人工智能芯片的数据复用技术应用

林广栋
（安徽芯纪元科技有限公司，安徽 230094）

摘要：阐述数据在片内和片外之间的传输过程所消耗的能量远大于数据计算所消耗的能量。而现代深度学习模型的数据量很大，无法全部放入片内存储器，必然存在数据在片内和片外之间的传输。因此，减少数据的移动成为设计人工智能芯片的重要原则。数据复用是利用数据在同一种计算过程中，多次参与计算的特点，减少数据移动的方法。探讨数据复用的方案，提出一种衡量数据复用程度的方法。
关键词：人工智能芯片，深度学习，加速核，数据复用，卷积层，全连接层。

中图分类号：TP183　文章编号：1674-2583(2023)04-0030-05
DOI：10.19339/j.issn.1674-2583.2023.04.011
文献引用格式：林广栋.人工智能芯片的数据复用技术应用[J].集成电路应用, 2023, 40(04): 30-34.

Survey on Data Reuse Methods of Artificial Intelligence Chips

LIN Guangdong
(Anhui Siliepoch Technology Company, Anhui 230094, China.)

Abstract — This paper describes that energy spent in data transfer in and out of chip is much more than energy spent in computation the data involved. The data volume of modern deep learning models is huge and cannot be fully loaded onto chip, and the data transfer in and out of chip is inevitable. Therefore, to avoid data transfer is an important design rule of artificial intelligence chips. Data reuse is a method that make use of repeatedly occurrence of data during computing to reduce transfer times. This paper introduce the concept of data reuse and its application in artificial intelligence chips, and propose a method to measure data reuse degree of hardware architecture.
Index Terms — artificial intelligence chips, deep learning, accelerating core, data reuse, convolution layer, fully conncted layer.

0 引言

随着集成电路制造工艺的迅速发展，单一芯片上可以容纳的晶体管数目大幅增加，集成电路设计进入了SoC（System On Chip）时代。越来越多的厂商选择把计算单元密集的高性能计算阵列加速器作为SoC系统的一部分集成到芯片中。这种做法把上层的控制软件放到SoC系统中的CPU上执行，而把高密集的计算任务放到专用加速器上执行，兼具上层软件的灵活性、丰富的运行时软件生态环境、专用加速器带来的超强算力等优点。

1 研究背景

早在1990年代，就有高性能计算阵列加速器以可重构计算的形态出现。早期的高性能计算阵列加速器包括Berkely大学提出的PADDI架构（1990年）、Goldstein等提出的PipeRench架构（1998年）、Kaiserslantern大学提出的KressArray架构（1996年）、Mirsky等人提出的MATRIX架构（1996年）、MIT提出的Raw架构（1997年）、HP实验室提出的CHESS架构（1999年）等等。但这些高性能计算阵列加速器的大部分用途仅限于学术研究，没有在工业界得到实用。

根据高性能计算阵列加速器基本功能处理单元的功能粒度，可以把高性能计算阵列加速器分为三类：细粒度计算阵列、粗粒度计算阵列、混合粒度计算阵列。功能处理单元的功能粒度指其基本功能的抽象层次，也可以用其操作数的位宽来代表。细粒度计算阵列中，功能处理单元的基本功能定义在比特层次，仅能执行基本的固定操作。粗粒度计算阵列中，功能处理单元包含完整的功能部件，如算术逻辑单元、乘法器等等，能够实现算法层次的功能。混合粒度计算阵列中，功能处理单元既可以实现比特层次的功能，又可以实现算法层次的功能。细粒度计算阵列的代表为CHESS架构；粗粒度计算阵列的代表为Raw架构；混合粒度计算阵列的代表为PipeRench架构。

近年来，随着以深度学习为代表的高性能计算应用对算力的需求越来越强烈，使用二维计算阵列构建的高性能计算加速器成为国内外学术界与工业界的热点，并在谷歌、华为、寒武纪等商业企业推出的人工智能芯片中得到广泛应用。目前工业界流行的高性能计算阵列加速器大多数是细粒度计算阵列，并专为深度学习推理计算而设计。

作者简介：林广栋，安徽芯纪元科技有限公司；研究方向：人工智能技术。
收稿日期：2023-01-23；修回日期：2023-03-22。

在设计针对深度学习的人工智能芯片时，功耗、性能、面积是衡量芯片优劣的重要标准。而其中功耗、性能与架构设计时需要完成的数据传输量密切相关。如图1所示，数据在计算单元之间、芯片内部与芯片外部之间传输所消耗的能量远大于计算的能量[1]。尤其是数据从DRAM搬移到芯片内部的计算单元，其消耗的能量甚至是单纯的乘累加计算的上百倍。显然，减少数据在芯片内部与芯片外部的移动，将可显著减少芯片的功耗。另一方面，芯片的带宽是固定的，若计算时需要的数据必须由片外传输进入芯片，将会受到有限的芯片带宽的限制，使芯片的算力不能全部发挥。人们常常用屋顶线模型来描述一款芯片的算力极限以及带宽极限。屋顶线模型中，当计算任务的计算密度（计算量除以数据量）比较小时，任务是带宽受限的，其最高性能受芯片的带宽限制；当计算任务的计算密度较高时，任务转换为算力受限的，其最高性能受芯片的峰值算力限制。综上所述，尽可能减少数据在芯片内部与外部之间的流动，是人工智能芯片架构设计的原则之一。

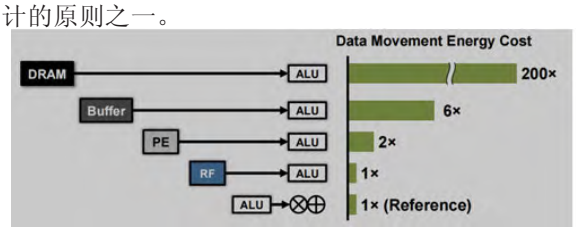

图1 消耗在计算和不同类型的数据传输上的能量比例关系

随着以深度学习为代表的人工智能算法的发展，深度学习模型的数据量不断提高，表1给出了典型深度学习模型的参数量与计算量。

而目前的用于推理计算的人工智能芯片中，芯片内部的存储器容量最多也仅为数MB量级，目前流行的深度学习模型是无法全部放入芯片内部的，必然存在数据在片内与片外的传输。

表1 典型深度学习模型参数量与计算量

深度学习模型	参数量/MB	计算量/GFLOPS
Alexnet	233	0.7
VGG-16	528	15.5
Resnet-50	98	3.9
Resnet-101	170	7.6
Resnet-152	230	11.3
Rfcn-res50	122	79
Rfcn-res101	194	117
Ssd-pascal-vggvd-512	104	91
Deeplab-res101-v2	505	346

减少数据移动的一种有效方法是数据复用。在深度学习领域，普遍存在的一个现象是一份数据会多次参与计算。因此，利用该特点，尽可能使数据只移动一次但被多次使用，就是数据复用方法。

2 数据复用方法分类

近年来用于人工智能芯片的高性能计算阵列加速器一般又可称为深度学习加速核。根据数据复用的方式，现代流行的深度学习加速核硬件架构常常分为三类：（1）输入静止（input stationary）：输入在计算单元（Processing Element，PE）上保持不动，变换权重和输出；（2）输出静止（output stationary）：输出作为部分累加和保存在PE的累加器中，变换权重和输入；（3）权重静止（weight stationary）：权重保持在PE中不动，变化输入，作为输出的部分和在PE之间传递。这三种分类方式根据计算阵列在数据流上最大利用哪一部分数据的复用来分类，也可以称为输入复用（input reuse）、输出复用（output reuse）、权重复用（weight reuse）。另外，eyeriss架构[2]的提出者还提出了“行静止”（Row Stationary）数据流，该架构可以同时实现权重、输入数据的复用。

国内，华为公司研制了用于人工智能加速核的达芬奇架构，代表性产品昇腾310的性能达到16TOPS@INT8。根据其公开的专利，达芬奇架构的计算阵列如图2所示[3]。

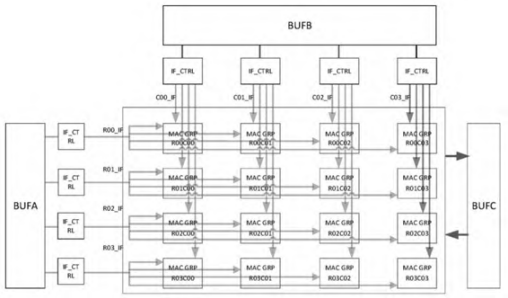

图2 华为达芬奇架构示意图

华为达芬奇架构属于“输出复用”的架构。该架构通过把乘累加（MAC）单元构建为二维的阵列，实现算力的叠加。通过在行方向上和列方向上分别广播左矩阵和右矩阵的元素，并把乘法结果累加到本地，该计算阵列可以高效实现矩阵乘法运算。该架构结合把卷积计算转换为矩阵运算的im2col操作，也可以高效实现二维卷积计算。

寒武纪公司是国内较早进行针对深度学习的计算阵列研究的公司，其研制的ShiDianNao架构提出利用二维PE阵列处理计算机视觉类深度学习模型，专注于执行卷积神经网络中的二维卷积计算。ShiDianNao的计算阵列架构如图3所示。根据公开

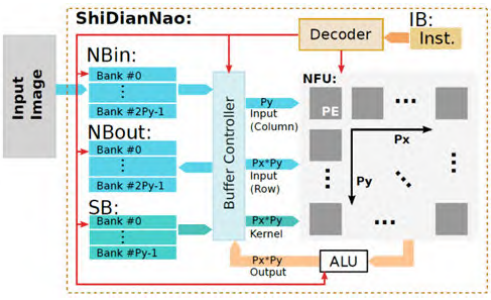

图3 ShiDianNao计算阵列架构

的论文，该架构变换向二维计算阵列输入的特征图和权重，卷积计算的部分和累加在PE内部，属于“输出复用”的架构[4]。

芯原微电子股份有限公司是一家专门提供GPU、深度学习加速核IP的供应商，其深度学习加速核IP在多家人工智能芯片厂商中得到应用。其核心的卷积模块同样采用计算阵列的方式实现。根据其公开专利，其卷积模块的核心计算阵列架构如图4所示[5]。该架构可以高效实现卷积神经网络中常见的二维卷积计算。

该架构把卷积核的权重预先存储在寄存器中不动，由于二维卷积计算对于一张输入特征图的权重是共享的，所以对输入特征图的所有滑动窗口，都可以共用该权重。计算时，整张输入特征图像水流一样流过计算阵列，并同时输出卷积计算结果。显然，该架构属于“权重复用”架构。

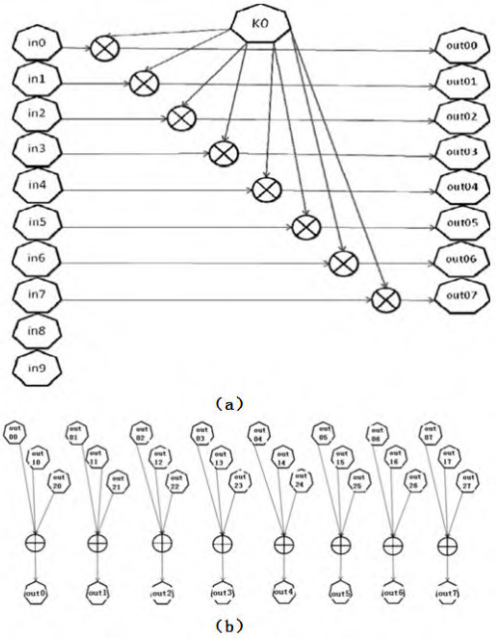

图4 芯原微电子公司专利中公开的卷积计算核心计算阵列架构

图4为芯原微电子公司专利中公开的用于卷积计算的计算阵列架构。图4（a）为计算一维卷积的计算阵列示意图；图4（b）为把多个一维卷积的结果累加为二维卷积结果的计算阵列示意图。

国外方面，谷歌公司利用脉动计算阵列实现了用于深度学习加速的芯片TPU。TPU芯片的核心计算单元同样是使用计算阵列的方式实现的，其论文中公开的计算阵列架构如图5所示[6]。

与达芬奇架构不同，脉动阵列架构中，权重存储在PE内部不动，乘法计算的部分和并不累加在PE内部，而是不断向下方的相邻PE传递。最终的乘累加结果在整个计算阵列的下方输出。TPU的计算阵列架构属于“权重复用”架构。

图6是eyeriss架构的示意图。该架构中，每行

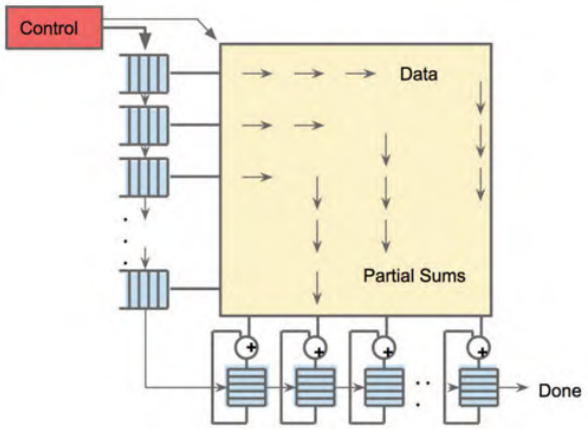

图5 TPU使用的脉动阵列架构示意图

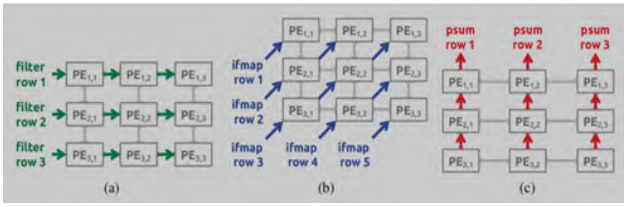

图6 eyeriss架构数据流

权重从PE阵列的一行输入，而输入数据沿PE阵列的对角线传播，两者在交汇的每个PE处相乘，每一列的部分和累加起来，正好形成一个二维卷积的计算结果。该架构属于其特有的“行静止”架构。

3 硬件上利用数据复用的方法

数据复用在硬件上有四种实现方法：空间上的广播；空间上的归约；时间上的广播；时间上的归约[7]。所谓广播就是把一份数据传输至多个位置。空间上的广播即传输一份数据到多个计算单元，从而避免了数据反复的传输，实现数据复用。所谓归约就是指把多个计算结果通过某种方式合并到一起，得到一份数据，同样可以避免数据多次传输，实现数据复用。空间上的归约一种常用实现方式是加法树，即把多个位置的数据通过加法树合并到一起，得到这些数据的和。时间上的广播指把一份数据在时间维度上传输到多个使用该数据的时机。一种简单的实现方式就是把一份会反复使用的数据存储在寄存器中不动，需要使用时，直接从寄存器中取用即可，避免该数据被反复从片外存储器导入寄存器。时间上的归约，指在时间维度上把多份数据合并到一起，产生一份数据。时间上的归约的常见实现方式是累加器：通过累加器，把在时间上多次计算得到的结果与累加器的结果累加，把累加结果再存储到累加器中，实现把时间上多次出现的数据合并到一起的效果，同样可以避免数据反复传输。

以人工智能芯片领域常见的二维计算阵列为例，常见的数据复用方式既包括空间上的广播，又包括时间上的归约。如图7所示，每列的数据广播到该列上所有的计算单元；每行的数据广播到该行

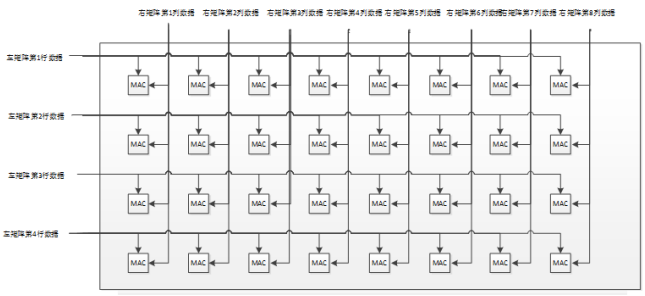

图7 常见二维计算阵列中的数据复用方式

上所有的计算单元，显然，这种数据复用方式是空间上的广播。每个计算单元内部，每个周期的计算结果与计算单元内部的累加器执行累加操作，这种数据复用方式属于时间上的归约。

另一种常用的硬件上实现数据复用的方式是加法树。如图8是一种常见的矩阵与向量相乘的硬件架构，该架构的每一行计算一个输出结果，把多个输入与每个输入对应的权重向量点乘的结果累加起来。通过设置加法树，多个乘法结果可以快速累加在一起，得到最终的结果，而不需要反复导入和导出计算单元。

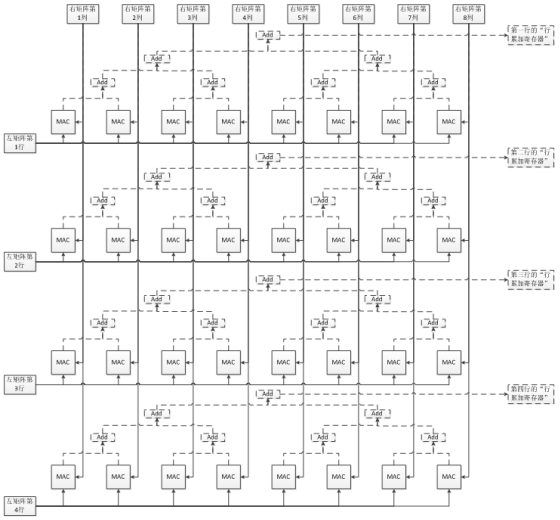

图8 利用加法树实现计算结果的累加，达到空间上归约的效果

4 二维卷积运算中的数据复用机会分析

二维卷积运算是卷积神经网络中最常见的操作，且计算量很大，约占卷积神经网络整体计算量的90%以上。在不考虑分组卷积、空洞卷积的情况下，一般二维卷积计算的计算如式（1）。

$$result_{b,co,oi,oj} = \sum_{ci=0}^{CI}\sum_{ky=0}^{KY}\sum_{kx=0}^{KX} x_{b,ci,oi*stride\_y+ky,oj*stride\_x+kx} * w_{co,ci,ky,kx} \quad (1)$$

对于空洞卷积或分组卷积，该公式将更为复杂，但基本的计算模式类似。上式中，输入通道数记为CI，输入通道循环索引记为ci；输出通道数记为CO，输出通道循环索引记为co；输出坐标高度方向上的索引记为oi，宽度方向上的索引记为oj；权重数据高度方向上的索引记为ky，宽度方向上的索引记为kx；高度方向上的步长为stride_y，宽度方向上的步长为stride_x；批处理索引记为b，批总数量记为B。

如果用伪代码的方式，上式可以用如下伪代码实现。

```
for(b=0;b<B;b++){
  for(co=0;co<CO;co++){
for(oi=0;oi<OH;oi++){
  for(oj=0;oj<OW;oj++){
    for(kx=0;kx<KX;kx++){
      for(ky=0;ky<KY;ky++){
Result[b][co][oi][oj]+=x[b][ci][oi*stride_y+ky][oj*stride_x+kx]*w[co][ci][ky][kx]
        }
      }
    }
  }
  }
  }
```

观察上式，可以发现如下几处数据复用机会。

（1）在计算输出特征图的不同位置的结果时，会使用相同的权重。即权重数据在计算不同位置的输出特征图时会反复使用，可考虑使用数据复用方法利用该特点。

（2）当步长为1时，同一个输入数据会在计算局部几个相邻的卷积结果时反复使用。例如，若卷积核大小为3×3，高度方向和宽度方向的卷积步长都是1，则同一个输入数据在计算相邻的9个输出结果时都会用到。以上2种数据复用机会都利用了二维卷积的特点，也称为卷积数据复用。

（3）在计算不同输出通道的结果时，使用的是同样的输入通道的输入数据。即同一份输入通道的输入数据，可以通过空间广播或时间广播的方式用于计算多个输出通道的结果。这种数据复用机会称为输入数据复用。

（4）对于某一个输出通道的结果而言，多个输入通道的二维卷积是独立计算，并把计算结果累加起来的。这里存在使用空间上的归约（加法树）或时间上的归约（累加器）的方法来实现数据复用的机会；这种数据复用机会称为部分和累加复用。

（5）当同时处理多批数据时，使用的是同样的权重数据。可以复用权重数据用于同时处理多批输入数据。这种数据复用机会称为权重复用。

各种不同类型的人工智能芯片卷积计算单元核心架构设计，都是分别利用以上几个特点，以硬件

数据复用的方式减少数据传输而设计的。

5 全连接层的数据复用机会分析

全连接层在卷积神经网络中使用相对较少，但在新兴的transformer模型中使用非常广泛。全连接层由矩阵乘法实现，其计算为式（2）。

$$result_{b,i} = \sum_{j=0}^{N} w_{i,j} \times x_{b,j} + bias_i \quad (2)$$

其中，数据的批次索引记为b，批的总数量记为B；每一批输出数据共M个，每一个的索引记为i；每一批输入数据共N个，每一个索引记为j。

上述计算过程用伪代码的方式可以表示如下。

```
for(b=0;b<B;b++){
  for(i=0;i<M;i++){
for(j=0;j<N;j++){
result[b][i]+=x[b][j]*w[i][j]
}
result[b][i]+=bias[i]
}
}
```

分析上述计算过程，可以看出有如下数据复用机会。（1）同一份输入数据用于多个输出数据的计算。即可以把输入数据通过数据复用的方式同时用于计算多个输出结果，减少数据搬移。这种数据复用机会称为输入数据复用。（2）每个输出结果的计算是一个累加的过程，各部分和可以独立计算，再以空间归约的方式（如加法树）或时间归约（累加器）的方式实现数据复用。这种数据复用机会称为部分和累加复用。（3）当同时处理多批数据时，使用的是同样的权重数据。因此，可以复用权重数据用于同时处理多批输入数据。这种数据复用机会称为权重复用。

6 数据复用程度衡量方式

目前，据本文作者所知，还没有一种统一的衡量硬件架构数据复用程度的标准。本文提出一种标准，可以衡量硬件架构的数据复用程度。

设硬件被用来处理某种算子C，其实现数据复用的手段是利用该算子中A元素的数据复用。假设每个A元素在理论上执行算子C的过程中平均会被使用N次，而该硬件架构中，平均每个元素A被导入硬件后会被使用M次，则该硬件架构在处理算子C时的数据复用率为式（3）。

$$\text{data\_reuse}_A = \frac{M}{N} \quad (3)$$

显然，数据复用率越接近100%，说明该硬件架构设计得越优秀。当数据复用率超过100%时，说明该硬件架构设计中存在冗余，存在删减的余地。当然，针对不同的算子，同一个硬件架构的数据复用率不同。一个硬件架构在声明数据复用率的同时，应强调该数据复用率是针对何种算子达到的。

7 结语

数据复用是人工智能芯片领域重要的概念，也是设计人工智能加速核的重要原则。本文介绍了人工智能芯片中数据复用的分类方式、常见主流产品的分类、以及常见算子的数据复用利用方法。未来，稀疏卷积、空洞卷积、可形变卷积等特殊卷积计算方式的应用将会越来越广泛，下一步的研究方向是设计针对这些新算子的高效数据复用的硬件架构。本文提出一种衡量硬件架构数据复用程度的方法，供同行参考。

参考文献

[1] Sze V,Chen Y H,Emer J,et al.Hardware for machine learning:Challenges and opportunities[C].2018 IEEE Custom Integrated Circuits Conference(CICC).IEEE,2018.

[2] Chen Y H,Emer J,Sze V.Eyeriss:A Spatial Architecture for Energy-Efficient Dataflow for Convolutional Neural Networks[C].International Symposium on Computer Architecture(ISCA),IEEE Computer Society,2016.

[3] 华为技术有限公司.矩阵乘法器[P].中国:ZL201711499179.X,2017.

[4] Du Z,Fasthuber R,Chen T,et al.ShiDianNao:Shifting vision processing closer to the sensor[J].Acm Sigarch Computer Architecture News,2015,43(3):92-104.

[5] 芯原微电子（上海）股份有限公司.基于多核的卷积神经网络加速方法及系统、存储介质及终端[P].中国:ZL201711273248.5,2017.

[6] Norman P.Jouppi,Cliff Young,Nishant Patil,David Patterson,et al.In-Datacenter Performance Analysis of a Tensor Processing Unit[C].In Proceedings of ISCA '17,Toronto,ON,Canada,2017.

[7] Hyoukjun Kwon,Prasanth Chatarasi,Michael Pellauer,Angshuman Parashar,Vivek Sarkar,Tushar Krishna.2019 Understanding Reuse,Performance,and Hardware Cost of DNN Dataflows:A Data-Centric Approach UsingMAESTRO[C].In The 52nd Annual IEEE/ACM International Symposium onMicroarchitecture(MICRO-52),Columbus,OH,USA,ACM,NY,USA,2019.