, performing the one-dimensional

chunking on the target data by the first dimension based on the target chunking size to obtain a plurality of target data blocks corresponding to the target data, where k is a positive integer; in response to determining that the plurality of target data blocks and the initial data meet a set chunking condition, taking $k+1$ times of the minimal granularity corresponding to the target data in the first dimension as an updated target chunking size, returning to the step of performing the one-dimensional chunking on the target data by the first dimension until determining that the plurality of target data blocks and the initial data do not meet the set chunking condition, and determining k times of the minimal granularity corresponding to the first dimension of the target data as the result of chunking; and in response to the plurality of target data blocks generated with k equal to 1 and the initial data not meeting the set chunking condition, determining the result of chunking as a failure of the one-dimensional chunking.

[0024] By the above method, the target chunking size increases constantly, and the result of chunking with high the memory utilization of the computing device is determined by continuous try, facilitating to reduce the waste of memory resources of the computing device.

[0025] In a possible implementation, in response to the specified dimension being two dimensions, and the dimension parameter including a second dimension, by taking the constant data and the input data respectively as the target data, performing a two-dimensional chunking on the target data based on the determined first dimension and second dimension of the target data to obtain the result of chunking includes: determining y times of the minimal granularity corresponding to the target data in the first dimension as a first target chunking size, and performing the one-dimensional chunking on the target data by the first dimension based on the first target chunking size to obtain a plurality of intermediate data blocks corresponding to the target data, where y is a positive integer; determining x times of the minimal granularity corresponding to the target data in the second dimension as a second target chunking size, and performing the two-dimensional chunking on each intermediate data block by the second dimension based on the second target chunking size to obtain a plurality of target data blocks respectively corresponding to the plurality of intermediate data blocks, where x is a positive integer; in response to determining that the plurality of target data blocks and the initial data meet the set chunking condition, taking $x+1$ times of the minimal granularity corresponding to the target data in the second dimension as an updated second target chunking size, returning to the step of performing the two-dimensional chunking on each intermediate data block by the second dimension based on the second target chunking size until determining that the plurality of target data blocks and the initial data do not meet the set chunking condition, and determining x times of the minimal granularity corresponding to the target data in the second dimension as the result of chunking.

[0026] In a possible implementation, in response to the parameter data corresponding to the network layer to be processed further including output data, determining that the plurality of target data blocks and the initial data meet the set chunking conditions includes: in response to determining that the initial data, the output data, and each target data block meet the memory requirement of the computing device respectively, and that the initial data, the output data, and each target data block meet the DMA transmission requirement of the computing device respectively, determining that the plurality of target data blocks and the initial data meet the set chunking condition.

[0027] By the above method, when the initial data, the output data and each target data block meet the memory requirement of the computing device and the DMA transmission requirement in the computing device, it is determined that a plurality of target data blocks and the initial data satisfy the set chunking condition, thereby ensuring that the chunking strategy is matched with the operational requirement of the computing device.

[0028] For the effects of the following apparatuses and electronic device, please refer to the description of the above methods, which is not repeated here.

[0029] In a second aspect, the present disclosure provides a neural network operation apparatus, which includes: a first determining module configured to determine a network layer to be processed in a target neural network; a second determining module configured to determine a target operator and a target chunking strategy corresponding to the network layer to be processed in the target neural network from a plurality of determined operators and a plurality of chunking strategies, wherein each operator of the plurality of operators is used to implement a corresponding function of the network layer to be processed, and each chunking strategy of the plurality of chunking strategies matches with the operational requirement of a computing device for operating the target neural network; and an operation module configured to operate the target neural network containing the target operator based on the target chunking strategy corresponding to each network layer to be processed.

[0030] In a third aspect, the present disclosure provides an electronic device, which includes a processor, a memory and a bus; the memory stores machine readable instructions executable by the processor; and when the electronic device operates, the processor communicates with the memory through the bus, and the machine readable instructions, when executed by the processor, perform steps of the neural network operation method according to the first aspect or any implementation.

[0031] In a fourth aspect, the present disclosure provides a computer readable storage medium; and the computer readable storage medium has computer program instructions stored thereon, and the computer program instructions, when executed by a processor, perform the

steps of the neural network operation method according to the first aspect or any implementation.

[0032] In a fifth aspect, the present disclosure provides a computer program, which includes computer readable codes, when the computer readable codes are executed on the electronic device, the processor in the electronic device performs the above method.

[0033] To make the above purpose, features and advantages of the present disclosure more apparent and understood, embodiments are listed below and described in detail in conjunction with accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0034] To describe the technical solutions in the embodiments of the present disclosure more clearly, the drawings required to be used in embodiments are presented simply below. The drawings described here are incorporated into the specification and constitute a part of the specification. The drawings illustrate embodiments in conformity with the present disclosure and are used to explain the technical solutions of the present disclosure together with the specification. It should be understood that the following drawings only show some embodiments of the present disclosure, so the embodiments should not be considered as limiting the scope, and for those ordinary skilled in the art, other relevant drawings can also be obtained according to the provided drawings without contributing creative labor.

[0035] FIG. 1 illustrates a flow chart of a neural network operation method according to an embodiment of the present disclosure.

[0036] FIG. 2 illustrates a flow chart of determining a target operator and a target chunking strategy corresponding to a network layer to be processed in a target neural network in the neural network operation method according to an embodiment of the present disclosure.

[0037] FIG. 3 illustrates a flow chart of determining a target candidate operator corresponding to the network layer to be processed from a plurality of operators, and determining a target candidate chunking strategy matched with the target candidate operator from a plurality of chunking strategies in the neural network operation method according to an embodiment of the present disclosure.

[0038] FIG. 4 illustrates a schematic diagram of software and hardware scheduling of a computing device in the neural network operation method according to an embodiment of the present disclosure.

[0039] FIG. 5 illustrates a schematic diagram of architecture of a neural network operation apparatus according to an embodiment of the present disclosure.

[0040] FIG. 6 illustrates a structure diagram of an electronic device according to an

embodiment of the present disclosure.

DETAILED DESCRIPTION

5 **[0041]** To make the purposes, technical solutions and advantages of embodiments of the present disclosure clearer, the technical solutions in the embodiments of the present disclosure are described clearly and completely in conjunction with accompanying drawings in the embodiments of the present disclosure. Apparently, the described embodiments are merely some embodiments of the present disclosure, not all embodiments. Components of the
10 embodiments of the present disclosure, which are generally described and illustrated in the accompanying drawings herein, may be arranged and designed in various different configurations. Therefore, the following detailed description of the embodiments of the present disclosure provided in the accompanying drawings is not intended to limit the scope of the claimed disclosure, but merely represents selected embodiments of the present disclosure. All
15 other embodiments obtained by those skilled in the art based on the embodiments of the present disclosure without creative efforts shall fall within the protection scope of the present disclosure.

[0042] Generally, a computing device that depends on direct memory access (DMA) operation for data transmission is low in data cache efficiency or has no data cache, so that
20 when the computing device is used for inferring large-scale neural networks, due to the limited memory of the computing device, problems such as chunking and scheduling of single-layer tasks of the large-scale neural networks may be encountered.

[0043] When in implementation, and when the computing device operates the large-scale neural networks, an official inference library set by a manufacturer of the computing device
25 may be used to operate the large-scale neural network on the computing device; and however, the official inference library is used for the specific basic neural network, after the basic neural network is optimized by users, the official inference library may be unable to use, or the efficiency of operating the optimized neural network by using the official inference library may be low. The official inference library is an available inference deployment solution, for
30 example, the official inference library may be a cdnn library of ceva dsp.

[0044] Further, for the optimized neural network, secondary development may be performed on the official inference library, so that the developed inference library can be suitable for the optimized neural network; however, the development process is high in cost and low in efficiency; and moreover, the developed inference library is only suitable for the
35 optimized neural network and not suitable for other neural networks, so that the developed

inference library is low in multiplex ratio.

[0045] Therefore, in order to solve the above problem, embodiments of the present disclosure provide a neural network operation method and apparatus, an electronic device and a storage medium.

5 **[0046]** The defects of the above solution are results obtained by the inventor through practice and careful research. Therefore, the discovery process of the above problems and solutions to the above problems proposed below in the present disclosure should be the contributions made by the inventor to the present invention in the process of the present disclosure.

10 **[0047]** It should be noted that the same reference numerals and letters indicate the same items in the following drawings, so once an item is defined in one drawing, the item does not need to be further defined and explained in the following drawings.

[0048] In order to facilitate the understanding of the embodiments of the present disclosure, firstly, a neural network operation method disclosed by the embodiments of the present disclosure is described in detail. An executing main body of the neural network operation method provided by the embodiments of the present disclosure is generally a computer device with certain computing power. The computer device may be a computing device operating the neural network and may also be other computing devices. The computer device, for example, includes: a terminal device, a server or other processing devices; and the terminal device may be user equipment (UE), a mobile device, a user terminal, a terminal, a cellular phone, a cordless telephone, a personal digital assistant (PDA), a handheld device, a computing device, a vehicle-mounted device, a wearable device, etc. In some possible implementations, the neural network operation method may be implemented in a method of calling computer readable instructions stored in a storage through a processor.

25 **[0049]** FIG. 1 illustrates a flow chart of a neural network operation method provided by an embodiment of the present disclosure. The method includes steps S101-S103 as follows:

[0050] S101: determining a network layer to be processed in a target neural network.

[0051] S102: determining a target operator and a target chunking strategy corresponding to the network layer to be processed in the target neural network from a plurality of determined operators and a plurality of determined chunking strategies.

30 **[0052]** Each operator in a plurality of operators is used to implement a corresponding function of the network layer to be processed, and each chunking strategy in the plurality of chunking strategies matches with an operational requirement of a computing device for operating the target neural network.

35 **[0053]** S103: operating the target neural network containing the target operator based on

the target chunking strategy corresponding to the network layer to be processed.

[0054] In the above method, after the network layer to be processed in the target neural network is determined, the target operator and the target chunking strategy corresponding to the network layer to be processed may be determined from a plurality of determined operators and chunking strategies; and because the chunking strategy meets the operational requirement of the computing device, when the target neural network containing the target operator is operated based on the target chunking strategy corresponding to the network layer to be processed, the operational requirement of the computing device can be met. At the same time, since the target chunking strategy may perform the chunking on the parameter data of the target operator corresponding to the matched network layer to be processed, the resource consumption for operating the network layer to be processed based on the chunked parameter data is minimal, for example, the resource consumption may be represented by total computational overhead, that is, when the operational requirement of the computing device is met, the efficiency of operating the target neural network containing the target operator based on the target chunking strategy corresponding to at least one network layer to be processed is high.

[0055] In a possible implementation, the target neural network is used to implement an image processing task, and the image processing task includes at least one of image identification, image classification, image segmentation and key point detection.

[0056] The steps S101-S103 are described in detail below.

[0057] For the step S101: the target neural network may be any neural network after graph hierarchic optimization (i.e., graph optimization); the neural network after the graph optimization is a neural network whose computation graph is already determined, that is, the neural network after the graph optimization is the neural network whose task and parameter size of each network layer are already determined; and the parameter size of each network layer may be a size of parameter data included in the network layer. For example, the task of a first network layer of the target neural network may be feature extraction, and when the first network layer includes input data, the parameter size of the input data may be $256 \times 256 \times 128$. The task and parameter size of each network layer may be set according to the actual situation, which is only an example here.

[0058] The network layer to be processed may be a network operator (OP) layer to be processed on the target neural network. For example, the network layer to be processed may be a network OP layer with a size greater than a set threshold in the target neural network or may be a network OP layer selected by a user according to the requirement. The number of the determined network layer to be processed may be one or more.

[0059] For an example, each network OP layer may be approximated as a convolutional layer, for example, a fully-connected layer may be approximated as the convolutional layer whose feature graph is consistent with the size of a convolutional kernel, and a conventional layer with no weight may be enabled to be a convolutional layer with a weight of 0, etc.

5 **[0060]** For the step S102: the number of the network layers to be processed may be multiple, and the target operator and target chunking strategy corresponding to the network layer to be processed may be determined for each network layer to be processed. Each chunking strategy in the plurality of chunking strategies is matched with the operational requirement of the computing device for operating the target neural network; and each operator
10 in a plurality of operators is used to implement a corresponding function of the network layer to be processed, and each operator may correspond to one operation or one basic network structure unit. For example, the preset operator, for example, includes a convolutional operator, a pooling operator, a fully-connected operator, etc. The computing device is a device that directly processes inference computation of the target neural network, for example, the
15 computing device may be digital signal processing (DSP), etc.

[0061] In a possible implementation, the chunking strategy is used to perform chunking on the parameter data of the target operator corresponding to the network layer to be processed; and among the plurality of chunking strategies, the resource consumption for operating the network layer to be processed based on the parameter data obtained by chunking the parameter
20 data of the target operator by using the target chunking strategy is minimal.

[0062] The minimal resource consumption here may refer to that the operation time of the network layer to be processed is minimal. When in implementation, the target chunking strategy of each network layer to be processed is used to perform the chunking on the parameter data of the target operator corresponding to the network layer to be processed, so
25 that the resource consumption for the computing device to operate each network layer to be processed based on the chunked parameter data is minimal, for example, the resource consumption may be represented by the total computational overhead, that is, the total computational overhead of operating each network layer to be processed is minimal. The parameter data of the operator may include input/output data and constant data; the
30 input/output data may include input data and output data; and the constant data may include weight data and/or deviation data.

[0063] Exemplarily, the input data may be three-dimensional data, for example, which may include a width dimension, a height dimension and an input channel dimension; the output data may be three-dimensional data, for example, which may include an output width
35 dimension, an output height dimension and an output channel dimension; the weight data may

be four-dimensional data, for example, which may include a width dimension, a height dimension, an input channel dimension and an output channel dimension; and the deviation data may be one-dimensional data, for example, which may include an output channel dimension. The dimension information of the input data, the output data, the weight data and the deviation data may be set according to the actual situation, which is only an example here.

[0064] In implementations, when there are a plurality of network layers to be processed, the target operator and the target candidate chunking strategy of each network layer to be processed may be determined layer by layer according to a sequence of the network layers to be processed in the target neural network for each network layer to be processed in the target neural network; or the target operator and the target candidate chunking strategy of each network layer to be processed in each network layer to be processed may also be determined randomly. For example, when it is necessary to determine whether the data arrangement on the input data of the current network layer to be processed is consistent with a set target data arrangement, it is necessary to utilize the output data of the network layer to be processed before the current network layer to be processed, and it is necessary to determine the target candidate operator and the target candidate chunking strategy corresponding to each network layer to be processed layer by layer.

[0065] In an optional implementation, referring to FIG. 2, in a case where there are a plurality of network layers to be processed, the target operator and the target chunking strategy corresponding to the network layer to be processed in the target neural network are determined from a plurality of determined operators and chunking strategies, which includes:

[0066] S201: for each network layer to be processed in the target neural network, determining the target candidate operator corresponding to the network layer to be processed from a plurality of operators, and determining the target candidate chunking strategy matched with the target candidate operator from the plurality of chunking strategies.

[0067] S202: in a case of a plurality of target candidate operators and/or a plurality of target candidate chunking strategies corresponding to any network layer to be processed, determining the target operator and the target chunking strategy corresponding to each network layer to be processed based on the target candidate operators and target candidate chunking strategies corresponding to the respective network layers to be processed respectively.

[0068] Firstly, the target candidate operator corresponding to each network layer to be processed and the target candidate chunking strategy matched with the target candidate operator may be determined here, so that the local optimization for the target candidate operator and the target candidate chunking strategy of each network layer to be processed is implemented; and further, in a case where there are a plurality of target candidate operators

and/or a plurality of target candidate chunking strategies corresponding to any network layer to be processed, in comprehensive consideration of each network layer to be processed, the target operators and the target chunking strategies corresponding to the respective network layers to be processed in the target neural network are determined, so that the global optimization for the target operators and the target chunking strategies for each network layers to be processed is implemented.

[0069] An operator set and a chunking strategy set may be preset here; and the operator set includes all the set operators, and the chunking strategy set includes all the set chunking strategies. In order to improve the efficiency of determining the target operator and the target chunking strategy of the network layer to be processed, for each network layer to be processed, a plurality of operators and a plurality of chunking strategies corresponding to the network layer to be processed may be determined from the operator set and the chunking strategy set. A plurality of operators corresponding to different network layers to be processed may be the same or different; and a plurality of chunking strategies corresponding to different network layers to be processed may be the same or different. A plurality of operators and a plurality of chunking strategies corresponding to each network layer to be processed may be determined according to the actual situation.

[0070] Exemplarily, a plurality of operators and/or a plurality of chunking strategies corresponding to each network layer to be processed may be determined based on historical experience data. For example, based on the historical experience data, it may be determined that a plurality of operators corresponding to the network layer I to be processed include an operator I, an operator II and an operator III, and the plurality of chunking strategies correspondingly include a chunking strategy I, a chunking strategy II and a chunking strategy IV; and a plurality of operators corresponding to the network layer II to be processed include an operator I, an operator II, an operator IV and an operator V, and the plurality of chunking strategies correspondingly include a chunking strategy II and a chunking strategy V.

[0071] In the step S201, in an optional implementation, referring to FIG. 3, for each network layer to be processed in the target neural network, the target candidate operator corresponding to the network layer to be processed is determined from a plurality of operators, and the target candidate chunking strategy matched with the target candidate operator is determined from the plurality of chunking strategies, which includes:

[0072] S301: for the network layer to be processed, determining one or more first candidate operators from a plurality of operators.

[0073] S302: based on a resource consumption situation for the first candidate operator in each chunking strategy from the plurality of chunking strategies, selecting one or more target

candidate operators corresponding to the network layer to be processed from the first candidate operators and one or more target candidate chunking strategies corresponding to the target candidate operators from the plurality of chunking strategies.

[0074] After one or more first candidate operators corresponding to the network layer to be processed are determined for each network layer to be processed, one or more target candidate operators corresponding to the network layer to be processed and the target candidate chunking strategy corresponding to the target candidate operator may be selected from at least one first candidate operator and the plurality of chunking strategies based on the resource consumption situation of the first candidate operator in each chunking strategy of the plurality of chunking strategies, for example, the first candidate operator and the chunking strategy with minimal resource consumption may be selected as the target candidate operator and the target candidate chunking strategy, so that the local optimization of the target candidate operator and the target candidate chunking strategy corresponding to each network layer to be processed is implemented.

[0075] For the step S301, for each network layer to be processed in the target neural network, one or more first candidate operators corresponding to the network layer to be processed may be determined from a plurality of operators. For example, the operator capable of completing a task may be selected from a plurality of operators as the first candidate operator corresponding to the network layer to be processed according to the task of each network layer to be processed; or the first candidate operator corresponding to the network layer to be processed may also be determined by the user based on the demand of the target neural network.

[0076] For the step S302, the resource consumption situation of each first candidate operator in each chunking strategy may be determined firstly, and then based on the resource consumption situation of the first candidate operator in each chunking strategy in plurality of chunking strategies, one or more target candidate operators corresponding to the network layer to be processed and one or more target candidate chunking strategies corresponding to the target candidate operators are determined from at least one candidate operator and the plurality of chunking strategies. The resource consumption situation is the resource consumption data when the computing device operates the first candidate operator based on the chunking strategy; and for example, the resource consumption situation may be represented by a computational overhead value, and the computational overhead value represents the time taken by the computing device for operating the network layer to be processed containing the target operator.

[0077] Example I, if the first candidate operator corresponding to the network layer I to

be processed includes a first candidate operator I and a first candidate operator II, and the chunking strategy corresponding to the network layer to be processed includes a chunking strategy I, a chunking strategy II and a chunking strategy III; for the first candidate operator I, the computational overhead value corresponding to the chunking strategy I, the computational overhead value corresponding to the chunking strategy II and the computational overhead value corresponding to the chunking strategy III may be computed; and for the first candidate operator II, the computational overhead value corresponding to the chunking strategy I, the computational overhead value corresponding to the chunking strategy II and the computational overhead value corresponding to the chunking strategy III may be computed; and based on the 6 computational overhead values obtained by computing, the target candidate operator and the target candidate chunking strategy corresponding to the network layer I to be processed may be determined.

[0078] In an optional implementation, after a plurality of computational overhead values corresponding to each first candidate operator are obtained, the computational overhead values may be used directly to determine at least one target candidate operator and target candidate chunking strategy corresponding to the network layer to be processed.

[0079] For example, at least one target candidate operator and target candidate chunking strategy corresponding to the network layer to be processed may be determined by utilizing the computational overhead value in the two ways below:

[0080] In a first way, the first candidate operator and chunking strategy with minimal overhead values are selected from the respective computational overhead values obtained by computing are selected as the target candidate operator and the target candidate chunking strategy corresponding to the network layer to be processed.

[0081] The description is continued with the above example I; and after 6 computational overhead values are obtained, the minimal computational overhead value is selected; and for example, if the computational overhead value of the first candidate operator I in the chunking strategy I is minimal, the first candidate operator I is determined as the target candidate operator corresponding to the network layer I to be processed, and the chunking strategy I is determined as the target candidate chunking strategy corresponding to the network layer I to be processed.

[0082] In a second way, an overhead threshold may be set, the candidate overhead value that is less than the overhead threshold is selected from a plurality of computational overhead values obtained by computing and corresponding to the network layer to be processed, and the first candidate operator corresponding to the candidate overhead value is determined as the target candidate operator corresponding to the network layer to be processed, and the chunking

strategy corresponding to the candidate overhead value is determined as the target chunking strategy matched with the target candidate operator.

[0083] The description is continued with the example I. After the 6 computational overhead values are obtained, if the computational overhead value of the first candidate operator I in the chunking strategy I is less than the set overhead threshold, and the computational overhead value of the first candidate operator II in the chunking strategy III is less than the set overhead threshold, the first candidate operator I is determined as a target candidate operator corresponding to the network layer I to be processed, and the chunking strategy I is determined as the target candidate chunking strategy matched with the first candidate operator I; and the first candidate operator II is determined as a target candidate operator corresponding to the network layer I to be processed, and the chunking strategy III is determined as the target candidate chunking strategy matched with the first candidate operator III, that is, the target candidate operator and the target candidate chunking strategy corresponding to the network layer I to be processed are determined.

[0084] In another implementation, in the step S302, based on the resource consumption situation of the first candidate operator in each chunking strategy of the plurality of chunking strategies, one or more target candidate operators corresponding to the network layer to be processed and one or more target candidate chunking strategies corresponding to the target candidate operators are selected from the first candidate operators and the plurality of chunking strategies, which includes:

[0085] Step I: a target resource consumption situation satisfying a predetermined condition is selected from a plurality of resource consumption situations corresponding to the first candidate operator, wherein one first candidate operator corresponds to one resource consumption situation in one chunking strategy.

[0086] Step II: the chunking strategy corresponding to the target resource consumption situation is determined as the candidate chunking strategy, the network layer to be processed containing the second candidate operator corresponding to the target resource consumption situation is operated based on the candidate chunking strategy, and a test result corresponding to the candidate chunking strategy and the second candidate operator is determined.

[0087] Step III: one or more target candidate operators corresponding to the network layer to be processed and one or more target candidate chunking strategies corresponding to the target candidate operators are determined based on the test result.

[0088] By using the above method, firstly, the resource consumption situation may be used to select the second candidate operator and the candidate chunking strategy matched with the second candidate operator from the first candidate operators and the plurality of chunking

strategies; and the second candidate operator and the candidate chunking strategy are tested, and then the test result is used to determine at least one target candidate operator and target candidate chunking strategy corresponding to the network layer to be processed, so that the at least one determined target candidate operator and target candidate chunking strategy corresponding to the network layer to be processed are preferred options.

[0089] In the step I, one first candidate operator corresponds to one resource consumption situation in one chunking strategy. For example, when the chunking strategy corresponding to the first candidate operator I includes four types, the first candidate operator corresponds to 4 resource consumption situations.

[0090] The resource consumption situation represented by the computational overhead value is taken as an example for description below, and a predetermined condition may be set according to the actual requirement. For example, the predetermined condition may be a minimal overhead value; and/or the predetermined condition may be less than the set overhead threshold; and/or the predetermined condition may also be to select the minimal overhead value and the second smallest overhead value whose difference with the minimal overhead value is less than the set difference threshold.

[0091] For example, if the computational overhead value of the first candidate operator obtained by computing in a respective set chunking strategy includes a computational overhead value I 10, a computational overhead value II 12, a computational overhead value III 18 and a computational overhead value IV 20, the minimal overhead value may be selected from a plurality of computational overhead values, that is, the computational overhead value I is determined as the target overhead value; or the overhead threshold 15 may be set, and the computational overhead value I and the computational overhead value II are determined as the target overhead values; or a difference threshold 5 may be set, (it may be known that a difference between the computational overhead value II and the computational overhead value I is less than the set difference threshold), and the computational overhead value I and the computational overhead value II are determined as the target overhead values, that is, the target overhead value corresponds to the second candidate operator and the candidate chunking strategy matched with the second candidate operator.

[0092] In the step II, the second candidate operator and the candidate chunking strategy corresponding to the target overhead value (i.e., target resource consumption situation) are measured to obtain the test result corresponding to each target overhead value. For each target overhead value, the chunking strategy corresponding to the target resource consumption situation may be determined as a candidate chunking strategy; the network layer to be processed containing the second candidate operator corresponding to the target overhead value

is operated based on the candidate chunking strategy; and the test result corresponding to the target overhead value is determined, that is, the test result corresponding to the candidate chunking strategy and the second candidate operator is determined.

[0093] In the step III, one or more target candidate operators corresponding to the network layer to be processed and one or more target candidate chunking strategies corresponding to the target candidate operators may be determined based on the test result. For example, when the test result is the operation time, the second candidate operator with minimal operation time may be selected as the target candidate operator of the network layer to be processed, and the candidate chunking strategy corresponding to the second candidate operator with the minimal operation time may be determined as the target candidate operator. The first candidate operator and the second candidate operator may be operators capable of implementing the function of the network layer to be processed.

[0094] Or, the operation time threshold may also be set, and the test result that is less than the operation time threshold may be determined as the target test result; the second candidate operator corresponding to the target test result is determined as the target candidate operator, and the candidate chunking strategy corresponding to the target candidate operator in the target test result is determined as the target candidate chunking strategy.

[0095] In an optional implementation, the resource consumption situation is indicated by the computational overhead value, and the computational overhead value of the first candidate operator in each chunking strategy may be determined according to steps of:

[0096] Step I: a restricted scenario corresponding to the first candidate operator with a preset size is determined, wherein the restricted scenario is determined based on a computing time cost and a transmission time cost for data capacity corresponding to the first candidate operator with the preset size;

[0097] Step II: in a case where the restricted scenario is a bandwidth restricted scenario, based on a result of chunking of the chunking strategy, the total direct memory access (DMA) data transmission amount, the number of DMA tasks and the data conversion overhead corresponding to the first candidate operator in the chunking strategy are determined; and based on the total DMA data transmission amount, the number of DMA tasks, the data conversion overhead, and a DMA rate and DMA task overhead corresponding to the computing device, the computational overhead value of the first candidate operator in the chunking strategy is determined, wherein the data conversion overhead is the time consumed in performing a data arrangement conversion on the input data corresponding to the first candidate operator based on a target data arrangement corresponding to the first candidate operator;

[0098] Step III: in a case where the restricted scenario belongs to the computation restricted scenario, the computing time cost of parameter data corresponding to the first candidate operator in the chunking strategy, the number of operator calls for the first candidate operator, a total amount of initial data transmission, the number of DMA tasks and data conversion overhead are determined based on a result of chunking by the chunking strategy; and the computational overhead value for the first candidate operator in the chunking strategy is determined based on the computing time cost, the number of operator calls, the total amount of initial data transmission, the data conversion overhead, the DMA task overhead, the number of the DMA tasks and a DMA rate corresponding to the computing device.

[0099] In the step I, the restricted scenario corresponding to each first candidate operator with the preset size may be determined. The preset size may be a large size that is set according to the requirement.

[00100] When in implementation, the parameter data of the target operator corresponding to each network layer to be processed may be stored in other external storages except for the internal storage of the computing device, for example, may be stored in a double data rate synchronous dynamic random access memory (DDR); when each network layer to be processed is operated again, the DMA may acquire the parameter data (such as input data, constant data, etc.) of the target operator corresponding to the network layer to be processed from the DDR and transmit the acquired parameter data into the internal storage of the computing device; and after the computing device completes the computation, the DMA transmits a data result (i.e., output data) into the DDR so as to be used by the next network layer (the network layer may be the network layer to be processed) adjacent to the network layer to be processed in the target neural network. The DMA may utilize a PING-PONG scheduling strategy to transmit the acquired parameter data.

[00101] Thus it can be known that the DMA has one transmission time cost for transmitting the parameter data of the first candidate operator with the preset size, and one computing time cost for the computing device processes the parameter data of the first candidate operator with the preset size. If the computing time cost is greater than the transmission time cost, it means that when the DMA transmits the current parameter data to the internal storage of the computing device, the computing device does not completes the processing on the previous parameter data, and at the time, the computing device needs to wait; after the computing device completes the processing on the previous parameter data, the current parameter data is transmitted to the internal storage of the computing device, and the scenario corresponding to this situation is the computation restricted scenario; and if the computing time cost is less than the transmission time cost, it means that after the computing

device completes the processing on the previous parameter data, the DMA does not transmit the current parameter data to the internal storage of the computing device yet, and the DMA needs to wait until the current parameter data transmitted by the DMA is received, and the scenario corresponding to this situation may be the bandwidth restricted scenario.

5 **[00102]** When the restricted scenario is the bandwidth restricted scenario, a first overhead function corresponding to the bandwidth restricted scenario is used to calculate the overhead value; and when the restricted scenario is the computation restricted scenario, a second overhead function corresponding to the computation restricted scenario may be used to calculate the overhead value.

10 **[00103]** Exemplarily, the restricted scenario corresponding to the first candidate operator with the preset size may be determined according to the process below: for the first candidate operator with the preset size, the transmission time cost required for transmitting the parameter data corresponding to the first candidate operator with the preset size is determined, and the computing time cost of the computing device processing the parameter data of the first
15 candidate operator with the preset size is determined; and the restricted scenario corresponding to the first candidate operator is determined based on the transmission time cost and the computing time cost.

[00104] Exemplarily, the restricted scenario of the first candidate operator with the preset size may also be determined according to the process below: first, the target time required by
20 the computing device for operating the corresponding network layer to be processed based on the parameter data corresponding to the first candidate operator is determined, and the target data capacity of the parameter data corresponding to the first candidate operator is determined. Second, the data capacity that can be transmitted by the DMA within the target time is determined based on the DMA rate corresponding to the computing device and the target time.
25 Third, the restricted scenario is determined based on the comparison between the data capacity that can be transmitted by the DMA within the target time and the target data capacity, that is, when a ratio is less than or equal to 1, the restricted scenario is determined as the bandwidth restricted scenario; and when the ratio is greater than 1, the restricted scenario is determined as the computation restricted scenario.

30 **[00105]** The data capacity that can be transmitted by the DMA within the target time is related to a transmission rate, and the target data capacity is related to the computing speed; when the ratio is greater than 1, it means that the transmission rate is greater than the computing speed (that is, the transmission time cost is less than the computing time cost), i.e., the computation restricted scenario; and when the ratio is less than or equal to 1, it means that
35 the transmission rate is less than or equal to the computing speed (that is, the transmission time

cost is greater than or equal to the computing time cost), i.e., the bandwidth restricted scenario, and different ways may be selected to determine the computational overhead value subsequently for different restricted scenarios.

[00106] The target time corresponding to the parameter data of the first candidate operator on the computing device may be determined based on the preset size information of the parameter data of the first candidate operator, that is, the target time required by the computing device for operating the corresponding network to be processed based on the parameter data corresponding to the first candidate operator. The data capacity that can be transmitted by the DMA within the target time may be obtained by multiplying the DMA rate corresponding to the computing device with the target time.

[00107] Meanwhile, the target data capacity of the parameter data corresponding to the first candidate operator may be determined based on the preset size information of the parameter data of the first candidate operator. For example, when the first candidate operator is a convolutional operator, the target data capacity may be a sum of the constant data (including weight data and deviation data), output data and input data. The restricted scenario may be determined based on the calculated data capacity that can be transmitted by the DMA within the target time and a ratio to the target data capacity.

[00108] When in implementation, after the computing device is determined, the DMA task overhead corresponding to the computing device may be determined in a unit of second (s); and for example, a cycle consumed by creating each DMA task may be converted into time, that is, the DMA task overhead is obtained; and the DMA rate, i.e., transmission rate of the DMA, may be determined in a unit of Byte/s.

[00109] In the step II, the first overhead function may be used to determine the computational overhead value of the first candidate operator in the chunking strategy. The first overhead function may be as follows: $\text{computational overhead} = \text{total DMA data transmission amount} / \text{DMA rate} + \text{number of DMA tasks} \times \text{DMA task overhead} + \text{data conversion overhead}$.

[00110] When the bandwidth restricted scenario is determined, the total DMA data transmission amount (the unit is Byte) corresponding to the first candidate operator in the chunking strategy, the number of DMA tasks and the data conversion overhead (the unit is second (s)) may be determined according to the result of chunking. The total DMA data transmission amount may be determined according to the generated DMA task; the number of the DMA tasks may be determined based on the number of data blocks of the parameter data obtained by chunking the parameter data based on the chunking strategy; and for example, when one data block corresponds to one DMA task, and when the number of the generated data blocks is 10, the number of the DMA tasks is determined as 10. The total DMA data

transmission amount and the number of the DMA tasks may be determined according to the actual situation, which is only an example here. For example, when the first candidate operator is the convolutional operator, the number of the DMA tasks obtained after the result of chunking may be determined according to convolutional parameters corresponding to the convolutional operator such as the size of the convolution kernel, a convolutional step length and the like.

[00111] The data conversion overhead is the time consumed by performing the conversion in the data arrangement on the input data corresponding to the first candidate operator according to the target data arrangement corresponding to the first candidate operator. When the data arrangement on the input data of the first operator here is consistent with the target data arrangement corresponding to the first operator, the data conversion overhead is 0; and when the data arrangement on the input data of the first operator is inconsistent with the target data arrangement corresponding to the first operator, the data conversion overhead may be calculated according to the formula below: data conversion overhead=total data capacity of input data $\times$ 2/DMA rate. The total data capacity of the input data is all input data inputted to the network layer to be processed before the chunking.

[00112] In the step III, when the computation restricted scenario is determined, the computational overhead value of the first candidate operator in the chunking strategy may be calculated according to the second overhead function. The second overhead function is as follows: computational overhead=operator overhead conversion bandwidth $\times$ number of operator calls/DMA rate+number of DMA tasks $\times$ DMA task overhead+data conversion overhead.

[00113] The operator overhead conversion width is an operator transmission data volume determined based on the computing time cost for the first candidate operator with the preset size and the size of the parameter data corresponding to the first candidate operator in the chunking strategy. For example, if the preset size is 1024 $\times$ 1024 $\times$ 128, the computing time cost for the first candidate operator with the preset size is 10 ms, and the size of the chunked parameter data is 512 $\times$ 512 $\times$ 64, the computing time cost for the parameter data corresponding to the first candidate operator in the chunking strategy is 1.25 ms. Further, the operator overhead conversion bandwidth corresponding to the chunked first candidate operator may be determined according to the determined computing speed, and the computing time cost (such as 1.25 ms) of the parameter data corresponding to the first candidate operator in the chunking strategy.

[00114] Specifically, the computing time cost for the parameter data corresponding to the first candidate operator in the chunking strategy, the number of operator calls of the first

candidate operator, the total transmission amount of the initial data, the number of DMA tasks and the data conversion overhead may be determined based on the result of chunking. The number of operator calls may be determined based on the number of data blocks of the parameter data obtained by chunking the parameter data based on the chunking strategy; for example, if the number of the obtained data blocks of the parameter data is 10, the number of operator calls is determined as 10; the total transmission amount of the initial data is a data capacity of the initial data determined based on the chunking strategy; and the target data capacity, the number of operator calls and the total transmission amount of the initial data may be determined according to the actual situation.

[00115] In the step II and the step III, the target data capacity corresponding to the aligned parameter data of the first candidate operator, the number of operator calls, the total transmission amount of the initial data, the number of DMA tasks and the data conversion overhead may be obtained based on the result of chunking.

[00116] The determining process of the data conversion overhead in the step III is the same as that of the data conversion overhead in the step II, which is not described in detail here. The disclosed embodiment may be mainly applied to the bandwidth restricted scenario, that is, when the bandwidth restricted scenario is satisfied, the step II is used to determine the computational overhead value; and when the bandwidth restricted scenario is not satisfied (that is the computational overhead is satisfied), the step III may be used to determine the computational overhead value.

[00117] In the above implementation, the corresponding restricted scenario of the first candidate operator with the preset size may be determined, and different restricted scenarios correspond to different computational overhead value determining ways. For example, in the bandwidth restricted scenario, the computational overhead value may be determined based on the total DMA data transmission amount, the number of DMA tasks, the data conversion overhead, the DMA rate and the DMA task overhead; and in the computation restricted scenario, the computational overhead value may be determined based on the computing time cost, the number of operator calls, the total transmission amount of initial data, the data conversion overhead and the DMA rate.

[00118] In an optional implementation, prior to selecting one or more target candidate operators corresponding to the network layer to be processed from the first candidate operators and one or more target candidate chunking strategies corresponding to the target candidate operators from the plurality of chunking strategies, the method further includes: aligning the parameter data corresponding to the first candidate operator based on the determined minimal granularity information corresponding to the target neural network to obtain the aligned

parameter data corresponding to the first candidate operator, wherein the minimal granularity information includes the minimal granularities corresponding to the parameter data in different dimensions; and the size of the aligned parameter data in different dimensions is of an integer multiple of the minimal granularity in the respective dimension indicated by the minimal granularity information.

[00119] The minimal granularity information here includes the minimal granularity corresponding to the parameter data in different dimensions; and for example, when the parameter data includes weight data, the minimal granularity information corresponding to the weight data includes minimal granularity in a width dimension, the minimal granularity in a length dimension, the minimal granularity in an input channel dimension and the minimal granularity in an output channel dimension. The minimal granularity information may be determined according to the operational requirement of the computing device, and/or the user requirement, which is only an example here.

[00120] The parameter data corresponding to each first candidate operator may be aligned by utilizing the determined minimal granularity information corresponding to the target neural network to obtain the aligned parameter data corresponding to the first candidate operator, so that the size of the aligned parameter data in different dimensions is of an integer multiple of the minimal granularity in the corresponding dimension indicated by the minimal granularity information. For example, if the size of the width dimension indicated by the minimum granularity information is 32, and the size of the width dimension indicated by the parameter data is 33, the size of the width dimension of the generated aligned parameter data is 64; and when the size of the width dimension indicated by the parameter data is 31, the size of the width dimension of the generated aligned parameter data is 32.

[00121] The specific process of the aligning may be selected according to the actual requirement. For example, the conventional data aligning method (such as the padding method) may be used to align the parameter data to generate the aligned parameter data.

[00122] In another implementation, the computing device may also acquire the parameter data before the alignment from the DDR, then utilize garbage data for computation, select valid data from the data outputted by the computing device and input the valid data as the output data into DDR.

[00123] The parameter data corresponding to each first candidate operator may be aligned based on the minimal granularity information corresponding to the target neural network to obtain the aligned parameter data corresponding to the first candidate operator, and the size of the aligned parameter data in different dimensions is of an integer multiple of the minimal granularity in the corresponding dimension indicated by the minimal granularity information,

so that the occurrence probability of parameter data loss when the target neural network is operated based on the target chunking strategy may be reduced.

[00124] For the step S202: in an optional implementation, in the step S202, determining the target operator and the target chunking strategy corresponding to each network layer to be processed based on the target candidate operators and the target candidate chunking strategies corresponding to respective network layers includes:

[00125] S2021: determining a plurality of test networks corresponding to the target neural network based on the target candidate operators corresponding to the respective network layers to be processed and the target candidate chunking strategies corresponding to the target candidate operators, wherein each test network includes a target candidate operator corresponding to each network layer to be processed and a target candidate chunking strategy matched with the target candidate operator.

[00126] S2022: operating a plurality of test networks respectively to obtain a plurality of test results, wherein each test network corresponds to one test result.

[00127] S2023: selecting the target test network from the plurality of test networks based on the plurality of test results.

[00128] S2024: determining the target candidate operator and the target candidate chunking strategy of the network layer to be processed in the target test network as the target operator and the target chunking strategy corresponding to the network layer to be processed in the target neural network.

[00129] In the step S2021, exemplarily, if the target neural network includes a first network layer to be processed, a second network layer to be processed and a third network layer to be processed, the first network layer to be processed includes a target candidate operator I, a chunking strategy I corresponding to the target candidate operator I, a target candidate operator II, and a chunking strategy II corresponding to the target candidate operator II; the second network layer to be processed includes a target candidate operator III, a chunking strategy I corresponding to the target candidate operator, a target candidate operator IV and a chunking strategy I corresponding to the target candidate operator IV; and the third network layer to be processed includes a target candidate operator V and a chunking strategy III corresponding to the target candidate operator V.

[00130] The four test networks corresponding to the target neural network may be further obtained, and the first test network includes a target candidate operator I, a chunking strategy I corresponding to the target candidate operator I, a target candidate operator III, a chunking strategy I corresponding to the target candidate operator, a target candidate operator V and a chunking strategy III corresponding to the target candidate operator V. The second test network

includes a target candidate operator I, a chunking strategy I corresponding to the target candidate operator I, a target candidate operator IV, a chunking strategy I corresponding to the target candidate operator VI, a target candidate operator V and a chunking strategy III corresponding to the target candidate operator V. The third test network includes a target candidate operator II, a chunking strategy II corresponding to the target candidate operator II, a target candidate operator III, a chunking strategy I corresponding to the target candidate operator, a target candidate operator V and a chunking strategy III corresponding to the target candidate operator V. The fourth test network includes a target candidate operator II, a chunking strategy II corresponding to the target candidate operator II, a target candidate operator IV, a chunking strategy I corresponding to the target candidate operator IV, a target candidate operator V and a chunking strategy III corresponding to the target candidate operator V.

[00131] In the steps S2022 and S2023, the computing device may be controlled to operate a plurality of test networks respectively to determine the test result of each test network. For example, the test result may be the operation time corresponding to each test network. The target test network may be selected from the plurality of test networks based on the test results corresponding to the plurality of test networks. For example, the test network with minimal operation time may be selected as the target test network.

[00132] In the step S2024, the target candidate operator and the target candidate chunking strategy of each network layer to be processed included in the target test network may be determined as the target operator and the target chunking strategy corresponding to each network layer to be processed respectively in the target neural network.

[00133] For example, if the second test network is determined as the target test network, the target candidate operator I is determined as the target operator of the first network layer to be processed, and the chunking strategy I is the target chunking strategy corresponding to the first network layer to be processed; the target candidate operator IV is the target operator of the second network layer to be processed, and the chunking strategy I is the target candidate chunking strategy of the second network layer to be processed; and the target candidate operator V is the target operator of the third network layer to be processed, and the chunking strategy III is the target candidate chunking strategy of the third network layer to be processed.

[00134] In order to reduce the cost and operation resource consumed by operating the test network and improve the determining efficiency of the target operator and the target chunking strategy, when in implementation, the maximal number of the target operators matched with the target chunking strategy corresponding to each network layer to be processed may be set. For example, when the maximal number is set to be 2, each network layer to be processed may

include a target operator matched with the target chunking strategy, such as the target operator I matched with the target chunking strategy I; or each network layer to be processed may include two target operators matched with the target chunking strategy, for example, the two target operators matched with the target chunking strategy may be the target operator I matched with the target chunking strategy I and the target operator I matched with the target chunking strategy II; or the two target operators matched with the target chunking strategy may be the target operator I matched with the target chunking strategy I, and the target operator II matched with the target chunking strategy II; or the two target operators matched with the target chunking strategy may be the target operator I matched with the target chunking strategy I and the target operator II matched with the target chunking strategy I, etc.

[00135] In addition/alternatively, when in implementation, the number threshold of the test network corresponding to the target neural network may be set. For example, the number threshold may be set to be 100, and then the neural network includes 10 network layers to be processed; and if the number of the target operators matched with the target chunking strategy corresponding to the first to sixth network layers to be processed is 2, the number of local test networks composed of the target operators and the target chunking strategies corresponding to each network layer to be processed in the first to sixth network layers to be processed may be 64. Further, when the target operator and the target chunking strategy of the seventh network layer to be processed are determined, if the number of the target operators matched with the target chunking strategy corresponding to the seventh network layer to be processed is 2, the number of the local test networks composed of the target operators and the target chunking strategies corresponding to each network layer to be processed in the first to the seventh network layers to be processed may be 128, which is greater than the set number threshold; in this case, the number of the target operator matched with the target chunking strategy corresponding to each network layer to be processed in the seventh network layer to be processed, the eighth network layer to be processed, the ninth network layer to be processed and the tenth network layer to be processed may be only one.

[00136] In the above implementation, a plurality of test networks corresponding to the target neural network are determined based on the at least one target candidate operator corresponding to each network layer to be processed and the target candidate chunking strategy corresponding to the target candidate operator; then the plurality of test networks are operated by the computing device to determine the test result of each test network; the target test network is determined based on the test result; and for example, when the test result is the computational overhead, the test network with minimal computational overhead may be selected as the target test network, the target candidate operator and the target candidate

chunking strategy of each network layer to be processed in the target test network are determined as the target operator and the target chunking strategy corresponding to each network layer to be processed in the target neural network, so that the global optimization of the target operator and target chunking strategy is implemented.

5 **[00137]** In an optional implementation, when the specified dimension is one dimension, the dimension parameter is a first dimension; and when the specified dimension is N dimensions, the dimension parameter includes the first dimension to an Nth dimension, and N is the dimension greater than 2 and less than the constant data or input data. In a case where the parameter data includes input data and constant data, the plurality of chunking strategies
10 include at least one of:

15 **[00138]** Solution I: all input data is taken as the initial data, and one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain a result of chunking; and the initial data is the data written into an initial data region allocated by the direct memory access (DMA) task when the computing device operates the target neural network.

15 **[00139]** Solution II: all input data is taken as the initial data, and two-dimensional chunking is performed on the constant data based on the determined first dimension and second dimension of the constant data to obtain the result of chunking.

20 **[00140]** Solution III: all constant data is taken as the initial data, and one-dimensional chunking is performed on the input data based on the determined first dimension of the input data to obtain the result of chunking.

20 **[00141]** Solution IV: all constant data is taken as the initial data, and two-dimensional chunking is performed on the input data based on the determined first dimension and second dimension of the input data to obtain the result of chunking.

25 **[00142]** Solution V: part of the input data is taken as the initial data, and one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain the result of chunking, wherein the target size of the part of the input data is determined according to the minimal granularity of the input data in the first dimension.

30 **[00143]** Solution VI: part of the input data is taken as the initial data, and two-dimensional chunking is performed on the constant data based on the determined first dimension and second dimension of the constant data to obtain the result of chunking, wherein the target size of the part of the input data is determined based on the minimal granularity the input data in the first dimension.

35 **[00144]** Solution VII: part of the constant data is taken as initial data, and one-dimensional chunking is performed on the input data based on the determined first dimension of the input

data to obtain the result of chunking, wherein the target size of the part of the constant data is determined according to the minimal granularity of the constant data in the first dimension.

[00145] Solution VIII: part of the constant data is taken as the initial data, and two-dimensional chunking is performed on the input data based on the determined first dimension and second dimension of the input data to obtain the result of chunking; wherein the target size of the part of the constant data is determined based on the minimal granularity of the constant data in the first dimension.

[00146] All the input data may be taken as the initial data, and an application space of the initial data is allocated in the initial data region. The one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain the result of chunking. Or, the two-dimensional chunking is performed on the constant data based on the determined first dimension and second dimension of the constant data to obtain the result of chunking.

[00147] All the constant data may be taken as the initial data, and the one-dimensional chunking is performed on the input data based on the determined first dimension of the input data to obtain the result of chunking. Or the two-dimensional chunking is performed on the input data based on the determined first dimension and second dimension of the input data to obtain the result of chunking.

[00148] The part of the input data may also be taken as the initial data, and the one-dimensional chunking is performed on the input data based on the determined first dimension of the input data to obtain the result of chunking; or the two-dimensional chunking is performed on the input data based on the determined first dimension and second dimension of the input data to obtain the result of chunking.

[00149] In an optional implementation, in the solution V or the solution VI, the part of the input data is taken as the initial data, and the constant data is chunked in a specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking, which includes:

[00150] I: determining the target size of the part of the input data based on i times of the minimal granularity of the input data in the first dimension.

[00151] II: by taking respectively the part of the input data of the target size as the initial data, chunking the constant data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking.

[00152] Where i is a positive integer that makes the data capacity of the part of the input data and the data capacity of the constant data block determined based on the minimum granularity of the dimension parameters of the constant data meet the memory requirements of

the computing device after the target size of the part of the input data is determined.

[00153] The maximum value of i may be determined in an increasing manner here. The solution V is taken as an example for description below (that is, the one-dimensional chunking is taken as an example for description), i increases gradually from 1, that is, when $i=1$, the target size of the part of the input data is one time of the minimal granularity of the input data in the first dimension, the part of the input data of the target size is taken as the initial data, and the one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain a result of one-dimensional chunking.

[00154] When the result of one-dimensional chunking corresponding to $i=1$ indicates that the constant data fails to allocate, the solution V is an unavailable solution; when the result of one-dimensional chunking corresponding to $i=1$ indicates the constant data succeeds in allocation, the value of i plus 1 (that is, $i=2$), and return to the step of determining the target size of the part of the input data, that is, the target size is twice of the minimal granularity of the input data in the first dimension, the part of the input data of the target size is taken as the initial data, and the one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain the result of one-dimensional chunking. When the result of one-dimensional chunking corresponding to $i=2$ indicates that the constant data fails to allocate, the maximum value of i is determined as 1, and the increasing process is ended; and when the result of one-dimensional chunking indicates that the constant data succeeds in allocation, the value of i plus 1 (at this time, $i=3$), return to the step of determining the target size of the part of the input data again until the result of one-dimensional chunking indicates that the constant data fails to allocate. For example, when $i=6$, if it is determined that the result of one-dimensional chunking indicates that the constant data fails to allocate, the maximal value of i is determined as 5. When the maximum value of i is 5, five results of chunking may be obtained in this solution.

[00155] The result of chunking indicates that the constant data fails to allocate, which may be that after the constant data is chunked based on the minimal granularity in the first dimension, the obtained constant data blocks and the initial data cannot meet the memory requirement of the computing device. If the scheduling strategy is PING-PONG schedule, when twice of the data capacity of the constant data blocks after the chunking based on the minimal granularity in the first dimension is greater than the internal storage of a scheduling region of the computing device, the input data fails to allocate.

[00156] For example, when the maximum value of i is 5, the solution V may include 5 chunking strategies below:

[00157] Solution I: one time of the minimal granularity of the input data in the first

dimension is determined as the target size of the part of the input data, the part of the input data is taken as the initial data, and one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain the result of one-dimensional chunking;

5 **[00158]** Solution II: twice of the minimal granularity of the input data in the first dimension is determined as the target size of the part of the input data, the part of the input data is taken as the initial data, and the one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain the result of one-dimensional chunking;

10 **[00159]**

[00160] Solution V: five times of the minimal granularity of the input data in the first dimension is determined as the target size of the part of the input data, the part of the input data is taken as the initial data, and the one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain result of one-dimensional chunking.

[00161] Part of the constant data may also be taken as the initial data, and the one-dimensional chunking is performed on the input data based on the determined first dimension of the input data.

[00162] In an optional implementation, in the solution VII or the solution VIII, the part of the constant data is taken as the initial data, and the input data is chunked in a specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking, which includes:

[00163] I: determining the target size of part of the constant data based on j times of the minimal granularity of the constant data in the first dimension; and

25 **[00164]** II: taking respectively the part of the constant data of the target size as the initial data, and chunking the input data in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking.

[00165] The maximum value of j may be determined in an increasing manner here. The solution VII is taken as an example for description below, j increases gradually from 1, that is, when j=1, the target size of the part of the input data is one time of the minimal granularity of the input data in the first dimension, the part of the input data of the target size is taken as the initial data, and the one-dimensional chunking is performed on the constant data based on the determined first dimension of the constant data to obtain the result of first-dimensional chunking.

35 **[00166]** When the result of one-dimensional chunking corresponding to j=1 indicates that

the input data fails to allocate, the solution VII is an unavailable solution; and when the result of one-dimensional chunking corresponding to $j=1$ indicates that the input data succeeds in allocation, the value of j plus 1 (that is, $j=2$), and return to the step of determining the target size of the part of the input data until the obtained result of one-dimensional chunking indicates that the input data fails to allocate. For example, when $j=6$, if it is determined that the obtained result of one-dimensional chunking indicates that the input data fails to allocate, the maximal value of j is determined as 5. When the maximum value of j is 5, five results of chunking may be obtained in this solution.

[00167] The result of chunking indicates that the input data fails to allocate, which may be that after the input data is chunked based on the minimal granularity in the first dimension, the obtained input data block and the initial data cannot meet the memory requirement of the computing device. If the scheduling strategy is PING-PONG schedule, when twice of the data capacity of the input data block after chunking based on the minimal granularity in the first dimension is greater than the internal storage of a scheduling region of the computing device, the input data fails to allocate. For example, if the initial data and the scheduling data PING (the input data block after the chunking based on the minimal granularity in the first dimension) and the scheduling data PONG (the input data block after the chunking based on the minimal granularity in the first dimension) do not meet the memory requirement of the computing device, it is determined that the input data fails to allocate.

[00168] For example, when the maximum value of j is 6, the solution VII may include 6 chunking strategies below:

[00169] Solution I: one time of the minimal granularity of the constant data in the first dimension is determined as the target size of the part of the constant data, the part of the constant data is taken as the initial data, and the one-dimensional chunking is performed on the input data based on the determined first dimension of the input data to obtain the result of one-dimensional chunking;

[00170] Solution II: twice of the minimal granularity of the constant data in the first dimension is determined as the target size of the part of the constant data, the part of the constant data is taken as the initial data, and the one-dimensional chunking is performed on the input data based on the determined first dimension of the input data to obtain the result of one-dimensional chunking;

[00171]

[00172] Solution VI: six times of the minimal granularity of the constant data in the first dimension is determined as the target size of the part of the constant data, the part of the constant data is taken as the initial data, and the one-dimensional chunking is performed on the

input data based on the determined first dimension of the input data to obtain the result of one-dimension chunking.

[00173] The first dimension and second dimension for chunking the input data here may be set according to information such as operational requirement and/or operator type; and the first dimension and second dimension for chunking the constant data may be set according to the information such as the operational requirement and/or operator type. For example, if the operator is a convolutional operator, the first dimension of the constant data may be an output channel (OC) dimension, and the second dimension may be an input channel (IC) dimension.

[00174] By setting a plurality of chunking strategies here, each network layer to be processed can select a better target operator and a better target chunking strategy matched with the target operator.

[00175] In an optional implementation, in a case where the specified dimension is one dimension, and the dimension parameter includes the first dimension, by taking the constant data and the input data respectively as the target data, performing the one-dimensional chunking on the target data based on the determined first dimension of the target data to obtain the result of one-dimensional chunking includes:

[00176] A1: determining k times of the minimal granularity corresponding to the target data in the first dimension as the target chunking size, performing the one-dimensional chunking on the target data by the first dimension based on the target chunking size to obtain a plurality of target data blocks corresponding to the target data; where k is a positive integer.

[00177] A2: in a case where it is determined that the plurality of target data blocks and the initial data satisfy a set chunking condition, taking k+1 times of the minimal granularity corresponding to the target data in the first dimension as an updated target chunking size, returning to the step of performing the one-dimensional chunking on the target data by the first dimension based on the target chunking size until it is determined that the plurality of target data blocks and the initial data do not satisfy the set chunking condition, and determining the k times of the minimal granularity corresponding to the target data in the first dimension as the result of chunking.

[00178] A3: in a case where the plurality of target data blocks generated with $k = 1$ and the initial data do not satisfy the set chunking conditions, determining the result of chunking as a failure of one-dimensional chunking.

[00179] By using the above method, the target chunking size increases constantly, and the result of chunking making the internal storage utilization rate of the computing device high is determined by a way of keeping trying, which is conducive to reducing the waste of internal storage resources of the computing device.

[00180] In the step A1, k is a positive integer. From $k=1$, the minimal granularity corresponding to the target data in the first dimension is determined as the target chunking size, and according to the target chunking size, the one-dimensional chunking is performed on the target data according to the first dimension to obtain a plurality of target data blocks
5 corresponding to the target data. The size of the first dimension of each obtained target data block is consistent with the target chunking size, and the size of other dimensions of each target data block except for the first dimension is consistent with the size of the corresponding dimension of the target data.

[00181] For example, if the minimal granularity in the first dimension is 32, and the size information of the target data is $64 \times 64 \times 128$, the target chunking size is 32, the one-dimensional chunking is performed on the target data according to the first dimension according to the target chunking size to obtain a plurality of target data blocks, and the size of each target data block may be $32 \times 64 \times 128$. The number of the target data blocks may be determined according to the actual situation.

[00182] The first dimension may be set according to the requirement, for example, the first dimension of the input data may be a width (W) dimension, and the second dimension may be an input channel (IC) dimension; and the first dimension of the constant data may be the output channel (OC) dimension, and the second dimension may be the input channel (IC) dimension.

[00183] Further, whether the plurality of target data blocks and the initial data satisfy the set chunking condition may be judged; if yes, twice of the minimal granularity corresponding to the target data in the first dimension is taken as the updated target chunking size, and return to the step of performing the one-dimensional chunking on the target data according to the first dimension according to the target chunking size until it is determined that a plurality of target data blocks and the initial data do not satisfy the set chunking condition, and the k times of the minimal granularity corresponding to the target data in the first dimension is determined as the result of chunking. For example, when $k=5$, and it is determined that the plurality of target data blocks and the initial data generated when $k=5$ do not satisfy the set chunking condition, four times of the minimal granularity corresponding to the target data in the first dimension is
25 determined as the result of chunking. That is, when the network layer to be processed is operated, the four times of the minimal granularity in the first dimension may be taken as the target chunking size, and the one-dimensional chunking is performed on the target data corresponding to the target operator of the network layer to be processed according to the target chunking size.

[00184] If the plurality of target data blocks generated when $k=1$ and the initial data do not

satisfy the set chunking condition, the result of chunking is determined as a failure of one-dimensional chunking.

[00185] In an optional implementation, in a case where the specified dimension is two dimensions, and the dimension parameter includes the second dimension, the constant data and the input data are taken as the target data respectively, performing a two-dimensional chunking on the target data based on the determined first dimension and second dimension of the target data to obtain the result of chunking includes:

[00186] B1: determining y times of the minimal granularity corresponding to the target data in the first dimension as the first target chunking size, and based on the first target chunking size, performing the one-dimensional chunking on the target data by the first dimension to obtain a plurality of intermediate data blocks corresponding to the target data, where y is a positive integer.

[00187] B2: determining x times of the minimal granularity corresponding to the target data in the second dimension as a second target chunking size, and performing the two-dimensional chunking on each intermediate data block by the second dimension based on the second target chunking size to obtain a plurality of target data blocks respectively corresponding to a plurality of intermediate data blocks; where x is a positive integer.

[00188] B3: in a case where it is determined that the plurality of target data blocks and the initial data satisfy the set chunking condition, taking $x+1$ times of the minimal granularity corresponding to the target data in the second dimension as an updated second target chunking size, returning to the step of performing the two-dimensional chunking on each intermediate data block by the second dimension based on the second target chunking size, until it is determined that the plurality of target data blocks and the initial data do not satisfy the set chunking condition; and determining the x times of the minimal granularity corresponding to the target data in the second dimension as the result of chunking.

[00189] In the step B1, y is a positive integer with an initial value of 1; for example, when the maximal value of y is set to be 3, y may be determined as 1, and the steps B1-B3 are executed to obtain one result of two-dimensional chunking; then y is determined as 2, and the steps B1-B3 are executed to obtain one result of two-dimensional chunking; and y is determined as 3, and the steps B1-B3 are executed to obtain one result of two-dimensional chunking, that is, three results of two-dimensional chunking may be obtained.

[00190] Here $y=1$ is taken as an example to describe the two-dimensional chunking process; if the minimal granularity corresponding to the first dimension is 32, the size of the target data is $128 \times 128 \times 256$, and then the one-dimensional chunking may be performed on the target data according to the first dimension based on the first target chunking size to obtain a

plurality of target intermediate data blocks corresponding to the target data; and the size of each target intermediate data block may be $32 \times 128 \times 256$. The number of the target intermediate data block may be determined according to the actual situation.

[00191] In the step B2, the description is continued with the example in the step B1, and x is a positive integer; from $x=1$, one time of the minimal granularity corresponding to the target data in the second dimension is determined as the second target chunking size; for example, if the minimal granularity of the second dimension is 32, the second target chunking size is 32; and based on the second target chunking size, two-dimensional chunking is performed on each intermediate data block according to the second dimension to obtain a plurality of target data blocks corresponding to each intermediate data block, that is, a plurality of target data blocks are obtained, and the size of each target data block may be $32 \times 32 \times 256$.

[00192] In the step B3, whether the plurality of target data blocks and the initial data satisfy the set chunking condition may be judged; if yes, twice (i.e., $x+1$ times) of the minimal granularity corresponding to the target data in the second dimension is taken as the updated second target chunking size, return to the step of performing the two-dimensional chunking on each intermediate data block according to the second dimension based on the second target chunking size until it is determined that the plurality of target data blocks and the initial data do not satisfy the set chunking condition; and the x times of the minimal granularity corresponding to the target data in the second dimension is determined as the result of chunking.

[00193] For example, when $k=3$, and it is determined that the plurality of target data blocks and the initial data generated when $k=3$ do not satisfy the set chunking condition, twice of the minimal granularity corresponding to the target data in the second dimension is determined as the result of chunking. That is, when the network layer to be processed is operated, the minimal granularity in the first dimension may be taken as the first target chunking size; twice of the minimal granularity in the second dimension may be taken as the second target chunking size; and based on the first target chunking size and the second target chunking size, the two-dimensional chunking is performed on the target data corresponding to the target operator of the network layer to be processed.

[00194] In an optional implementation, in a case where the parameter data corresponding to the network layer to be processed includes output data, determining that the plurality of target data blocks and the initial data satisfy the set chunking conditions includes: in a case where the initial data, the output data and each target data block respectively meets the memory requirement of the computing device, and the initial data, the output data, and each target data block respectively meets the DMA transmission requirement in the computing device,

determining that the plurality of target data blocks and the initial data satisfy the set chunking condition.

[00195] The memory requirement of the computing device here may be set according to the demand of the user and/or the demand of the computing device. For example, it may be determined whether a sum of data capacity of the initial data, the output data and each target data block is less than or equal to the set internal storage capacity of the computing device; and if yes, it is determined that the memory requirement of the computing device is met.

[00196] Or, whether the data capacity of the initial data is less than or equal to a first local internal storage capacity allocated for the initial data on the internal storage of the computing device may be determined, whether the data capacity of the output data is less than or equal to a second local internal storage capacity allocated for the output data on the internal storage of the computing device may be determined, and whether the data capacity of each target data block is less than or equal to a third local internal storage capacity allocated for the target data on the internal storage of the computing device may be determined; and if the initial data, the output data and each target data block each meets the requirement, it is determined that the memory requirement of the computing device is met.

[00197] When in implementation, a dedicated memory and a common memory may also be set; if the constant data is stored in the common memory, the input data and the output data are stored on the dedicated memory, and then whether the initial data, the output data and each target data block meet the corresponding memory requirement of the dedicated memory and the common memory may be judged; and if yes, it is determined that the memory requirement of the computing device is met. If the initial data is the input data, and the target data block is the target data block corresponding to the constant data, whether the data capacity of the initial data and the output data is less than or equal to the set internal storage capacity of the dedicated memory is judged, and whether each target data block is less than or equal to the set internal storage capacity of the common memory is judged; and if yes, it is determined that the memory requirement of the computing device is met.

[00198] Exemplarily, allocate attempt may be performed on the target data block, the initial data and the output data after each target data block is determined; and if the allocation attempt is successful, it is determined that the initial data, the output data and each target data block meet the memory requirement of the computing device.

[00199] The DMA transmission requirement may be determined according to the actual requirement. For example, if it is determined that the data capacity sum of the initial data, the output data and each target data block is less than or equal to the data capacity that can be transmitted by the DMA, that is, it is determined that the DMA task is established successfully,

it is determined that the DMA transmission requirement in the computing device is met.

[00200] When it is determined that the initial data, the output data and each target data block meet the memory requirement of the computing device, and the DMA transmission requirement in the computing device is met, it is determined that a plurality of target data blocks and the initial data satisfy the set chunking conditions.

[00201] By using the above method, when the initial data, the output data and each target data block meet the memory requirement of the computing device and the DMA transmission requirement in the computing device, it is determined that a plurality of target data blocks and the initial data satisfy the set chunking conditions, thereby ensuring that the chunking strategy is matched with the operational requirement of the computing device.

[00202] For the step S103, after the target operator and the target chunking strategy corresponding to each network layer to be processed in the target neural network are determined, the target neural network containing the target operator may be operated based on the target chunking strategy corresponding to at least one network layer to be processed.

[00203] For example, an image to be processed may be inputted into the target neural network, the computing device utilizes the target chunking strategy and the target operator corresponding to each network layer to be processed to perform feature extraction on the image to be processed and determine a detection result corresponding to the image to be processed; and for example, the detection result may be a type of the target object included in the image to be processed, position information of the target object, contour information of the target object, etc.

[00204] Exemplarily, FIG. 4 illustrates a schematic diagram of software and hardware scheduling of a computing device in a neural network operation method. The process for processing the parameter data of the network layer to be processed by utilizing PING-PONG schedule is described in conjunction with FIG. 4. The internal storage of the computing device may be divided into an initial data region, a scheduling data region PING, a scheduling data region PONG, an output data region PING and an output data region PONG. When the initial data is the input data, the scheduling data is the constant data; and when the initial data is the constant data, the scheduling data is the input data.

[00205] It can be seen from FIG. 4 that the computing device and the DMA are in parallel operations; DMA firstly transmits the initial data and scheduling PING (i.e., scheduling data PING) to a corresponding internal storage region of the computing device (that is, the initial data is transmitted to the internal storage region corresponding to the initial data region of the computing device, and the scheduling data PING is outputted to the internal storage region corresponding to the scheduling data region PING of the computing device); and the

computing device processes the initial data and scheduling PING, and at the same time, the DMA may also transmit the scheduling PONG (i.e., scheduling data PONG) to the internal storage region corresponding to the scheduling data PONG of the computing device.

[00206] After processing the initial data and the scheduling PING, the computing device generates output PING (i.e., output data PING), the output PING is arranged at the internal storage region corresponding to the output data region PING of the computing device, and the output PING is acquired by the DMA from the internal storage region corresponding to the output data region PING of the computing device, and then the output PING is transmitted to the corresponding external memory (such as DDR). Further, the computing device processes the received scheduling PONG, and at the same time, the DMA transmits the next scheduling PING to the internal storage region corresponding to the scheduling PING of the computing device, and the above process is repeated until the parameter data of the layer to be processed is processed completely.

[00207] It will be appreciated by those skilled in the art that in the above method of the specific implementation, the writing order of each step does not mean a strict execution order to constitute any limitation to the implementation process, and the specific execution order of each step should be determined by the functions and possible internal logics.

[00208] Based on the same concept, an embodiment of the present disclosure also provides a neural network operation apparatus. FIG. 5 illustrates a schematic diagram of an architecture of the neural network operation apparatus provided by an embodiment of the present disclosure. The neural network operation apparatus includes a first determining module 501, a second determining module 502 and an operation module 503, which are specifically as follows: the first determining module 501 is configured to determine a network layer to be processed in a target neural network; the second determining module 502 is configured to determine a target operator and a target chunking strategy corresponding to the network layer in the target neural network from a plurality of determined operators and chunking strategies, wherein each operator in the plurality of operators is used to implement a corresponding function of the network layer to be processed, and each chunking strategy in the plurality of chunking strategies matches with operational requirements of a computing device for operating the target neural network; and the operation module 503 is configured to operate the target neural network containing the target operator based on the target chunking strategy corresponding to the network layer to be processed.

[00209] In a possible implementation, the chunking strategy is used to perform chunking on parameter data of the target operator corresponding to the network layer to be processed; and among the plurality of chunking strategies, the resource consumption for operating the

network layer to be processed based on the parameter data obtained by chunking the parameter data of the target operator by using the target chunking strategy is minimal.

[00210] In a possible implementation, the second determining module 502 is configured to, in a case where there are a plurality of network layers to be processed, when determining the target operator and the target chunking strategy corresponding to the network layer to be processed in the target neural network from a plurality of determined operators and chunking strategies, determine a target candidate operator corresponding to the network layer to be processed from the plurality of operators, and determine a target candidate chunking strategy matched with the target candidate operator from the plurality of chunking strategies for each network layer to be processed in the target neural network; and determine the target operator and the target chunking strategy corresponding to each network layer to be processed based on the target candidate operators and target candidate chunking strategies corresponding to the respective network layers to be processed in a case where there are a plurality of target candidate operators and/or a plurality of target candidate chunking strategies corresponding to any network layer to be processed.

[00211] In a possible implementation, the second determining module 502 is configured to, when determining the target operator and the target chunking strategy corresponding to each network layer to be processed based on the target candidate operators and the target candidate chunking strategies corresponding to the respective network layers to be processed, determine a plurality of test networks corresponding to the target neural network based on the target candidate operators corresponding to the respective network layers to be processed and the target candidate chunking strategy corresponding to the target candidate operator, wherein each test network includes one of the target candidate operators corresponding to each network layer to be processed and a target candidate chunking strategy matched with the target candidate operator; operate a plurality of test networks respectively to obtain a plurality of test results, wherein each test network corresponds to one test result; select the target test network from the plurality of test networks based on the plurality of test results; and determine the target candidate operator and the target candidate chunking strategy of the network layer to be processed in the target test network as the target operator and the target chunking strategy respectively corresponding to the network layer to be processed in the target neural network.

[00212] In a possible implementation, the second determining module 502 is configured to, when determining the target candidate operator corresponding to the network layer to be processed from the plurality of operators, and determining the target candidate chunking strategy matched with the target candidate operator from the plurality of chunking strategies for each network layer to be processed in the target neural network, determine one or more first

candidate operators from the plurality of operators for the network layer to be processed; and select one or more target candidate operators corresponding to the network layer to be processed from the first candidate operators and one or more target candidate chunking strategies corresponding to the target candidate operator from the plurality of chunking strategies based on a resource consumption situation for the first candidate operator in each chunking strategy of the plurality of chunking strategies.

[00213] In a possible implementation, the resource consumption situation is represented by a computational overhead value, and the second determining module 502 is configured to determine the computational overhead value for the first candidate operator in each chunking strategy based on the steps of: determine a restricted scenario corresponding to the first candidate operator with a preset size, wherein the restricted scenario is determined based on a computing time cost and a transmission time cost for the data capacity corresponding to the first candidate operator with the preset size; determine a total direct memory access (DMA) data transmission amount, a number of DMA tasks and data conversion overhead corresponding to the first candidate operator in the chunking strategy based on a result of chunking of the chunking strategy in a case where the restricted scenario is a bandwidth restricted scenario; and determine the computational overhead value for the first candidate operator in the chunking strategy based on the total DMA data transmission amount, the number of the DMA tasks, the data conversion overhead, and a DMA rate corresponding to the computing device, and DMA task overhead; and wherein the data conversion overhead is time consumed in performing a data arrangement conversion on input data corresponding to the first candidate operator based on a target data arrangement corresponding to the first candidate operator; and determine the computing time cost for the parameter data corresponding to the first candidate operator in the chunking strategy, a number of operator calls for the first candidate operator, a total amount of initial data transmission, the number of DMA tasks and data conversion overhead based on the result of chunking of the chunking strategy in a case where the restricted scenario is a computation restricted scenario; and determine the computational overhead value of the first candidate operator in the chunking strategy based on the computing time cost, the number of operator calls, the total amount of initial data transmission, the data conversion overhead, the DMA task overhead, the number of DMA tasks and the DMA rate corresponding to the computing device.

[00214] In a possible implementation, the second determining module 502 is configured to, when selecting one or more target candidate operators corresponding to the network layer to be processed and one or more target candidate chunking strategies corresponding to the target candidate operator from the first candidate operators and the plurality of chunking strategies

based on the resource consumption situation of the first candidate operator in each chunking strategy in the plurality of chunking strategies, select a target resource consumption situation satisfying a predetermined condition from a plurality of resource consumption situations corresponding to the first candidate operator, wherein one first candidate operator corresponds to one resource consumption situation in one chunking strategy; determine the chunking strategy corresponding to the target resource consumption situation as the candidate chunking strategy, operate the network layer to be processed containing a second candidate operator corresponding to the target resource consumption situation based on the candidate chunking strategy, and determine a test result corresponding to the candidate chunking strategy and the second candidate operator; and determine one or more target candidate operators corresponding to the network layer to be processed and a target candidate chunking strategy corresponding to the target candidate operator based on the test result.

[00215] In a possible implementation, prior to selecting one or more target candidate operators corresponding to the network layer to be processed from the first candidate operators and the target candidate chunking strategies corresponding to the target candidate operators from the plurality of chunking strategies, the apparatus further includes: an aligning module 504, configured to align the parameter data corresponding to the first candidate operator based on determined minimal granularity information corresponding to the target neural network to obtain the aligned parameter data corresponding to the first candidate operator; wherein the minimal granularity information includes minimal granularities corresponding to the parameter data in different dimensions; and the size of the aligned parameter data in different dimensions of is an integer multiple of the minimal granularity in the corresponding dimension indicated by the minimal granularity information.

[00216] In a possible implementation, in a case where the parameter data includes input data and constant data, the plurality of chunking strategies include at least one of: all input data is taken as initial data, and the constant data is chunked in a specified dimension based on the determined dimension parameter of the constant data to obtain a result of chunking; and the initial data is the data written into an initial data region allocated by the direct memory access (DMA) task when the computing device operates the target neural network; all constant data is taken as the initial data, and the input data is chunked in the specified dimension according to the determined dimension parameter of the input data to obtain the result of chunking; part of the input data is taken as the initial data, and the constant data is chunked in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking; wherein a target size of the part of the input data is determined according to the minimal granularity of the input data in a first dimension; and part of the constant data is

taken as the initial data, and the input data is chunked in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking; wherein the target size of the part of the constant data is determined according to the minimal granularity of the constant data in the first dimension.

5 **[00217]** In a possible implementation, the part of the input data is taken as the initial data, and the constant data is chunked in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking, which includes: the target size of the part of input data is determined based on i times of the minimal granularity of the input data in the first dimension; the part of the input data of the target size is taken as the initial data,
10 and the constant data is chunked in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking; where i is a positive integer that makes the data capacity of the part of the input data and the data capacity of the constant data block determined based on the minimum granularity of the dimension parameter of the constant data meet the memory requirement of the computing device after the target size of the
15 part of the input data is determined.

[00218] In a possible implementation, the part of the constant data is taken as the initial data, and the input data is chunked in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking, which includes: the target size of the part of the constant data is determined based on j times of the minimal
20 granularity of the constant data in the first dimension; the part of the constant data of the target size is taken as the initial data, and the input data is chunked in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking; and where j is a positive integer that makes the data capacity of the part of the constant data and the data capacity of the input data block determined based on the minimum granularity of the
25 dimension parameter of the input data meet the memory requirement of the computing device after the target size of the part of the constant data is determined.

[00219] In a possible implementation, in a case where the specified dimension is one dimension, and the dimension parameter includes the first dimension, the constant data and the input data are taken as the target data respectively, and the one-dimensional chunking is
30 performed on the target data based on the determined first dimension of the target data, which includes: k times of the minimal granularity corresponding to the target data in the first dimension is determined as a target chunking size, and one-dimensional chunking is performed on the target data according to the first dimension based on the target chunking size to obtain a plurality of target data blocks corresponding to the target data; where k is a positive integer; in
35 a case where it is determined that the plurality of target data blocks and the initial data satisfy a

set chunking condition, $k+1$ times of the minimal granularity corresponding to the target data in the first dimension is taken as an updated target chunking size, and return to the step of performing the one-dimensional chunking on the target data according to the first dimension based on the target chunking size until it is determined that the plurality of target data blocks and the initial data do not satisfy the set chunking condition; and k times of the minimal granularity corresponding to the target data in the first dimension is determined as the result of chunking; and in a case where the initial data and the plurality of target data blocks generated when k is equal to 1 do not satisfy the set chunking conditions, the result of chunking is determined as a failure of one-dimensional chunking.

[00220] In a possible implementation, in a case where the specified dimension is two dimensions, and the dimension parameter includes a second dimension, the constant data and the input data are taken as the target data respectively, and the two-dimensional chunking is performed on the target data based on the determined first dimension and second dimension of the target data, which includes: y times of the minimal granularity corresponding to the target data in the first dimension is determined as a first target chunking size, and the one-dimensional chunking is performed on the target data according to the first dimension based on the first target chunking size to obtain a plurality of intermediate data blocks corresponding to the target data; where y is a positive integer; x times of the minimal granularity corresponding to the target data in the second dimension is determined as the second target chunking size; the two-dimensional chunking is performed on each intermediate data block according to the second dimension based on the second target chunking size to obtain a plurality of target data blocks corresponding to a plurality of intermediate data blocks; where x is a positive integer; and in a case where it is determined that the plurality of target data blocks and the initial data satisfy the set chunking condition, $x+1$ times of the minimal granularity corresponding to the target data in the second dimension is taken as the updated second target chunking size, and return to the step of performing the two-dimensional chunking on each intermediate data block according to the second dimension based on the second target chunking size until it is determined that the plurality of target data blocks and the initial data do not satisfy the set chunking condition; and x times of the minimal granularity corresponding to the target data in the second dimension is determined as the result of chunking.

[00221] In a possible implementation, in a case where the parameter data corresponding to the network layer to be processed includes output data, it is determined that the plurality of target data blocks and the initial data satisfy the set chunking block, which includes: in a case where the initial data, the output data and each target data block respectively meets the memory requirement of the computing device, and the initial data, the output data and each target data

block each meets the DMA transmission requirement in the computing device, it is determined that the plurality of target data blocks and the initial data satisfy the set chunking condition.

[00222] In some embodiments, functions or modules of the apparatus provided in the embodiments of the present disclosure may be used to execute the method described in the above method embodiments, which may be specifically implemented by referring to the above descriptions of the method embodiments, and are not repeated here for brevity.

[00223] Based on the same technical concept, an embodiment of the present disclosure further provides an electronic device. FIG. 6 is a schematic structure diagram of an electronic device provided by an embodiment of the present invention. The electronic device includes a processor 601, a memory 602 and a bus 603. The memory 602 is used to store execution instructions and includes an internal storage 6021 and an external storage 6022; the internal storage 6021 here may also be referred to as an internal storage and is used to temporarily store operation data in the processor 601 and data exchanged with the external storage 6022 such as a hard disk; the processor 601 exchanges data with the external storage 6022 through the internal storage 6021; and when the electronic device 600 operates, the processor 601 is communicated with the memory 602 through the bus 603, so that the processor 601 executes instructions below: determining a network layer to be processed in a target neural network; determining a target operator and a target chunking strategy corresponding to a network layer to be processed in the target neural network from a plurality of determined operators and respective chunking strategies, wherein each operator in the plurality of operators is used to implement a corresponding function of the network layer to be processed, and each chunking strategy in the plurality of chunking strategies is matched with the operational requirement of a computing device for operating the target neural network; and operating the target neural network containing the target operator based on the target chunking strategy corresponding to the network layer to be processed.

[00224] Furthermore, an embodiment of the present disclosure also provides a computer readable storage medium; and the computer readable storage medium has computer programs stored thereon, and the computer programs, when executed by a processor, implement steps of the neural network operating method in the above method embodiments. The storage medium may be a volatile or non-volatile computer readable storage medium.

[00225] An embodiment of the present disclosure also provides a computer program product. The computer program product carries program codes. Instructions included by the program codes may be used to execute the steps of the neural network operating method in the above method embodiments. For details, please refer to the above method embodiments, which are not repeated here.

[00226] The computer program product may be implemented specifically by hardware, software or a combination thereof. In an optional embodiment, the computer program product is specifically embodied as a computer storage medium. In another optional embodiment, the computer program product is specifically embodied as a software product, such as software development kit (SDK) and the like.

[00227] Those skilled in the art may clearly learn about that for the convenience and brevity of description, the specific working process of the above systems and apparatuses may refer to the corresponding processes in the foregoing method embodiments, which is not repeated here. In the embodiments provided in the present disclosure, it should be understood that the disclosed systems, apparatuses and methods may be implemented in other ways. The apparatus embodiments described above are merely illustrative. For example, the division of the units is only a logical function division, and there may be other divisions in actual implementation, for another example, a plurality of units or components may be combined or integrated into another system, or some features may be ignored or not implemented. In addition, the displayed or discussed mutual coupling or direct coupling or communication connection may be indirect coupling or communication connection through communication interfaces, apparatuses or units, and may be in electrical, mechanical or other forms.

[00228] The units serving as separate components may or may not be physically separated, and the components serving as unit display may or may not be physical units, that is, the units may be located in one place or may be distributed on a plurality of network units. Some or all of the units may be selected according to actual needs to achieve the objectives of the solutions of the embodiments.

[00229] Furthermore, all functional units in various embodiments of the present disclosure may be integrated into one processing unit, may also be physically separated, or two or more of the functional units may be integrated into one unit.

[00230] If the function is implemented in the form of a software functional unit and sold or taken as an independent product, it can be stored in a processor-executable non-volatile computer readable storage medium. Based on this understanding, the technical solutions of the present disclosure or the part contributing to the prior art or part of the technical solutions can be essentially embodied in a form of software products. The computer software product can be stored in a computer readable storage medium including a plurality of instructions for making a computer device (which may be a personal computer, a server, or a network device, etc.) execute all or some steps of the methods described in each embodiment of the present disclosure. The foregoing memory medium includes various media capable of storing the program codes, such as a USB flash drive, a portable hard disk, a Read-Only Memory (ROM),

a Random Access Memory (RAM), a magnetic disk or a compact disk, etc.

[00231] The above are only specific implementations of the present disclosure, but the protection scope of the present disclosure is not limited thereto. Any simple variations or replacements made by the technical skilled familiar with the prior art within the technical scope disclosed by the present disclosure shall be covered by the protection scope of the present disclosure. Therefore, the protection scope of the present disclosure should be subjected to the protection scope defined by the claims.

CLAIMS

1. A neural network operation method, comprising:

determining a network layer to be processed in a target neural network;

determining a target operator and a target chunking strategy corresponding to the network layer to be processed in the target neural network from a plurality of determined operators and a plurality of determined chunking strategies, wherein the plurality of operators are configured to implement corresponding functions of the network layer to be processed, and the plurality of chunking strategies match with operational requirements of a computing device for operating the target neural network; and

operating the target neural network containing the target operator based on the target chunking strategy corresponding to the network layer to be processed.

2. The method according to claim 1, wherein the chunking strategy is configured to perform chunking on parameter data of the target operator corresponding to the network layer to be processed; and

among the plurality of chunking strategies, resource consumption for operating the network layer to be processed based on parameter data obtained by chunking the parameter data of the target operator with the target chunking strategy is minimal.

3. The method according to claim 1 or 2, wherein based on a plurality of network layers to be processed, said determining the target operator and the target chunking strategy corresponding to the network layer to be processed in the target neural network from a plurality of determined operators and a plurality of determined chunking strategies includes:

for any network layer to be processed in the target neural network, determining one or more target candidate operators corresponding to the network layer to be processed from the plurality of operators, and determining one or more target candidate chunking strategies matched with the target candidate operator from the plurality of chunking strategies; and

in response to a plurality of target candidate operators and/or a plurality of target candidate chunking strategies corresponding to any network layer to be processed, determining the target operator and the target chunking strategy corresponding to the network layer to be processed based on the target candidate operators and the target candidate chunking strategies corresponding to the network layer to be processed.

4. The method according to claim 3, wherein said determining the target operator and the

target chunking strategy corresponding to the network layer to be processed based on the target candidate operators and the target candidate chunking strategies corresponding to the network layer to be processed includes:

determining a plurality of test networks corresponding to the target neural network based on the target candidate operator corresponding to the network layer to be processed and the target candidate chunking strategy corresponding to the target candidate operator, wherein any test network in the plurality of test networks includes one target candidate operator corresponding to the network layer to be processed and one target candidate chunking strategy matched with the target candidate operator;

operating the plurality of test networks respectively to obtain a plurality of test results, wherein any test network in the plurality of test networks corresponds to one test result;

selecting a target test network from the plurality of test networks based on the plurality of test results; and

determining the target candidate operator and the target candidate chunking strategy of the network layer to be processed in the target test network as the target operator and the target chunking strategy corresponding to the network layer to be processed in the target neural network.

5. The method according to claim 3 or 4, wherein said determining one or more target candidate operators corresponding to the network layer to be processed from the plurality of operators, and determining one or more target candidate chunking strategies matched with the target candidate operator from the plurality of chunking strategies for any network layer to be processed in the target neural network includes:

determining, for the network layer to be processed, one or more first candidate operators from the plurality of operators; and

selecting one or more target candidate operators corresponding to the network layer to be processed from the first candidate operators and the target candidate chunking strategies corresponding to the target candidate operators from the plurality of chunking strategies, based on a resource consumption situation for the first candidate operator in each chunking strategy of the plurality of chunking strategies.

6. The method according to claim 5, wherein the resource consumption situation is represented by a computational overhead value, and the computational overhead value for the first candidate operator in each chunking strategy is determined by:

determining a restricted scenario for the first candidate operator with a preset size,

wherein the restricted scenario is determined based on a computing time cost and a transmission time cost for a data capacity corresponding to the first candidate operator with the preset size;

in response to the restricted scenario being a bandwidth restricted scenario, determining a total direct memory access (DMA) data transmission amount, a number of DMA tasks, and data conversion overhead corresponding to the first candidate operator in the chunking strategy based on a result of chunking by the chunking strategy, and determining the computational overhead value for the first candidate operator in the chunking strategy based on the total DMA data transmission amount, the number of the DMA tasks, the data conversion overhead, a DMA rate corresponding to the computing device, and DMA task overhead, wherein the data conversion overhead is time consumed in performing a data arrangement conversion on input data corresponding to the first candidate operator based on a target data arrangement corresponding to the first candidate operator;

in response to the restricted scenario being a computation restricted scenario, determining the computing time cost for the parameter data corresponding to the first candidate operator in the chunking strategy, a number of operator calls for the first candidate operator, a total amount of initial data transmission, the number of DMA tasks, and the data conversion overhead based on a result of chunking by the chunking strategy, and determining the computational overhead value for the first candidate operator in the chunking strategy based on the computing time cost, the number of operator calls, the total amount of initial data transmission, the data conversion overhead, the DMA task overhead, the number of the DMA tasks, and the DMA rate corresponding to the computing device.

7. The method according to claim 5 or 6, wherein said selecting one or more target candidate operators corresponding to the network layer to be processed and the target candidate chunking strategies corresponding to the target candidate operators based on the resource consumption situation for the first candidate operator in each chunking strategy in the plurality of chunking strategies includes:

selecting a target resource consumption situation meeting a predetermined condition from a plurality of resource consumption situations corresponding to the first candidate operator, wherein one first candidate operator corresponds to one resource consumption situation in one chunking strategy;

determining the chunking strategy corresponding to the target resource consumption situation as the candidate chunking strategy, operating the network layer to be processed containing a second candidate operator corresponding to the target resource consumption

situation based on the candidate chunking strategy, and determining a test result corresponding to the candidate chunking strategy and the second candidate operator; and

determining one or more target candidate operators corresponding to the network layer to be processed and the one or more target candidate chunking strategies corresponding to the target candidate operators based on the test result.

8. The method according to any one of claims 5-7, wherein prior to selecting one or more target candidate operators corresponding to the network layer to be processed from the first candidate operators and the target candidate chunking strategies corresponding to the target candidate operators from the plurality of chunking strategies, the method further comprises:

aligning the parameter data corresponding to the first candidate operator based on determined minimal granularity information corresponding to the target neural network to obtain the aligned parameter data corresponding to the first candidate operator;

wherein the minimal granularity information includes minimal granularities corresponding to the parameter data in different dimensions; and in different dimensions the size of the aligned parameter data is of an integer multiple of the minimal granularity in the respective dimension indicated by the minimal granularity information.

9. The method according to any one of claims 1-8, wherein in response to the parameter data including input data and constant data, the plurality of chunking strategies include at least one of:

by taking all input data as initial data, chunking the constant data in a specified dimension based on a determined dimension parameter of the constant data to obtain a result of chunking, wherein the initial data is data written into an initial data region allocated by the direct memory access (DMA) task in response to the computing device operating the target neural network;

by taking all constant data as the initial data, chunking the input data in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking;

by taking part of the input data as the initial data, chunking the constant data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking, wherein a target size of the part of the input data is determined based on the minimal granularity of the input data in the first dimension; and

by taking part of the constant data as the initial data, chunking the input data in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking; wherein the target size of the part of the constant data is determined

based on the minimal granularity of the constant data in the first dimension.

10. The method according to claim 9, wherein said chunking, by taking part of the input data as the initial data, the constant data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking includes:

determining the target size of the part of the input data based on i times of the minimal granularity of the input data in the first dimension;

by taking respectively part of the input data of the target size as the initial data, chunking the constant data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking;

where i is a positive integer that makes the data capacity of the part of the input data and the data capacity of a constant data block determined based on the minimum granularity of the dimension parameter of the constant data meet a memory requirement of the computing device after the target size of the part of the input data is determined.

11. The method according to claim 9, wherein said chunking, by taking part of the constant data as the initial data, the input data in the specified dimension based on the determined dimension parameter of the constant data to obtain the result of chunking includes:

determining the target size of the part of the constant data based on j times of the minimal granularity of the constant data in the first dimension; and

taking respectively the part of the constant data of the target size as the initial data, and chunking the input data in the specified dimension based on the determined dimension parameter of the input data to obtain the result of chunking;

where j is a positive integer that makes the data capacity of the part of the constant data and the data capacity of the input data block determined based on the minimum granularity of the input data for the dimension parameters meet the memory requirement of the computing device after the target size of the part of the constant data is determined.

12. The method according to any one of claims 9-11, wherein in response to the specified dimension being one dimension and the dimension parameter including the first dimension, by taking the constant data and the input data respectively as the target data, performing a one-dimensional chunking on the target data based on the determined first dimension of the target data to obtain the result of chunking includes:

determining k times of the minimal granularity corresponding to the target data in the first dimension as a target chunking size, performing the one-dimensional chunking on the target

data by the first dimension based on the target chunking size to obtain a plurality of target data blocks corresponding to the target data; where k is a positive integer;

in response to determining that the plurality of target data blocks and initial data meet a set chunking condition, taking $k+1$ times of the minimal granularity corresponding the target data in the first dimension as an updated target chunking size, returning to the step of performing the one-dimensional chunking on the target data by the first dimension based on the target chunking size, until determining that the plurality of target data blocks and the initial data do not meet the set chunking condition, and determining k times of the minimal granularity corresponding to the target data in first dimension as the result of chunking; and

in response to the plurality of target data blocks generated with k equal to 1 and the initial data not meeting the set chunking condition, determining the result of chunking as a failure of one-dimensional chunking.

13. The method according to any one of claims 9-12, wherein in response to the specified dimension being two dimensions and the dimension parameter including a second dimension, by taking the constant data and the input data respectively as the target data, performing a two-dimensional chunking on the target data based on the determined first dimension and second dimension of the target data to obtain the result of chunking includes:

determining y times of the minimal granularity corresponding to the target data in the first dimension as a first target chunking size, and performing the one-dimensional chunking on the target data by the first dimension based on the first target chunking size to obtain a plurality of intermediate data blocks corresponding to the target data, where y is a positive integer;

determining x times of the minimal granularity corresponding to the target data in the second dimension as a second target chunking size, and performing the two-dimensional chunking on the at least one intermediate data block by the second dimension based on the second target chunking size to obtain a plurality of target data blocks corresponding to respective intermediate data blocks, where x is a positive integer; and

in response to determining that the plurality of target data blocks and the initial data meet the set chunking condition, taking $x+1$ times of the minimal granularity corresponding to the target data in the second dimension as an updated second target chunking size, returning to the step of performing the two-dimensional chunking on the at least one intermediate data block by the second dimension based on the second target chunking size, until determining that the plurality of target data blocks and the initial data do not meet the set chunking condition, and determining x times of the minimal granularity corresponding to the target data in the second dimension as the result of chunking.

14. The method according to claim 12 or 13, wherein in response to the parameter data corresponding to the network layer to be processed including output data, determining that the plurality of target data blocks and the initial data meet the set chunking condition includes:

5 in response to determining that the initial data, the output data, and at least one target data block respectively meets the memory requirement of the computing device, and that the initial data, the output data, and at least one target data block respectively meets a DMA transmission requirement of the computing device, determining that the plurality of target data blocks and the initial data meet the set chunking condition.

10 15. A neural network operation apparatus, comprising:

a first determining module, configured to determine a network layer to be processed in a target neural network;

15 a second determining module, configured to determine a target operator and a target chunking strategy corresponding to the network layer to be processed in the target neural network from a plurality of determined operators and a plurality of chunking strategies, wherein the plurality of operators are configured to implement corresponding functions of the network layer to be processed, and the plurality of chunking strategies match with operational requirements of a computing device for operating the target neural network; and

20 an operation module, configured to operate the target neural network containing the target operator based on the target chunking strategy corresponding to the network layer to be processed.

25 16. An electronic device, comprising: a processor, a memory and a bus, wherein the memory stores machine readable instructions executable by the processor, when the electronic device operates, the processor communicates with the memory through the bus, and the machine readable instructions, when executed by the processor, perform the steps of the neural network operation method according to any of claims 1-14.

30 17. A computer readable storage medium having computer programs stored thereon, wherein the computer programs, when executed by a processor, perform the steps of the neural network operation method according to any of claims 1-14.

35 18. A computer program, comprising computer readable codes, wherein when the computer readable codes are executed in an electronic device, a processor in the electronic

device performs a method for implementing any of claims 1-14.