(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 117396846 A
(43)申请公布日 2024.01.12

(21)申请号 202280038619.4
(22)申请日 2022.06.09
(30)优先权数据
17/350,550 2021.06.17 US
(85)PCT国际申请进入国家阶段日
2023.11.28
(86)PCT国际申请的申请数据
PCT/EP2022/065660 2022.06.09
(87)PCT国际申请的公布数据
WO2022/263277 EN 2022.12.22
(71)申请人 国际商业机器公司
地址 美国纽约
(72)发明人 C·里彻特纳 J·布拉德伯里 L·阿尔巴拉卡特
(74)专利代理机构 北京市中咨律师事务所 11247
专利代理师 刘薇 于静
(51)Int.Cl.
*G06F 9/30*(2006.01)

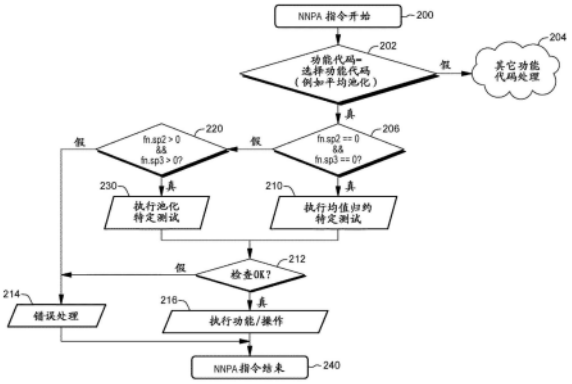
权利要求书3页 说明书30页 附图18页

(54)发明名称

具有不同操作参数验证的执行多个操作的单个功能

(57)摘要

获得要执行的功能的指示,其中该功能是指令的一个功能并被配置为执行多个操作。确定多个操作中的要执行的操作,并且使用一组值和对应的一组关系来验证一组功能特定参数。该组值和对应的一组关系基于要执行的操作。一组值和对应的一组关系将被用于要执行的操作,而另一组值和对应的一组关系将被用于多个操作中的另一个操作。




CN 117396846 A

1.一种用于促进计算环境内的处理的计算机程序产品,所述计算机程序产品包括:

一个或多个计算机可读存储介质和共同存储在所述一个或多个计算机可读存储介质上的程序指令,所述程序指令用于执行一种方法,所述方法包括:

获得要执行的功能的指示,所述功能是指令的一个功能并被配置为执行多个操作;

确定所述多个操作中的要执行的操作;以及

使用一组值和对应的一组关系来验证一组功能特定参数,其中,所述一组值和对应的一组关系基于所述要执行的操作,其中,一组值和对应的一组关系要被用于所述要执行的操作,而另一组值和对应的一组关系要被用于所述多个操作中的另一个操作。

2.根据权利要求1所述的计算机程序产品,其中,确定要执行的操作包括:对照至少一个特定值检查一个或多个功能特定参数,其中,基于所述一个或多个功能特定参数相对于所述至少一个特定值具有第一选择关系,所述操作是一个操作,并且基于所述一个或多个功能特定参数相对于所述至少一个特定值具有第二选择关系,所述操作是另一个操作。

3.根据权利要求2所述的计算机程序产品,其中,所述功能包括平均池化功能,所述一个或多个功能特定参数包括一个或多个步长值,其中,步长值是当计算一个或多个相邻输出张量元素时滑动窗口在输入张量上移动的量,所述至少一个特定值包括零,所述第一选择关系包括等于,并且基于所述一个或多个步长值等于零,所述操作是均值归约操作。

4.根据权利要求3所述的计算机程序产品,其中,所述第二选择关系包括大于,并且基于所述一个或多个步长值大于零,所述操作是池化操作。

5.根据前述权利要求中的任一项所述的计算机程序产品,其中,所述一组功能特定参数包括一个或多个选择维度窗口大小值,并且其中,选择维度窗口大小值指定滑动窗口包含的在选择维度上的元素数量,所述滑动窗口被配置为在所述功能的输入张量上移动以产生输出张量。

6.根据权利要求5所述的计算机程序产品,其中,所述功能包括平均池化功能,所述操作包括均值归约操作,并且要用于验证所述一组功能特定参数的所述一组值和对应的一组关系包括:选择输入张量的一个维度的一个值,对应的关系是等于;所述选择输入张量的另一个维度的另一个值,对应的关系是等于;以及选择值,对应的关系是小于或等于。

7.根据权利要求6所述的计算机程序产品,其中,所述验证包括:检查维度2窗口大小的值等于第一输入张量的维度2的值,维度3窗口大小的值等于所述第一输入张量的维度3的值,所述维度2窗口大小的值小于或等于所述选择值,并且所述维度3窗口大小的值小于或等于所述选择值。

8.根据权利要求5所述的计算机程序产品,其中,所述功能包括平均池化功能,所述操作包括池化操作,并且要用于验证所述一组功能特定参数的所述一组值和对应的一组关系包括:选择输入张量的一个维度的一个值,对应的关系是小于或等于;以及所述选择输入张量的另一个维度的另一个值,对应的关系是小于或等于。

9.根据权利要求8所述的计算机程序产品,其中,所述验证包括:检查维度2窗口大小的值小于或等于第一输入张量的维度2的值,以及维度3窗口大小的值小于或等于所述第一输入张量的维度3的值。

10.根据权利要求8所述的计算机程序产品,其中,所述方法进一步包括:确定填充类型是否被设置为特定类型,其中,所述填充类型指示窗口的哪些元素将要用于计算所述输出,

并且基于所述填充类型被设置为所述特定类型,执行所述验证。

11.根据权利要求10所述的计算机程序产品,其中,基于所述填充类型未被设置为所述特定类型,执行与输出张量的一个或多个维度相关的一个或多个检查。

12.根据前述权利要求中的任一项所述的计算机程序产品,其中,确定所述操作是基于输入张量的至少一个滑动窗口步长值,并且其中,所述一组功能特定参数包括输入张量的至少一个滑动窗口维度。

13.一种用于促进计算环境内的处理的计算机系统,所述计算机系统包括:

存储器;以及

与所述存储器通信的至少一个处理器,其中,所述计算机系统被配置为执行一种方法,所述方法包括:

获得要执行的功能的指示,所述功能是指令的一个功能并被配置为执行多个操作;

确定所述多个操作中的要执行的操作;以及

使用一组值和对应的一组关系来验证一组功能特定参数,其中,所述一组值和对应的一组关系基于所述要执行的操作,其中,一组值和对应的一组关系要被用于所述要执行的操作,而另一组值和对应的一组关系要被用于所述多个操作中的另一个操作。

14.根据权利要求13所述的计算机系统,其中,确定要执行的操作包括:对照至少一个特定值检查一个或多个功能特定参数,其中,基于所述一个或多个功能特定参数相对于所述至少一个特定值具有第一选择关系,所述操作是一个操作,并且基于所述一个或多个功能特定参数相对于所述至少一个特定值具有第二选择关系,所述操作是另一个操作。

15.根据权利要求14所述的计算机系统,其中,所述功能包括平均池化功能,所述一个或多个功能特定参数包括一个或多个步长值,其中,步长值是当计算一个或多个相邻输出张量元素时滑动窗口在输入张量上移动的量,所述至少一个特定值包括零,所述第一选择关系包括等于,并且基于所述一个或多个步长值等于零,所述操作是均值归约操作,并且其中,所述第二选择关系包括大于,并且基于所述一个或多个步长值大于零,所述操作是池化操作。

16.根据权利要求13至15中的任一项所述的计算机系统,其中,确定所述操作是基于输入张量的至少一个滑动窗口步长值,并且其中,所述一组功能特定参数包括输入张量的至少一个滑动窗口维度。

17.一种促进计算环境内的处理的计算机实现的方法,所述计算机实现的方法包括:

获得要执行的功能的指示,所述功能是指令的一个功能并被配置为执行多个操作;

确定所述多个操作中的要执行的操作;以及

使用一组值和对应的一组关系来验证一组功能特定参数,其中,所述一组值和对应的一组关系基于所述要执行的操作,其中,一组值和对应的一组关系要被用于所述要执行的操作,而另一组值和对应的一组关系要被用于所述多个操作中的另一个操作。

18.根据权利要求17所述的计算机实现的方法,其中,确定要执行的操作包括:对照至少一个特定值检查一个或多个功能特定参数,其中,基于所述一个或多个功能特定参数相对于所述至少一个特定值具有第一选择关系,所述操作是一个操作,并且基于所述一个或多个功能特定参数相对于所述至少一个特定值具有第二选择关系,所述操作是另一个操作。

19.根据权利要求18所述的计算机实现的方法,其中,所述功能包括平均池化功能,所述一个或多个功能特定参数包括一个或多个步长值,其中,步长值是当计算一个或多个相邻输出张量元素时滑动窗口在输入张量上移动的量,所述至少一个特定值包括零,所述第一选择关系包括等于,并且基于所述一个或多个步长值等于零,所述操作是均值归约操作,并且其中,所述第二选择关系包括大于,并且基于所述一个或多个步长值大于零,所述操作是池化操作。

20.根据权利要求17至19中的任一项所述的计算机实现的方法,其中,确定所述操作是基于输入张量的至少一个滑动窗口步长值,并且其中,所述一组功能特定参数包括输入张量的至少一个滑动窗口维度。

# 具有不同操作参数验证的执行多个操作的单个功能

## 背景技术

[0001] 一个或多个方面一般涉及促进计算环境内的处理,尤其涉及改进这样的处理。

[0002] 为了增强数据和/或计算密集型的计算环境中的处理,利用协处理器,诸如人工智能加速器(也称为神经网络处理器或神经网络加速器)。这样的加速器提供了在执行例如相关计算(诸如对矩阵或张量的计算)中使用的大量计算能力。

[0003] 作为示例,张量计算被用在复杂处理中,包括深度学习,它是机器学习的子集。深度学习或机器学习是人工智能的一个方面,被用于各种技术中,包括但不限于工程、制造、医学技术、汽车技术、计算机处理等。

[0004] 深度学习使用对张量数据进行操作的各种操作。每个操作是独立实现的,从而增加了开发和验证工作。

## 发明内容

[0005] 通过提供一种用于促进计算环境内的处理的计算机程序产品来克服现有技术的缺点,并且提供了附加的优点。计算机程序产品包括一个或多个计算机可读存储介质和共同存储在一个或多个计算机可读存储介质上以执行一种方法的程序指令。该方法包含获得要执行的功能的指示,其中该功能是指令的一个功能并被配置为执行多个操作。确定多个操作中的要执行的操作,并且使用一组值和对应的一组关系来验证一组功能特定参数。该组值和对应的一组关系基于要执行的操作。作为示例,一组值和对应的一组关系将被用于要执行的操作,而另一组值和对应的一组关系将被用于多个操作中的另一个操作。

[0006] 使用单个功能(例如,架构指令的单个功能)来执行多个操作,但降低了每一操作的参数验证、代码复杂度、代码重复和/或验证工作,从而改善系统性能。

[0007] 在一个示例中,确定要执行的操作包括:对照至少一个特定值检查一个或多个功能特定参数。基于一个或多个功能特定参数相对于至少一个特定值具有第一选择关系,操作是一个操作,并且基于一个或多个功能特定参数相对于至少一个特定值具有第二选择关系,操作是另一个操作。

[0008] 通过使用相同的功能特定参数但不同的关系来确定要执行的操作,降低了代码复杂度和验证工作量。

[0009] 作为示例,功能包括平均池化功能,一个或多个功能特定参数包括一个或多个步长值(stride value),其中步长值是当计算一个或多个相邻输出张量元素时滑动窗在输入张量上移动的量,至少一个特定值包括零,第一选择关系包括等于,基于一个或多个步长值等于零,操作是均值归约(mean-reduce)操作。

[0010] 进一步地,在一个示例中,第二选择关系包括大于,并且基于一个或多个步长值大于零,操作是池化操作。

[0011] 作为示例,一组功能特定参数包括一个或多个选择维度窗口大小值。选择维度窗口大小值指定滑动窗口包含的在选择维度上的元素数量,并且滑动窗口被配置为在功能的输入张量上移动以产生输出张量。

[0012] 在一个示例中,功能包括平均池化功能,操作包括均值归约操作,并且要用于验证一组功能特定参数的一组值和对应的一组关系包括:选择输入张量的一个维度的一个值,对应的关系是等于;选择输入张量的另一个维度的另一个值,对应的关系是等于;以及选择值,对应的关系是小于或等于。

[0013] 验证包括例如检查维度2窗口大小的值等于第一输入张量的维度2的值,维度3窗口大小的值等于第一输入张量的维度3的值,维度2窗口大小的值小于或等于选择值,并且维度3窗口大小的值小于或等于选择值。

[0014] 在一个示例中,功能包括平均池化功能,操作包括池化操作,并且要用于验证一组功能特定参数的一组值和对应的一组关系包括:选择输入张量的一个维度的一个值,对应的关系是小于或等于;以及选择输入张量的另一个维度的另一个值,对应的关系是小于或等于。

[0015] 验证包括例如检查维度2窗口大小的值小于或等于第一输入张量的维度2的值,以及维度3窗口大小的值小于或等于第一输入张量的维度3的值。

[0016] 在一个示例中,确定填充类型是否被设置为特定类型,其中填充类型指示窗口的哪些元素将被用于计算输出,并且对于一个或多个实施例,基于填充类型被设置为特定类型,执行验证。进一步地,在一个示例中,基于填充类型未被设置为特定类型,执行与输出张量的一个或多个维度相关的一个或多个检查。

[0017] 在一个示例中,确定操作是基于输入张量的至少一个滑动窗口步长值,并且一组功能特定参数包括输入张量的至少一个滑动窗口维度。

[0018] 本文还描述并要求保护与一个或多个方面相关的计算机实现的方法和系统。进一步地,本文还描述并要求保护与一个或多个方面相关的服务。

[0019] 通过本文所描述的技术来实现附加的特征和优点。其它实施例和方面在本文中被详细描述,并被视为所主张的方面的一部分。

## 附图说明

[0020] 在说明书结尾处的权利要求中作为示例特别指出并清楚地要求保护一个或多个方面。从结合附图的以下详细描述中,一个或多个方面的前述和目的、特征和优点将变得显而易见,在附图中:

[0021] 图1A描绘了用于结合和使用本发明的一个或多个方面的计算环境的一个示例;

[0022] 图1B描绘了根据本发明的一个或多个方面的图1A的处理器的进一步细节;

[0023] 图2描绘了根据本发明的一个或多个方面的与执行指令的单个功能相关联的处理的一个示例,该指令被配置为执行多个操作,但能够检查用于多个操作的不同参数条件;

[0024] 图3A描绘了根据本发明的一个或多个方面的神经网络处理辅助指令的格式的一个示例;

[0025] 图3B描绘了根据本发明的一个或多个方面的由神经网络处理辅助指令使用的通用寄存器的一个示例;

[0026] 图3C描绘了根据本发明的一个或多个方面的由神经网络处理辅助指令支持的功能代码的示例;

[0027] 图3D描绘了根据本发明的一个或多个方面的由神经网络处理辅助指令使用的另

一个通用寄存器的一个示例;

[0028] 图3E描绘了根据本发明的一个或多个方面的由神经网络处理辅助指令的查询功能使用的参数块的一个示例;

[0029] 图3F描绘了根据本发明的一个或多个方面的由神经网络处理辅助指令的一个或多个非查询功能使用的参数块的一个示例;

[0030] 图3G描绘了根据本发明的一个或多个方面的由神经网络处理辅助指令使用的张量描述符的一个示例;

[0031] 图4描绘了根据本发明的一个或多个方面的神经网络处理(NNP)数据类型1数据类型的格式的一个示例;

[0032] 图5A-图5C描绘了根据本发明的一个或多个方面的由神经网络处理辅助指令使用的输入数据布局的示例;

[0033] 图6A-图6C描绘了根据本发明的一个或多个方面的与图5A-图5C的输入数据布局对应的示例输出;

[0034] 图7A-图7C描绘了根据本发明的一个或多个方面的促进计算环境内的处理的一个示例;

[0035] 图8A描绘了用于结合和使用本发明的一个或多个方面的计算环境的另一个示例;

[0036] 图8B描绘了根据本发明的一个或多个方面的图8A的存储器的进一步细节的一个示例;

[0037] 图8C描绘了根据本发明的一个或多个方面的图8A的存储器的进一步细节的另一个示例;

[0038] 图9A描绘了用于结合和使用本发明的一个或多个方面的计算环境的又一示例;

[0039] 图9B描绘了根据本发明一个或多个方面的图9A的存储器的进一步细节;

[0040] 图10描绘了根据本发明的一个或多个方面的云计算环境的一个实施例;以及

[0041] 图11描绘了根据本发明的一个或多个方面的抽象模型层的一个示例。

## 具体实施方式

[0042] 根据本发明的一个或多个方面,提供了一种促进计算环境内的处理的能力。作为示例,提供了被配置为实现多个功能的指令,并且至少一个功能被配置为执行多个操作,其中每个操作具有不同的参数验证。通过使用一个功能来执行多个操作,但能够检查多个操作之间不同的参数边界条件,减少了代码复杂性、代码复制和/或验证工作。

[0043] 作为示例,被配置为实现多个操作的功能是平均池化功能,并且多个操作包括例如在深度学习中使用的均值归约操作和池化操作。平均池化功能执行不同的操作,但在算法上被简化为使用相同的输入张量和功能特定参数但具有不同的相对约束的公共算法功能。

[0044] 在一个示例中,被配置为执行多个操作的功能由指令发起。作为示例,指令是神经网络处理辅助指令,其是被配置为执行多个功能的单个指令(例如,在硬件/软件接口处的单个架构硬件机器指令)。每个功能被配置为单个指令(例如,单个架构指令)的一部分,从而减少了系统资源的使用和复杂性,并且提高了系统性能。进一步地,至少一个功能(AVGPOOL2D功能,其示例将在下面描述)被配置为基于输入数据(诸如由指令提供的功能特

定参数(例如,功能特定参数2和3,将在下面描述)的值)实施多个操作(例如,均值归约和池化)。

[0045] 指令可以是通用处理器指令集架构(ISA)的一部分,其由处理器(诸如通用处理器)上的程序分派。它可以由通用处理器执行,和/或指令的一个或多个功能可由耦合到通用处理器的或作为通用处理器的一部分的专用处理器(诸如被配置用于某些功能的协处理器)执行。其它变型也是可能的。

[0046] 参考图1A描述了用于结合和使用本发明的一个或多个方面的计算环境的一个实施例。作为示例,计算环境基于由纽约阿蒙克的国际商业机器公司提供的**z/Architecture®**指令集架构。在题为"z/Architecture Principles of Operation(z/Architecture操作原理)"的IBM出版物No.SA22-7832-12(第十三版,2019年9月)中描述了z/Architecture指令集架构的一个实施例,该出版物通过引用被整体并入本文。然而,z/Architecture指令集架构仅是一个示例架构;国际商业机器公司和/或其他实体的其他架构和/或其他类型的计算环境可以包括和/或使用本发明的一个或多个方面。z/Architecture和IBM是国际商业机器公司在至少一个管辖权内的商标或注册商标。

[0047] 参考图1A,计算环境100包括例如以通用计算设备的形式示出的计算机系统102。计算机系统102可以包括但不限于一个或多个通用处理器或处理单元104(例如,中央处理单元(CPU))、至少一个专用处理器(例如,神经网络处理器105)、存储器106(也称为系统存储器、主存储器、主存储、中央存储或存储,作为示例)、以及一个或多个输入/输出(I/O)接口108,它们经由一个或多个总线和/或其它连接彼此耦合。例如,处理器104、105和存储器106经由一条或多条总线110耦合到I/O接口108,并且处理器104、105经由一条或多条总线111彼此耦合。

[0048] 总线111例如是存储器或高速缓存一致性总线,并且总线110表示例如若干类型的总线结构中的任何一种或多种,包括存储器总线或存储器控制器、外围总线、加速图形端口、以及使用各种总线架构中的任何一种的处理器或局部总线。作为示例而非限制,这些体系结构包括工业标准体系结构(ISA)、微通道体系结构(MCA)、增强型ISA(EISA)、视频电子标准协会(VESA)局部总线、和外围部件互连(PCI)。

[0049] 作为示例,一个或多个专用处理器(例如,神经网络处理器)可以与一个或多个通用处理器分离但耦合到一个或多个通用处理器,和/或可以被嵌入在一个或多个通用处理器内。许多变型是可能的。

[0050] 存储器106可以包括例如高速缓存112,诸如共享高速缓存,其可以经由例如一个或多个总线111耦合到处理器104的本地高速缓存114和/或神经网络处理器105。进一步地,存储器106可以包括一个或多个程序或应用116以及至少一个操作系统118。示例操作系统包括由纽约阿蒙克的国际商业机器公司提供的**z/OS®**操作系统。z/OS是国际商业机器公司在至少一个管辖权内的商标或注册商标。也可使用由国际商业机器公司和/或其它实体提供的其它操作系统。存储器106还可以包括一个或多个计算机可读程序指令120,其可以被配置为执行本发明的各方面的实施例的功能。

[0051] 此外,在一个或多个实施例中,存储器106包括处理器固件122。处理器固件包括例如处理器的微码或毫码。它包括例如在实现更高级的机器代码时使用的硬件级指令和/或

数据结构。在一个实施例中,它包括例如通常作为包括可信软件的微码或毫码、底层硬件专用的微码或毫码递送的专有代码,并且控制操作系统对系统硬件的访问。

[0052] 计算机系统102可以经由例如I/O接口108与一个或多个外部设备130(诸如用户终端、磁带驱动器、指示设备、显示器和一个或多个数据存储设备134等)通信。数据存储设备134可以存储一个或多个程序136、一个或多个计算机可读程序指令138、和/或数据等。计算机可读程序指令可以被配置为执行本发明的各方面的实施例的功能。

[0053] 计算机系统102还可以经由例如I/O接口108与网络接口132通信,这使得计算机系统102能够与一个或多个网络通信,诸如局域网(LAN)、通用广域网(WAN)和/或公共网络(例如,因特网),从而提供与其他计算设备或系统的通信。

[0054] 计算机系统102可以包括和/或耦合到可移除/不可移除、易失性/非易失性计算机系统存储介质。例如,它可以包括和/或耦合到不可移除的非易失性磁介质(通常称为“硬盘驱动器”)、用于从可移除的非易失性磁盘(例如,“软盘”)读取和向其写入的磁盘驱动器、和/或用于从可移除的非易失性光盘(诸如CD-ROM、DVD-ROM或其它光学介质)读取或向其写入的光盘驱动器。应当理解,其它硬件和/或软件组件可以与计算机系统102结合使用。示例包括但不限于:微码或毫码、设备驱动器、冗余处理单元、外部磁盘驱动器阵列、RAID系统、磁带驱动器和数据档案存储系统等。

[0055] 计算机系统102可以与许多其它通用或专用计算系统环境或配置一起操作。适合与计算机系统102一起使用的公知的计算系统、环境和/或配置的示例包括但不限于个人计算机(PC)系统、服务器计算机系统、瘦客户端、胖客户端、手持式或膝上型设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品、网络PC、小型计算机系统、大型计算机系统、以及包括任何上述系统或设备的分布式云计算环境等。

[0056] 在一个示例中,处理器(例如,处理器104和/或处理器105)包括用于执行指令的多个功能组件(或其子集)。如图1B所示,这些功能组件包括例如指令取组件150,用于取得要执行的指令;指令解码单元152,用于对所取得的指令进行解码并获得经解码的指令的操作数;一个或多个指令执行组件154,用于执行经解码的指令;存储器访问组件156,用于在需要时访问存储器以用于指令执行;以及写回组件158,用于提供所执行的指令的结果。一个或多个组件可以在指令处理中访问和/或使用一个或多个寄存器160。进一步地,根据本发明的一个或多个方面,一个或多个组件可以包括一个或多个其它组件的至少一部分或可以访问一个或多个其它组件,该一个或多个其它组件用于基于执行单个功能而执行具有不同参数检查的多个操作,和/或用于执行例如神经网络处理辅助指令的神经网络处理辅助处理(或可以使用本发明的一个或多个方面的其它处理),如本文所描述的。一个或多个其他组件可以包括例如单个功能、多个操作-不同参数验证组件170和/或神经网络处理辅助组件172(和/或一个或多个其他组件)。

[0057] 根据本发明的一个或多个方面,执行能够执行多个功能的指令,并且至少一个功能实现具有不同参数验证的多个操作。参考图2进一步描述该处理的示例。

[0058] 参考图2,在一个示例中,在处理器(诸如通用处理器104)上发起指令(诸如神经网络处理辅助(NNPA)指令(或另一个指令))200。确定要执行的功能。这例如通过检查指令的功能代码来确定。如果要执行的功能不是选择功能代码,例如指定平均池化功能的功能代码(例如NNPA_AVGPOOL2D),则执行其它处理204。

[0059] 返回到查询202,然而,如果功能代码指定选择功能代码,例如指定平均池化功能的功能代码,则处理继续,如在此所描述的。在一个示例中,该处理由发起指令的通用处理器执行。然而,在其他实施例中,该处理可以由专用处理器(诸如神经网络加速器(例如,神经网络加速器105))或由另一通用处理器、专用处理器或其他处理器来执行。其它变型也是可能的。

[0060] 在一个示例中,基于指定AVGPOOL2D功能,通过所指定的操作来减小输入张量,例如输入张量1,以概括输入的窗口。窗口是例如具有定义的尺寸的输入张量的选择部分。通过在例如输入张量的维度2和3上移动2D滑动窗口来选择输入的窗口。窗口的概要是输出张量中的元素。通过例如由指令提供的功能特定参数(例如,功能特定参数4和功能特定参数5)来描述滑动窗口维度,在此描述其示例。

[0061] 在处理功能时,在一个实施例中,确定要执行的操作,因为该功能被配置为执行多个操作。在一个示例中,通过检查作为指令的输入的选择功能特定参数(例如,在与指令一起使用的参数块中)来确定要执行的操作。作为示例,选择功能特定参数是功能特定参数2(也称为fn.sp2)和功能特定参数3(也称为fn.sp3),其中每一个参数指定例如滑动窗口步长。滑动窗口步长或步长是当计算相邻输出张量元素时滑动窗口在输入张量1上移动的量。

[0062] 在一个示例中,确定功能特定参数2和功能特定参数3的值是否等于选择值,诸如零(206)。如果例如由fn.sp2指定的维度2步长的值和由fn.sp3指定的维度3步长的值等于选择值(例如零),则将执行均值归约操作,并因此,执行均值归约特定测试210。这些测试包括例如检查:

[0063] 填充类型等于选择填充类型,诸如有效。例如,检查作为功能的输入所提供特定的功能特定参数(例如功能特定参数1(也称为fn.sp1))的值。如果所指定的填充类型是有效,则窗口中的所有元素被添加到用于计算结果输出元素的集合。

[0064] 功能特定参数4(fn.sp4)的值是否等于第一输入张量的维度2(例如,e2)的值(例如,检查fn.sp4==in1.e2)。例如,对照in1.e2的值检查fn.sp4中的维度2窗口大小(也称为滑动窗口值)。

[0065] 功能特定参数5(fn.sp5)的值是否等于第一输入张量的维度3(例如,e3)的值(例如,检查fn.sp5==in1.e3)。例如,对照in1.e3的值检查fn.sp5中的维度3窗口大小(又称为滑动窗口值)。

[0066] 功能特定参数4的值是否小于或等于选择值(例如1024)(例如,fn.sp4<=1024)。例如,将维度2窗口大小与选择值(例如1024)进行比较。

[0067] 功能特定参数5的值是否小于或等于选择值(例如1024)(例如,fn.sp5<=1024)。例如,将维度3窗口大小与选择值(例如1024)进行比较。

[0068] 可以执行附加的、更少的和/或其他测试。

[0069] 如果测试不符合要求212,则执行错误处理214。然而,如果测试符合要求212,则执行功能/所选择的操作(例如AVGPOOL2D功能的均值归约操作)216。例如,在一个实施例中,通用处理器(例如,通用处理器104)发起神经网络处理辅助指令,并且对于某些功能(诸如非查询功能),像AVGPOOL2D功能一样,通用处理器向专用处理器(例如,神经网络处理器105)提供信息,诸如要执行的功能/操作的指示和用于输入数据(例如,一个或多个输入张量)的存储器地址信息,以使得专用处理器可以执行功能/操作,如在此所描述的。当功能完

成时,处理返回到通用处理器以完成指令。在其他实施例中,通用处理器或专用处理器发起指令,执行功能/操作并完成指令。其它变型是可能的。

[0070] 返回到查询206,如果例如功能特定参数2和功能特定参数3的值不等于选择值,诸如零,则进一步检查fn.sp2和fn.sp3的值是否例如大于选择值(例如零)(220)。如果例如由fn.sp2指定的维度2步长和由fn.sp3指定的维度3步长的值不大于选择值(例如0),则执行错误处理214。然而,如果由fn.sp2指定的维度2步长和由fn.sp3指定的维度3步长的值大于选择值(例如零),则将执行池化操作,并因此执行池化特定测试230。这些测试的示例包括例如检查:

[0071] fn.sp2和fn.sp3的值是否小于或等于特定值,诸如30。例如,将维度2步长值和维度3步长值与特定值(例如30)进行比较。

[0072] 由功能特定参数1(fn.sp1)指定的填充类型是否等于选择填充类型,诸如有效。如果fn.sp1中的值是例如有效,则检查在fn.sp4中指定的滑动窗口值(也称为维度2窗口大小)是否小于或等于第一输入张量的维度2的值(in1.e2)以及在fn.sp5中指定的滑动窗口值(也称为维度3窗口大小)是否小于或等于第一输入张量的维度3的值(in1.e3)。

[0073] 如果在一个示例中在fn.sp1中指定的填充类型不等于选择填充类型,诸如有效,则检查例如输出张量的维度2(E2)的值(例如,out.e2)是否等于向上取整(ceil)(in1.e2/fn.sp4)的值,并且检查输出张量的维度3(e3)的值(例如,out.e3)是否等于向上取整(in1.e3/fn.sp5)的值。也就是说,

[0074] $$\mathbf{01D2IS} = \left\lceil \frac{\mathbf{I1D2IS}}{\mathbf{D2S}} \right\rceil$$

[0075] $$\mathbf{01D3IS} = \left\lceil \frac{\mathbf{I1D3IS}}{\mathbf{D3S}} \right\rceil$$

[0076] 其中:

[0077] IxDyIS在张量描述符x中定义的输入张量x的维度y索引大小。

[0078] OxDyIS在张量描述符x中定义的输出张量x的维度y索引大小。

[0079] D2S维度2步长。

[0080] D3S维度3步长。

[0081] 可以执行附加的、更少和/或其他测试。

[0082] 如果测试不符合要求,则执行错误处理214。然而,如果这些测试符合要求212,则执行功能/所选择的操作(例如,AVGPOOL2D功能的池化操作)216,如本文所描述的。

[0083] 在完成功能/操作后,处理返回到通用处理器,并且指令完成240。

[0084] 如所指示的,在一个示例中,AVGPOOL2D功能被实现为指令(诸如神经网络处理辅助指令)的一部分。参考图3A-图3G描述与神经网络处理辅助指令、AVGPOOL2D功能、以及均值归约和池化操作有关的进一步细节。首先参考图3A,在一个示例中,神经网络处理辅助指令300具有RRE格式,该RRE格式表示具有扩展操作码(opcode)的寄存器和寄存器操作。在一个示例中,神经网络处理辅助指令300包括指示神经网络处理辅助操作的操作码(opcode)字段302(例如,位0-15)。在一个示例中,指令的位16-31被保留,并且将包含零。在本文关于指令、指令的功能和/或操作的描述中,指示了特定位置、特定字段和/或字段的特定大小(例如,特定字节和/或位)。然而,可以提供其他位置、字段和/或大小。进一步地,尽管可以

指定将位设置为特定值,例如一或零,但这仅是示例。在其它示例中,如果位被设置,则位可被设置为不同值,诸如相反值或另一个值。许多变型是可能的。

[0085] 在一个示例中,指令使用由指令隐含指定的多个通用寄存器。例如,神经网络处理辅助指令300使用隐含的寄存器,通用寄存器0和通用寄存器1,其示例分别参考图3B和图3D来描述。

[0086] 参考图3B,在一个示例中,通用寄存器0包括功能代码字段和可在指令完成时被更新的状态字段。作为示例,通用寄存器0包括响应代码字段310(例如,位0-15)、异常标志字段312(例如,位24-31)和功能代码字段314(例如,位56-63)。进一步地,在一个实例中,通用寄存器0的位16-23和32-55被保留并将包含零。一个或多个字段由指令所执行的特定功能使用。在一个示例中,并非所有字段都被所有功能使用。每个字段被如下描述:

[0087] 响应代码(RC)310:该字段(例如,位位置0-15)包含响应代码。当神经网络处理辅助指令的执行以条件代码(例如一)完成时,响应代码被存储。当遇到无效的输入条件时,非零值被存储到响应代码字段,它指示在执行期间识别的无效的输入条件的原因,并且设置所选择的条件代码,例如1。在一个示例中,被存储到响应代码字段的代码如下定义:

[0088]

| 响应代码 | 含义 |
|---|---|
| 0001 | 模型不支持如由参数块版本号指定的参数块的格式。 |
| 0002 | 在机器上未定义或安装所指定的功能。 |
| 0010 | 不支持所指定的张量数据布局格式。 |
| 0011 | 不支持所指定的张量数据类型。 |
| 0012 | 所指定的单个张量维度大于最大维度索引大小。 |
| 0013 | 所指定的张量的大小大于最大张量大小。 |
| 0014 | 所指定的张量地址未在 4K 字节边界上对齐。 |
| 0015 | 功能特定保存区域地址未在 4K 字节边界上对齐。 |
| F000-FFFF | 功能特定响应代码。这些响应代码是针对某些功能所定义的。 |

[0089] 异常标志(EF)312:该字段(例如,位位置24-31)包括异常标志。如果在执行指令期间检测到异常条件,则对应的异常标志控件(例如,位)将被设置为例如一;否则,控件保持不变。异常标志字段在指令的第一次调用之前被初始化为零。保留的标志在指令的执行期间不变。在一个示例中,被存储到异常标志字段的标志如下定义:

[0090]

| EF(位) | 含义 |
|---|---|
| 0 | 范围违反。当在输入张量中检测到非数字值或者非数字值被存储到输出张量时,设置该标志。该标志例如仅在指令以条件代码(例如,0)完成时有效。 |
| 1-7 | 保留 |

[0091] 功能代码(FC)314:该字段(例如,位位置56-63)包括功能代码。在图3C中描绘了用于神经网络处理辅助指令的所分配的功能代码的示例。所有其它功能代码未被分配。如果未分配的或未安装的功能代码被指定,则设置响应代码(例如0002(十六进制))和选择条件代码(例如1)。在执行期间不修改该字段。

[0092] 如所指示的,除了通用寄存器0之外,神经网络处理辅助指令还使用通用寄存器1,其示例在图3D中描绘。作为示例,24位寻址模式中的位40-63、31位寻址模式中的位33-63或64位寻址模式中的位0-63包括参数块的地址320。通用寄存器1的内容指定例如存储中的参数块的最左侧字节的逻辑地址。参数块将在双字边界上被指定;否则,识别规范异常。对于所有功能,不修改通用寄存器1的内容。

[0093] 作为示例,在访问寄存器模式中,访问寄存器1指定包含参数块、输入张量、输出张量和功能特定保存区域的地址空间。

[0094] 在一个示例中,参数块可取决于待执行的由指令所指定的功能而具有不同的格式。例如,指令的查询功能具有一种格式的参数块,而指令的其它功能具有另一种格式的参数块。在另一个示例中,所有功能使用相同的参数块格式。其它变型也是可能的。

[0095] 作为示例,参数块和/或参数块中的信息被存储在存储器中、硬件寄存器中和/或存储器和/或寄存器的组合中。其它示例也是可能的。

[0096] 参考图3E描述由查询功能(诸如NNPA查询可用功能(QAF)操作)使用的参数块的一个示例。如图所示,在一个示例中,NNPA查询可用功能参数块330包括例如:

[0097] 已安装功能向量332:参数块的该字段(例如,字节0-31)包括已安装功能向量。在一个示例中,已安装功能向量的位0-255分别对应于神经网络处理辅助指令的功能代码0-255。当位是例如一时,对应的功能被安装;否则,功能未被安装。

[0098] 已安装参数块格式向量334:参数块的该字段(例如,字节32-47)包括已安装参数块格式向量。在一个示例中,已安装参数块格式向量的位0-127对应于用于神经网络处理辅助指令的非查询功能的参数块格式0-127。当位是例如一时,对应的参数块格式被安装;否则,参数块格式未被安装。

[0099] 已安装数据类型336:参数块的该字段(例如,字节48-49)包括已安装数据类型向量。在一个示例中,已安装数据类型向量的位0-15对应于被安装的数据类型。当位是例如一时,对应的数据类型被安装;否则,数据类型未被安装。示例数据类型包括(附加的、更少的和/或其他数据类型是可能的):

| [0100] | 位 | 数据类型 |
|---|---|---|
| [0101] | 0 | NNP数据类型1 |
| [0102] | 1-15 | 保留 |

[0103] 已安装数据布局格式338:参数块的该字段(例如,字节52-55)包括已安装数据布局格式向量。在一个示例中,已安装数据布局格式向量的位0-31对应于被安装的数据布局格式。当位是例如一时,对应的数据布局格式被安装;否则,数据布局格式未被安装。示例数据布局格式包括(附加的、更少的和/或其他数据类型是可能的):

[0104]

| 位 | 数据布局格式 |
|---|---|
| 0 | 4D 特征张量 |
| 1 | 4D 核张量 |
| 2-31 | 保留 |

[0105] 最大维度索引大小340:参数块的该字段(例如,字节60-63)包括例如32位无符号二进制整数,其指定在用于任何指定张量的指定维度索引大小中的最大元素数量。在另一个示例中,最大维度索引大小指定在用于任何指定张量的指定维度索引大小中的最大字节数。其它示例也是可能的。

[0106] 最大张量大小342:参数块的该字段(例如,字节64-71)包括例如32位无符号二进制整数,其指定任何指定张量中的包括张量格式所需的任何填充字节的字节的最大数量。在另一个示例中,最大张量大小指定任何指定张量中的包括张量格式所需的任何填充的全部元素的最大数量。其它示例也是可能的。

[0107] 已安装NNP数据类型1转换向量344:参数块的该字段(例如,字节72-73)包括已安装NNP数据类型1转换向量。在一个示例中,已安装NNP数据类型1转换向量的位0-15对应于从/到NNP数据类型1格式的已安装数据类型转换。当位是一时,对应的转换被安装;否则,转换未被安装。

[0108] 可以指定附加的、更少的和/或其他转换。

[0109]

| 位 | 数据类型 |
|---|---|
| 0 | 保留 |
| 1 | BFP 微小格式 |
| 2 | BFP 短格式 |
| 3-15 | 保留 |

[0110] 尽管参考图3E描述了用于查询功能的参数块的一个示例,但是也可以使用用于查询功能(包括NNPA查询可用功能操作)的参数块的其他格式。在一个示例中,格式可以取决于要执行的查询功能的类型。进一步地,参数块和/或参数块的每个字段可以包括附加的、更少的和/或其他信息。

[0111] 除了用于查询功能的参数块之外,在一个示例中,存在用于非查询功能(诸如神经网络处理辅助指令的非查询功能)的参数块格式。参考图3F描述由非查询功能(诸如神经网络处理辅助指令的AVGPOOL2D功能)使用的参数块的一个示例。

[0112] 如图所示,在一个示例中,由例如神经网络处理辅助指令的非查询功能所采用的参数块350包括例如:

[0113] 参数块版本号352:参数块的该字段(例如,字节0-1)指定参数块的版本和大小。在一个示例中,参数块版本号的位0-8被保留并将包含零,并且参数块版本号的位9-15包含指定参数块的格式的无符号二进制整数。查询功能提供了指示可用的参数块格式的机制。当所指定的参数块的大小或格式不被模型支持时,在通用寄存器0中存储响应代码(例如,十

六进制的0001),并且指令通过设置条件代码(例如条件代码1)来完成。参数块版本号由程序指定,并在指令的执行期间不被修改。

[0114] 模型版本号354:参数块的该字段(例如,字节2)是标识执行指令(例如,特定的非查询功能)的模型的无符号二进制整数。当继续标志(在下面描述)是一时,模型版本号可以是用于解释参数块的继续状态缓冲器字段(在下面描述)的内容以恢复操作的目的的操作的输入。

[0115] 继续标志356:参数块的该字段(例如,位63)在例如一时指示操作部分地完成,并在继续状态缓冲器的内容可用于恢复操作。程序将继续标志初始化为零,并在为了恢复操作的目的而要重新执行指令的情况下不修改继续标志;否则,结果是不可预测的。

[0116] 如果继续标志在操作开始时被设置,并且参数块的内容自初始调用以来已经改变,则结果是不可预测的。

[0117] 功能特定保存区域地址358:参数块的该字段(例如,字节56-63)包括功能特定保存区域的逻辑地址。在一个示例中,功能特定保存区域地址将在4K字节边界上对齐;否则,在通用寄存器0中设置响应代码(例如十六进制的0015),并且指令以例如1的条件代码完成。地址受到当前寻址模式的限制。功能特定保存区域的大小取决于功能代码。

[0118] 当整个功能特定保存区域与程序事件记录(PER)存储区域指定相重叠时,在适用时,针对功能特定保存区域识别PER存储更改事件。当仅有功能特定保存区域的一部分与PER存储区域指定相重叠时,它是模型相关的,其中发生以下情况:

[0119] *在适用时,针对整个功能特定保存区域识别PER存储更改事件。

[0120] *在适用时,针对所存储的功能特定保存区域的部分识别PER存储更改事件。

[0121] 当整个参数块与PER存储区域指定相重叠时,在适用时,针对参数块识别PER存储更改事件。当仅有参数块的一部分与PER存储区域指定相重叠时,它是模型相关的,其中发生以下情况:

[0122] *在适用时,针对整个参数块识别PER存储更改事件。

[0123] *在适用时,针对所存储的参数块的部分,识别PER存储更改事件。

[0124] 当适用时,针对参数块识别PER零地址检测事件。在一个示例中,零地址检测不应用于张量地址或功能特定保存区域地址。

[0125] 输出张量描述符(例如,1-2)360/输入张量描述符(例如,1-3)365:参考图3G描述张量描述符的一个示例。在一个示例中,张量描述符360、365包括:

[0126] 数据布局格式382:张量描述符的该字段(例如,字节0)指定数据布局格式。有效的数据布局格式包括例如(附加的、更少的和/或其他数据布局格式是可能的):

[0127]

| 格式 | 描述 | 对齐(字节) |
|---|---|---|
| 0 | 4D特征张量 | 4096 |
| 1 | 4D核张量 | 4096 |
| 2-255 | 保留 | -- |

[0128] 如果不支持的或保留的数据布局格式被指定,则在通用寄存器0中存储例如0010(十六进制)的响应代码,并且指令通过设置例如1的条件代码来完成。

[0129] 数据类型384:该字段(例如,字节1)指定张量的数据类型。下面描述所支持的数据类型的示例(附加的、更少的和/或其他数据类型是可能的):

| [0130] 值 | 数据类型 | 数据大小(位) |
|---|---|---|
| [0131] 0 | NNP数据类型1 | 16 |
| [0132] 1-255 | 保留 | -- |

[0133] 如果不支持的或保留的数据类型被指定,则在通用寄存器0中存储例如0011(十六进制)的响应代码,并且指令通过设置条件代码(例如1)来完成。

[0134] 维度1-4索引大小386:整体地,维度索引大小一到四指定4D张量的形状。每个维度索引大小将要大于零且小于或等于最大维度索引大小(340,图3E);否则,在通用寄存器0中存储例如0012(十六进制)的响应代码,并且指令通过设置条件代码(例如1)完成。总张量大小将要小于或等于最大张量大小(342,图3E);否则,在通用寄存器0中存储,例如0013(十六进制)的响应代码,并且指令通过设置条件代码(例如1)完成。

[0135] 在一个示例中,为了确定具有NNPA数据类型1元素的4D特征张量中的字节数(即,总张量大小),使用以下:维度索引4*维度索引3*向上取整(维度索引2/32)*32*向上取整(维度索引1/64)*64*2。

[0136] 张量地址388:张量描述符的该字段(例如,字节24-31)包括张量的最左侧字节的逻辑地址。地址受到当前寻址模式的限制。

[0137] 如果地址未在相关联的数据布局格式的边界上对齐,则在通用寄存器0中存储例如0014(十六进制)的响应代码,并且指令通过设置条件代码(例如1)完成。

[0138] 在访问寄存器模式中,访问寄存器1指定包含存储中的所有活动输入和输出张量的地址空间。

[0139] 返回到图3F,在一个示例中,参数块350进一步包括可由特定功能使用的功能特定参数1-5(370),如本文所描述的。

[0140] 进一步地,在一个示例中,参数块350包括继续状态缓冲器字段375,其包括如果要恢复该指令的操作则要使用的数据(或数据的位置)。

[0141] 作为操作的输入,参数块的保留字段应包含零。当操作结束时,保留字段可以被存储为零或保持不变。

[0142] 尽管参考图3F描述了用于非查询功能的参数块的一个示例,但是可以使用用于非查询功能(包括神经网络处理辅助指令的非查询功能)的参数块的其他格式。在一个示例中,格式可以取决于要执行的功能的类型。进一步地,尽管参考图3G描述了张量描述符的一个示例,但是可以使用其他格式。进一步地,可以使用用于输入张量和输出张量的不同格式。其它变型是可能的。

[0143] 在下面描述关于由神经网络处理辅助指令的一个实施例所支持的各种功能的进一步细节:

[0144] 功能代码0:NNPA-QAF(查询可用功能)

[0145] 神经网络处理辅助(NNPA)查询功能提供了一种机制,用于指示所选择的信息,例如已安装功能的可用性、已安装参数块格式、已安装数据类型、已安装数据布局格式、最大维度索引大小和最大张量大小。信息被获得,并被放置在所选择的位置,诸如参数块(例如参数块330)中。当操作结束时,参数块的保留字段可被存储为零或可保持不变。

[0146] 在执行查询功能的一个实施例时,处理器(诸如通用处理器104)获得与特定处理器(诸如神经网络处理器(诸如神经网络处理器105)的特定模型)有关的信息。处理器或机器的特定模型具有某些能力。处理器或机器的另一种模型可以具有附加的、更少的和/或不同的能力和/或是具有附加的、更少的和/或不同的能力的不同代(例如,当前代或未来代)。所获得的信息被放置在参数块(例如,参数块330)或可由一个或多个应用访问和/或与一个或多个应用一起使用的其他结构中,该一个或多个应用可以在进一步的处理中使用该信息。在一个示例中,参数块和/或参数块的信息被保持在存储器中。在其它实施例中,参数块和/或信息可被保持在一个或多个硬件寄存器中。作为另一个示例,查询功能可以是由操作系统执行的特权操作,其使应用程序编程接口可用于使该信息可用于应用程序或非特权程序。在又一示例中,查询功能由专用处理器(诸如神经网络处理器105)执行。其它变型是可能的。

[0147] 信息例如通过执行查询功能的处理器的固件来获得。固件知道特定处理器(例如,神经网络处理器)的特定模型的属性。该信息可以被存储在例如控制块、寄存器和/或存储器中,和/或以其他方式可由执行查询功能的处理器访问。

[0148] 所获得的信息包括例如关于特定处理器的至少一个或多个数据属性的模型相关的详细信息,包括例如特定处理器的所选择的模型的一个或多个已安装或支持的数据类型、一个或多个已安装或支持的数据布局格式和/或一个或多个已安装或支持的数据大小。该信息是模型相关的,因为其它模型(例如,先前的模型和/或未来的模型)可能不支持相同的数据属性,诸如相同的数据类型、数据大小和/或数据布局格式。当查询功能(例如,NNPA-QAF功能)的执行完成时,作为示例,设置条件代码0。在一个示例中,条件代码1、2和3不适用于查询功能。下面描述与所获得的信息有关的进一步信息。

[0149] 如所指示的,在一个示例中,所获得的信息包括关于例如神经网络处理器的特定模型的一个或多个数据属性的模型相关的信息。数据属性的一个示例是神经网络处理器的已安装数据类型。例如,神经网络处理器(或其他处理器)的特定模型可以支持一个或多个数据类型,例如NNP数据类型1数据类型(也称为神经网络处理数据类型1数据类型)和/或其他数据类型。NNP数据类型1数据类型是16位浮点格式,其对于深度学习训练和推理计算提供了许多优点,包括例如:保持深度学习网络的准确性;消除简化舍入模式的非规格格式和角情况的处理;自动舍入到最近值以用于算术运算;以及无穷大和非数(NaN)的特殊实体被组合成一个值(NINF),该值被算术运算接受并处理。NINF提供用于指数溢出和无效操作(例如除以零)的更好的默认值。这允许许多程序继续运行,而不隐藏这种错误,并且不使用专门的异常处理程序。其它模型相关的数据类型也是可能的。

[0150] 在图4中描绘NNP数据类型1数据类型的格式的一个示例。如所描绘的,在一个示例中,NNP数据类型1数据可以以格式400表示,该格式包括例如符号402(例如,位0)、指数+31 404(例如,位1-6)和小数406(例如,位7-15)。

[0151] 以下描述NNP数据类型1格式的示例性特性:

[0152]

| 特性 | NNP 数据类型 1 |
|---|---|
| 格式长度（位） | 16 位 |
| 有偏指数长度（位） | 6 位 |
| 小数长度（位） | 9 位 |
| 精度（p） | 10 位 |
| 最大左单位视图指数（Emax） | 32 |
| 最小左单位视图指数（Emin） | -31 |
| 左单位视图（LUV）偏置 | 31 |
| Nmax | $(1-2^{-9}) \times 2^{33} \approx 8.6 \times 10^{9}$ |
| Nmin | $(1+2^{-9}) \times 2^{-31} \approx 4.6 \times 10^{-10}$ |
| Dmin | --- |

[0153]

[0154] 其中,≈表明值是近似的,Nmax是(在大小上)最大可表示有限数,Nmin是(在大小上)最小可表示数。

[0155] 以下描述关于NNP数据类型1数据类型的进一步细节:

[0156] 有偏指数:上面示出了用于允许指数被表示为无符号数的偏移。有偏指数类似于二进制浮点格式的特征,除了没有特殊的含义被附着到全零和全一的有偏指数之外,如下面参照NNP数据类型1数据类型的类所描述的。

[0157] 有效数:NNP数据类型1数字的二进制点被认为在最左侧的小数位的左边。在二进制点的左边有隐含的单位位,它对于规格数被认为是一,而对于零被认为是零。在左边附有隐含的单位位的小数是该数的有效数。

[0158] 规格NNP数据类型1的值是有效数乘以基数2的无偏指数次幂。

[0159] 非零数的值:非零数的值如下所示:

[0160]

| 数字类 | 值 |
|---|---|
| 规格数 | $\pm 2e^{-31} \times (1.f)$ |

[0161]

[0162] 其中,e是以十进制表示的有偏指数,f是以二进制表示的小数。

[0163] 在一个实施例中,存在三类NNP数据类型1数据,包括数字和相关的非数字实体。每个数据项包括符号、指数和有效数。指数被偏移,以使得所有有偏指数是非负的无符号数,并且最小有偏指数是零。有效数包括显式的小数和在二进制点的左边的隐含的单位位。对于正数,符号位是零,对于负数,符号位是一。

[0164] 所允许的所有非零有限数具有唯一的NNP数据类型1表示。不存在非规格数,该数可能允许相同值的多个表示,并且不存在非规格算术运算。这三类包括例如:

[0165]

| 数据类 | 符号 | 有偏指数 | 单位位* | 小数 |
|---|---|---|---|---|
| 零 | ± | 0 | 0 | 0 |

[0166]

| | | | | |
|---|---|---|---|---|
| 规格数 | ± | 0 | 1 | 非0 |
| 规格数 | ± | 非0，非全一 | 1 | 任意 |
| 规格数 | ± | 全一 | - | 非全一 |
| NINF | ± | 全一 | - | 全一 |

[0167] 其中:-表明不适用,*表明隐含了单位位,NINF是非数或无穷大。

[0168] 下面描述关于每一类的进一步细节:

[0169] 零:零具有有偏指数零和零小数。隐含的单位位是零。

[0170] 规格数:规格数可以具有任意值的有偏指数。当有偏指数是0时,小数将是非零。当有偏指数是全一时,小数不是全一。其它有偏指数值可具有任意小数值。隐含的单位位对于所有规格数是一。

[0171] NINF:NINF由全一的有偏指数和全一的小数表示。NINF表示不在NNP数据类型1(即,具有6个指数位和9个小数位的被设计用于深度学习(DL)的16位浮点)的可表示值的范围内的值。通常,NINF仅在计算期间被传播,以使得它们在结束时将保持可见。

[0172] 尽管在一个示例中支持NNP数据类型1数据类型,但也可支持其他专用或非标准数据类型以及一个或多个标准数据类型,包括但不限于:IEEE 754短精度、二进制浮点16位、IEEE半精度浮点、8位浮点、4位整数格式和/或8位整数格式,仅举几例。这些数据格式对于神经网络处理具有不同的质量。作为示例,较小的数据类型(例如,较少的位)可以被更快地处理并使用较少的高速缓存/存储器,而较大的数据类型在神经网络中提供较高的结果准确性。要支持的数据类型可在查询参数块中(例如,在参数块330的已安装数据类型字段336中)具有一个或多个分配的位。例如,在已安装数据类型字段中指示特定处理器所支持的专用或非标准数据类型,但没有指示标准数据类型。在其他实施例中,也指示一个或多个标准数据类型。其它变型是可能的。

[0173] 在一个特定示例中,已安装数据类型字段336的位0被保留用于NNP数据类型1数据类型,并且当它被设置为例如1时,它指示处理器支持NNP数据类型1。作为示例,已安装数据类型的位向量被配置为表示多达16种数据类型,其中,位被分配给每个数据类型。然而,在其他实施例中的位向量可以支持更多或更少的数据类型。进一步地,可以配置其中一个或多个比特被分配给数据类型的向量。许多示例是可能的和/或附加的,更少和/或其他数据类型可以在向量中被支持和/或指示。

[0174] 在一个示例中,查询功能获得在模型相关的处理器上安装的数据类型的指示,并通过例如在参数块330的已安装数据类型字段336中设置一个或多个位来将该指示放置在参数块中。进一步地,在一个示例中,查询功能获得已安装数据布局格式的指示(另一个数据属性),并通过例如在已安装数据布局格式字段338中设置一个或多个位来将该信息放置在参数块中。示例性数据布局格式包括例如4D特征张量布局和4D核张量布局。在一个示例中,4D特征张量布局由在此所指示的功能使用,并在一个示例中,卷积函数使用4D核张量布局。这些数据布局格式以增加神经网络处理辅助指令的功能的执行的处理效率的方式在用于张量的存储装置中布置数据。例如,为了有效地操作,神经网络处理辅助指令使用以特定数据布局格式提供的输入张量。尽管提供了示例性布局,但是可以针对本文所描述的功能

和/或其他功能提供附加的、更少的和/或其他布局。

[0175] 用于特定处理器模型的布局的使用或可用性由已安装数据布局格式(例如,参数块330的字段338)的向量提供。向量例如是已安装数据布局格式的位向量,其允许CPU向应用程序传达支持哪些布局。例如,位0被保留用于4D特征张量布局,并且当它被设置为例如1时,它指示处理器支持4D特征张量布局;并且位1被保留用于4D核张量布局,并且当它被设置为例如1时,它指示处理器支持4D核张量布局。在一个示例中,已安装数据布局格式的位向量被配置成表示多达16个数据布局,其中位被分配给每个数据布局。然而,在其他实施例中的位向量可以支持更多或更少的数据布局。进一步地,可以配置其中一个或多个位被分配给数据布局的向量。许多示例是可能的。下面描述关于4D特征张量布局和4D内核张量布局的进一步细节。再次,现在或将来可以使用其他布局来优化性能。

[0176] 在一个示例中,神经网络处理辅助指令用4D张量(即具有4维的张量)操作。这些4D张量是从在此描述的以例如行为主的通用输入张量获得的,即,当按增加存储器地址的顺序枚举张量元素时,被称为E1的内部维度将首先步进通过从0开始到E1索引大小1的E1索引大小值,之后,E2维度的索引将被增加,并且重复步进通过E1维度。被称为E4维度的外部维度的索引最后被增加。

[0177] 具有较低维度数量的张量(例如3D或1D张量)将被表示为4D张量,其中该4D张量的一个或多个维度超过被设置为1的原始张量维度。

[0178] 在此描述将具有维度E4、E3、E2、E1的以行为主的通用4D张量转换为4D特征张量布局(这里也称为NNPA数据布局格式0 4D特征张量):

[0179] 所得到的张量例如可以被表示为例如64元素向量的4D张量或者具有以下维度的5D张量:

[0180] E4,$\lceil E1/64 \rceil$,E3,$\lceil E2/32 \rceil *32$,64,其中,⌈ ⌉表示向上取整(ceil)函数。(另一种方式是:E4*E3*ceil(E2/32)*32*ceil(E1/64)*64个元素)。

[0181] 通用张量的元素[e4][e3][e2][e1]可被映射到所得到的5D张量的以下元素:

[0182] [e4]$[\lfloor e1/64 \rfloor]$[e3][e2][e1MOD 64],其中,⌊ ⌋是向下取整(floor)函数,并且mod是模。(另一种方式是:元素$(E3*e2_limit*e1_limit*e4x) + (e2_limit*e3x*64)+(e2x*64)+(\lfloor e1x/64 \rfloor *e2_limit*E3*64) + (e1x \bmod 64)$,其中$e2_limit = \lceil E2/32 \rceil *32$,并且$e1_limit = \lceil E1/64 \rceil *64$。)

[0183] 所得到的张量可大于通用张量。在通用张量中没有对应元素的所得到的张量的元素被称为填充(pad)元素。

[0184] 考虑64元素向量的NNPA数据布局格式0 4D特征张量的元素[fe4][fe1][fe3][fe2][fe0]或其元素的5D张量的等效表示。该元素是填充元素或者是它在具有维度E4、E3、E2、E1的通用4D张量中的对应元素,可以用下式确定:

[0185] • 如果fe2≥E2,则这是E2(或页)填充元素

[0186] • 否则,如果fe1*64+fe0≥E1,则这是E1(或行)填充元素

[0187] • 否则,通用4D张量中的对应元素是:

[0188] [fe4][fe3][fe2][fe1*64+fe0]。

[0189] 对于基于卷积神经网络的人工智能模型,特征张量的4维的含义通常可以被映射为:

[0190] • E4:N-小批量的大小

[0191] • E3:H-3D张量/图像的高度

[0192] • E2:W-3D张量/图像的宽度

[0193] • E1:C-3D张量的通道或类别

[0194] 对于基于机器学习或循环神经网络的人工智能模型,4D特征张量的4维的含义通常可以被映射到:

[0195] • E4:T-时间步长或模型的数量

[0196] • E3:保留,一般被设置为1

[0197] • E2:Nmb-小批量大小

[0198] • E1:L-特征NNPA数据布局格式0提供例如具有4k字节数据块(页)的二维数据局部性以及用于所产生的张量的外部维度的4k字节块数据对齐。

[0199] 填充元素字节对于输入张量被忽略,而对于输出张量是不可预测的。填充字节上的PER存储更改是不可预测的。

[0200] 图5A-图5C示出了用于4D特征张量布局的输入数据布局的一个示例,其具有维度E1、E2、E3和E4,图6A-图6C描绘了用于4D特征张量布局的示例输出。参考图5A,示出了3D张量500,其具有维度E1、E2和E3。在一个示例中,每个3D张量包括多个2D张量502。每个2D张量502中的数字描述其每个元素在存储器中的位置的存储器偏移。输入被用于在存储器中布置原始张量(例如,图5A-图5C的原始4D张量)的数据,如图6A-图6C所示,其对应于图5A-图5C。

[0201] 在图6A中,作为示例,存储器单元600(例如,存储器页)包括预选数量(例如,32)的行602,其中每一行由例如e2_page_idx标识;并且每一行具有预选数量(例如64)的元素604,每个元素由例如e1_page_idx标识。如果行不包括预选数量的元素,则它被填充606,称为行或E1填充;且如果存储器单元不具有预先选定数量的行,则它被填充608,称为页或E2填充。作为示例,行填充是例如零或其他值,并且页填充是例如现有值、零或其他值。

[0202] 在一个示例中,行的输出元素基于它的对应输入在E1方向上的元素位置而在存储器中(例如,在页中)提供。例如,参考图5A,在图6A的页0的行0中示出了所示的三个矩阵的元素位置0、1和2(例如,在每个矩阵中的相同位置处的元素位置),等等。在该示例中,4D张量很小,并且表示4D张量的每个2D张量的所有元素都适合一页。然而,这仅仅是一个示例。2D张量可以包括一个或多个页。如果基于4D张量的重新格式化而创建了2D张量,则2D张量的页数是基于4D张量的大小。在一个示例中,一个或多个向上取整函数被用于确定2D张量中的行数以及每行中的元素数目,这将指示要使用多少页。其它变型是可能的。

[0203] 除了4D特征张量布局之外,在一个示例中,神经网络处理器可以支持4D核张量布局,其重新排列4D张量的元素以在执行某些人工智能(例如,神经网络处理辅助)操作(例如卷积)时减少存储器访问和数据收集步骤的数量。作为示例,具有维度E4、E3、E2、E1的以行为主通用4D张量被转换成NNPA数据布局格式1 4D核张量(4D核张量),如在此所描述的:

[0204] 所得到的张量可以被表示为例如64元素向量的4D张量或者具有以下维度的5D张量:

[0205] $\lceil E1/64 \rceil$,E4,E3,$\lceil E2/32 \rceil *32$,64,其中,⌈ ⌉表示向上取整函数。(另一种方式是:E4*E3*ceil(E2/32)*32*ceil(E1/64)*64个元素)

[0206] 通用张量的元素[e4][e3][e2[e1]可被映射到所得到的5D张量的以下元素:

[0207] $\lfloor e1/64 \rfloor$ [e4] [e3] [e2] [e1 MOD 64],其中,⌊ ⌋表示向下取整函数,而mod是模。另一种方式是:元素$(\lfloor e1x/64 \rfloor *E4*E3*E2*e2_limit*64)+(e4x*E3*e2_limit*64)+(e3x*e2_limit*64)+(e2x*64)+(e1x \bmod 64)$,其中,$e2\_limit = \lceil E2/32 \rceil *32$, $e1\_limit = \lceil E1/64 \rceil *64$。

[0208] 所得到的张量可大于通用张量。在通用张量中没有对应元素的所得到的张量的元素被称为填充元素。

[0209] 考虑64元素向量的NNPA数据布局格式1 4D特征张量的元素[fe1][fe4][fe3][fe2][fe0]或其元素的5D张量的等效表示。该元素是填充元素或者是它在具有维度E4、E3、E2、E1的通用4D张量中的对应元素,可以用下式确定:

[0210] • 如果fe2≥E2,则这是E2(或页)填充元素

[0211] • 否则,如果fe1*64+fe0≥E1,则这是E1(或行)填充元素

[0212] • 否则,通用4D张量中的对应元素是:

[0213] [fe4][fe3][fe2][fe1*64+fe0]

[0214] 对于基于卷积神经网络的人工智能模型,核张量的4维的含义通常可以被映射为:

[0215] • E4:H-3D张量/图像的高度

[0216] • E3:W-3D张量/图像的宽度

[0217] • E2:C-3D张量的通道数量

[0218] • E1:核的数量

[0219] NNPA数据布局格式1提供例如在4k字节的数据块(页)内的二维内核并行性以及用于生成张量的外部维度的4k字节块数据对齐,以用于有效处理。

[0220] 对于输入张量,忽略填充字节。填充字节上的PER存储更改是不可预测的。

[0221] 同样,尽管示例性的数据布局格式包括4D特征张量布局和4D核张量布局,但是处理器(例如,神经网络处理器105)可以支持其他数据布局格式。获得所支持的数据布局的指示,并通过在例如字段338中设置一个或多个位来将该指示放置在查询参数块中。

[0222] 根据本发明的一个或多个方面,查询参数块还包括其它数据属性信息,包括例如用于数据的支持大小信息。处理器(诸如神经网络处理器)通常具有基于内部缓冲器大小、处理单元、数据总线结构、固件限制等的限制,这些限制可限制张量维度的最大大小和/或张量的总大小。因此,查询功能提供将这些限制传达给应用程序的字段。例如,处理器基于执行查询功能而获得各种数据大小,诸如最大维度索引大小(例如,65,536个元素)和最大张量大小(例如,8GB),并将该信息分别包括在参数块(例如,参数块330)的字段340和342中。另外,更少的和/或其他大小信息也可以由处理器(例如,神经网络处理器105)支持,并因此被获得并被放置在参数块中,例如,字段340、342和/或其他字段。在其它实施例中,限制可以更小或更大,和/或大小可以采用其它单位,诸如字节而不是元素、元素而不是字节等。进一步地,其它实施例允许每个维度的不同最大大小,而不是对于所有维度是相同的最

大值。许多变型是可能的。

[0223] 根据本发明的一个或多个方面,提供了查询功能,其传达与所选择的处理器(例如,神经网络处理器105)的特定模型有关的详细信息。详细信息包括例如与特定处理器有关的模型相关信息。(处理器也可支持标准数据属性,例如标准数据类型、标准数据布局等,它们是由查询功能所隐含的并且不一定呈现;尽管在其它实施例中,查询功能可指示数据属性的所有或各种所选择的子集等),尽管提供了示例信息,但在其它实施例中可提供其它信息。所获得的信息(可以对于处理器的不同模型和/或不同的处理器而不同)用于执行人工智能和/或其他处理。人工智能和/或其他处理可以采用例如神经网络处理辅助指令的一个或多个非查询功能。在处理中采用的特定非查询功能通过执行神经网络处理辅助指令一次或多次并且指定非查询特定功能来执行。

[0224] 神经网络处理辅助指令所支持的非查询功能的示例包括AVGPOOL2D功能和MAXPOL2D功能,其中的每一个在下面描述(在一个或多个实施例中支持附加的、更少的和/或其他功能)。

[0225] 功能代码80:NNPA-MAXPOL2D

[0226] 功能代码81:NNPA-AVGPOOL2D

[0227] 当NNPA-MAXPOOL2D或NNPA-AVGPOOL2D功能被指定时,由输入张量1描述符(例如,参见图3G)描述的输入张量1通过所指定的操作被减少以概括输入的窗口。输入的窗口通过在维度2和3上移动2D滑动窗口来选择。窗口的概要是输出张量的元素。滑动窗口维度由例如功能特定参数4和功能特定参数5来描述,当计算相邻输出张量元素时,滑动窗口在输入张量1上移动的量被称为步长。滑动窗口步长由例如功能特定参数2和功能特定参数3指定。当NNPA-MAXPOOL2D操作被指定时,对窗口执行下面定义的Max操作。当NNPA-AVGPOOL2D操作被指定时,对窗口执行下面定义的AVG操作。如果所指定的填充类型是有效,则窗口中的所有元素被添加到用于计算结果输出元素的集合。如果所指定的填充类型是相同,则取决于窗口的位置,仅仅来自窗口的元素子集可被添加到用于计算结果输出元素的集合(例如,可以忽略在张量的边界之外的那些元素)。

[0228] 在一个示例中,CollectElements操作将元素添加到元素集合中,并递增集合中的元素的数量。每当窗口起始位置移动时,集合被清空。执行操作不需要的元素是否被访问是不可预测的。

[0229] Max操作:在一个示例中,通过将窗口中的元素集合的所有元素彼此进行比较并返回最大值来计算该集合的最大值。

[0230] AVG(均值)操作:在一个示例中,窗口中的元素集合的平均值被计算为例如集合中的所有元素的总和除以集合中的元素的数量。

[0231] 在一个示例中,如下分配字段:

[0232] *池化功能特定参数1控制填充类型。例如,功能特定参数1的位29-31包括指定填充类型的填充(PAD)字段。示例类型包括例如:

[0233]

| PAD | 填充类型 |
| --- | --- |
| 0 | 有效 |
| 1 | 相同 |
| 2-7 | 保留 |

[0234] 如果针对PAD字段指定保留值,则报告例如F000(十六进制)的响应代码,并且操作以的条件代码(例如1)完成。

[0235] 在一个示例中,功能特定参数1的位位置0-28被保留并将包含零。

[0236] *功能特定参数2包含例如指定维度2步长(D2S)的32位无符号二进制整数,该D2S指定滑动窗口在维度2(也称为e2)上移动的元素数量。

[0237] *功能特定参数3包含例如指定维度3步长(D3S)的32位无符号二进制整数,该D3S指定滑动窗口在维度3(也称为e3)上移动的元素数量。

[0238] *功能特定参数4包含例如指定维度2窗口大小(D2WS)的32位无符号二进制整数,该D2WS指定滑动窗口包含的在维度2上的元素数量。

[0239] *功能特定参数5包含例如指定维度3窗口大小(D3WS)的32位无符号二进制整数,该D3WS指定滑动窗口包含的在维度3上的元素数量。

[0240] 在一个示例中,在功能特定参数2-5中指定的值要小于或等于最大维度索引大小,并且在功能特定参数4-5中指定的值要大于例如零;否则,报告响应代码,例如0012(十六进制),并且操作以条件代码(例如1)完成。

[0241] 如果维度2步长和维度3步长都是零,并且维度2窗口大小或维度3窗口大小大于例如1024,则存储响应代码,例如F001(十六进制)。如果维度2步长和维度3步长都大于例如零,并且维度2窗口大小或维度3窗口大小大于例如64,则存储响应代码,例如F002(十六进制)。如果维度2步长和维度3步长都大于例如零,并且维度2步长或维度3步长大于例如30,则存储响应代码,例如F003(十六进制)。如果维度2步长和维度3步长都大于例如零,并且输入张量维度2索引大小或输入张量维度3索引大小大于例如1024,则存储响应码,例如F004(十六进制)。对于所有上述条件,指令以条件代码(例如1)完成。

[0242] 在一个示例中,如果在任何所指定的张量描述符中所指定的数据布局没有指定4D特征张量(例如,数据布局=0),或者如果在任何所指定的张量描述符中的数据类型没有指定NNP数据类型1(例如,数据类型=0),则在通用寄存器0中分别设置响应代码,例如0010(十六进制)或0011(十六进制),并且指令以条件代码(例如1)完成。

[0243] 在一个示例中,以下条件将为真,否则,识别一般操作数数据异常:

[0244] *输入张量和输出张量的维度4索引大小和维度1索引大小将是相同的。

[0245] *输入张量和输出张量的数据布局和数据类型将是相同的。

[0246] *如果维度2步长和维度3步长都是零(指定例如AVGPOOL2D功能的平均归约操作),则在一个示例中,以下附加条件将为真:

[0247] *输入张量维度2索引大小将等于维度2窗口大小。

[0248] *输入张量的维度3索引大小将等于维度3窗口大小。

[0249] *输出张量的维度2索引大小和维度3索引大小将是一。

[0250] *所指定的填充将是有效

[0251] *在一个示例中,如果维度2步长或维度3步长是非零,则两个步长都将是非零。

[0252] *如果维度2步长和维度3步长都大于零(指定例如AVGPOOL2D功能的池化操作),则在一个示例中,以下附加条件将为真:

[0253] *当所指定的填充是有效时,维度2窗口大小将小于或等于输入张量的维度2索引大小。

[0254] *当所指定的填充是有效时,维度3窗口大小将小于或等于输入张量的维度3索引大小。

[0255] *当所指定的填充是相同时,要满足输入张量和输出张量的维度2索引大小与维度3索引大小之间的以下关系(池化相同填充):

[0256] $$\mathbf{O1D2IS} = \left\lceil \frac{\mathbf{I1D2IS}}{\mathbf{D2S}} \right\rceil$$

[0257] $$\mathbf{O1D3IS} = \left\lceil \frac{\mathbf{I1D3IS}}{\mathbf{D3S}} \right\rceil$$

[0258] 其中:

[0259] IxDyIS在张量描述符x中定义的输入张量x的维度y索引大小。

[0260] OxDyIS在张量描述符x中定义的输出张量x的维度y索引大小。

[0261] D2S维度2步长。

[0262] D3S维度3步长。

[0263] *当所指定的填充是有效时,要满足输入张量和输出张量的维度2索引大小与维度3索引大小之间的以下关系(池化有效填充):

[0264] $$\mathbf{O1D2IS} = \left\lceil \frac{(\mathbf{I1D2IS} - \mathbf{D2WS} + \mathbf{1})}{\mathbf{D2S}} \right\rceil$$

[0265] $$\mathbf{O1D3IS} = \left\lceil \frac{(\mathbf{I1D3IS} - \mathbf{D3WS} + \mathbf{1})}{\mathbf{D3S}} \right\rceil$$

[0266] 其中,D2WS是维度2窗口大小,D3WS是维度3窗口大小。

[0267] 输出张量描述符2、输入张量描述符2和3、以及功能特定保存区域地址字段被忽略。

[0268] 对于神经网络处理辅助指令,在一个实施例中,如果输出张量与任何输入张量或参数块重叠,则结果是不可预测的。

[0269] 作为示例,当尝试执行神经网络处理辅助指令并且参数块未在例如双字边界上被指定时,识别规范异常。

[0270] 当尝试执行神经网络处理辅助指令且存在(例如)张量描述符不一致时,识别一般操作数数据异常。

[0271] 用于神经网络处理辅助指令的所得条件代码包括例如:0-正常完成;1-设置响应代码;2--;3-CPU确定的处理数据量。

[0272] 在一个实施例中,神经网络处理辅助指令的执行优先级包括例如:

[0273] 1.-7.异常,其中对于一般情况具有与程序中断条件的优先级相同的优先级。

[0274] 8.A由于未分配或未安装的功能代码被指定的条件代码1。

[0275] 8.B由于参数块未在双字边界上被指定的规范异常。

[0276] 9.针对访问参数块的的访问异常。

[0277] 10.由于模型不支持参数块的指定格式的条件代码1。

[0278] 11.A由于不支持所指定的张量数据布局的条件代码1。

[0279] 11.B由于张量描述符之间的数据布局不同的一般操作数数据异常。

[0280] 12.A由于除了被包括在上面的项8.A、10和11.A和下面的12.B.1中的条件之外的条件的条件代码1。

[0281] 12.B.1由于用于NNPA-RELU和NNPA-CONVOLUTION的无效输出张量数据类型的条件代码1。

[0282] 12.B.2针对用于NNPA-RELU功能特定参数1和NNPA-CONVOLUTION功能特定参数4的无效值的一般操作数数据异常。

[0283] 13.A针对访问输出张量的访问异常。

[0284] 13.B针对访问输入张量的访问异常。

[0285] 13.C针对访问功能特定保存区域的访问异常。

[0286] 14.条件代码0。

[0287] 如本文所描述的,单个指令(例如,神经网络处理辅助指令)被配置为执行多个功能,包括查询功能和多个非查询功能。至少一个非查询功能(AVGPOOL2D功能)被配置为实现多个操作(例如,均值归约和池化)。通过使用一个功能来执行多个操作,消除例如重复编码和验证。要执行的特定操作取决于功能的选择输入参数的值。相同的输入参数被用于功能的两个操作,但是,选择参数的不同值指示要执行的操作导致其它输入参数的不同边界检查。尽管均值归约和池化操作不同地执行,但是它们可以在算法上被归约为具有相同输入张量和功能特定参数但具有不同的相对约束的公共算法操作。一个不同是检查在两个操作之间不同的条件(例如,步长、窗口大小)。如本文所描述的,提供了指令的单个功能,其执行多个操作并在一些参数实现对两个操作的不同边界检查。这至少减少了代码复杂性、代码复制和验证工作。

[0288] 本发明的一个或多个方面紧密地依赖于计算机技术,并促进计算机内的处理,从而提高其性能。使用被配置为执行各种功能的单个架构机器指令通过降低复杂度、减少资源的使用并提高处理速度来改善计算环境内的性能。使用单个功能来实现多个操作降低了复杂性、资源的使用、编码和/或验证工作,并提高了系统性能。指令、功能和/或操作可以用于许多技术领域,例如计算机处理、医疗处理、工程、汽车技术、制造等。通过提供优化,通过例如减少错误和/或执行时间来改进这些技术领域。

[0289] 参考图7A-图7C描述与本发明的一个或多个方面相关的促进计算环境内的处理的一个实施例的进一步细节。

[0290] 参考图7A,获得要执行的功能的指示,其中该功能是指令的一个功能并且被配置为执行多个操作700。确定多个操作中的要执行的操作702,并使用一组值和对应的一组关系来验证一组功能特定参数704。该组值和对应的一组关系基于要执行的操作706。作为示例,一组值和对应的一组关系将被用于要执行的操作708,而另一组值和对应的一组关系将被用于多个操作中的另一个操作710。

[0291] 使用单个功能(例如,架构指令的单个功能)来执行多个操作,但减少了每一操作的参数验证、代码复杂度、代码重复和/或验证工作,从而改善系统性能。

[0292] 在一个示例中,确定要执行的操作包括对照至少一个特定值检查一个或多个功能特定参数720。基于一个或多个功能特定参数相对于至少一个特定值具有第一选择关系,操作是一个操作722,并且基于一个或多个功能特定参数相对于至少一个特定值具有第二选择关系,操作是另一个操作724。

[0293] 通过使用相同的功能特定参数但不同的关系来确定要执行的操作,降低了代码复杂度和验证工作量。

[0294] 作为示例,功能包括平均池化功能,一个或多个功能特定参数包括一个或多个步长值,其中,步长值是当计算一个或多个相邻输出张量元素时滑动窗口在输入张量上移动的量,至少一个特定值包括零,第一选择关系包括等于,并且基于一个或多个步长值等于零,操作是均值归约操作726。

[0295] 进一步地,在一个示例中,参考图7B,第二选择关系包括大于,并且基于一个或多个步长值大于零,操作是池化操作728。

[0296] 作为示例,该组功能特定参数包括一个或多个选择维度窗口大小值730。选择维度窗口大小值指定滑动窗口包含的在选择维度上的元素数量732,并且滑动窗口被配置为在功能的输入张量上移动以产生输出张量734。

[0297] 在一个示例中,功能包括平均池化功能,操作包括均值归约操作,并且要用于验证一组功能特定参数的一组值和对应的一组关系包括:选择输入张量的一个维度的一个值,对应的关系是等于;选择输入张量的另一个维度的另一个值,对应的关系是等于;以及选择值,对应的关系是小于或等于740。

[0298] 验证包括例如检查维度2窗口大小的值等于第一输入张量的维度2的值,维度3窗口大小的值等于第一输入张量的维度3的值,维度2窗口大小的值小于或等于选择值,以及维度3窗口大小的值小于或等于选择值746。

[0299] 在一个示例中,参考图7C,功能包括平均池化功能,操作包括池化操作,并且要用于验证一组功能专用参数的一组值和对应的一组关系包括:选择输入张量的一个维度的一个值,对应的关系是小于或等于;以及选择输入张量的另一个维度的另一个值,对应的关系是小于或等于750。

[0300] 验证包括例如检查维度2窗口大小的值小于或等于第一输入张量的维度2的值,以及维度3窗口大小的值小于或等于第一输入张量的维度3的值756。

[0301] 在一个示例中,确定填充类型是否被设置为特定类型,其中填充类型指示窗口的哪些元素将被用于计算输出,并且对于一个或多个实施例,例如池化操作,基于填充类型被设置为特定类型,执行验证760。进一步地,在一个示例中,基于填充类型未被设置为特定类型,执行与输出张量的一个或多个维度相关的一个或多个检查762。

[0302] 在一个示例中,确定操作是基于输入张量的至少一个滑动窗口步长值770,并且该组功能特定参数包括输入张量的至少一个滑动窗口维度772。

[0303] 其它变型和实施例是可能的。

[0304] 本发明的各方面可由许多类型的计算环境使用。参考图8A描述用于结合和使用本发明的一个或多个方面的计算环境的另一个示例。作为示例,图8A的计算环境基于由纽约

阿蒙克的国际商业机器公司提供的z/Architecture®指令集架构。然而,z/Architecture指令集架构仅是一个示例架构。同样,计算环境可以基于其他架构,包括但不限于Intel® x86架构、国际商业机器公司的其他架构、和/或其他公司的其他架构。Intel是Intel公司或其子公司在美国和其他国家的商标或注册商标。

[0305] 在一个示例中,计算环境10包括中央电子复合体(CEC)11。中央电子复合体11包括多个组件,例如存储器12(也称为系统存储器、主存储器、主存储、中央存储、存储),其耦合到一个或多个处理器,诸如一个或多个通用处理器(也称为中央处理单元(CPU)13)和一个或多个专用处理器(例如神经网络处理器31),并耦合到输入/输出(I/O)子系统14。

[0306] 作为示例,一个或多个专用处理器可以与一个或多个通用处理器分离,和/或至少一个专用处理器可以被嵌入在至少一个通用处理器内。其它变型也是可能的。

[0307] I/O子系统14可以是中央电子复合体的一部分或者与其分离。它引导主存储器12与耦合到中央电子复合体的输入/输出控制单元15和输入/输出(I/O)设备16之间的信息流。

[0308] 可以使用许多类型的I/O设备。一种特定类型是数据存储设备17,数据存储设备17可以存储一个或多个程序18、一个或多个计算机可读程序指令19、和/或数据等。计算机可读程序指令可以被配置为执行本发明的各方面的实施例的功能。

[0309] 中央电子复合体11可以包括和/或耦合到可移除/不可移除的易失性/非易失性计算机系统存储介质。例如,它可以包括和/或耦合到不可移除的非易失性磁介质(通常称为“硬盘驱动器”)、用于从可移除的非易失性磁盘(例如,“软盘”)读取和向其写入的磁盘驱动器、和/或用于从可移除的非易失性光盘(诸如CD-ROM、DVD-ROM或其它光学介质)读取或向其写入的光盘驱动器。应当理解,其它硬件和/或软件组件可以与中央电子复合体11结合使用。示例包括但不限于:微码或毫码、设备驱动器、冗余处理单元、外部磁盘驱动器阵列、RAID系统、磁带驱动器和数据档案存储系统等。

[0310] 进一步地,中央电子复合体11可以与许多其它通用或专用计算系统环境或配置一起操作。可适合与中央电子复合体11一起使用的公知的计算系统、环境和/或配置的示例包括但不限于个人计算机(PC)系统、服务器计算机系统、瘦客户端、胖客户端、手持或膝上型设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品、网络PC、小型计算机系统、大型计算机系统、以及包括任何上述系统或设备的分布式云计算环境等。

[0311] 中央电子复合体11在一个或多个实施例中提供逻辑分区和/或虚拟化支持。在一个实施例中,如图8B所示,存储器12包括例如一个或多个逻辑分区20、管理逻辑分区的管理程序21、以及处理器固件22。管理程序21的一个示例是由纽约阿蒙克的国际商业机器公司提供的处理器资源/系统管理器(PR/SM™)。PR/SM是国际商业机器公司在至少一个管辖权内的商标或注册商标。

[0312] 每个逻辑分区20能够用作单独的系统。即,每个逻辑分区可以被独立地重置,运行客机操作系统23,诸如由纽约阿蒙克的国际商业机器公司提供的z/OS®操作系统或其他控制代码24(诸如耦合设施控制代码(CFCC)),并且与不同的程序25一起操作。尽管z/OS操作系统是作为示例提供的,但是根据本发明的一个或多个方面,也可以使用由国际商业机器公司和/或其它公司提供的其它操作系统。

[0313] 存储器12耦合到例如CPU13(图8A),它是可被分配给逻辑分区的物理处理器资源。例如,逻辑分区20可以包括一个或多个逻辑处理器,每个逻辑处理器代表可被动态分配给逻辑分区的物理处理器资源13的全部或一部分。

[0314] 在又一实施例中,中央电子复合体提供虚拟机支持(支持或不支持逻辑分区)。如图8C所示,中央电子复合体11的存储器12包括例如一个或多个虚拟机26、管理虚拟机的虚拟机管理器(例如管理程序27)以及处理器固件28,管理程序27的一个示例是由纽约阿蒙克的国际商业机器公司提供的z/VM®管理程序。。管理程序有时被称为主机。z/VM是国际商业机器公司在至少一个管辖权内的商标或注册商标。

[0315] 中央电子复合体的虚拟机支持提供了操作大量虚拟机26的能力,每个虚拟机能够与不同的程序29一起操作并且运行客机操作系统30,诸如Linux®操作系统。每个虚拟机26能够用作单独的系统。也就是说,每个虚拟机可以被独立地重置,运行客机操作系统,并且与不同的程序一起操作。在虚拟机中运行的操作系统或应用程序看起来可以访问整个系统,但是实际上,它仅有一部分是可用的。尽管z/VM和Linux作为示例被提供,但是根据本发明的一个或多个方面,可以使用其他虚拟机管理器和/或操作系统。注册商标Linux®是根据来自Linux基金会(其是Linus Torvalds的独家被许可人)的分许可在全球基础上使用的。

[0316] 参考图9A描述用于结合和使用本发明的一个或多个方面的计算环境的另一个实施例。在该示例中,计算环境36包括例如经由例如一个或多个总线40和/或其它连接彼此耦合的本机中央处理单元(CPU)37、存储器38、以及一个或多个输入/输出设备和/或接口39。作为示例,计算环境36可包括由纽约阿蒙克的国际商业机器公司提供的PowerPC®处理器;由加利福尼亚帕洛阿托的惠普公司提供的具有Intel®Itanium®II处理器的HP Superdome;和/或基于由国际商业机器公司、惠普公司、英特尔公司、Oracle和/或其它公司提供的架构的其它机器。PowerPC是国际商业机器公司在至少一个管辖权内的商标或注册商标。Itanium是英特尔公司或其子公司在美国和其他国家的商标或注册商标。

[0317] 本机中央处理单元37包括一个或多个本机寄存器41,诸如在环境内的处理期间使用的一个或多个通用寄存器和/或一个或多个专用寄存器。这些寄存器包括表示在任何特定时间点的环境的状态的信息。

[0318] 此外,本机中央处理单元37执行被存储在存储器38中的指令和代码。在一个特定示例中,中央处理单元执行被存储在存储器38中的仿真器代码42。该代码使得在一个架构中被配置的计算环境能够仿真另一个架构。例如,仿真器代码42允许基于除了z/Architecture指令集架构之外的架构的机器(诸如PowerPC处理器、HP Superdome服务器或其他)仿真z/Architecture指令集架构,并执行基于z/Architecture指令集架构所开发的软件和指令。

[0319] 参考图9B描述与仿真器代码42有关的进一步细节。被存储在存储器38中的客机指令43包括(例如,与机器指令相关的)软件指令,其被开发为在不同于本机CPU 37的架构中执行。例如,客机指令43可能已被设计为在基于z/Architecture指令集架构的处理器上执行,但是替代地在本机CPU 37(其可以是例如Intel Itanium II处理器)上被仿真。在一个示例中,仿真器代码42包括指令取例程44,以从存储器38获得一个或多个客机指令43,并可

选地为所获得的指令提供本地缓冲。它还包括指令转换例程45,以确定已获得的客机指令的类型,并将客机指令转换成一个或多个对应的本机指令46。该转换包括例如识别要由客机指令执行的功能以及选择(一个或多个)本机指令来执行该功能。

[0320] 进一步地,仿真器代码42包括仿真控制例程47以使得本机指令被执行。仿真控制例程47可以使本机CPU 37执行仿真一个或多个先前获得的客机指令的本机指令的例程,并在这种执行结束时将控制返回到指令取例程以仿真下一个客机指令或一组客机指令的获得。本机指令46的执行可以包括将数据从存储器38加载到寄存器中;将数据从寄存器存储回存储器;或者执行由转换例程确定的某种类型的算术或逻辑运算。

[0321] 每个例程例如以软件实现,该软件被存储在存储器中并由本机中央处理单元37执行。在其他示例中,一个或多个例程或操作以固件、硬件、软件或其某种组合实现。被仿真的处理器的寄存器可以使用本机CPU的寄存器41或者通过使用存储器38中的位置来仿真。在实施例中,客机指令43、本机指令46和仿真器代码42可以驻留在相同的存储器中,或者可以被分布在不同的存储器设备之间。

[0322] 根据本发明的一个或多个方面,可以被仿真的指令包括本文所描述的神经网络辅助处理指令。进一步地,根据本发明的一个或多个方面,可以仿真神经网络处理的其它指令、功能、操作和/或一个或多个方面。

[0323] 上述的计算环境仅仅是可以使用的计算环境的示例。可以使用其他环境,包括但不限于非分区环境、分区环境、云环境和/或仿真环境;实施例不限于任何一种环境。尽管在此描述了计算环境的各种示例,但是本发明的一个或多个方面可以与许多类型的环境一起使用。这里提供的计算环境仅仅是示例。

[0324] 每个计算环境能够被配置成包括本发明的一个或多个方面。

[0325] 一个或多个方面可以涉及云计算。

[0326] 应当理解,尽管本公开包括关于云计算的详细描述,但是本文所陈述的教导的实现不限于云计算环境。相反,本发明的实施例能够结合现在已知的或以后开发的任何其它类型的计算环境来实现。

[0327] 云计算是一种服务交付模型,用于实现对共享的可配置计算资源(例如,网络、网络带宽、服务器、处理、存储器、存储、应用、虚拟机和服务)池的方便、按需的网络访问,可配置计算资源可以以最小的管理成本或与服务提供商进行最少的交互来快速供应和释放。该云模型可以包括至少五个特性、至少三个服务模型和至少四个部署模型。

[0328] 特征如下:

[0329] 按需自助式服务:云的消费者可以单方面自动地按需提供计算能力(诸如服务器时间和网络存储),而无需与服务提供者进行人工交互。

[0330] 广泛的网络接入:能力在网络上可用并通过促进异构的瘦或厚客户端平台(例如,移动电话、膝上型计算机和PDA)的使用的标准机制来接入。

[0331] 资源池化:提供商的计算资源被归入资源池以使用多租户模型来服务多个消费者,其中不同的物理和虚拟资源根据需求被动态地分配和再分配。一般情况下,消费者不能控制或不知道所提供的资源的确切位置,但是可以在较高抽象程度上指定位置(例如国家、州或数据中心),因此具有位置无关性。

[0332] 迅速弹性:可以迅速且有弹性地(在一些情况下自动地)提供能力以快速向外扩展

并被迅速释放以快速缩小。对于消费者,可用于提供的能力通常看起来是无限的,并可以在任何时间以任何数量购买。

[0333] 可测量的服务:云系统通过利用在适于服务类型(例如,存储、处理、带宽和活动用户账户)的某一抽象程度的计量能力,自动地控制和优化资源使用。可以监视、控制和报告资源使用情况,为所利用的服务的提供者和消费者双方提供透明度。

[0334] 服务模型如下:

[0335] 软件即服务(SaaS):向消费者提供的能力是使用提供者在云基础架构上运行的应用。可通过诸如网络浏览器的瘦客户机接口(例如,基于网络的电子邮件)来从各种客户机设备访问应用。除了有限的特定于用户的应用配置设置以外,消费者既不管理也不控制包括网络、服务器、操作系统、存储、或甚至单个应用能力等的底层云基础架构。

[0336] 平台即服务(PaaS):向消费者提供的能力是在云基础架构上部署消费者创建或获得的应用,这些应用是使用由提供商支持的编程语言和工具创建的。消费者既不管理也不控制包括网络、服务器、操作系统或存储的底层云基础架构,但对其部署的应用具有控制权,对应用托管环境配置可能也具有控制权。

[0337] 基础设施即服务(IaaS):向消费者提供的能力是提供消费者能够在其中部署并运行包括操作系统和应用的任意软件的处理、存储、网络和其它基础计算资源。消费者既不管理也不控制底层云基础架构,但对操作系统、存储、所部署的应用具有控制权,对所选择的网络组件(例如,主机防火墙)可能具有有限的控制权。

[0338] 部署模型如下:

[0339] 私有云:云基础设施单独为某个组织运行。它可以由该组织或第三方管理,并且可以存在于该组织内部或外部。

[0340] 共同体云:云基础设施被若干组织共享,并支持具有共同利害关系(例如,任务、安全要求、政策和合规考虑)的特定共同体。它可以由该组织或第三方管理,并且可以存在于该组织内部或外部。

[0341] 公共云:云基础设施可用于一般公众或大型产业群,并由销售云服务的组织拥有。

[0342] 混合云:云基础设施由两个或更多云(私有云、共同体云或公共云)组成,这些云依然是独特实体,但是通过使数据和应用能够移植的标准化技术或私有技术(例如,用于云之间的负载平衡的云突发)绑定在一起。

[0343] 云计算环境是面向服务的,特点集中在无状态性、低耦合性、模块性和语义的互操作性。计算的核心是包括互连节点网络的基础架构。

[0344] 现在参考图10,描绘了说明性云计算环境50。如图所示,云计算环境50具有云消费者所使用的本地计算设备可以与其通信的一个或多个云计算节点52。这些本地计算设备的示例包括但不限于个人数字助理(PDA)或蜂窝电话54A、台式计算机54B、膝上型计算机54C、和/或汽车计算机系统54N。节点52可彼此通信。它们可以被物理地或虚拟地分组(未示出)在一个或多个网络中,诸如如上所描述的私有云、共同体云、公共云或混合云、或其组合。这允许云计算环境50提供基础设施即服务、平台即服务和/或软件即服务,而云消费者不需要为其在本地计算设备上维护资源。应当理解,图10中所示的计算设备54A-N的类型仅仅是说明性的,并且计算节点52和云计算环境50可通过任何类型的网络和/或网络可寻址连接(例如,使用网络浏览器)与任何类型的计算机化设备通信。

[0345] 现在参考图11,示出了由云计算环境50(图10)提供的一组功能抽象层。应当预先理解,图11中所示的组件、层和功能仅仅是说明性的,并且本发明的实施例不限于此。如所描绘的,提供了以下层和相应的功能:

[0346] 硬件和软件层60包括硬件和软件组件。硬件组件的示例包括:大型机61;基于RISC(精简指令集计算机)架构的服务器62;服务器63,刀片服务器64;存储设备65;以及网络和联网组件66。在一些实施例中,软件组件包括网络应用服务器软件67和数据库软件68。

[0347] 虚拟化层70提供抽象层,从该抽象层可以提供虚拟实体的以下示例:虚拟服务器71;虚拟存储72;虚拟网络73,包括虚拟专用网络;虚拟应用程序和操作系统74;以及虚拟客户端75。

[0348] 在一个示例中,管理层80可以提供以下描述的功能。资源供应81功能提供用于在云计算环境内执行任务的计算资源和其他资源的动态获取。计量和定价82功能提供对在云计算环境中使用资源的成本跟踪,并为这些资源的消耗提供账单或发票。在一个示例中,这些资源可以包括应用软件许可。安全功能为云消费者和任务提供身份验证,并为数据和其他资源提供保护。用户门户83功能为消费者和系统管理员提供对云计算环境的访问。服务水平管理84功能提供云计算资源分配和管理,以满足所需的服务水平。服务水平协议(SLA)计划和履行85功能提供对根据SLA针对其预测未来需求的云计算资源的预安排和采购。

[0349] 工作负载层90可以利用云计算环境的功能的示例。在该层中,可提供的工作负载和功能的示例包括但不限于:地图绘制和导航91;软件开发及生命周期管理92;虚拟教室的教学提供93;数据分析处理94;交易处理95;以及神经网络处理辅助处理96。

[0350] 本发明的各方面可以是任何可能的技术细节集成水平的系统、方法和/或计算机程序产品。计算机程序产品可以包括其上具有计算机可读程序指令的计算机可读存储介质(或多个介质),所述计算机可读程序指令用于使处理器执行本发明的各方面。

[0351] 计算机可读存储介质可以是能够保留和存储由指令执行设备使用的指令的有形设备。计算机可读存储介质可以是例如但不限于电子存储设备、磁存储设备、光存储设备、电磁存储设备、半导体存储设备或前述的任何合适的组合。计算机可读存储介质的更具体示例的非穷举列表包括以下:便携式计算机磁盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦除可编程只读存储器(EPROM或闪存)、静态随机存取存储器(SRAM)、便携式光盘只读存储器(CD-ROM)、数字多功能盘(DVD)、记忆棒、软盘、诸如上面记录有指令的打孔卡或凹槽中的凸起结构的机械编码装置,以及上述的任何适当组合。如本文所使用的计算机可读存储介质不应被解释为暂时性信号本身,诸如无线电波或其他自由传播的电磁波、通过波导或其他传输介质传播的电磁波(例如,通过光纤线缆的光脉冲)、或通过导线传输的电信号。

[0352] 本文描述的计算机可读程序指令可以从计算机可读存储介质下载到相应的计算/处理设备,或者经由网络(例如因特网、局域网、广域网和/或无线网络)下载到外部计算机或外部存储设备。网络可以包括铜传输电缆、光传输光纤、无线传输、路由器、防火墙、交换机、网关计算机和/或边缘服务器。每个计算/处理设备中的网络适配卡或网络接口从网络接收计算机可读程序指令,并转发计算机可读程序指令以存储在相应计算/处理设备内的计算机可读存储介质中。

[0353] 用于执行本发明的操作的计算机可读程序指令可以是汇编指令、指令集架构

(ISA)指令、机器相关指令、微代码、固件指令、状态设置数据、集成电路的配置数据、或者以一种或多种编程语言(包括面向对象的编程语言,例如Smalltalk、C++等)和过程编程语言(例如“C”编程语言或类似的编程语言)的任意组合编写的源代码或目标代码。计算机可读程序指令可以完全在用户的计算机上执行,部分在用户的计算机上执行,作为独立的软件包执行,部分在用户的计算机上并且部分在远程计算机上执行,或者完全在远程计算机或服务器上执行。在后一种情况下,远程计算机可以通过任何类型的网络(包括局域网(LAN)或广域网(WAN))连接到用户的计算机,或者可以连接到外部计算机(例如,使用因特网服务提供商通过因特网)。在一些实施例中,为了执行本发明的各方面,包括例如可编程逻辑电路、现场可编程门阵列(FPGA)或可编程逻辑阵列(PLA)的电子电路可以通过利用计算机可读程序指令的状态信息来执行计算机可读程序指令以使电子电路个性化。

[0354] 在此参考根据本发明实施例的方法、装置(系统)和计算机程序产品的流程图和/或框图描述本发明的各方面。将理解,流程图和/或框图的每个框以及流程图和/或框图中的框的组合可以由计算机可读程序指令来实现。

[0355] 这些计算机可读程序指令可以被提供给计算机或其他可编程数据处理装置的处理器以产生机器,使得经由计算机或其他可编程数据处理装置的处理器执行的指令创建用于实现流程图和/或框图的一个或多个框中指定的功能/动作的装置。这些计算机可读程序指令还可以存储在计算机可读存储介质中,其可以引导计算机、可编程数据处理装置和/或其他设备以特定方式工作,使得其中存储有指令的计算机可读存储介质包括制品,该制品包括实现流程图和/或框图的一个或多个框中指定的功能/动作的方面的指令。

[0356] 计算机可读程序指令还可以被加载到计算机、其他可编程数据处理装置或其他设备上,以使得在计算机、其他可编程装置或其他设备上执行一系列操作步骤,以产生计算机实现的过程,使得在计算机、其他可编程装置或其他设备上执行的指令实现流程图和/或框图的一个或多个框中指定的功能/动作。

[0357] 附图中的流程图和框图示出了根据本发明的各种实施例的系统、方法和计算机程序产品的可能实现的架构、功能和操作。在这点上,流程图或框图中的每个框可以表示指令的模块、段或部分,其包括用于实现指定的逻辑功能的一个或多个可执行指令。在一些替代实施方案中,框中所注明的功能可不按图中所注明的次序发生。例如,连续示出的两个框实际上可以作为一个步骤来实现,同时、基本同时、以部分或全部时间重叠的方式执行,或者这些框有时可以以相反的顺序执行,这取决于所涉及的功能。还将注意,框图和/或流程图图示的每个框以及框图和/或流程图图示中的框的组合可以由执行指定功能或动作或执行专用硬件和计算机指令的组合的专用的基于硬件的系统来实现。

[0358] 除了上述之外,可以由提供客户环境的管理的服务提供商来提供、部署、管理、服务等一个或多个方面。例如,服务提供商可以针对一个或多个客户创建、维护、支持等执行一个或多个方面的计算机代码和/或计算机基础设施。作为回报,服务提供商可以例如在订阅和/或费用协议下从客户接收支付。附加或替代地,服务提供商可以从向一个或多个第三方销售广告内容中接收支付。

[0359] 在一个方面,可以部署应用以执行一个或多个实施例。作为一个示例,应用的部署包括提供可操作以执行一个或多个实施例的计算机基础设施。

[0360] 作为另一方面,可以部署计算基础设施,包括将计算机可读代码集成到计算系统

中,其中代码与计算系统结合能够执行一个或多个实施例。

[0361] 作为又一方面,可以提供一种用于集成计算基础设施的过程,包括将计算机可读代码集成到计算机系统中。计算机系统包括计算机可读介质,其中计算机介质包括一个或多个实施例。代码与计算机系统结合能够执行一个或多个实施例。

[0362] 尽管上文描述了各种实施例,但这些仅是实例。例如,其它架构的计算环境可用于结合和/或使用一个或多个方面。进一步地,可以使用不同的指令、功能和/或操作。另外,可以使用不同类型的寄存器和/或不同的寄存器。进一步地,可以支持其他数据格式、数据布局和/或数据大小。在一个或多个实施例中,可以使用一个或多个通用处理器、一个或多个专用处理器或通用处理器和专用处理器的组合。许多变型是可以的。

[0363] 本文描述了各个方面。此外,在不背离本发明的各方面的精神的情况下,许多变型是可能的。应当注意,除非另外不一致,否则本文所述的每个方面或特征及其变型可与任何其它方面或特征组合。

[0364] 进一步地,其它类型的计算环境也可以受益并被使用。作为示例,适合于存储和/或执行程序代码的数据处理系统是可用的,其包括通过系统总线直接或间接耦合到存储器元件的至少两个处理器。存储器元件包括例如在程序代码的实际执行期间采用的本地存储器、大容量存储装置和高速缓冲存储器,该高速缓冲存储器提供至少一些程序代码的临时存储以便减少在执行期间必须从大容量存储装置检索代码的次数。

[0365] 输入/输出或I/O设备(包括但不限于键盘、显示器、定点设备、DASD、磁带、CD、DVD、拇指驱动器和其它存储介质等)可以直接或通过中间I/O控制器耦合到系统。网络适配器也可以耦合到系统,以使数据处理系统能够通过中间专用或公共网络耦合到其它数据处理系统或远程打印机或存储设备。调制解调器、电缆调制解调器和以太网卡只是几种可用的网络适配器类型。

[0366] 本文所用的术语仅是为了描述特定实施例的目的,而不是旨在进行限制。如本文所用,单数形式“一”、“一个”和“该”旨在也包括复数形式,除非上下文另有明确指示。还将理解,术语“包括”和/或“包含”在本说明书中使用时指定所陈述的特征、整数、步骤、操作、元件和/或组件的存在,但不排除一个或多个其它特征、整数、步骤、操作、元件、组件和/或其群组的存在或添加。

[0367] 如果存在,下面的权利要求中的所有装置或步骤加上功能元件的对应结构、材料、动作和等同物旨在包括用于与具体要求保护的其它要求保护的元件组合执行功能的任何结构、材料或动作。已经出于说明和描述的目的呈现了对一个或多个实施例的描述,但是该描述不旨在是穷尽的或者限于所公开的形式。许多修改和变化对于本领域普通技术人员来说是显而易见的。选择和描述实施例是为了最好地解释各个方面和实际应用,并且使本领域的其他普通技术人员能够理解具有适合于所设想的特定用途的各种修改的各种实施例。

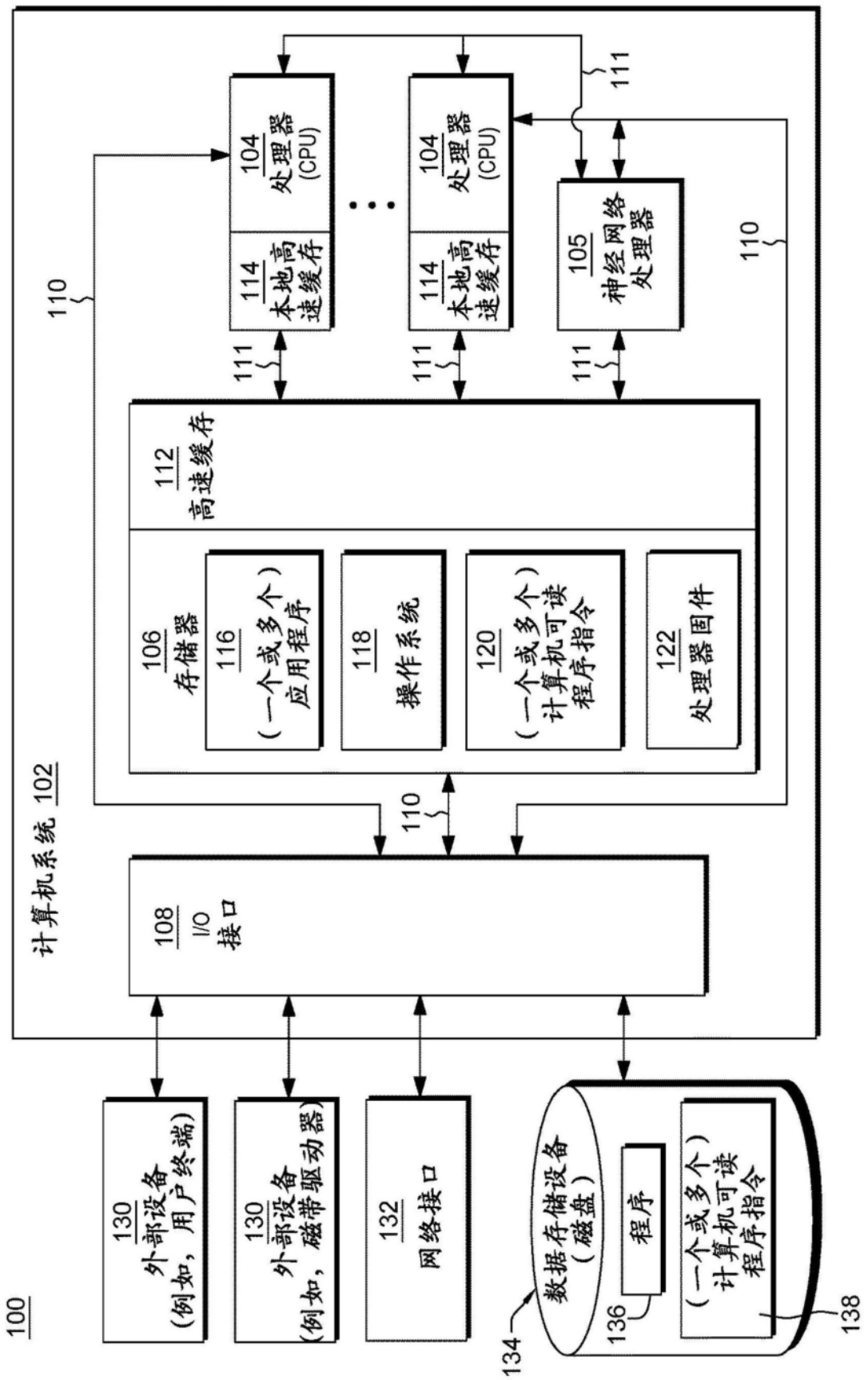

100
计算机系统 102
104 处理器 (CPU)
104 处理器 (CPU)
105 神经网络处理器
114 本地高速缓存
114 本地高速缓存
110
111
112 高速缓存
106 存储器
116 (一个或多个) 应用程序
118 操作系统
120 (一个或多个) 计算机可读程序指令
122 处理器固件
108 I/O 接口
130 外部设备 (例如，用户终端)
130 外部设备 (例如，磁带驱动器)
132 网络接口
数据存储设备 (磁盘)
程序
(一个或多个) 计算机可读程序指令
134
136
138

图1A

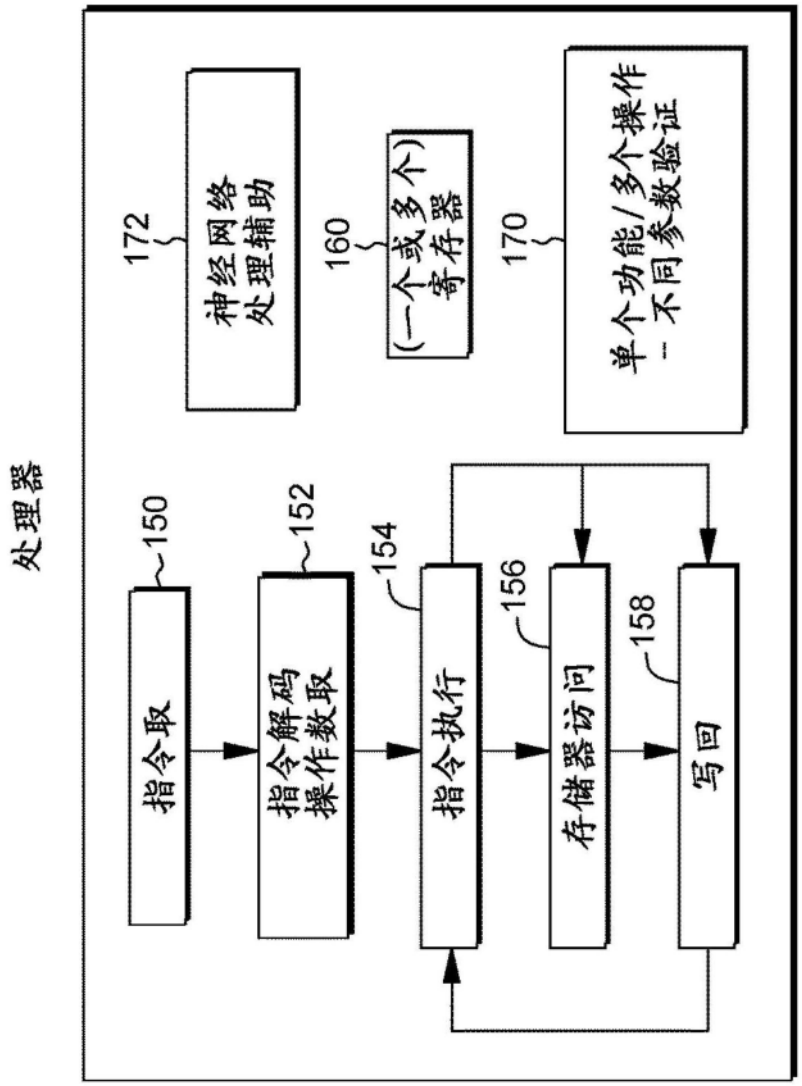

处理器
指令取
150
指令解码
操作数取
152
指令执行
154
存储器访问
156
写回
158
神经网络
处理辅助
172
(一个或多个)
寄存器
160
单个功能/多个操作
- 不同参数验证
170

图1B

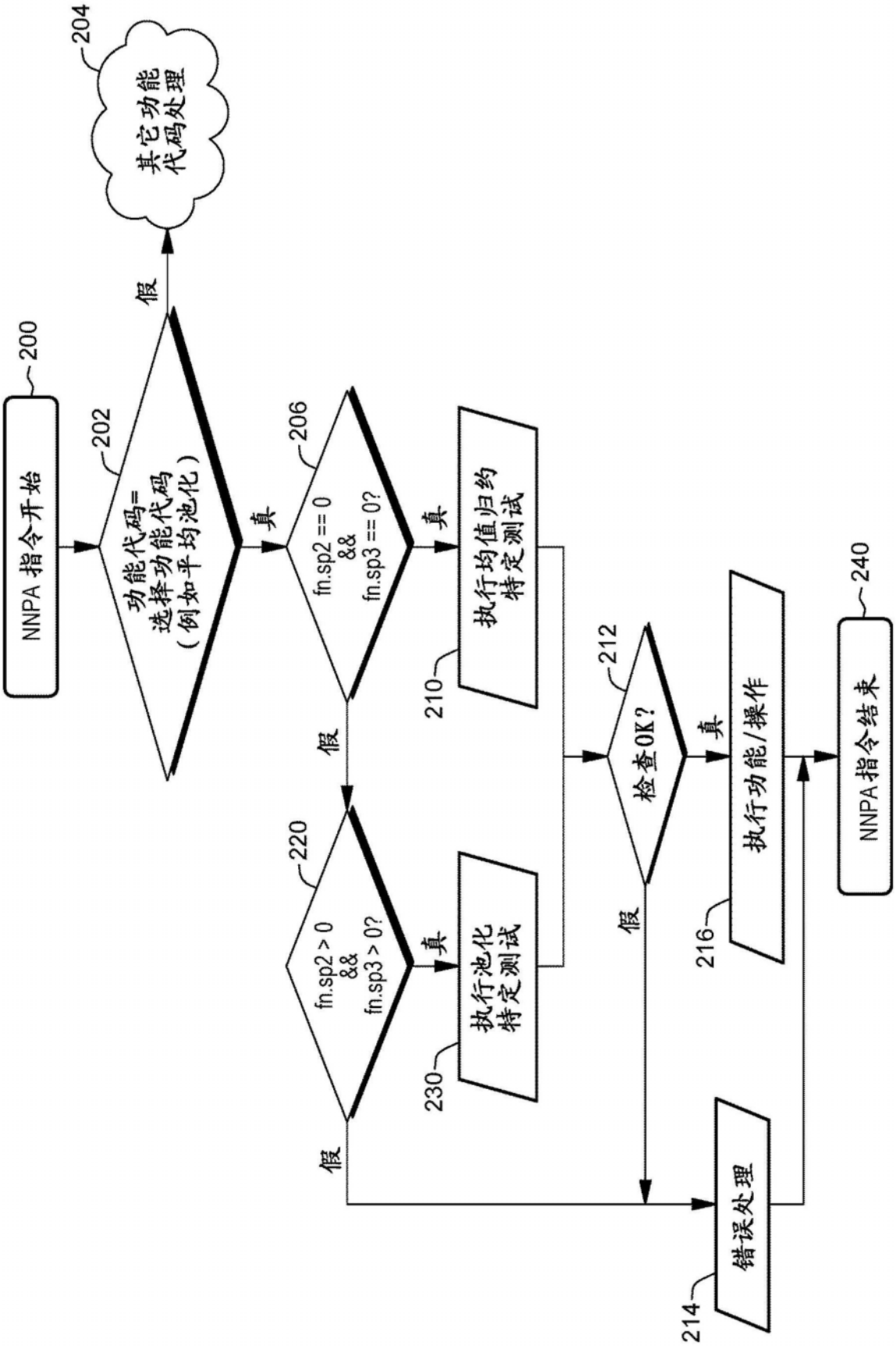

NNPA 指令开始
200
功能代码=
选择功能代码
(例如平均池化)
202
假
其它功能
代码处理
204
真
fn.sp2 == 0
&&
fn.sp3 == 0?
206
真
执行均值归约
特定测试
210
假
fn.sp2 > 0
&&
fn.sp3 > 0?
220
真
执行池化
特定测试
230
假
检查OK?
212
真
执行功能/操作
216
假
错误处理
214
NNPA指令结束
240

图2


图3A


图3B

314

| 代码（十进制） | 代码（十六进制） | 功能 | 参数块大小（字节） |
|---|---|---|---|
| 0 | 0 | NNPA - QAF | 256 |
| 16 | 10 | NNPA - ADD | 4096 |
| 17 | 11 | NNPA - SUB | 4096 |
| 18 | 12 | NNPA - MUL | 4096 |
| 19 | 13 | NNPA - DIV | 4096 |
| 20 | 14 | NNPA - MIN | 4096 |
| 21 | 15 | NNPA - MAX | 4096 |
| 32 | 20 | NNPA - LOG | 4096 |
| 33 | 21 | NNPA - EXP | 4096 |
| 49 | 31 | NNPA - RELU | 4096 |
| 50 | 32 | NNPA - TANH | 4096 |
| 51 | 33 | NNPA - SIGMOID | 4096 |
| 52 | 34 | NNPA - SOFTMAX | 4096 |
| 64 | 40 | NNPA - BATCHNORM | 4096 |
| 80 | 50 | NNPA - MAXPOOL2D | 4096 |
| 81 | 51 | NNPA - AVGPOOL2D | 4096 |
| 96 | 60 | NNPA - LSTMACT | 4096 |
| 97 | 61 | NNPA - GRUACT | 4096 |
| 112 | 70 | NNPA - CONVOLUTION | 4096 |
| 113 | 71 | NNPA - MATMUL-OP | 4096 |
| 114 | 72 | NNPA - MATMUL-OP-BCAST23 | 4096 |

图3C

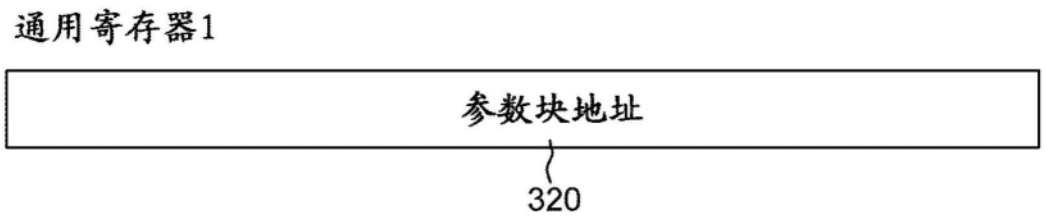



图3D

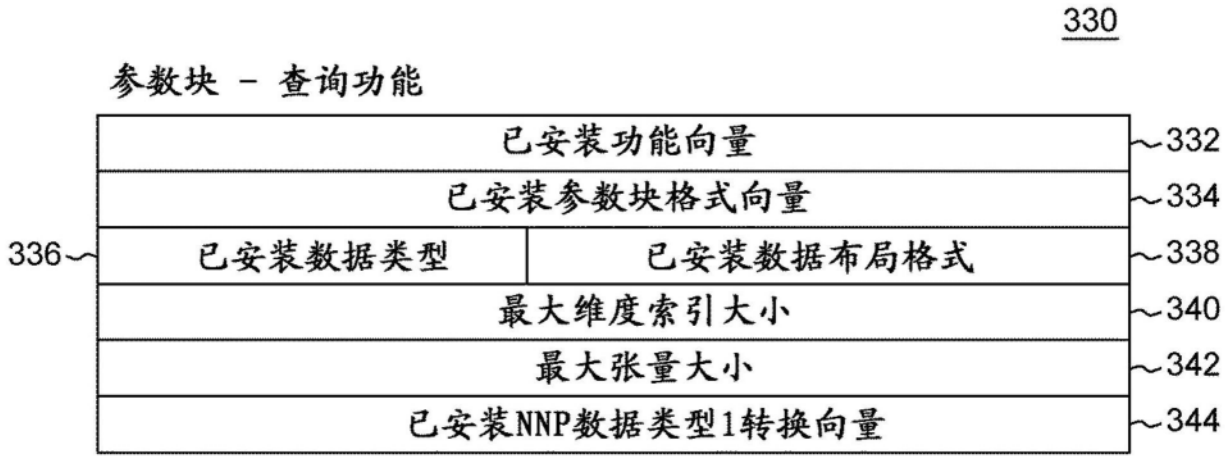

330
参数块 – 查询功能
已安装功能向量
332
已安装参数块格式向量
334
336
已安装数据类型
已安装数据布局格式
338
最大维度索引大小
340
最大张量大小
342
已安装NNP数据类型1转换向量
344

图3E


350
参数块 – 非查询功能
352
参数块版本号
模型版本号
354
继续标志
356
功能特定保存区域地址
358
输出张量描述符1
360
输出张量描述符2
输入张量描述符1
365
输入张量描述符2
输入张量描述符3
功能特定参数1
功能特定参数2
370
功能特定参数3
功能特定参数4
功能特定参数5
继续状态缓冲器
375

图3F

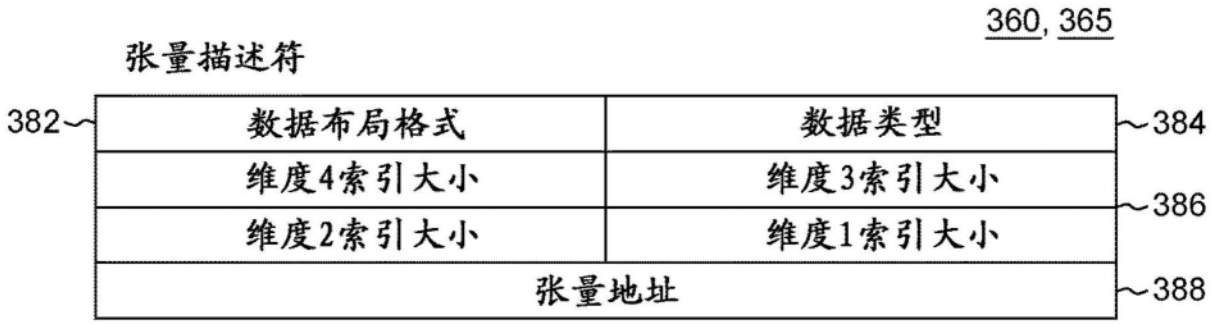

360, 365
张量描述符
382
数据布局格式
数据类型
384
维度4索引大小
维度3索引大小
386
维度2索引大小
维度1索引大小
张量地址
388

图3G

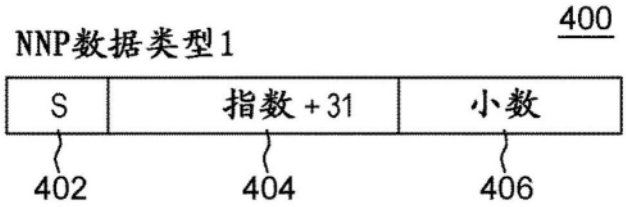

400
NNP数据类型1
S
指数 +31
小数
402
404
406

图4

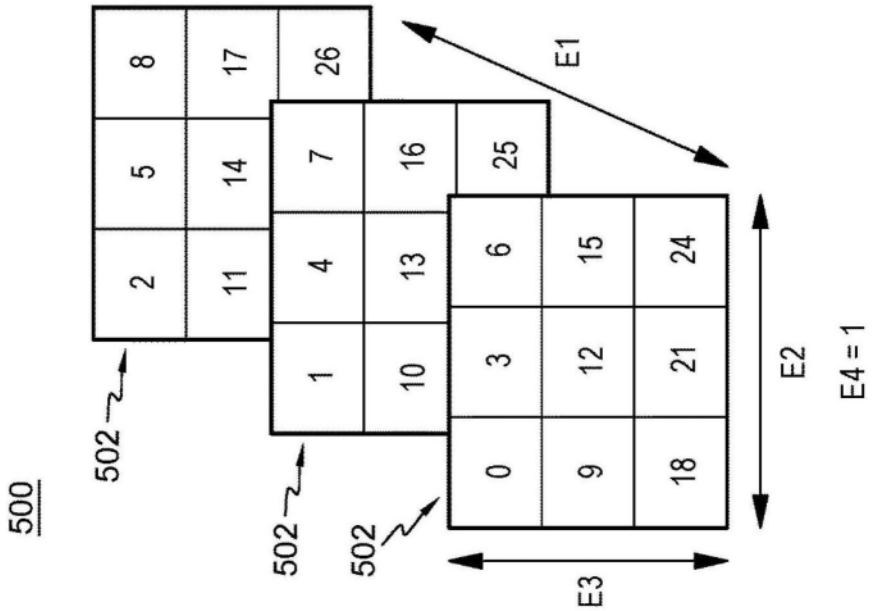

500
502
502
502
E1
E2
E3
E4 = 1
8
17
26
5
14
7
16
25
2
11
4
13
6
15
24
1
10
3
12
21
0
9
18

图5A

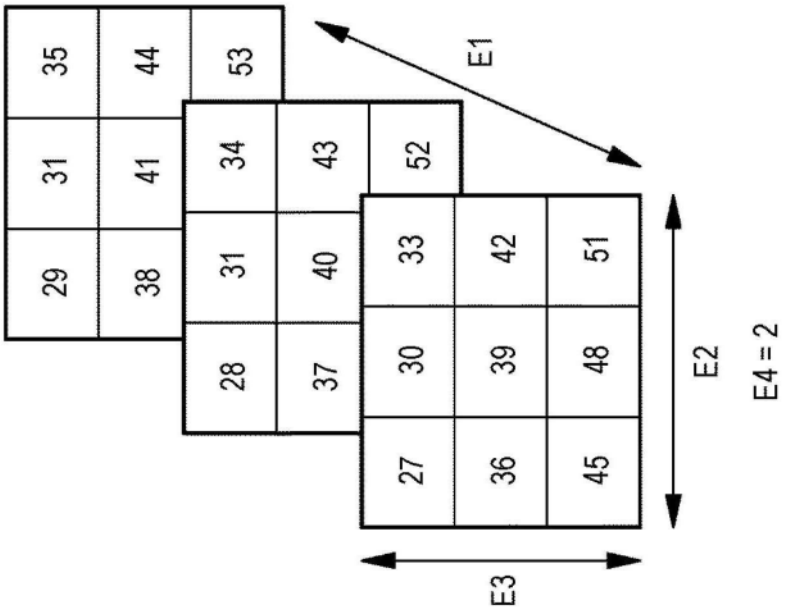

35
44
53
31
41
34
43
52
29
38
31
40
33
42
51
28
37
30
39
48
27
36
45
E1
E2
E4 = 2
E3

图5B

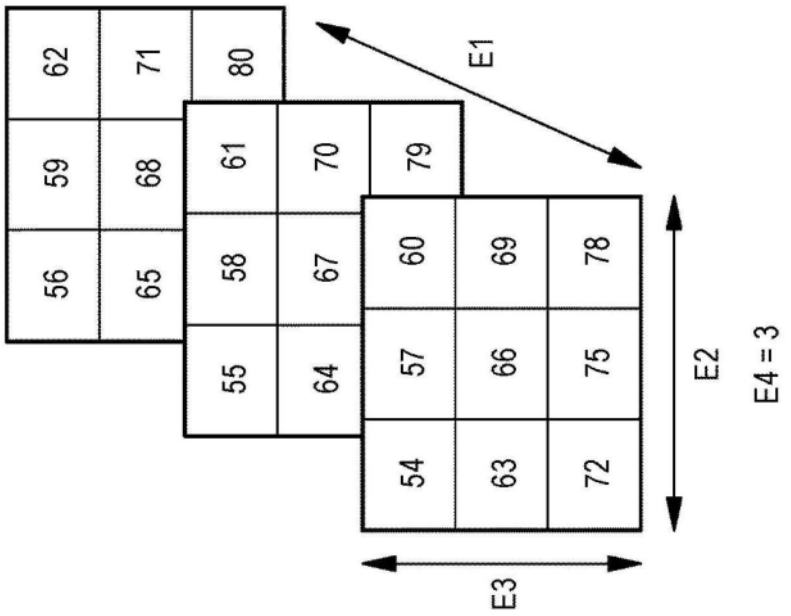

62
71
80
59
68
61
70
79
56
65
58
67
60
69
78
55
64
57
66
75
54
63
72
E1
E2
E4 = 3
E3

图5C

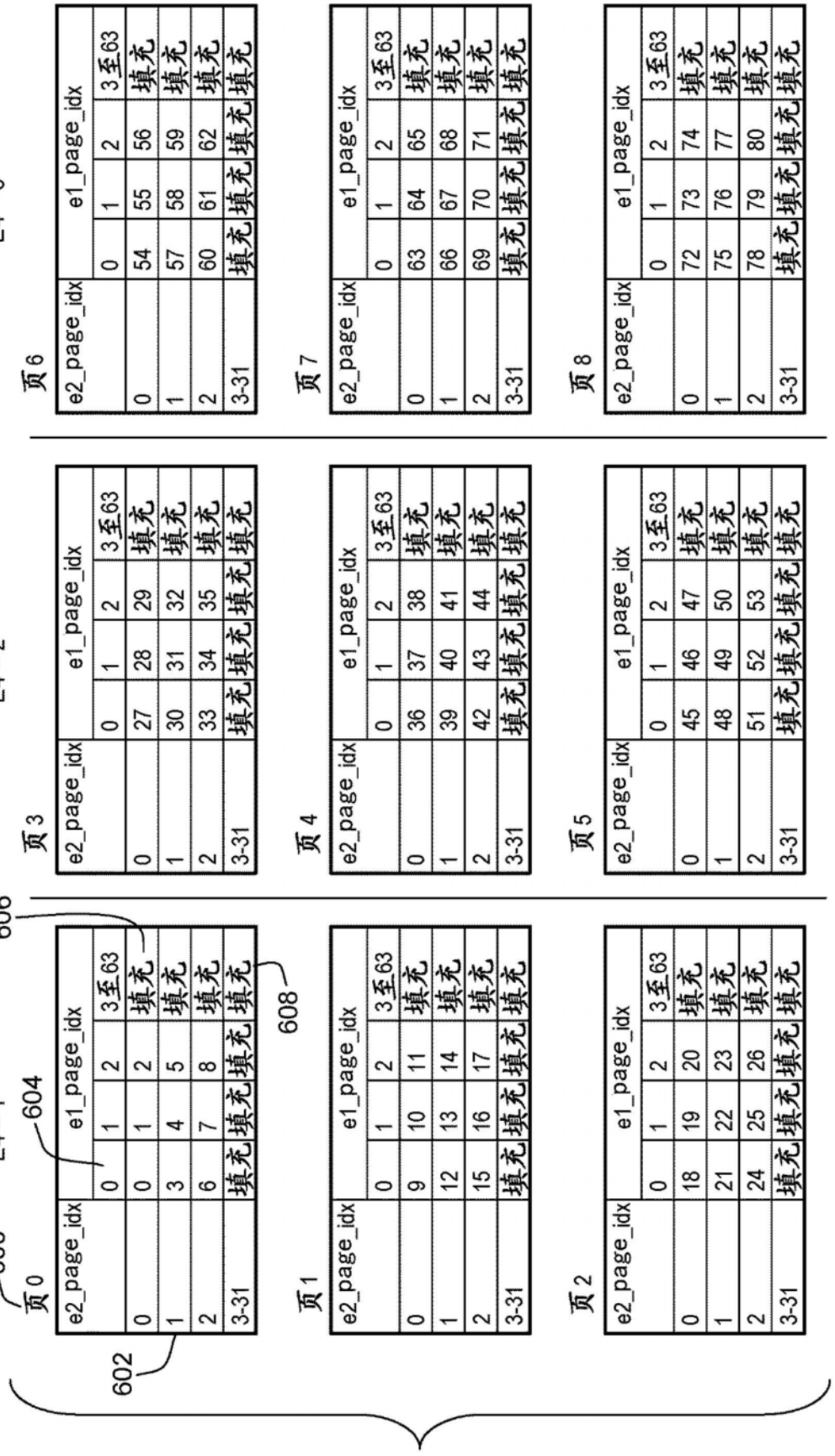

600
602
604
606
608
E4 = 1
E4 = 2
E4 = 3
页 0
页 1
页 2
页 3
页 4
页 5
页 6
页 7
页 8
e2_page_idx
e1_page_idx
0
1
2
3至63
3-31
填充

图 6A 图 6B 图 6C

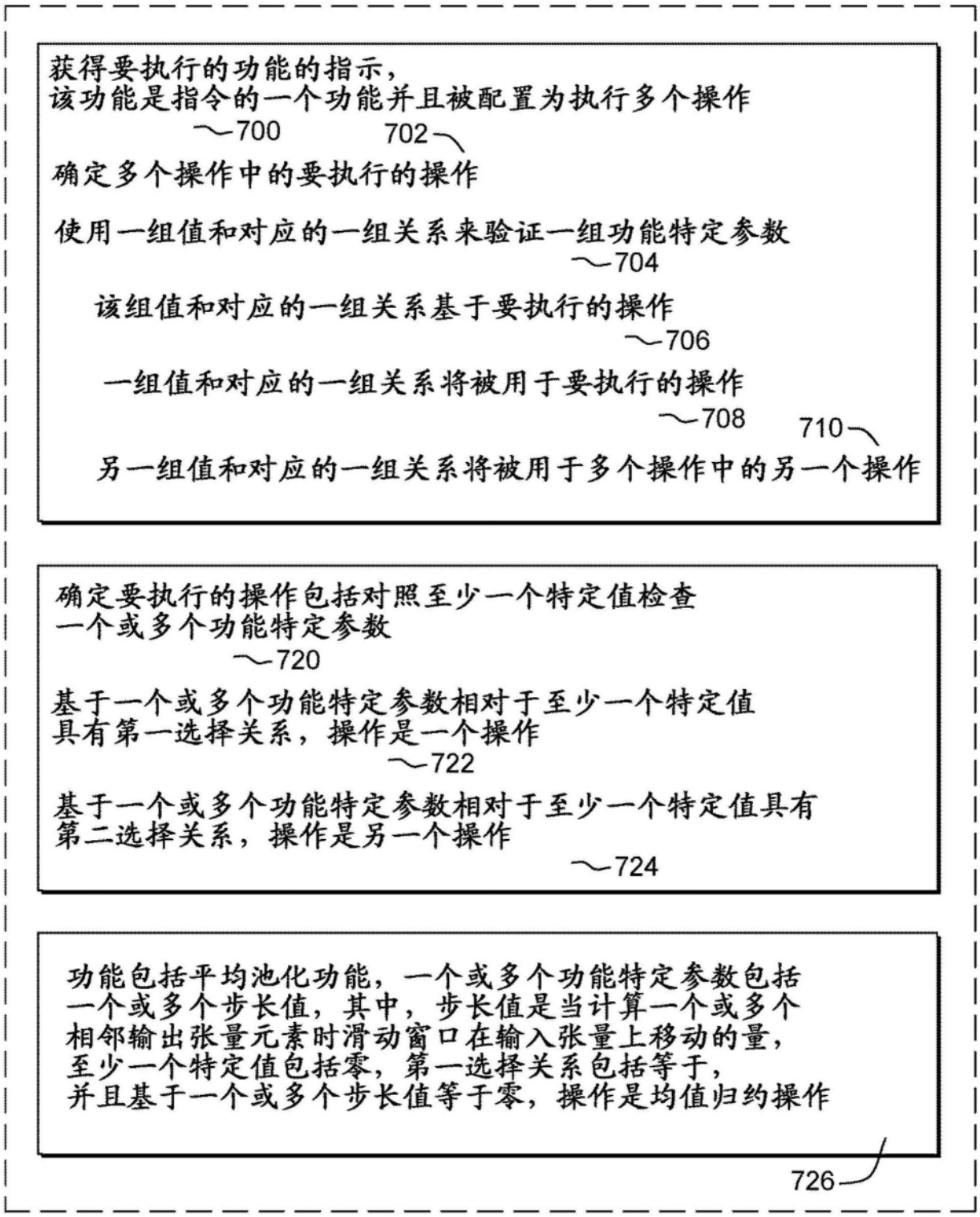

获得要执行的功能的指示，
该功能是指令的一个功能并且被配置为执行多个操作
700
702
确定多个操作中的要执行的操作
使用一组值和对应的一组关系来验证一组功能特定参数
704
该组值和对应的一组关系基于要执行的操作
706
一组值和对应的一组关系将被用于要执行的操作
708
710
另一组值和对应的一组关系将被用于多个操作中的另一个操作
确定要执行的操作包括对照至少一个特定值检查
一个或多个功能特定参数
720
基于一个或多个功能特定参数相对于至少一个特定值
具有第一选择关系，操作是一个操作
722
基于一个或多个功能特定参数相对于至少一个特定值具有
第二选择关系，操作是另一个操作
724
功能包括平均池化功能，一个或多个功能特定参数包括
一个或多个步长值，其中，步长值是当计算一个或多个
相邻输出张量元素时滑动窗口在输入张量上移动的量，
至少一个特定值包括零，第一选择关系包括等于，
并且基于一个或多个步长值等于零，操作是均值归约操作
726

图7A

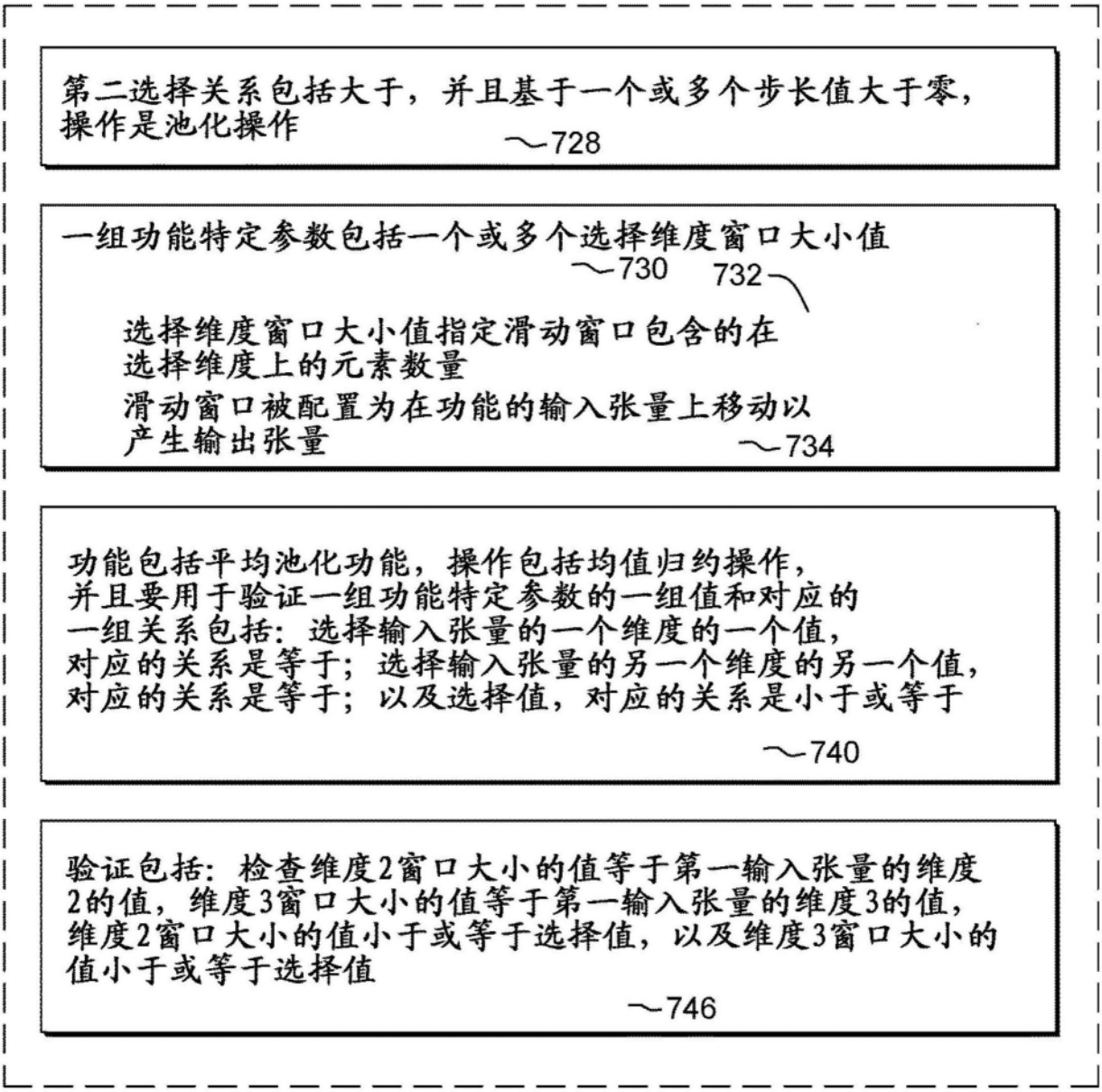

第二选择关系包括大于，并且基于一个或多个步长值大于零，操作是池化操作
728
一组功能特定参数包括一个或多个选择维度窗口大小值
730
732
选择维度窗口大小值指定滑动窗口包含的在选择维度上的元素数量
滑动窗口被配置为在功能的输入张量上移动以产生输出张量
734
功能包括平均池化功能，操作包括均值归约操作，并且要用于验证一组功能特定参数的一组值和对应的一组关系包括：选择输入张量的一个维度的一个值，对应的关系是等于；选择输入张量的另一个维度的另一个值，对应的关系是等于；以及选择值，对应的关系是小于或等于
740
验证包括：检查维度2窗口大小的值等于第一输入张量的维度2的值，维度3窗口大小的值等于第一输入张量的维度3的值，维度2窗口大小的值小于或等于选择值，以及维度3窗口大小的值小于或等于选择值
746

图7B

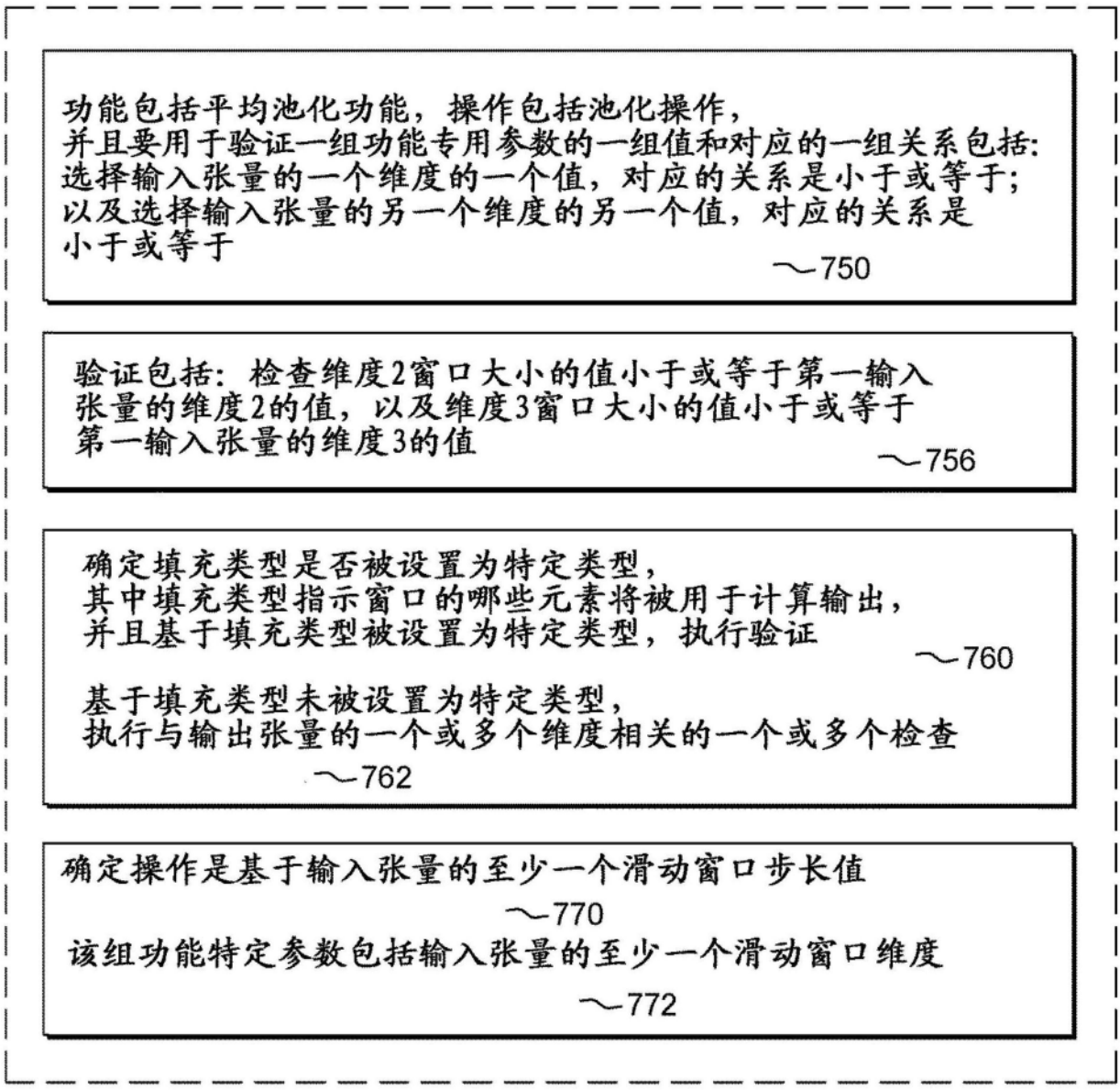

功能包括平均池化功能，操作包括池化操作，
并且要用于验证一组功能专用参数的一组值和对应的一组关系包括：
选择输入张量的一个维度的一个值，对应的关系是小于或等于；
以及选择输入张量的另一个维度的另一个值，对应的关系是
小于或等于
750
验证包括：检查维度2窗口大小的值小于或等于第一输入
张量的维度2的值，以及维度3窗口大小的值小于或等于
第一输入张量的维度3的值
756
确定填充类型是否被设置为特定类型，
其中填充类型指示窗口的哪些元素将被用于计算输出，
并且基于填充类型被设置为特定类型，执行验证
760
基于填充类型未被设置为特定类型，
执行与输出张量的一个或多个维度相关的一个或多个检查
762
确定操作是基于输入张量的至少一个滑动窗口步长值
770
该组功能特定参数包括输入张量的至少一个滑动窗口维度
772

图7C

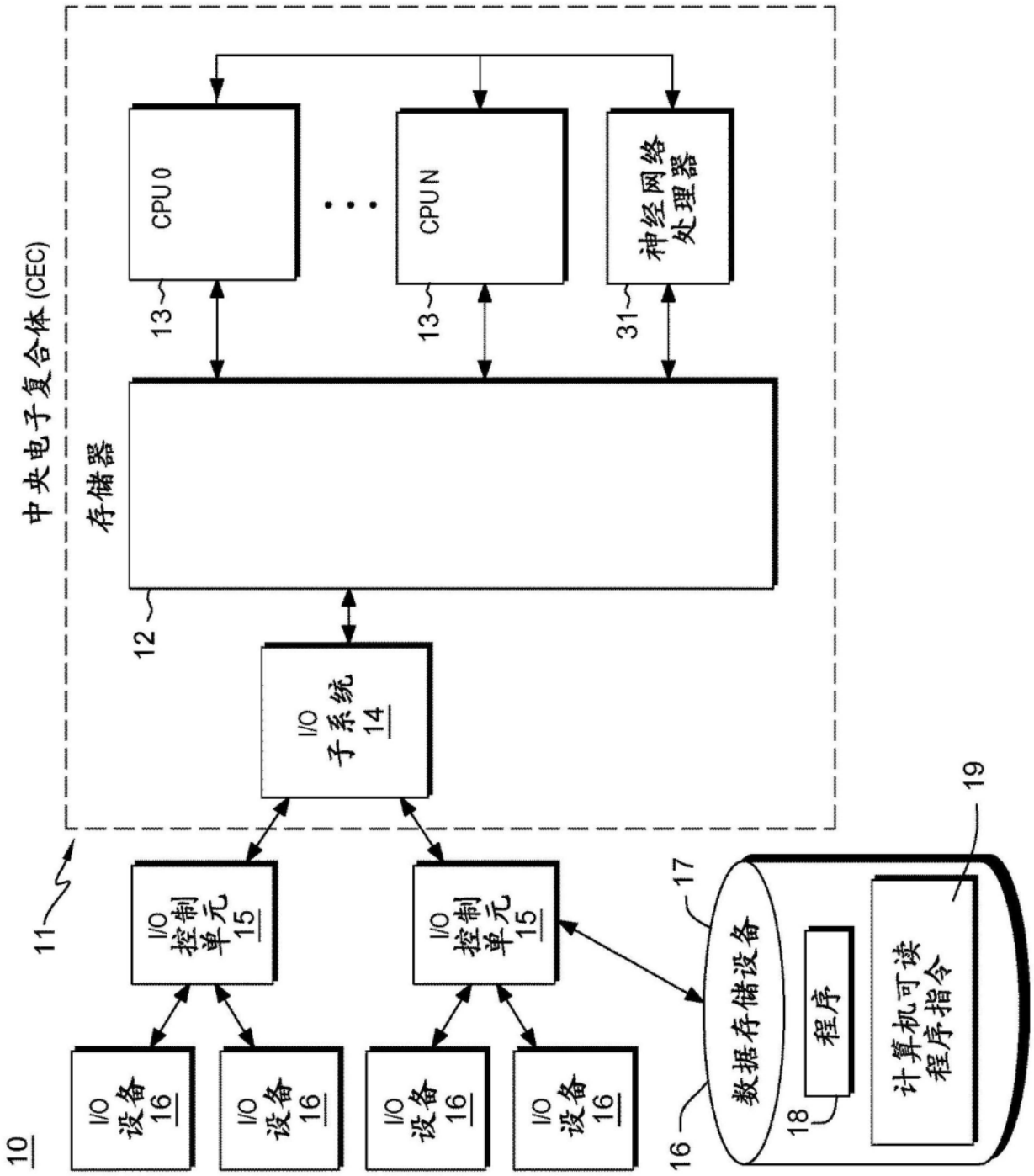

10
11
中央电子复合体 (CEC)
CPU 0
CPU N
神经网络处理器
13
13
31
存储器
12
I/O 子系统 14
I/O 控制单元 15
I/O 控制单元 15
I/O 设备 16
I/O 设备 16
I/O 设备 16
I/O 设备 16
17
数据存储设备
16
18
程序
19
计算机可读程序指令

图8A

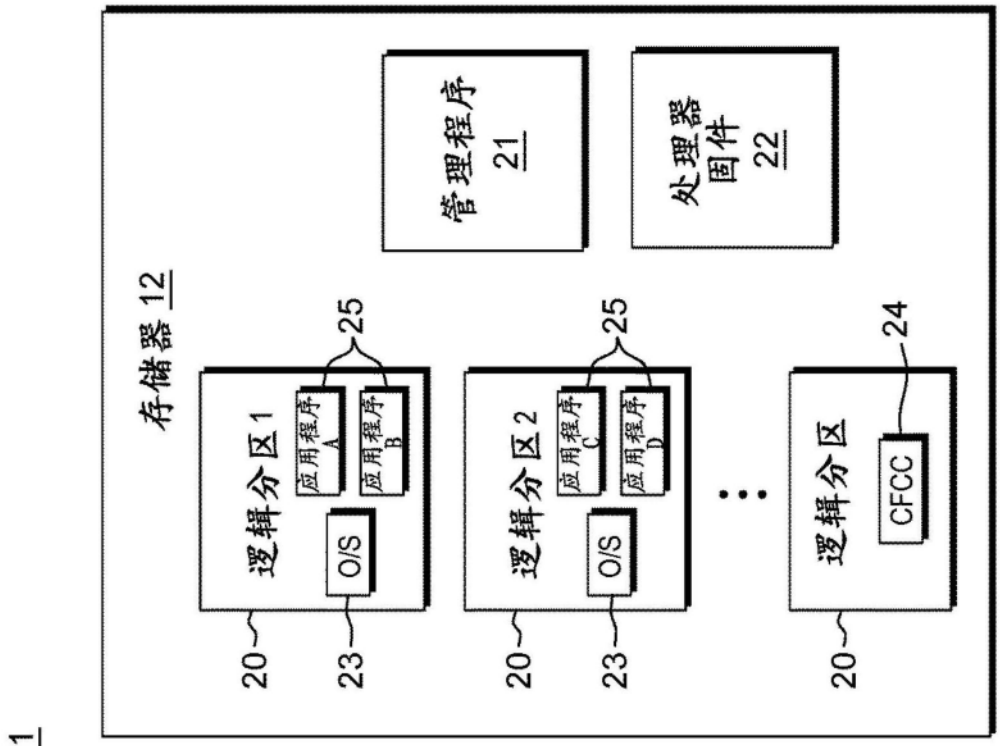

11
存储器 12
管理程序 21
处理器固件 22
逻辑分区 1
应用程序 A
应用程序 B
O/S
逻辑分区 2
应用程序 C
应用程序 D
O/S
逻辑分区
CFCC
20
23
25
24

图8B

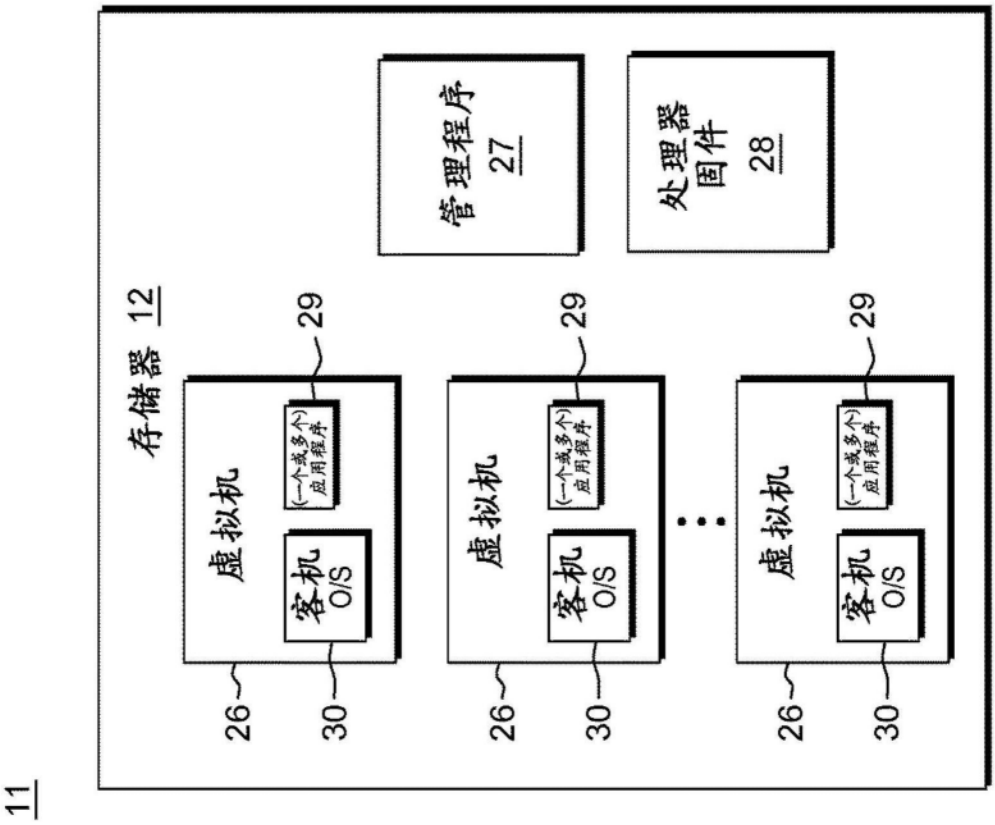

11
存储器 12
管理程序 27
处理器固件 28
虚拟机
客机 O/S
(一个或多个)应用程序
29
26
30

图8C

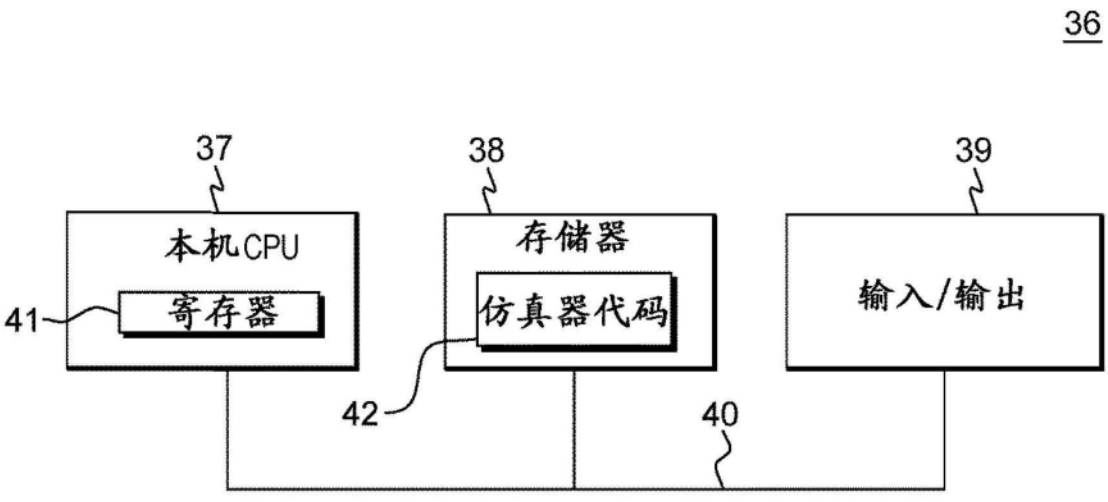

36
37
38
39
本机CPU
寄存器
41
存储器
仿真器代码
42
输入/输出
40

图9A

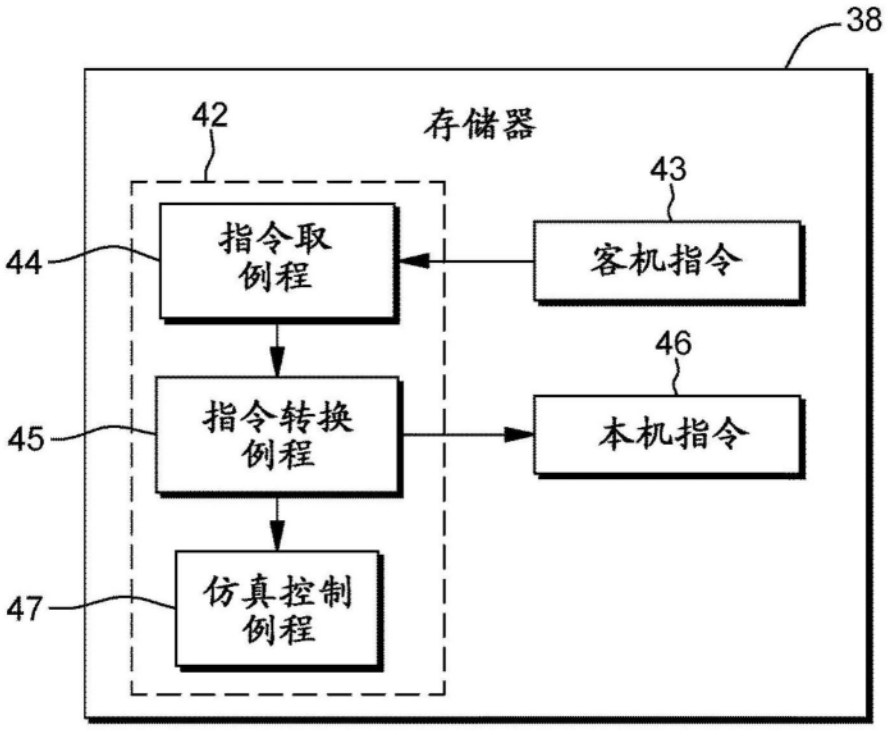



图9B

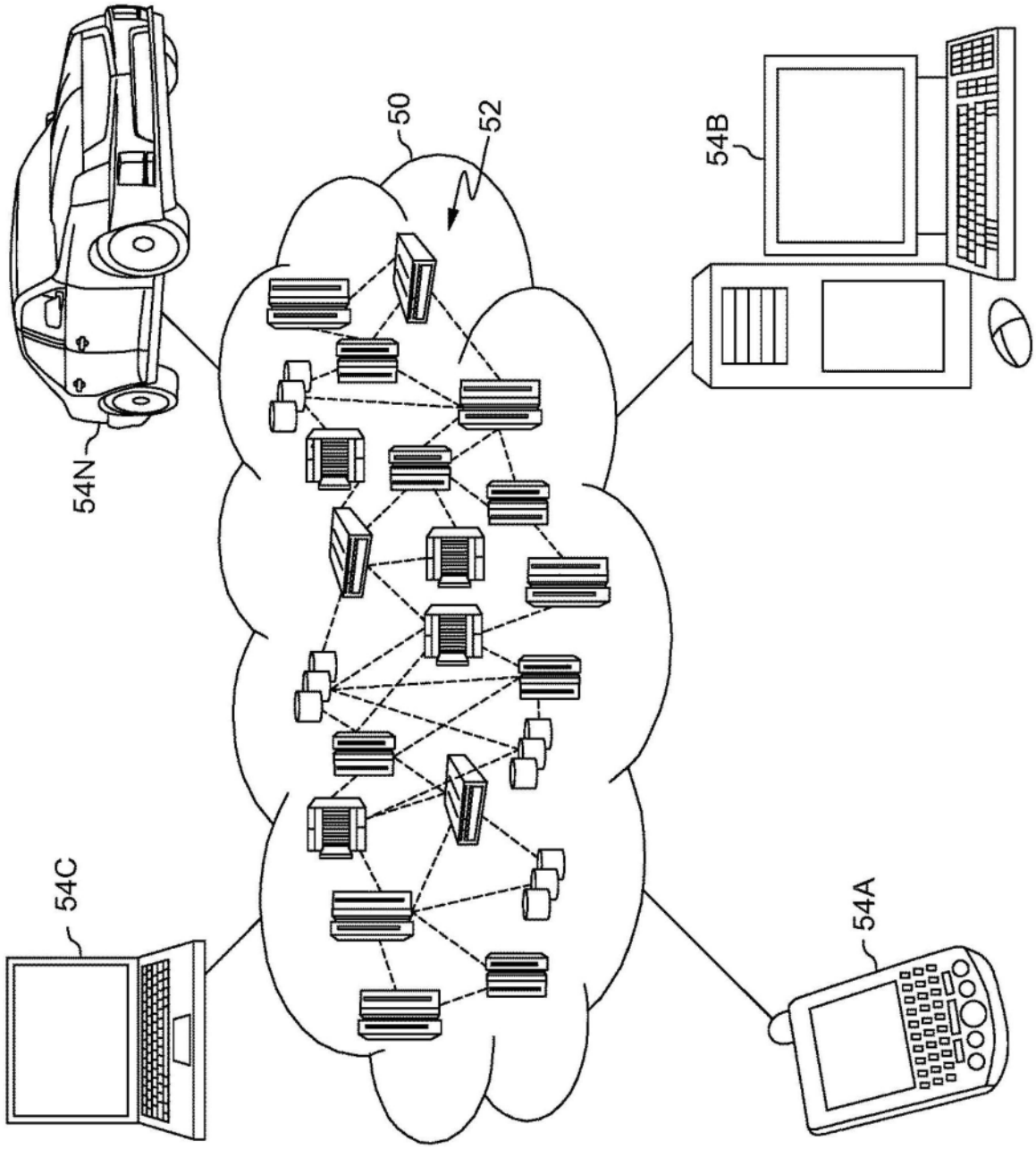



图10

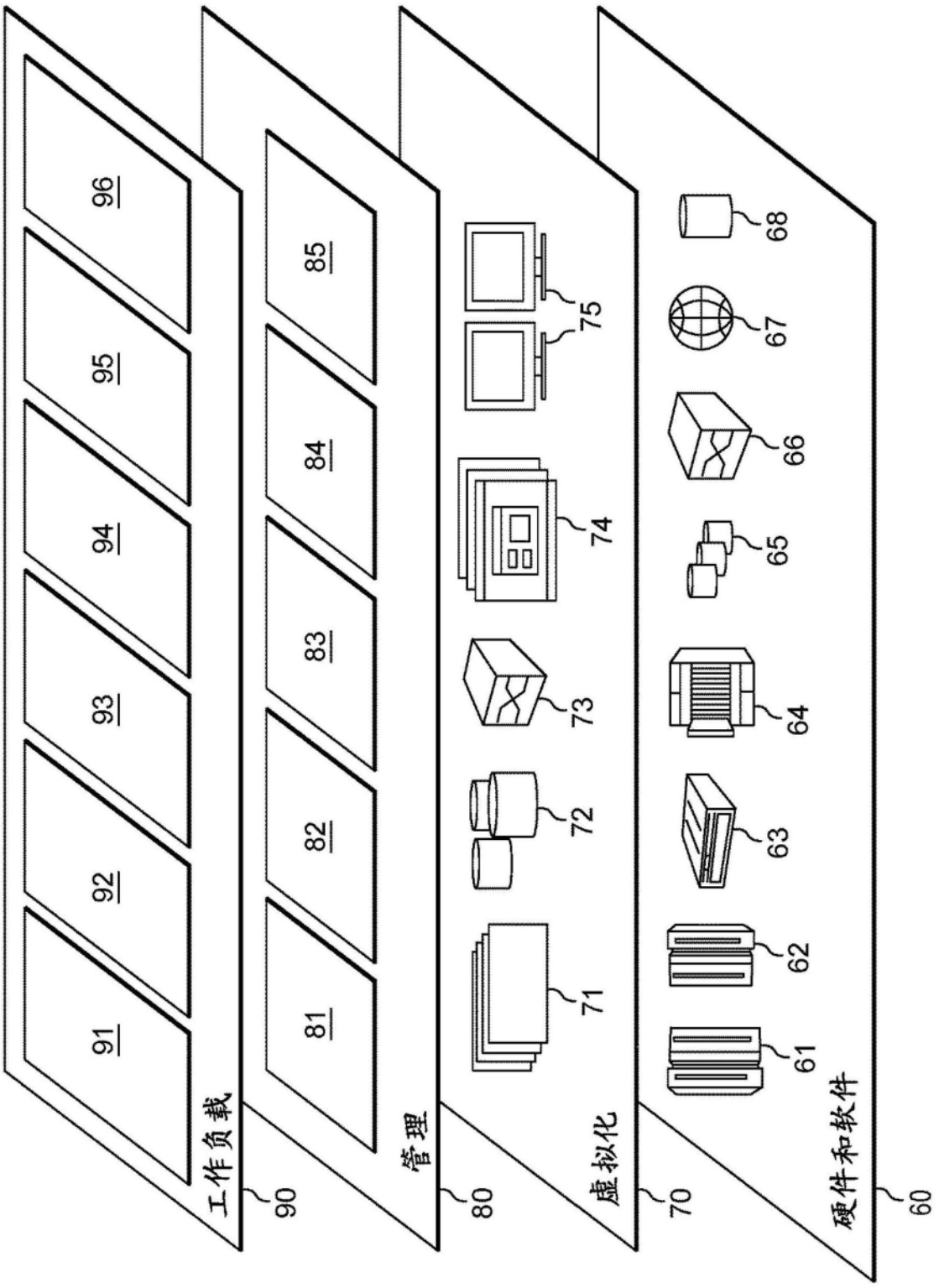

工作负载
90
91
92
93
94
95
96
管理
80
81
82
83
84
85
虚拟化
70
71
72
73
74
75
硬件和软件
60
61
62
63
64
65
66
67
68

图11