

多输入浮点数处理方法、装置、处理器及计算机设备

本申请要求于 2021 年 12 月 29 日提交中国专利局，申请号为 2021116348560、发明名称为“多输入浮点数处理方法、装置、处理器及计算机设备”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及数据处理技术领域，特别是涉及一种多输入浮点数处理方法、装置、处理器、计算机设备和存储介质。

背景技术

随着计算机技术的发展，AI（Artificial Intelligence，人工智能）技术也在飞速发展。在 AI 技术领域，通常通过 AI 处理器实现 AI 算法。在 AI 处理器中，矩阵运算单元是核心的数据处理器件，矩阵运算单元的性能和算力直接决定 AI 处理器的性能优劣。而在矩阵运算单元中，多输入的浮点运算单元是决定性能优劣的关键。

传统方案中的多输入的浮点运算方式，若要达到无中间精度损失的目的，移位后的数据位宽会非常宽，因此通常需要多个高位数的移位器才能保障中间精度无损。而过大的移位范围使得硬件开销巨大，导致处理器需要占据更多的硬件资源。

发明内容

本申请的各种实施例提供一种多输入浮点数处理方法、装置、处理器、计算机设备和存储介质。

根据本申请的各种实施例，提供了一种多输入浮点数处理方法。所述方法包括：

获取与目标任务对应的多个待处理的浮点数，并分别提取每个浮点数中指数部分的指数值和尾数部分的尾数值；

依据各浮点数的指数值大小，对多个所述浮点数进行排序，得到排序结果，基于所述排序结果，从多个具有不同预设位数的移位器中，为每个浮点数分别分配移位器；

对于每个浮点数，均通过为所述浮点数分配的移位器，对相应浮点数的尾数值进行移位处理，得到移位结果；及

基于各所述移位结果，确定与所述目标任务对应的浮点数处理结果。

根据本申请的各种实施例，提供了一种多输入浮点数处理装置。所述装置包括：

获取模块，用于获取与目标任务对应的多个待处理的浮点数，并分别提取每个浮点数中指数部分的指数值和尾数部分的尾数值；

分配模块，用于依据各浮点数的指数值大小，对多个所述浮点数进行排序，得到排序结果，基于所述排序结果，从多个具有不同预设位数的移位器中，为每个浮点数分别分配移位器；

移位模块，用于对于每个浮点数，均通过为所述浮点数分配的移位器，对相应浮点数的尾数值进行移位处理，得到移位结果；及

确定模块，用于基于各所述移位结果，确定与所述目标任务对应的浮点数处理结果。

根据本申请的各种实施例，提供了一种处理器。所述处理器包括：

至少一个具有不同预设位数的移位器，每个移位器被分别分配用于对多个待处理的浮点数中的其中一个浮点数的尾数部分的尾数值进行移位处理；其中，为每个浮点数所分配

的移位器是根据各浮点数的指数部分的指数值大小进行排序所得到的排序结果确定的；

逻辑处理单元，用于对各所述移位器进行移位处理所得到的多个移位结果进行逻辑处理，得到浮点数处理结果。

根据本申请的各种实施例，提供了一种计算机设备。所述计算机设备包括存储器和处理器，所述存储器存储有计算机可读指令，所述处理器执行所述计算机可读指令时实现上述多输入浮点数处理方法的步骤。

根据本申请的各种实施例，提供了一种计算机可读存储介质。所述计算机可读存储介质，其上存储有计算机程序，所述计算机程序被处理器执行时实现上述多输入浮点数处理方法的步骤。

根据本申请的各种实施例，提供了一种计算机程序产品。所述计算机程序产品，包括计算机程序，该计算机程序被处理器执行时实现上述多输入浮点数处理方法的步骤。

本申请的一个或多个实施例的细节在下面的附图和描述中提出。本申请的其它特征、目的和优点将从说明书、附图以及权利要求书变得明显。

附图说明

图1为一个或多个实施例中多输入浮点数处理方法的流程示意图；

图2为一个或多个实施例中浮点数的格式示意图；

图3为一个或多个实施例中分配移位器的步骤的流程示意图；

图4为一个或多个实施例中移位处理的步骤的流程示意图；

图5为另一个或多个实施例中移位处理的步骤的流程示意图；

图6为一个或多个实施例中基于各移位结果确定与目标任务对应的浮点数处理结果的步骤的流程示意图；

图7为一个或多个实施例中分段压缩处理的步骤的流程示意图；

图8为一个或多个实施例中分段压缩处理的原理示意图；

图9为一个或多个实施例中基于多个分段压缩结果，确定与目标任务对应的浮点数处理结果的流程示意图；

图10为一个或多个实施例中选择器输出选择结果的步骤的流程示意图；

图11为一个或多个实施例中浮点数处理结果进行规格化处理的步骤的流程示意图；

图12为一个或多个实施例中现有的多浮点数加法计算的流程示意图；

图13为一个或多个实施例中数据位宽的构成示意图；

图14为一个或多个实施例中本申请应用于多浮点数加法计算的流程示意图；

图15为一个或多个实施例中多输入浮点数处理装置的结构框图；

图16为一个或多个实施例中处理器的结构框图；

图17为一个或多个实施例中逻辑处理单元的结构框图；

图18为一个或多个实施例中计算机设备的内部结构图。

为了更好地描述和说明这里公开的那些发明的实施例和/或示例，可以参考一幅或多幅附图。用于描述附图的附加细节或示例不应当被认为是所公开的发明、目前描述的实施例和/或示例以及目前理解的这些发明的最佳模式中的任何一者的范围的限制。

具体实施方式

为了使本申请的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本

申请进行进一步详细说明。应当理解，此处描述的具体实施例仅仅用以解释本申请，并不用于限定本申请。

本申请提供一种多输入浮点数处理方法、装置、处理器、计算机设备、存储介质和计算机程序产品，通过优化浮点数的运算方式和相应的处理逻辑，不仅能够高效处理多个浮点数的运算，还能通过减少移位器的面积，有效降低 AI 处理器的面积，并降低了时序上的关键路径，提升了 AI 处理器的主频，使得应用了 AI 处理器的单个芯片可提供更高的算力。

在一些实施例中，如图 1 所示，提供了一种多输入浮点数处理方法，本实施例以该方法应用于计算机设备进行举例说明，可以理解的是，计算机设备可以是终端，也可以是服务器。其中，服务器可以是独立的物理服务器，也可以是多个物理服务器构成的服务器集群或者分布式系统，还可以是提供云计算服务的云服务器。终端可以但不限于是各种个人计算机、笔记本电脑、智能手机、平板电脑、物联网设备、或者便携式可穿戴设备等中的一种或多种。物联网设备可为智能音箱、智能电视、智能空调、或者智能车载设备等中的一种或多种。便携式可穿戴设备可为智能手表、智能手环、或者头戴设备等中的一种或多种。示例性地，本申请实施例中提供的多输入浮点数处理方法可以应用于计算机设备中的数据处理器件，例如处理器和传感器等。

本实施例中，该方法包括以下步骤：

步骤 S102，获取与目标任务对应的多个待处理的浮点数，并分别提取每个浮点数中指数部分的指数值和尾数部分的尾数值。

其中，目标任务指的是计算机设备为达到某个目标所执行的计算处理任务，其中，计算处理包括但不限于加、减、乘、或者除等数学运算。例如，目标任务可以是在神经网络训练的过程中的计算处理任务，包括但不限于卷积求和任务或相似度计算任务等。又如，目标任务也可以是云计算或分布式计算任务，用于对多个数据进行计算。

浮点数是属于有理数中某特定子集的数的数字表示，在计算机中用以近似表示任意某个实数。以目前通用的浮点数格式为例，浮点数的格式一般都遵循由微处理器标准委员会（MSC，Microprocessor Standards Committee）制定的 IEEE 二进制浮点数算术标准（ANSI/IEEE Std 754-1985，通常简称为 IEEE 754）。

IEEE 754 标准规定了在计算机内存中以二进制的方式存储十进制浮点数的具体标准，并制定了四种表示浮点数值的方式：单精度浮点数、双精度浮点数、延伸单精度浮点数、以及延伸双精度浮点数。

如图 2 所示，IEEE 754 将浮点数表示为符号、指数部分、以及指数部分三个部分。其中，符号 S 占据最高位，用于表示浮点数为正数或负数；“0”表示浮点数为正数，“1”表示浮点数为负数。指数部分中的指数值 E（通常也可被称为阶码）由移码（又称增码或偏置码）表示，以 32 位单精度浮点数为例，指数值 E 的范围为 8bit（比特），表示 0~255 个指数值。指数值用于指示小数点的位置。浮点数的尾数部分的尾数值 M 由原码表示，同样以 32 位单精度浮点数为例，尾数值 M 的范围为 24bit，确定了浮点数所能表示的实数的精度。需要说明的是，由于原码的最高位必须是有效位（也就是必须为 1），因此，计算机内存中通常会省略（或隐藏）该尾数部分的最高位以节省存储空间，由此即可由图示中 23bit 的尾数部分来表示 24bit 的信息。

为了描述简便且易于理解，以下实施例中所列出的举例说明均符合 IEEE 754 标准，但并不应当理解为对本申请实施例适用范围的限制。

在一些实施例中，待处理的多个浮点数存储于存储器中，存储器可以为计算机设备中设置的内部存储器，也可以是独立于计算机设备并与计算机设备通信连接的外部存储器。

具体地，计算机设备从存储器中获取两个或两个以上待处理的浮点数，并依据浮点数所遵循的格式，提取每个浮点数的指数部分的指数值和位数部分的尾数值。其中，指数值用于后续对浮点数的排序，以此来确定为各个浮点数分配的移位器的大小；尾数值则是用于具体进行移位处理的部分。

再以 32 位单精度浮点数为例，计算机设备将第 31 位（最高位）对应的数值提取作为符号位的值，将第 30 位至第 23 位对应的数值提取作为指数部分的指数值，将第 22 位至第 0 位（最低位）对应的数值提取作为尾数部分的尾数值。承前所述，计算机设备采用隐含尾数部分最高位 1 的表示方式来表示尾数值，故计算机设备提取的位数是第 22 位至第 0 位。

步骤 S104，依据各浮点数的指数值大小，对多个浮点数进行排序，得到排序结果，基于排序结果，从多个具有不同预设位数的移位器中，为每个浮点数分配移位器。

由于各个浮点数的指数值不同，各个浮点数的量纲可能并不一致。为了实现对浮点数的计算，首先需要将不同的浮点数统一至同一量纲中。在此，通过设置多个移位器实现对各个浮点数的移位处理，从而使得各个浮点数均处于同一量纲。移位器的数量基于浮点数的数量而确定。例如，移位器的数量可以与浮点数的数量相同；又如，由于指数值最大值对应的浮点数无需进行移位处理，移位器的数量可以是浮点数的数量减 1，以减少所需要消耗的硬件资源。

各个移位器被预先设置有不同的位数，每个移位器的预设位数包括尾数值的最大范围加上该移位器在进行移位处理时的最大可移位范围。例如，50bit 的移位器的位数为 50 位，由于尾数值最大为 24 位，去除尾数值所占据的空间，则该移位器的最大可移位范围 w 为 26 位。在后续的移位过程中，移位器在其最大可移位范围之内对尾数值进行移动。

为了尽可能地减少硬件开销，在一些实施例中，多个移位器所具有的不同预设位数处于第一预设范围内，且各个移位器的预设位数在第一预设范围内呈均匀分布。具体地，将多个移位器中预设位数的最大值作为第一预设范围，其余各个移位器的移位范围均在该第一预设范围内。各个移位器的预设位数之间呈均匀分布，包括按照一定倍数递增或递减、呈等差数列或等比数列等。示例性地，预先设置有 a 个 q （例如 $q=24\text{bit}+w\text{bit}$ ， $w=26$ ）位的移位器、 b 个 $2*w$ 位的移位器、 c 个 $3*w$ 位的移位器... x 个 $n*w$ 位的移位器，其中， n 的值为浮点数的个数减 1。

为了以示区分，本申请实施例中将移位器的预设位数所位于的范围称为第一预设范围，将压缩器具有的压缩比的数值所位于的范围称为第二预设范围。需要说明的是，上述术语“第一”和“第二”在本申请中用来描述不同的数值范围，但是这些数值范围不应被这些术语限制。这些术语仅用来将一个数值范围与另一个数值范围进行区分。例如，第一预设范围可以被称作第二预设范围，并且类似地，第二预设范围可以被称作第一预设范围，而不脱离各种所描述的实施例的范围，但是除非上下文以其他方式明确指出，否则

它们所指的并不是同一个范围。类似的情况还有下文中的第一域段、第二域段和第三域段，第一符号标识和第二符号标识，第一移位方向和第二移位方向，等等。

具体地，计算机设备在获取了各个浮点数的指数值以后，按照各个浮点数的指数值的大小，对各个浮点数进行排序，得到排序结果。示例性地，计算机设备按照指数值从小到大的顺序，对各个浮点数进行排序，得到排序结果。又如，计算机设备按照指数值从大到小的顺序，对各个浮点数进行排序，得到排序结果。由于指数值最大的浮点数无需进行移位处理，因此，除了该指数值最大的浮点数，计算机设备按照所得到的浮点数的排序结果，先后依次分别为其余每个浮点数分别分配一个移位器。

上述实施例中，通过为不同的浮点数分配不同的移位器，相较于现有技术中无论各个浮点数的大小、均使用同一位数的多个移位器来进行移位处理而言，显著地减少了硬件开销，并且时序特性好，效率高。

在另一些实施例中，多个移位器所具有的不同预设位数在第一预设范围内也可以呈非均匀分布。例如，在第一预设范围内，设置一个移位器的预设位数为 q 位，其余各个移位器的预设位数均为 $n*w$ 位。又如，在第一预设范围内，各个移位器的预设位数呈指数型递增或递减。再如，在第一预设范围内，每 x 个移位器的预设位数相同，且整体呈递增或递减的趋势。

步骤 S106，对于每个浮点数，均通过为浮点数分配的移位器，对相应浮点数的尾数值进行移位处理，得到移位结果。

具体地，对于一个浮点数，计算机设备利用为该浮点数分配的移位器，通过在移位器的预设位数内对该浮点数的尾数值进行移位处理，得到该浮点数的移位结果。由于指数值最大的浮点数无需进行移位处理，计算机设备对于其余各个浮点数均利用各自分配的移位器进行移位处理后，得到多个移位结果。

其中，移位处理指的是对尾数值朝着某一个移位方向移动一定的距离。移位方向包括左移和右移。移位处理中所移动的位数为该浮点数与浮点数最大值之间的差异。示例性地，对于浮点数 A ，其指数值与最大指数值的差值为 b ，则计算机设备将浮点数 A 的尾数值朝左移或右移 b 位。其中，最大指数值为多个浮点数的指数值中的最大值。

步骤 S108，基于各移位结果，确定与目标任务对应的浮点数处理结果。

具体地，移位完成后，计算机设备可根据具体的目标任务进行后续处理，具体可以是根据各个浮点数各自对应的移位结果，依次对各个移位结果进行压缩处理，得到压缩结果；计算机设备再对压缩结果进行后处理，从而得到与目标任务对应的浮点数处理结果。其中，后处理包括标准化处理和舍入处理等。例如，后处理用于对压缩结果进行规格化处理，以使得该压缩结果符合 IEEE 754 所规定的格式，再通过舍入处理，得到最终的浮点数处理结果。

上述多输入浮点数处理方法中，预先设计多个具有不同预设位数的移位器，在对待处理的多个浮点数进行处理时，可依据各浮点数的指数值大小对多个浮点数进行排序，从而基于排序结果从多个具有不同预设位数的移位器中为每个浮点数分别分配对应的移位器，再利用所分配的移位器对多个浮点数进行移位处理，以基于移位处理结果得到目标任务所对应的浮点数处理结果。这样，引入有效移位的思想，对排序靠后的尾数值进行有效移位，排序靠前的尾数值移位范围小（甚至无需进行移位），在保障了无中间精度损失的前提下，

大大减小了移位器的面积开销,进而节省了处理器的硬件开销,在硬件资源有限的前提下,能很好的兼顾浮点数处理效率和准确性。

承前所述,在一些实施例,多个移位器的数量与多个浮点数的数量相同。如图3所示,基于排序结果从多个具有不同预设位数的移位器中为每个浮点数分别分配对应的移位器,包括:

步骤 S302, 确定排序结果中每个浮点数的排序序号。

步骤 S304, 确定与各排序序号分别对应的预设位数, 并将多个移位器分别按照所具有的预设位数, 分配至与相应预设位数对应的排序序号所指定的浮点数。

具体地, 计算机设备根据各个浮点数的排序结果, 确定各个浮点数各自所属的排序序号。排序序号指示浮点数位于排序结果中的第几位。每个排序序号预先设置有相对应的预设位数, 例如预先设置有第一位对应于 x 位的移位器, 第二位对应于 y 位的移位器...等。对于一个浮点数, 计算机设备根据该浮点数所属的排序序号所对应的预设位数, 进而确定具有该预设位数的移位器, 由此, 确定该浮点数与该移位器之间的关联关系, 计算机设备即可将该移位器分配至处理该浮点数。

示例性地, 对于排序结果中第一位的浮点数, 其无需进行移位; 对于第二位的浮点数, 计算机设备为其分配一个 q 位的移位器; 对于第三位的浮点数, 计算机设备为其分配一个 $2q$ 位的移位器...依次类推, 直到除第一位的浮点数以外的其他浮点数均完成分配。

本实施例中, 通过为不同的浮点数分配不同的移位器, 显著地减少了硬件开销, 并且时序特性好, 效率高。

在为每个浮点数分配了相应的移位器后, 计算机设备利用移位器对浮点数进行移位处理。其中, 在移位处理的过程中, 移位器对浮点数的尾数值进行移位的位数可以根据其指数值与最大指数值之间的差异而确定。为此, 在一些实施例中, 如图4所示, 对于每个浮点数, 均通过为浮点数分配的移位器, 对相应浮点数的尾数值进行移位处理, 得到移位结果, 包括:

步骤 S402, 基于各浮点数的指数值分别与最大指数值的差异, 确定每个浮点数各自的尾数值所对应的移位位数, 最大指数值为多个浮点数的指数值中的最大值。

步骤 S404, 对于每个浮点数, 均通过为浮点数分配的移位器, 基于相应浮点数的尾数值所对应的移位位数, 对相应浮点数的尾数值进行移位处理, 得到移位结果。

具体地, 对于一个浮点数, 计算机设备基于该浮点数的指数值与最大指数值的差异, 确定每个浮点数各自的尾数值所对应的移位位数, 再利用分配给该浮点数的移位器, 结合所确定的移位位数进行移位处理。其中, 差异可以是指数值与最大指数值的差值, 也可以是指数值与最大指数值的比值, 也可以是指数值与最大指数值的差值的倍数等。

示例性地, 浮点数 A 的指数值 E_1 与最大指数值 $E_{\max}$ 的差值为 x , 则计算机设备确定浮点数 A 的尾数值 M_1 所对应的移位位数为 x 位。在确定移位位数后, 计算机设备利用对浮点数分配的移位器, 对浮点数的尾数值移位 x 位, 得到该浮点数的移位结果。除了最大指数值 $E_{\max}$ 对应的浮点数(该浮点数无需进行移位)以外, 计算机设备对其余各个浮点数均进行移位处理, 得到多个移位结果。

本实施例中, 通过为不同的浮点数分配不同的移位器, 显著地减少了硬件开销, 节省了移位处理所需要的移位器资源。

在移位过程中，可能会遇到所确定的移位位数比移位器的预设位数还要大的情况。在此，在一些实施例中，如图 5 所示，对于每个浮点数，均通过为浮点数分配的移位器，基于相应浮点数的尾数值所对应的移位位数，对相应的尾数值进行移位处理，得到移位结果，包括：

步骤 S502，对于每个浮点数，确定浮点数的尾数值所对应的移位位数是否位于移位范围之内，移位范围与浮点数分配的移位器的预设位数相匹配。

步骤 S504，在移位位数位于移位范围之内的情况下，则通过为浮点数分配的移位器，在相应移位范围内，对浮点数中构成尾数值的每一位尾数成员均朝着同一个移位方向移动移位位数；其中，移位方向包括左移或右移。

步骤 S506，在移位位数位于移位范围之外的情况下，则通过为浮点数分配的移位器，对浮点数中构成尾数值的每一位尾数成员均朝着同一个移位方向移动预设位数。

步骤 S508，将每个浮点数进行移位处理后得到的尾数值，作为各浮点数各自的移位结果。

具体地，对于一个浮点数，计算机设备通过执行上述步骤 S502，获取该浮点数的尾数值所对应的移位位数。同时，计算机设备确定为该浮点数所分配的移位器，并获取该移位器的预设位数。计算机设备将移位位数与预设位数进行比较，判断该移位位数是否位于该预设位数所对应的移位范围之内。在移位位数位于移位范围之内的情况下，则计算机设备利用该移位器对尾数值朝着移位方向移动移位位数；在移位位数位于移位范围之外的情况下，则计算机设备利用该移位器对尾数值朝着移位方向移动预设位数。计算机设备对该浮点数的尾数值进行移位处理后，所得到的尾数值作为该浮点数的移位结果。除了最大指数值对应的浮点数（无需进行移位）以外，计算机设备对其余各个浮点数均进行如上述的移位处理，得到多个移位结果。

示例性地，对于浮点数 A，计算机设备确定其尾数值 M_1 所对应的移位位数为 22 位，并且为浮点数 A 分配的移位器 S_A 的预设位数为 50 位，即移位范围为 26 位。则计算机设备判断浮点数 A 的移位位数在移位器 S_A 的移位范围之内，因此计算机设备利用该移位器 S_A ，对浮点数 A 的尾数值的每一位尾数成员均朝着同一个移位方向移动 22 位，例如，均朝右移 22 位。

又如，对于浮点数 B，计算机设备确定其尾数值 M_2 所对应的移位位数为 56 位，并且为浮点数 B 分配的移位器 S_B 的预设位数为 76 位，即移位范围为 52 位。则计算机设备判断浮点数 B 的移位位数在移位器 S_B 的移位范围之外，换言之，浮点数 B 的移位位数已经超出了移位器 S_B 能够移位的最大移位范围。因此计算机设备利用该移位器 S_B ，对浮点数 B 的尾数值的每一位尾数成员均朝着同一个移位方向移动 52 位。

本实施例中，通过分别对浮点数分配不同预设位数的移位器进行移位处理，无需使用较大位宽的移位器来保障中间精度无损，节省了移位器的硬件资源，减少了移位器的面积开销。

承前所述，在确定多个浮点数的移位结果后，计算机设备进而对浮点数进行压缩处理。为了减少所使用的压缩器的面积、并且减少时序路径，在一些实施例中，如图 6 所示，基于各移位结果确定与目标任务对应的浮点数处理结果，包括：

步骤 S602，基于不同预设位数的移位器所处的第一预设范围，对第一预设范围进行

划分得到多个域段，并确定与多个域段分别对应的压缩器，其中，不同的压缩器具有不同的预设压缩比。

压缩比指的是压缩器的输入的数量与输出的数量的比值。例如，预设压缩比为 $(n:2)$ 的压缩器，其输入的数量为 n 个，输出的数量为2个；又如，预设压缩比为 $(n:3)$ 的压缩器，其输入的数量为 n 个，输出的数量为3个。

在一些实施例中，多个压缩器所具有的不同预设压缩比的数值均处于第二预设范围内，且各个压缩器的预设压缩比的数值在第二预设范围内呈均匀分布。第二预设范围是基于浮点数的数量得到的。具体地，预先设置的压缩器的压缩比在 $(n:2)$ 为范围最大值所确定的第二预设范围内，该第二预设范围内例如还包括 $(n-1):2$ 、 $(n-2):2$ 等。其中， n 为浮点数的数量。

具体地，计算机设备对于各个具有不同预设位数的移位器所处的第一预设范围进行划分，得到多个域段。例如，按照各个移位器之间的最大预设位数，确定从最高位至该最大预设位数之间的范围，并将该范围等分为多个域段；其中，各个域段的范围相等。

示例性地，对于有4个浮点数的压缩过程，对于最高位的域段，计算机设备确定该域段对应的压缩器的压缩比为4:2；对于次高位的域段，计算机设备确定该域段对应的压缩器的压缩比为3:2；对于最低位和次低位的域段，由于无需进行压缩，因此无需分配压缩器。由于最低位和次低位的域段内的尾数值无需进行压缩，因此压缩器的数量为浮点数的数量减2。

步骤S604，对于每个域段，分别确定处于相应域段内的多个域内移位结果，其中，单个域内移位结果为单个浮点数所对应的移位结果中的域内部分。

具体地，对于一个域段，计算机设备确定处于该域段内、所有的浮点数的移位结果在该域段内的部分，该部分称之为域内部分。例如，最高位的域段例如为最高的 w 位，则在该域段内，计算机设备获取各个浮点数的移位结果的最高 w 位的值，所获取的全部浮点数的域内移位结果即为该域段内的多个域内移位结果。

步骤S606，通过各压缩器，分别将处于相应压缩器所对应的域段内的多个域内移位结果进行分段压缩处理，得到多个分段压缩结果。

具体地，计算机设备利用为各个域段所分配的压缩器，对划分的各个域段内的多个域内移位结果分别进行分段压缩处理，得到多个分段压缩结果。

示例性地，对于有 n 个浮点数的压缩过程，对于最高位的域段，计算机设备利用压缩比为 $(n:2)$ 的压缩器进行分段压缩处理；对于次高位的域段，计算机设备利用压缩比为 $(n-1):2$ 的压缩器进行分段压缩处理...依次类推；而对于最低位和次低位的域段，则无需进行压缩处理。

步骤S608，基于多个分段压缩结果，确定与目标任务对应的浮点数处理结果。

具体地，计算机设备基于各个域段对应的分段压缩结果，对各个分段压缩结果进行进一步处理，从而确定与目标任务对应的浮点数处理结果。例如，计算机设备对于分段压缩处理后的多个分段压缩结果，按照位数的高低顺序将各个分段压缩结果拼接形成完整的压缩结果，再将该压缩结果输入至加法器中进行处理，最终得到与目标任务对应的浮点数处理结果。

本实施例中，通过划分多个域段并进行分段压缩，低位的域段压缩器的输入少（甚至

无需进行压缩),大大减小了压缩器的面积,并且减小了时序路径。

如图7所示,在一些实施例中,通过各压缩器,分别将处于相应压缩器所对应的域段内的多个域内移位结果进行分段压缩处理,得到多个分段压缩结果,包括:

步骤 S702,对于每个压缩器,均将各自对应的域段内的多个域内移位结果作为相应压缩器的输入。

步骤 S704,每个压缩器分别按照各自所对应的预设压缩比,对各自的输入进行分段压缩处理,得到本位结果和进位结果,本位结果和进位结果构成相应分段器所对应的分段压缩结果。

具体地,对于一个压缩器,计算机设备将该压缩器对应的域段内的多个域内移位结果,作为该压缩器的输入,然后按照该压缩器的预设压缩比,对该域段内的多个域内移位结果进行分段压缩处理,得到本位结果和进位结果。其中,本位结果为各个域内移位结果经压缩后得到的和的值,进位结果为该和的值的进位值。

例如,如图8所示,以4个浮点数(浮点数A、浮点数B、浮点数C、以及浮点数D)的压缩过程为例,对于最高w位的域段,计算机设备利用4:2的压缩器,对该域段内的4个域内移位结果进行分段压缩处理,得到本位结果和进位结果;当本位结果中存在进位时,则相应的进位结果为1,反之为0;对于次高w位的域段,计算机设备利用3:2的压缩器对该域段内的3个域内移位结果进行分段压缩处理。而对于最低位和次低位的域段,则无需进行压缩处理。

本实施例中,通过按照域段划分来分别设置不同的压缩器进行压缩,大大减小了压缩器的面积,减少了硬件资源的开销。

承前所述,计算机设备在获得各个域段对应的分段压缩结果后,会对各个分段压缩结果进行进一步处理,从而确定最终的浮点数处理结果。现有技术中,在压缩得到 carry (对应于本申请实施例中的本位结果)和 sum (对应于本申请实施例中的进位结果)后,计算机设备利用一个 CPA (Carry Propagate Adder, 进位传导加法器)对 carry 和 sum 进行进位传递的加法,得到浮点数处理结果。

然而,现有技术中所采用的方式需要对全位宽的两个输入(carry 和 sum)进行进位传递的加法,至少需要一个128bit的加法器,所占用的硬件资源较多,且时序较差。为此,计算机设备分别对不同的域段分配不同的加法器进行加法处理;由于是分段进行加法处理,还需要考虑各个域段的进位情况。因此,为了减少占用的硬件资源,在一些实施例中,如图9所示,基于多个分段压缩结果,确定与目标任务对应的浮点数处理结果,包括:

步骤 S902,对于划分得到的多个域段中未经过压缩处理、且仅对应有单个域内移位结果的第一域段,将单个域内移位结果作为第一域段的选择结果。

步骤 S904,对于划分得到的多个域段中未经过压缩处理、且对应有多于一个域内移位结果的第二域段,基于处于第二域段内的多于一个的域内移位结果,生成第二域段的真值结果和拟值结果。

步骤 S906,对于划分得到的多个域段中经压缩处理的第三域段,基于各第三域段分别对应的分段压缩结果,生成相应第三域段的真值结果和拟值结果。

步骤 S908,按照域段的位段高低,从最低位的域段开始依次确定各域段所对应的选择结果,直至得到最高位的域段的选择结果,其中,域段中除第一域段外的其他域段的选

择结果为相应域段的真值结果和拟值结果中的一种。

步骤 S910, 基于各个域段的选择结果, 确定与目标任务对应的浮点数处理结果。

对于各个域段, 计算机设备首先分别对该域段内的分段压缩结果进行加法处理, 得到真值结果和拟值结果。其中, 真值结果为对域内移位结果进行加法处理得到的实际的求和结果, 拟值结果为对域内移位结果进行加法处理时、假设有进位的情况下进行计算得到的模拟的求和结果。然后, 计算机设备将各个域段内的真值结果和拟值结果, 输入至选择器中进行选择, 最终由选择器确定输出哪一结果。

具体地, 计算机设备在划分的多个域段中, 最低位的域段(称之为第一域段)无需经过压缩处理, 且该域段内只有一个域内移位结果, 则计算机设备无需对该域段进行加法处理, 也无需进行选择, 因为无需分配加法器和选择器, 即可直接将该域内移位结果作为该第一域段的选择结果。

对于次低位的域段(称之为第二域段)也无需经过压缩处理。而该第二域段内对应有多于一个域内移位结果, 需要进行加法处理。则计算机设备对第二域段内的全部域内移位结果进行加法处理, 计算得到第二域段的真值结果和拟值结果。

除了最低位的域段和次低位的域段以外, 其余各个域段(称之为第三域段)中的域内移位结果均经过压缩处理, 且第三域段内对应有多于一个域内移位结果。因此, 对于每一个第三域段, 计算机设备对第三域段内的分段压缩结果进行加法处理, 生成第三域段的真值结果和拟值结果。

在确定了除第一域段以外的各个域段的真值结果和拟值结果后, 计算机设备将各个域段的真值结果和拟值结果, 分别输入至一个选择器中, 由选择器确定输出的结果为真值结果或拟值结果。为了提高效率, 计算机设备按照域段的位段高低, 从最低位的域段开始, 依次确定最低位域段、次低位域段...最高位域段的选择结果。其中, 最低位域段所对应的选择结果也就是该域段内的原始的域内移位结果。由于最低位域段未经过压缩和加法处理, 不存在进位的情况, 因此次低位域段的选择结果为真值结果。对于各个第三域段, 较高位域段的选择结果需考虑相邻域段中较低位域段是否存在进位的情况; 当较低位域段存在进位时, 则选择器输出较高位域段的选择结果为拟值结果; 反之, 则选择器输出较高位域段的选择结果为真值结果。

示例性地, 如图 10 所示, 对于最高位域段, 计算机设备通过次高位域段的选择器输入的进位情况, 该最高位域段的选择器输出真值结果或拟值结果。同理, 对于次高位域段, 计算机设备也作如上处理。对于次低位域段, 由于最低位域段不存在进位, 因此次低位域段的选择器输出真值结果。对于最低位域段, 由于无需进行压缩和加法处理, 因此直接为原始的域内移位结果。

由此, 计算机设备通过选择器输出了全部域段对应的选择结果。

本实施例中, 通过采用分段相加的策略, 分别对不同的域段分配不同的加法器进行分段加法处理, 再分别利用多个选择器根据各个域段的进位情况进行进位传递, 相较于现有技术中使用全位宽进行进位传递的加法, 减少了需要占用的硬件资源, 并且有效降低了关键路径的长度, 时序特性好。

之后, 计算机设备将各个域段下的选择结果进行拼接, 从而得到完整的第一预设范围内的浮点数处理结果。为此, 在一些实施例中, 基于各个域段的选择结果, 确定与目标任

务对应的浮点数处理结果，包括：将每个域段的选择结果，按照各域段的位段高低依次进行拼接，得到与目标任务对应的浮点数处理结果。具体地，计算机设备按照各个域段的位段高低，由高至低（或由低至高）依次将相邻两个域段下的选择结果一一拼接起来，由此得到整个第一预设范围下的浮点数处理结果。

本实施例中，由于采取了分段压缩和分段相加的策略，因此通过将得到的各个域段的结果再进行拼接，从而得到完整的浮点数处理结果，这种方式无需使用全位宽的压缩器和加法器，降低了关键路径的长度，时序特性好。

浮点数的有效精度高，更适合于科学计算与工程计算。在科学计数法中，假如不对浮点数的表示作出明确规定，那么一个浮点数在计算机中的编码表示即会不唯一，则不利于计算机进行识别和处理。例如十进制数可以表示成 1.11×10^0 、 0.111×10^1 、 0.0111×10^2 等多种表示形式。而由于规格化的浮点数具有唯一的表示形式，因此，在浮点运算中需要对浮点数进行规格化。

为了保证所得到的浮点数处理结果符合规格化的浮点数标准，在得到浮点数处理结果以后，在一些实施例中，计算机设备还对该浮点数处理结果进行规格化处理，以使浮点数处理结果符合预设的浮点数标准。通过对浮点数处理结果进行规格化处理，以保证所得到的浮点数处理结果符合规格化的浮点数标准，使得计算机处理时无需再针对各个浮点数处理结果进行识别和转换，处理效率更高，避免了浮点数的编码表示不唯一导致的计算不准确的问题。

其中，规格化处理又称格式化输出，是指把一个浮点数按指定的格式进行转换。规格化处理后的浮点数处理结果的尾数 M 的绝对值应满足 $1/r \leq |M| < 1$ ，其中 r 为基数，通常 r 为 2 或 8 或 16。

规格化处理是指通过调整一个非规格化的浮点数的尾数值和指数的大小，使非零的浮点数在尾数值的最高位上保证为有效值。在一些实施例中，如图 11 所示，对浮点数处理结果进行规格化处理，包括：

步骤 S1102，确定浮点数处理结果中的第一符号标识和第二符号标识。

步骤 S1104，在第一符号标识与第二符号标识相同的情况下，则对浮点数处理结果中的尾数值按照第一移位方向进行移位处理。

步骤 S1106，在第一符号标识与第二符号标识不同的情况下，则对浮点数处理结果中的尾数值按照第二移位方向进行移位处理；其中，第二移位方向与第一移位方向相反。

浮点数的规格化处理有左规和右规两种方式，其中，左规指的是当浮点数运算的结果为非规格化时要进行规格化处理，将尾数值左移一位、阶码减 1（基数 $r=2$ 时）；左规可能要进行多次。右规指的是当浮点数运算的结果中，尾数值出现溢出时，将尾数值右移一位、阶码加 1（基数 $r=2$ 时）。右规只需进行一次。

具体地，计算机设备获取浮点数处理结果中的第一符号标识和第二符号标识。在第一符号标识与第二符号标识相同（也就是第一符号标识和第二符号标识构成 00 或 11）的情况下，则对浮点数处理结果中的尾数值按照第一移位方向进行移位处理。在第一符号标识与第二符号标识不同（也就是第一符号标识和第二符号标识构成 01 或 10）的情况下，则对浮点数处理结果中的尾数值按照第二移位方向进行移位处理；其中，第二移位方向与第一移位方向相反。以 IEEE 754 标准下的浮点数规格化方式为例，则第一移位方向为左移，

第二移位方向为右移。

示例性地，当计算机设备判断两个符号标识相同，说明没有溢出。但是浮点数处理结果的最高数值位与符号标识相同，则此时需要进行左规处理，即将尾数值进行左移，直到最高数值位与符号标识的数值不同为止。示例性地，对于如下两种情况的浮点数处理结果： $111\times\times\times$ 和 $000\times\times\times$ 。 $111\times\times\times$ 左移一位的结果为 $11\times\times\times 0$ ； $000\times\times\times$ 左移一位的结果为 $00\times\times\times 0$ ，最后将指数值减去移动的位数。

当计算机设备判断两个符号标识不相同，说明运算结果溢出，此时需要进行右规处理，即把浮点数处理结果的尾数进行右移，直至没有溢出时停止；再将指数值增加移动的位数。示例性地，对于如下两种情况的浮点数处理结果： $01\times\times\times\times$ 和 $10\times\times\times\times$ 。 $01\times\times\times\times$ 右移一位的结果为 $001\times\times\times$ ； $10\times\times\times\times$ 右移一位的结果为 $110\times\times\times$ ，最后将指数值增加 1。

本实施例中，通过对浮点数处理结果进行规格化处理，充分利用了尾数值的有效位数，提高了浮点数运算的精度。

在一些实施例中，在对浮点数进行规格化处理后，可能会在尾数部分的低位上增加一些值，需要对这些增加的值进行舍入处理，例如将 1.2349999 向下舍入到 1.23，或者将 1.2350001 向上舍入到 1.24 等。IEEE 754 标准中规定了如下几种舍入的方式：就近舍入到偶数、向上舍入、向下舍入、以及朝 0 的方向舍入。当然并不局限于此，不同的浮点数标准可能制定有不同的舍入方式，在实际应用中可以根据需求选择适当的舍入方式进行舍入。

本申请还提供一种应用场景，该应用场景应用上述的多输入浮点数处理方法。在一些实施例中，该多输入浮点数处理方法在该应用场景的应用例如为处理目标任务，其中，目标任务为神经网络处理任务中的其中一个子任务，神经网络处理任务至少包括卷积处理任务和相似度处理任务中的一种。在该应用场景下，上述的多输入浮点数处理方法还包括：基于浮点数处理结果执行神经网络处理任务中后续的各子任务，得到神经网络处理结果。

具体而言，在神经网络训练和应用的过程中，例如需要对于图像数据、音频数据、以及文本数据等进行计算处理；而神经网络所处理的图像数据、音频数据、以及文本数据等，在计算机中通常由浮点数进行表示，例如，一张图像中每个像素由 0~255 之间的 32 位单精度浮点数进行表示，其中 255 代表白色，0 代表黑色，又如，读取的音频数据为采样范围内的浮点数。

在神经网络处理任务中，有多个子任务涉及到对浮点数的处理。例如，对所输入的图像数据进行卷积或反卷积操作等。以卷积操作为例，卷积是两个数据在某范围内相乘后求和的结果。在卷积过程中，计算机设备可以利用处理器执行上述的多输入浮点数处理方法，实现对多个浮点数的求和，得到浮点数处理结果；然后再基于浮点数处理结果完成后续的卷积操作；由此，在神经网络处理任务中，计算机设备可以根据浮点数处理结果执行神经网络处理任务中后续的各子任务，得到神经网络处理结果，例如输出处理后的图像数据。

又如，上述多输入浮点数处理方法还可以用于神经网络处理任务中的相似度处理任务。以图像数据为例，在相似度处理任务中，需要将处理得到的图像数据与预先设置的标准图像数据进行比较，从而计算二者之间的相似度。在计算相似度的过程中，计算机设备可以利用上述多输入浮点数处理方法，对图像数据对应的浮点数进行求差计算，从而得到

浮点数处理结果。根据所得到的浮点数处理结果，计算机设备可以据此继续执行神经网络处理任务中后续的各子任务，例如根据浮点数处理结果所指示的差值，对图像数据进行分类等，最终输出神经网络处理结果（例如为图像的分类结果）等。

当然并不局限于此，本领域人员清楚的是，在不背离本申请所揭示的发明构思和思想的情况下，任何适用于多个浮点数的计算处理任务均可以作为上述目标任务，例如云计算场景下的数据计算任务、或者智能传感器（例如边缘传感器）所执行数据处理过程中的计算任务等。

本实施例中，通过将上述多输入浮点数处理方法应用在神经网络处理等任务中，能够实现神经网络的高精度浮点数处理，提高了神经网络的计算性能。

为了尽可能地阐述清楚本申请的发明思想，此处以一个具体执行多个浮点数的加法计算的示例来进行说明，并详细说明与传统方式的不同与优越之处。

在一个具体地示例中，传统的浮点数处理方式的流程如图 12 所示，计算机设备首先获取 n 个浮点数的指数（exponent），并通过比较器比较得到最大指数值；同时，计算机设备获取 n 个浮点数的尾数值（mantissa）。然后，对于一个浮点数，计算机设备将该浮点数的指数与最大指数值之间的差值，以及该浮点数的尾数值，输入至一个固定大小的移位器中，以利用该移位器对该浮点数进行移位处理。对于其他浮点数，计算机设备均作同样处理。由于每个浮点数的尾数值有大有小，为了保证无中间精度损失，所利用的移位器的范围需要为最大值 $m*w$ 位， m 为浮点数的数量减 1。

在移位过程中，要达到无中间精度损失的目的，移位后的数据位宽会非常宽。以 5 个 32 位的单精度浮点数为例，移位后的数据位宽需要达到 128bit 才能保证中间精度无损（具体 128bit 构成如图 13 所示，其中每个浮点数扩展后的尾数为 24bit，预留 2bit 为进位传递位）。过大的移位范围使得硬件开销巨大，并且时序很差，计算机中的处理器难以实现高主频。

同样，移位完成后，计算机设备将所得到的移位结果输入至一个 $n:2$ 的压缩器进行压缩（这里 n 为多输入浮点加法器的输入个数），得到两个输出：carry 和 sum，然后对 carry 和 sum 进行进位传递的加法，最终的加法结果送入 normalize 单元完成浮点数的标准化操作，最后将标准化后的数据进行舍入操作，最终得到浮点数的加法结果。这其中压缩器对全位宽使用了 $n:2$ 的压缩，以 5 个 32 位的单精度浮点数为例，压缩器使用了 128bit 的 5:2 压缩器，压缩器的面积开销巨大。而在后续的加法阶段，也是对全位宽的 2 输入（carry 和 sum）进行进位传递的加法，同样以 5 个 32 位的单精度浮点数为例，需要一个 128bit 的加法器，时序较差。

而相较于传统方式而言，如图 14 所示，本申请实施例中提供的多输入浮点数处理方法，整个处理流程主要包括：计算机设备首先获取 n 个浮点数的指数（exponent），并通过比较器比较得到最大指数值；同时，计算机设备获取 n 个浮点数的尾数值（mantissa）。在输入至移位器中进行移位处理之前，计算机设备根据各个浮点数的指数值大小，对各个浮点数进行排序处理。即，对于指数最大的浮点数，无需进行移位操作；对于排序第二位的浮点数，只需移位 w 位；对于排序第三位的浮点数，只需移位 $2*w$ 位……依此类推，对于指数最小（排序最后一位）的浮点数，其移位的范围为 $m*w$ 。以 5 个 32 位的单精度浮点数为例， $m=4$ ， $w=26$ 。按照排序结果，计算机设备动态分配各个浮点数的移位器，而无

需对于每个浮点数均使用同样固定大小的移位器。相较于现有技术而言，大大减少了移位器的面积开销。

然后，计算机设备将各个浮点数的尾数值输入至相对应的移位器中进行移位处理。所得到的移位结果再分别输入至相应的压缩器中。具体而言，计算机设备分别对每个域段设置不同压缩比的压缩器。即，对于最高的 w bit，使用一个 n:2 的压缩器；对于次高的 w bit，使用一个(n-1):2 的压缩器...依此类推，对于最低的 w bit 和次低的 w bit，无需压缩器进行压缩。因此，也无需分配压缩器。相较于现有技术而言，大大减少了压缩器的面积开销。

然后，对于各个域段，计算机设备再分别利用一个 CPA 进行加法操作。对于最低的 w bit，无需进行加法操作，直接获取该 w 位的移位结果；对于次低的 w bit，由于域段内存在两个浮点数的移位结果（对应于前述实施例中的域内移位结果），则计算机设备利用一个 CPA 进行加法操作。由于该 w 位（次低位）的后 w 位（最低位）没有进位，因此，该 w 位输出实际的加和结果。而对于其他域段，根据该域段的后一个域段（相邻的低位域段）是否有进位，由 MUX（选择器）选择是输出实际的加和结果（对应于前述实施例中的真值结果），或包含了进位的加和结果（即前述实施例中的拟值结果），由此得到多个选择结果。

最后，计算机设备对各个选择结果进行拼接，得到最终的加法结果。再进行后处理，对加法结果进行规格化处理，使得其符合 IEEE 754 所规定的格式，再进行舍入操作，得到最终结果。

相较于传统方式而言，一方面引入有效移位的思想，对排序后的尾数进行有效移位，排序靠前的尾数移位范围小（甚至无需进行移位），大大减小了移位器的面积开销。另一方面，基于所引入的有效移位的思想，对全位宽划分域段并分别进行分段压缩，低位的域段压缩器的输入少（甚至无需进行压缩），大大减小了压缩器的面积，并且减小了时序路径。而在最终的加法阶段，采用分段相加的策略，有效降低了关键路径的长度。由此，能够以最小硬件面积实现无损多输入浮点数处理，可大幅度减小整个处理器的面积，有效提升处理器的内核频率。

以 5 个 32 位的单精度浮点数为例，以 Global Foundries（格罗方德，简称 GF）12nm 制程技术为面积的估算标准，本发明需要的移位器资源与现有技术的对比如下表 1 所示：

现有技术中所使用的移位器资源	现有技术中所使用的移位器面积 (um ²)	本发明中所使用的移位器资源	本发明中所使用的移位器面积 (um ²)
128bit 移位器 5 个	800	50bit 移位器 1 个	62.5
/	/	76bit 移位器 1 个	95
/	/	102bit 移位器 1 个	127.5
/	/	128bit 移位器 1 个	160

合计	800	/	445
----	-----	---	-----

表 1

可以看到，本发明中所使用的移位器资源仅为现有技术的 55.6%，极大地节省了移位器的面积开销。

而在压缩阶段，本发明中所使用的压缩器资源与现有技术中所使用的压缩器资源的对比如下表 2 所示：

现有技术中所使用的压缩器资源	现有技术中所使用的压缩器面积(um ²)	本发明中所使用的压缩器资源	本发明中所使用的压缩器面积(um ²)
128bit 的 5:2 压缩器 1 个	192	24bit 的 5:2 压缩器 1 个	36
/	//	26bit 的 4:2 压缩器 1 个	26
	/	26bit 的 3:2 压缩器 1 个	13
合计	192	/	75

表 2

可以看到，本发明中所使用的压缩器资源为现有技术的 39%，极大地节省了压缩器的面积开销。

此外，在时序方面，现有技术中从移位处理到加法器输出的关键路径为：128bit shift（移位器）-> 5:2 compressor（压缩器）-> 128bit adder（加法器）；而本发明中从移位处理到加法器输出的关键路径为：128bit shift（移位器）-> 5:2 compressor（压缩器）-> 24bit adder（加法器），有效降低了关键路径的长度，可以实现较高的处理器主频。

应该理解的是，虽然如上所述的各实施例所涉及的流程图中的各个步骤按照箭头的指示依次显示，但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明，这些步骤的执行并没有严格的顺序限制，这些步骤可以以其它的顺序执行。而且，如上所述的各实施例所涉及的流程图中的至少一部分步骤可以包括多个步骤或者多个阶段，这些步骤或者阶段并不必然是在同一时刻执行完成，而是可以在不同的时刻执行，这些步骤或者阶段的执行顺序也不必然是依次进行，而是可以与其它步骤或者其它步骤中的步骤或者阶段的至少一部分轮流或者交替地执行。

基于同样的发明构思，本申请实施例还提供了一种用于实现上述所涉及的多输入浮点数处理方法的多输入浮点数处理装置。该装置所提供的解决问题的实现方案与上述方法中所记载的实现方案相似，故下面所提供的一个或多个多输入浮点数处理装置实施例中的具体限定可以参见上文中对于多输入浮点数处理方法的限定。

在一些实施例中，如图 15 所示，提供了一种多输入浮点数处理装置 1500，该装置可以采用软件模块或硬件模块，或者是二者的结合成为计算机设备的一部分，该装置具体包括：获取模块 1501、分配模块 1502、移位模块 1503 和确定模块 1504，其中：

获取模块 1501，用于获取与目标任务对应的多个待处理的浮点数，并分别提取每个

浮点数中指数部分的指数值和尾数部分的尾数值。

分配模块 1502，用于依据各浮点数的指数值大小对多个浮点数进行排序，得到排序结果，基于所述排序结果，从多个具有不同预设位数的移位器中，为每个浮点数分别分配移位器。

移位模块 1503，用于对于每个浮点数，均通过为浮点数分配的移位器，对相应浮点数的尾数值进行移位处理，得到移位结果。

确定模块 1504，用于基于各移位结果，确定与目标任务对应的浮点数处理结果。

在一些实施例中，多个移位器所具有的不同预设位数均处于第一预设范围内，且各个移位器的预设位数在第一预设范围内呈均匀分布。

在一些实施例中，多个移位器的数量与多个浮点数的数量相同，分配模块还用于确定排序结果中每个浮点数的排序序号；确定与各排序序号分别对应的预设位数，并将多个移位器分别按照所具有的预设位数，分配至与相应预设位数对应的排序序号所指定的浮点数。

在一些实施例中，移位模块还用于基于各浮点数的指数值分别与最大指数值的差异，分别确定每个浮点数各自的尾数值所对应的移位位数，最大指数值为多个浮点数的指数值中的最大值；对于每个浮点数，均通过为浮点数分配的移位器，基于相应浮点数的尾数值所对应的移位位数，对相应的尾数值进行移位处理，得到移位结果。

在一些实施例中，移位模块还用于对于每个浮点数，确定浮点数的尾数值所对应的移位位数是否位于移位范围之内，所述移位范围与为浮点数分配的移位器的预设位数相匹配；在移位位数位于移位范围之内，则通过为浮点数分配的移位器，在相应移位范围内，对浮点数中构成尾数值的每一位尾数成员均朝着同一个移位方向移动移位位数；其中，移位方向包括左移或右移；在移位位数位于移位范围之外的情况下，则通过为浮点数分配的移位器，对浮点数中构成尾数值的每一位尾数成员均朝着同一个移位方向移动预设位数；将每个浮点数进行移位处理后得到的尾数值，作为各浮点数各自的移位结果。

在一些实施例中，确定模块还包括压缩单元，该压缩单元用于基于不同预设位数的移位器所处的第一预设范围，对第一预设范围划分得到多个域段，并确定与多个域段分别对应的压缩器，其中，不同的压缩器具有不同的预设压缩比；对于每个域段，分别确定处于相应域段内的多个域内移位结果，其中，单个域内移位结果为单个浮点数所对应的移位结果中的域内部分；通过各压缩器，分别将处于相应压缩器所对应的域段内的多个域内移位结果进行分段压缩处理，得到多个分段压缩结果；基于多个分段压缩结果，确定与目标任务对应的浮点数处理结果。

在一些实施例中，多个压缩器所具有的不同预设压缩比的数值均处于第二预设范围内，且各个压缩器的预设压缩比的数值在第二预设范围内呈均匀分布。

在一些实施例中，压缩单元还用于对于每个压缩器，均将各自对应的域段内的多个域内移位结果作为相应压缩器的输入；每个压缩器分别按照各自所对应的预设压缩比，对各自的输入进行分段压缩处理，得到本位结果和进位结果，本位结果和进位结果构成相应分段器所对应的分段压缩结果。

在一些实施例中，确定模块还包括加法单元，该加法单元用于对于划分得到的多个域段中未经过压缩处理、且仅对应有单个域内移位结果的第一域段，将单个域内移位结果作

为第一域段的选择结果；对于划分得到的多个域段中未经过压缩处理、且对应有多个域段内移位结果的第二域段，基于处于第二域段内的多个域段内移位结果，生成第二域段的真值结果和拟值结果；对于划分得到的多个域段中经压缩处理的第三域段，基于各第三域段分别对应的分段压缩结果，生成相应第三域段的真值结果和拟值结果；按照域段的位段高低，从最低位的域段开始依次确定各域段所对应的选择结果，直至得到最高位的域段的选择结果，其中，域段中除第一域段外的其他域段的选择结果为相应域段的真值结果和拟值结果中的一种；基于各个域段的选择结果，确定与目标任务对应的浮点数处理结果。

在一些实施例中，确定模块还包括拼接单元，该拼接单元用于将每个域段的选择结果，按照各域段的位段高低依次进行拼接，得到与目标任务对应的浮点数处理结果。

在一些实施例中，上述装置还包括后处理模块，该后处理模块用于对浮点数处理结果进行规格化处理，以使浮点数处理结果符合预设的浮点数标准。

在一些实施例中，后处理模块还用于确定浮点数处理结果中的第一符号标识和第二符号标识；在第一符号标识与第二符号标识相同的情况下，则对浮点数处理结果中的尾数值按照第一移位方向进行移位处理；在第一符号标识与第二符号标识不同的情况下，则对浮点数处理结果中的尾数值按照第二移位方向进行移位处理；其中，第二移位方向与第一移位方向相反。

在一些实施例中，目标任务为神经网络处理任务中的其中一个子任务，神经网络处理任务至少包括卷积处理任务或相似度处理任务中的一种；上述装置还包括任务模块，该任务模块用于基于浮点数处理结果执行神经网络处理任务中后续的各子任务，得到神经网络处理结果。

关于多输入浮点数处理装置的具体限定可以参见上文中对于多输入浮点数处理方法的限定。上述多输入浮点数处理装置中的各个模块可全部或部分通过软件、硬件及其组合来实现。上述各模块可以硬件形式内嵌于或独立于计算机设备中的处理器中，也可以以软件形式存储于计算机设备中的存储器中，以便于处理器调用执行以上各个模块对应的操作。

基于同样的发明构思与思想，本申请实施例还提供一种处理器，用于实现上述实施例中所涉及的多输入浮点数处理方法。在一些实施例中，如图16所示，处理器1600包括至少一个具有不同预设位数的移位器1601和逻辑处理单元1602。在不同的实施情景下，处理器依据应用到不同计算机设备中的需求可以呈现多种封装结构。

其中，至少一个具有不同预设位数的移位器 1601，每个移位器被分别分配用于对多个待处理的浮点数中的其中一个浮点数的尾数部分的尾数值进行移位处理；其中，每个浮点数所对应分配的移位器是根据各浮点数的指数部分的指数值大小进行排序所得到的排序结果确定的。

逻辑处理单元 1602，用于对各移位器进行移位处理所得到的多个移位结果进行逻辑处理，得到浮点数处理结果。

在一些实施例中，至少一个移位器所具有的不同预设位数均处于第一预设范围内，且各个移位器的预设位数在第一预设范围内呈均匀分布。

在一些实施例中，如图 17 所示，逻辑处理单元 1602 包括：

至少一个具有不同预设压缩比的压缩器 16021，分别与移位器相连，每个压缩器被分

配用于对经移位处理得到的多个移位结果进行分段压缩处理，得到分段压缩结果。

至少一个加法器 16022，分别与压缩器 16021 相连，用于根据分段压缩结果生成多个域段的真值结果和拟值结果。

至少一个选择器 16023，分别与加法器 16022 相连，用于根据多个域段的真值结果和拟值结果确定各个域段的选择结果；选择结果用于拼接生成浮点数处理结果。

在一些实施例中，至少一个压缩器所具有的不同预设压缩比的数值均处于第二预设范围内，且各个压缩器的预设压缩比的数值在第二预设范围内呈均匀分布。

关于处理器的具体限定可以参见上文中对于多输入浮点数处理方法的限定。上述处理器中的各个元件可全部或部分通过门电路、开关电路等其他电路元器件的组合来实现。上述各元件可集成式内嵌于计算机设备的处理器中，以便于处理器调用执行以上各个元件对应的操作。

在一些实施例中，提供了一种计算机设备，该计算机设备可以是包含上述处理器的服务器或终端。其中，服务器可以是独立的物理服务器，也可以是多个物理服务器构成的服务器集群或者分布式系统，还可以是提供云计算服务的云服务器。终端可以是智能手机、平板电脑、笔记本电脑、台式计算机、智能音箱、智能手表、车载终端、智能电视等。该计算机设备的内部结构图可以如图 18 所示。该计算机设备包括通过系统总线连接的处理器、存储器和网络接口。其中，该计算机设备的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统、计算机可读指令和数据库。该内存储器为非易失性存储介质中的操作系统和计算机可读指令的运行提供环境。该计算机设备的数据库用于存储浮点数。该计算机设备的网络接口用于与外部的终端通过网络连接通信。该计算机可读指令被处理器执行时以实现一种多输入浮点数处理方法。

本领域技术人员可以理解，图 18 中示出的结构，仅仅是与本申请方案相关的部分结构的框图，并不构成对本申请方案所应用于其上的计算机设备的限定，具体的计算机设备可以包括比图中所示更多或更少的部件，或者组合某些部件，或者具有不同的部件布置。

在一些实施例中，还提供了一种计算机设备，包括存储器和处理器，存储器中存储有计算机可读指令，该处理器执行计算机可读指令时实现上述各方法实施例中的步骤。

在一些实施例中，提供了一种计算机可读存储介质，存储有计算机程序，该计算机程序被处理器执行时实现上述各方法实施例中的步骤。

在一个实施例中，提供了一种计算机程序产品，包括计算机程序，该计算机程序被处理器执行时实现上述各方法实施例中的步骤。

本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机程序来指令相关的硬件来完成，所述的计算机程序可存储于一非易失性计算机可读存储介质中，该计算机程序在执行时，可包括如上述各方法的实施例的流程。其中，本申请所提供的各实施例中所使用的对存储器、数据库或其它介质的任何引用，均可包括非易失性和易失性存储器中的至少一种。非易失性存储器可包括只读存储器（Read-Only Memory, ROM）、磁带、软盘、闪存、光存储器、高密度嵌入式非易失性存储器、阻变存储器（ReRAM）、磁变存储器（Magnetoresistive Random Access Memory, MRAM）、铁电存储器（Ferroelectric Random Access Memory, FRAM）、相变存储器（Phase Change Memory,

PCM)、石墨烯存储器等。易失性存储器可包括随机存取存储器(Random Access Memory, RAM)或外部高速缓冲存储器等。作为说明而非局限,RAM可以是多种形式,比如静态随机存取存储器(Static Random Access Memory, SRAM)或动态随机存取存储器(Dynamic Random Access Memory, DRAM)等。本申请所提供的各实施例中所涉及的数据库可包括关系型数据库和非关系型数据库中至少一种。非关系型数据库可包括基于区块链的分布式数据库等,不限于此。本申请所提供的各实施例中所涉及的处理器可为通用处理器、中央处理器、图形处理器、数字信号处理器、可编程逻辑器、基于量子计算的数据处理逻辑器等,不限于此。

以上实施例的各技术特征可以进行任意的组合,为使描述简洁,未对上述实施例中的各个技术特征所有可能的组合都进行描述,然而,只要这些技术特征的组合不存在矛盾,都应当认为是本说明书记载的范围。

以上所述实施例仅表达了本申请的几种实施方式,其描述较为具体和详细,但并不能因此而理解为对发明专利范围的限制。应当指出的是,对于本领域的普通技术人员来说,在不脱离本申请构思的前提下,还可以做出若干变形和改进,这些都属于本申请的保护范围。因此,本申请专利的保护范围应以所附权利要求为准。

权利要求书

1、一种多输入浮点数处理方法，由计算机设备执行，所述方法包括：

获取与目标任务对应的多个待处理的浮点数，并分别提取每个浮点数中指数部分的指数值和尾数部分的尾数值；

依据各浮点数的指数值大小，对多个所述浮点数进行排序，得到排序结果，基于所述排序结果，从多个具有不同预设位数的移位器中，为每个浮点数分别分配移位器；

对于每个浮点数，均通过为所述浮点数分配的移位器，对相应浮点数的尾数值进行移位处理，得到移位结果；及

基于各所述移位结果，确定与所述目标任务对应的浮点数处理结果。

2、根据权利要求1所述的方法，其特征在于，所述多个移位器所具有的不同预设位数均处于第一预设范围内，且各个移位器的预设位数在所述第一预设范围内呈均匀分布。

3、根据权利要求1所述的方法，其特征在于，所述多个移位器的数量与所述多个浮点数的数量相同，所述基于所述排序结果，从多个具有不同预设位数的移位器中，为每个浮点数分别分配移位器，包括：

确定所述排序结果中每个浮点数的排序序号；及

确定与各所述排序序号分别对应的预设位数，并将多个移位器分别按照所具有的预设位数，分配至与相应预设位数对应的排序序号所指定的浮点数。

4、根据权利要求1所述的方法，其特征在于，所述对于每个浮点数，均通过为所述浮点数分配的移位器，对相应浮点数的尾数值进行移位处理，得到移位结果，包括：

基于各浮点数的指数值分别与最大指数值的差异，分别确定每个浮点数各自的尾数值所对应的移位位数，所述最大指数值为多个浮点数的指数值中的最大值；及

对于每个浮点数，均通过为所述浮点数分配的移位器，基于相应浮点数的尾数值所对应的移位位数，对相应浮点数的尾数值进行移位处理，得到移位结果。

5、根据权利要求4所述的方法，其特征在于，所述对于每个浮点数，均通过为所述浮点数分配的移位器，基于相应浮点数的尾数值所对应的移位位数，对相应浮点数的尾数值进行移位处理，得到移位结果，包括：

对于每个浮点数，确定所述浮点数的尾数值所对应的移位位数是否位于移位范围之内，所述移位范围与为浮点数分配的移位器的预设位数相匹配；

在所述移位位数位于所述移位范围之内，则通过与所述浮点数对应分配的移位器，在相应移位范围内，对所述浮点数中构成尾数值的每一位尾数成员，均朝着同一个移位方向移动所述移位位数；其中，所述移位方向包括左移或右移；

在所述移位位数位于所述移位范围之外的情况下，则通过与所述浮点数对应分配的移位器，对所述浮点数中构成尾数值的每一位尾数成员均朝着同一个移位方向移动所述预设位数；及

将每个浮点数进行移位处理后得到的尾数值，作为各浮点数各自的移位结果。

6、根据权利要求1所述的方法，其特征在于，所述基于各所述移位结果，确定与所述目标任务对应的浮点数处理结果，包括：

基于所述不同预设位数的移位器所处的第一预设范围，对所述第一预设范围进行划分得到多个域段，并确定与多个域段分别对应的压缩器，其中，不同的压缩器具有不同的

预设压缩比;

对于每个域段,分别确定处于相应域段内的多个域内移位结果,其中,单个域内移位结果为单个浮点数所对应的移位结果中的域内部分;

通过各所述压缩器,分别将处于相应压缩器所对应的域段内的多个域内移位结果,进行分段压缩处理,得到多个分段压缩结果;及

基于所述多个分段压缩结果,确定与所述目标任务对应的浮点数处理结果。

7、根据权利要求6所述的方法,其特征在于,所述多个压缩器所具有的不同预设压缩比的数值均处于第二预设范围内,且各个压缩器的预设压缩比的数值在所述第二预设范围内呈均匀分布。

8、根据权利要求6所述的方法,其特征在于,所述通过各所述压缩器,分别将处于相应压缩器所对应的域段内的多个域内移位结果,进行分段压缩处理,得到多个分段压缩结果,包括:

对于每个压缩器,均将各自对应的域段内的多个域内移位结果,作为相应压缩器的输入;及

每个压缩器分别按照各自所对应的预设压缩比,对各自的输入进行分段压缩处理,得到本位结果和进位结果,所述本位结果和进位结果构成相应分段器所对应的分段压缩结果。

9、根据权利要求6所述的方法,其特征在于,所述基于所述多个分段压缩结果,确定与所述目标任务对应的浮点数处理结果,包括:

对于划分得到的多个域段中未经过压缩处理、且仅对应有单个域内移位结果的第一域段,将所述单个域内移位结果作为所述第一域段的选择结果;

对于划分得到的多个域段中未经过压缩处理、且对应有多于一个域内移位结果的第二域段,基于处于所述第二域段内的多于一个的域内移位结果,生成所述第二域段的真值结果和拟值结果;

对于划分得到的多个域段中经压缩处理的第三域段,基于各第三域段分别对应的分段压缩结果,生成相应第三域段的真值结果和拟值结果;

按照域段的位段高低,从最低位的域段开始,依次确定各域段所对应的选择结果,直至得到最高位的域段的选择结果,其中,所述域段中除所述第一域段外的其他域段的选择结果为相应域段的真值结果和拟值结果中的一种;及

基于各个域段的选择结果,确定与所述目标任务对应的浮点数处理结果。

10、根据权利要求9所述的方法,其特征在于,所述基于各个域段的选择结果,确定与所述目标任务对应的浮点数处理结果,包括:

将每个域段的选择结果,按照各域段的位段高低依次进行拼接,得到与所述目标任务对应的浮点数处理结果。

11、根据权利要求1所述的方法,其特征在于,所述方法还包括:

对所述浮点数处理结果进行规格化处理,以使所述浮点数处理结果符合预设的浮点数标准。

12、根据权利要求11所述的方法,其特征在于,所述对所述浮点数处理结果进行规格化处理,包括:

确定所述浮点数处理结果中的第一符号标识和第二符号标识;

在所述第一符号标识与所述第二符号标识相同的情况下,则对所述浮点数处理结果中的尾数值按照第一移位方向进行移位处理;及

在所述第一符号标识与所述第二符号标识不同的情况下,则对所述浮点数处理结果中的尾数值按照第二移位方向进行移位处理;其中,所述第二移位方向与所述第一移位方向相反。

13、根据权利要求1至12任意一项所述的方法,其特征在于,所述目标任务为神经网络处理任务中的其中一个子任务,所述神经网络处理任务至少包括卷积处理任务或相似度处理任务中的一种,所述方法还包括:

基于所述浮点数处理结果,执行所述神经网络处理任务中后续的各子任务,得到神经网络处理结果。

14、一种多输入浮点数处理装置,其特征在于,所述装置包括:

获取模块,用于获取与目标任务对应的多个待处理的浮点数,并分别提取每个浮点数中指数部分的指数值和尾数部分的尾数值;

分配模块,用于依据各浮点数的指数值大小对多个所述浮点数进行排序,得到排序结果,基于所述排序结果,从多个具有不同预设位数的移位器中,为每个浮点数分别分配移位器;

移位模块,用于对于每个浮点数,均通过为所述浮点数分配的移位器,对相应浮点数的尾数值进行移位处理,得到移位结果;及

确定模块,用于基于各所述移位结果,确定与所述目标任务对应的浮点数处理结果。

15、一种处理器,其特征在于,包括:

至少一个具有不同预设位数的移位器,每个移位器被分别分配用于对多个待处理的浮点数中的其中一个浮点数的尾数部分的尾数值进行移位处理;其中,每个浮点数所对应分配的移位器是根据各浮点数的指数部分的指数值大小进行排序所得到的排序结果确定的;及

逻辑处理单元,用于对各所述移位器进行移位处理所得到的多个移位结果进行逻辑处理,得到浮点数处理结果。

16、根据权利要求15所述的处理器,其特征在于,所述至少一个移位器所具有的不同预设位数均处于第一预设范围内,且各个移位器的预设位数在所述第一预设范围内呈均匀分布。

17、根据权利要求15所述的处理器,其特征在于,所述逻辑处理单元包括:

至少一个具有不同预设压缩比的压缩器,分别与所述移位器相连,每个压缩器被分配用于对经移位处理得到的多个移位结果进行分段压缩处理,得到分段压缩结果;

至少一个加法器,分别与所述压缩器相连,用于根据所述分段压缩结果生成多个域段的真值结果和拟值结果;及

至少一个选择器,分别与所述加法器相连,用于根据多个域段的真值结果和拟值结果确定各个域段的选择结果;所述选择结果用于拼接生成浮点数处理结果。

18、一种计算机设备,包括存储器和处理器,所述存储器存储有计算机可读指令,其特征在于,所述处理器执行所述计算机可读指令时实现权利要求1至13中任一项所述的

方法的步骤。

19、一种计算机可读存储介质，存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现权利要求1至13中任一项所述的方法的步骤。

20、一种计算机程序产品，包括计算机程序，其特征在于，该计算机程序被处理器执行时实现权利要求1至13中任一项所述的方法的步骤。