



(12) 发明专利

(10) 授权公告号 CN 113330421 B

(45) 授权公告日 2024. 01. 16

(21) 申请号 201980088183.8

(22) 申请日 2019.11.04

(65) 同一申请的已公布的文献号
申请公布号 CN 113330421 A

(43) 申请公布日 2021.08.31

(30) 优先权数据
16/184,985 2018.11.08 US

(85) PCT国际申请进入国家阶段日
2021.07.07

(86) PCT国际申请的申请数据
PCT/EP2019/080136 2019.11.04

(87) PCT国际申请的公布数据
W02020/094586 EN 2020.05.14

(73) 专利权人 莫维迪乌斯有限公司
地址 荷兰北荷兰省

(72) 发明人 F·康纳 D·伯纳德 N·汉拉恩

(74) 专利代理机构 永新专利商标代理有限公司
72002

专利代理师 刘文灿

(51) Int.Cl.
G06F 9/30 (2006.01)
G06F 17/10 (2006.01)
G06F 17/16 (2006.01)

(56) 对比文件
US 2018121377 A1, 2018.05.03
CN 107741842 A, 2018.02.27
CN 108268320 A, 2018.07.10
US 2018101531 A1, 2018.04.12
US 2018309451 A1, 2018.10.25
车文博;刘衡竹;田甜.M-DSP中高性能浮点
乘加器的设计与实现.计算机应用.2016, (第08
期), 全文.

审查员 薛梅

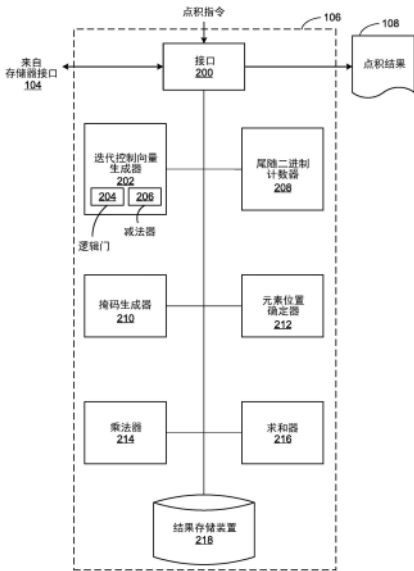
权利要求书6页 说明书17页 附图6页

(54) 发明名称

点积计算器及其操作方法

(57) 摘要

公开了使用稀疏向量执行点积计算的方法、装置、系统和制品。示例点积计算器包括：用于确定控制向量的尾随二进制计数的计数器，该控制向量对应于对第一稀疏向量的第一位图和第二稀疏向量的第二位图的第一逻辑与运算的第一结果。示例点积计算器还包括：掩码生成器，其用于基于尾随二进制计数来生成掩码向量。示例点积计算器还包括：接口，该接口基于对第一位图和掩码向量的第二逻辑与运算的第二结果来访问第一稀疏向量的第一值，并基于对第二位图和掩码向量的第三逻辑与运算的第三结果来访问第二稀疏向量的第二值。示例点积计算器还包括：用于将第一值与第二值相乘以生成乘积的乘法器。



1. 一种点积计算器,包括:

计数器,其用于确定控制向量的尾随二进制计数,所述控制向量对应于对第一向量的第一位图和第二向量的第二位图的第一逻辑与运算的第一结果;

掩码生成器,其用于基于所述尾随二进制计数来生成掩码向量;

接口,其用于:

基于对所述第一位图和所述掩码向量的第二逻辑与运算的第二结果,访问与所述第一向量相对应的第一稀疏向量的第一值;以及

基于对所述第二位图和所述掩码向量的第三逻辑与运算的第三结果,访问与所述第二向量相对应的第二稀疏向量的第二值;以及乘法器,其用于将所述第一值与所述第二值相乘以生成乘积。

2. 根据权利要求1所述的点积计算器,其中:

所述第一位图用于标识所述第一向量的第一元素分别对应于零值还是非零值,并且所述第二位图用于标识所述第二向量的第二元素分别对应于零值还是非零值;并且

所述第一稀疏向量对应于所述第一向量的非零值,并且所述第二稀疏向量对应于所述第二向量的非零值。

3. 根据权利要求1所述的点积计算器,还包括逻辑门,所述逻辑门用于:基于以所述第一位图和所述第二位图为输入的所述第一逻辑与运算,来生成所述控制向量。

4. 根据权利要求1所述的点积计算器,其中,所述掩码生成器用于:生成所述掩码向量以在最低有效位中包括一定数量的第一二进制值,所述数量对应于所述尾随二进制计数,所述掩码生成器用于生成所述掩码向量以与所述第一向量具有相同维度。

5. 根据权利要求1所述的点积计算器,还包括:元素位置确定器,其用于:

确定所述第二结果中的第一数量的二进制值,所述接口用于基于与所述第一数量的二进制值相对应的第一地址来访问所述第一值;以及

确定所述第三结果中的第二数量的二进制值,所述接口用于基于与所述第二数量的二进制值相对应的第二地址来访问所述第二值。

6. 根据权利要求1所述的点积计算器,还包括用于存储所述乘积的存储装置。

7. 根据权利要求1所述的点积计算器,还包括:

减法器,其用于通过从与所述控制向量的二进制位相对应的值中减去一来生成差向量;以及

迭代控制向量生成器,其用于:

生成与对所述控制向量和所述差向量的逻辑与运算相对应的更新后的控制向量;以及确定所述更新后的控制向量的所有元素是否对应于相同的二进制值。

8. 根据权利要求7所述的点积计算器,其中,当所述迭代控制向量生成器确定所述更新后的控制向量的所有元素对应于相同的二进制值时,所述接口用于输出所述乘积作为点积结果。

9. 根据权利要求7所述的点积计算器,其中,所述尾随二进制计数是第一尾随二进制计数,所述掩码向量是第一掩码向量,所述乘积是第一乘积,并且当所述迭代控制向量生成器确定所述更新后的控制向量的所有元素不对应于相同的二进制值时:

所述计数器用于确定所述更新后的控制向量的第二尾随二进制计数;

所述掩码生成器用于生成与所述第二尾随二进制计数相对应的第二掩码向量；
所述接口用于：

基于对所述第一位图和所述第二掩码向量的第四逻辑与运算的第四结果，访问所述第一稀疏向量的第三值；

基于对所述第二位图和所述第二掩码向量的第五逻辑与运算的第五结果，访问所述第二稀疏向量的第四值；以及

所述乘法器用于将所述第三值与所述第四值相乘以生成第二乘积，所述点积计算器还包括：

求和器，其用于将所述第一乘积与所述第二乘积求和。

10. 根据权利要求9所述的点积计算器，其中，所述差向量是第一差向量，所述更新后的控制向量是第一更新后的控制向量，并且所述元素是第一元素，并且：

所述减法器用于：通过从所述更新后的控制向量的二进制值的值转换中减去一，来生成第二差向量；以及

所述迭代控制向量生成器用于：生成与对所述更新后的控制向量和所述第二差向量的逻辑与运算相对应的第二更新后的控制向量；并且

所述接口用于：当所述第二更新后的控制向量的所有第二元素对应于相同的二进制值时，输出所述第一乘积与所述第二乘积之和作为点积结果。

11. 至少一个非暂时性计算机可读存储介质，其包括指令，所述指令在被执行时使机器至少：

确定控制向量的尾随二进制计数，所述控制向量对应于对第一向量的第一位图和第二向量的第二位图的第一逻辑与运算的第一结果；

基于所述尾随二进制计数来生成掩码向量；以及

将(A)与所述第一向量相对应的第一稀疏向量的第一值与(B)与所述第二向量相对应的第二稀疏向量的第二值相乘以生成乘积，所述第一稀疏向量的第一值基于对所述第一位图和所述掩码向量的第二逻辑与运算的第二结果，所述第二稀疏向量的第二值基于对所述第二位图和所述掩码向量的第三逻辑与运算的第三结果。

12. 根据权利要求11所述的计算机可读存储介质，其中：

所述第一位图用于分别标识所述第一向量的零值或非零值，并且所述第二位图用于分别标识所述第二向量的零值或非零值；并且

所述第一稀疏向量对应于所述第一向量的非零值，并且所述第二稀疏向量对应于所述第二向量的非零值。

13. 根据权利要求11所述的计算机可读存储介质，其中，所述指令使所述机器：基于以第一位图和所述第二位图作为输入的所述第一逻辑与运算来生成所述控制向量。

14. 根据权利要求11所述的计算机可读存储介质，其中，所述指令使所述机器：生成所述掩码向量，以在最低有效位中包括一定数量的第一二进制值，所述数量对应于所述尾随二进制计数，所述掩码向量具有与所述第一向量相同的维度。

15. 根据权利要求11所述的计算机可读存储介质，其中，所述指令使所述机器：

确定所述第二结果中的第一数量的二进制值；

基于与所述第一数量的二进制值相对应的第一地址，访问所述第一值；

确定所述第三结果中的第二数量的二进制值;以及

基于与所述第二数量的二进制值相对应的第二地址,访问所述第二值。

16. 根据权利要求11所述的计算机可读存储介质,其中,所述指令使所述机器:将所述第一值和所述第二值的所述乘积存储在本地存储器中。

17. 根据权利要求11所述的计算机可读存储介质,其中,所述指令使所述机器:

通过从与所述控制向量的二进制位相对应的值中减去一,来生成差向量;

生成与对所述控制向量和所述差向量的逻辑与运算相对应的更新后的控制向量;以及
确定所述更新后的控制向量的所有元素是否对应于相同的二进制值。

18. 根据权利要求17所述的计算机可读存储介质,其中,当所述更新后的控制向量的所有元素对应于相同的二进制值时,所述指令使所述机器:输出所述第一值和所述第二值的所述乘积作为点积结果。

19. 根据权利要求17所述的计算机可读存储介质,其中,所述尾随二进制计数是第一尾随二进制计数,所述掩码向量是第一掩码向量,所述乘积是第一乘积,并且当所述更新后的控制向量的所有元素不对应于相同的二进制值时,所述指令使所述机器:

确定所述更新后的控制向量的第二尾随二进制计数;

生成与所述第二尾随二进制计数相对应的第二掩码向量;

基于对所述第一位图和所述第二掩码向量的第四逻辑与运算的第四结果,访问所述第一稀疏向量的第三值;

基于对所述第二位图和所述第二掩码向量的第五逻辑与运算的第五结果,访问所述第二稀疏向量的第四值;以及

将所述第三值与所述第四值相乘以生成第二乘积;以及

将所述第一乘积与所述第二乘积求和。

20. 根据权利要求19所述的计算机可读存储介质,其中,所述差向量是第一差向量,所述更新后的控制向量是第一更新后的控制向量,并且所述元素是第一元素,并且,所述指令使所述机器:

通过从所述更新后的控制向量的二进制值的值转换中减去一,来生成第二差向量;

生成与所述更新后的控制向量与所述第二差向量之间的逻辑与功能相对应的第二更新后的控制向量;以及

当所述第二更新后的控制向量的所有第二元素对应于相同的二进制值时,输出所述第一乘积与所述第二乘积之和作为点积结果。

21. 一种用于确定两个向量之间的点积的方法,所述方法包括:

利用至少一个逻辑电路,确定控制向量的尾随二进制计数,所述控制向量对应于对第一向量的第一位图和第二向量的第二位图的第一逻辑与运算的第一结果;

利用所述至少一个逻辑电路,生成与所述尾随二进制计数相对应的掩码向量;

基于对所述第一位图和所述掩码向量的第二逻辑与运算的第二结果,访问与所述第一向量相对应的第一稀疏向量的第一值;以及

基于对所述第二位图和所述掩码向量的第三逻辑与运算的第三结果,访问与所述第二向量相对应的第二稀疏向量的第二值;以及

利用所述至少一个逻辑电路,将所述第一值与所述第二值相乘以生成乘积。

22. 根据权利要求21所述的方法, 其中:

所述第一位图对应于所述第一向量的第一元素是对应于零值还是非零值, 并且所述第二位图对应于所述第二向量的第二元素是对应于零值还是非零值; 并且

所述第一稀疏向量对应于所述第一向量的非零值, 并且所述第二稀疏向量对应于所述第二向量的非零值。

23. 根据权利要求21所述的方法, 还包括: 基于以所述第一位图和所述第二位图作为输入的所述第一逻辑与运算, 来生成所述控制向量。

24. 根据权利要求21所述的方法, 还包括生成所述掩码向量以在最低有效位中包括一定数量的第一二进制值, 所述数量对应于所述尾随二进制计数, 所述掩码向量具有与所述第一向量相同的维度。

25. 根据权利要求21所述的方法, 还包括:

确定所述第二结果中的第一数量的二进制值;

基于与所述第一数量的二进制值相对应的第一地址, 访问所述第一值;

确定所述第三结果中的第二数量的二进制值; 以及

基于与所述第二数量的二进制值相对应的第二地址, 访问所述第二值。

26. 根据权利要求21所述的方法, 还包括: 将所述第一值和所述第二值的所述乘积存储在本地存储器中。

27. 根据权利要求21所述的方法, 还包括:

通过从与所述控制向量的二进制位相对应的值中减去一, 来生成差向量;

生成与对所述控制向量和所述差向量的逻辑与运算相对应的更新后的控制向量; 以及

确定所述更新后的控制向量的所有元素是否对应于相同的二进制值。

28. 根据权利要求27所述的方法, 还包括: 当所述更新后的控制向量的所有元素对应于相同的二进制值时, 输出所述第一值和所述第二值的所述乘积作为点积结果。

29. 根据权利要求27所述的方法, 其中, 所述尾随二进制计数是第一尾随二进制计数, 所述掩码向量是第一掩码向量, 所述乘积是第一乘积, 并且还包括, 当所述更新后的控制向量的所有元素不对应于相同的二进制值时:

确定所述更新后的控制向量的第二尾随二进制计数;

生成与所述第二尾随二进制计数相对应的第二掩码向量;

基于对所述第一位图和所述第二掩码向量的第四逻辑与运算的第四结果, 访问所述第一稀疏向量的第三值;

基于对所述第二位图和所述第二掩码向量的第五逻辑与运算的第五结果, 访问所述第二稀疏向量的第四值;

将所述第三值与所述第四值相乘以生成第二乘积; 以及

将所述第一乘积与所述第二乘积求和。

30. 根据权利要求29所述的方法, 其中, 所述差向量是第一差向量, 所述更新后的控制向量是第一更新后的控制向量, 并且所述元素是第一元素, 并且还包括:

通过从所述更新后的控制向量的二进制值的值转换中减去一, 来生成第二差向量;

生成与对所述更新后的控制向量和所述第二差向量的逻辑与运算相对应的第二更新后的控制向量; 以及

当所述第二更新后的控制向量的所有第二元素对应于相同的二进制值时,输出所述第一乘积与所述第二乘积之和作为点积结果。

31.一种点积计算器,包括:

第一单元,其用于确定控制向量的尾随二进制计数,所述控制向量对应于对第一向量的第一位图和第二向量的第二位图的第一逻辑与运算的第一结果;

第二单元,其用于基于所述尾随二进制计数来生成掩码向量;

第三单元,其用于:

基于对所述第一位图和所述掩码向量的第二逻辑与运算的第二结果,访问与所述第一向量相对应的第一稀疏向量的第一值;以及

基于对所述第二位图和所述掩码向量的第三逻辑与运算的第三结果,访问与所述第二向量相对应的第二稀疏向量的第二值;以及第四单元,其用于将所述第一值与所述第二值相乘以生成乘积。

32.根据权利要求31所述的点积计算器,其中:

所述第一位图用于标识所述第一向量的第一元素是分别对应于零值还是非零值,并且所述第二位图用于标识所述第二向量的第二元素是分别对应于零值还是非零值;以及

所述第一稀疏向量对应于所述第一向量的非零值,并且所述第二稀疏向量对应于所述第二向量的非零值。

33.根据权利要求31所述的点积计算器,还包括第五单元,其用于基于以所述第一位图和所述第二位图为输入的所述第一逻辑与运算来生成所述控制向量。

34.根据权利要求31所述的点积计算器,其中,所述第二单元包括用于生成所述掩码向量以在最低有效位中包括一定数量的第一二进制值的单元,所述数量对应于所述尾随二进制计数,所述第二单元包括用于生成所述掩码向量以具有与所述第一向量相同的维度的单元。

35.根据权利要求31所述的点积计算器,还包括第六单元,其用于:

确定所述第二结果中的第一数量的二进制值,所述第三单元包括用于基于与所述第一数量的二进制值相对应的第一地址来访问所述第一值的单元;以及

确定所述第三结果中的第二数量的二进制值,所述第三单元包括用于基于与所述第二数量的二进制值相对应的第二地址来访问所述第二值的单元。

36.根据权利要求31所述的点积计算器,还包括用于存储所述乘积的第七单元。

37.根据权利要求31所述的点积计算器,还包括:

第八单元,其用于通过从与所述控制向量的二进制位相对应的值中减去一来生成差向量;以及

第九单元,其用于:

生成与对所述控制向量和所述差向量的逻辑与运算相对应的更新后的控制向量;以及确定所述更新后的控制向量的所有元素是否对应于相同的二进制值。

38.根据权利要求37所述的点积计算器,其中,所述第三单元包括用于在所述第九单元确定所述更新后的控制向量的所有元素对应于相同的二进制值时,输出所述乘积作为点积结果的单元。

39.根据权利要求37所述的点积计算器,其中,所述尾随二进制计数是第一尾随二进制

计数,所述掩码向量是第一掩码向量,所述乘积是第一乘积,并且当所述第九单元确定所述更新后的控制向量的所有元素不对应于相同的二进制值时:

所述第一单元包括用于确定所述更新后的控制向量的第二尾随二进制计数的单元;

所述第二单元包括用于生成与所述第二尾随二进制计数相对应的第二掩码向量的单元;

所述第三单元包括用于执行以下操作的单元:

基于对所述第一位图和所述第二掩码向量的第四逻辑与运算的第四结果,访问所述第一稀疏向量的第三值;以及

基于对所述第二位图和所述第二掩码向量的第五逻辑与运算的第五结果,访问所述第二稀疏向量的第四值;以及

第四单元,包括用于将所述第三值与所述第四值相乘以生成第二乘积的单元,所述点积计算器还包括:

求和器,其用于将所述第一乘积与所述第二乘积求和。

40. 根据权利要求39所述的点积计算器,其中,所述差向量是第一差向量,所述更新后的控制向量是第一更新后的控制向量,并且所述元素是第一元素,并且:

所述第八单元包括用于通过从所述更新后的控制向量的二进制值的值转换中减去一来生成第二差向量的单元;以及

所述第九单元包括用于生成与对所述更新后的控制向量和所述第二差向量的逻辑与运算相对应的第二更新后的控制向量的单元;以及

所述第三单元包括用于当所述第二更新后的控制向量的所有第二元素对应于相同的二进制值时,输出所述第一乘积与所述第二乘积之和作为点积结果的单元。

点积计算器及其操作方法

[0001] 相关申请

[0002] 该专利源自于要求2018年11月8日提交的序号为16/184,985的美国专利申请的权益的申请。序号为16/184,985的美国专利申请的全部内容通过引用合并于本文中。特此要求序号为16/184,985的美国专利申请的优先权。

技术领域

[0003] 本公开总体上涉及处理器,并且更具体地,涉及点积计算器及其操作方法。

背景技术

[0004] 近年来,对图像处理能力的需求已经超出了大功率专用台式机硬件的范围,并且已经成为对个人和/或其他移动设备的期望。移动设备通常包括受尺寸约束、温度管理约束和/或功率约束限制的处理能力。

附图说明

[0005] 图1是示例卷积神经网络引擎的图的块实现。

[0006] 图2是图1的示例处理器的点积计算器的示例实现方式的框图。

[0007] 图3是图1的点积计算器的示例实现方式的硬件图。

[0008] 图4是由图1、图2和/或图3的示例点积计算器执行的点积协议的示例。

[0009] 图5是表示示例机器可读指令的流程图,该示例机器可读指令可以被执行以实现图1和/或图2的示例点积计算器,以使用两个向量的位图确定两个向量的点积。

[0010] 图6是被构造为执行图5的指令以实现图1和图2的示例点积计算器的示例处理平台的框图。

[0011] 附图未按比例绘制。通常,在整个附图和随附的书面描述中将使用相同的附图标记来指代相同或相似的部分。

具体实施方式

[0012] 典型的计算系统,包括个人计算机和/或移动设备,采用高级图像处理或计算机视觉算法来使人类视觉可以执行的任务自动化。计算机视觉任务包括获取、处理、分析和/或理解数字图像。这样的任务部分地促进了从数字图像中提取尺寸数据以产生数字和/或符号信息。计算机视觉算法可以使用数字和/或符号信息来做出决策和/或以其他方式执行与三维(3-D)姿态估计、事件检测、对象识别、视频跟踪等相关联的操作。为了支持增强现实(AR)、虚拟现实(VR)、机器人技术和/或其他应用,因此重要的是快速(例如,实时或近实时)且高效地执行这样的任务。

[0013] 先进的图像处理或计算机视觉算法有时采用卷积神经网络(CNN或ConvNet)。CNN是一种深层的人工神经网络,其通常用于对图像进行分类,通过相似性(例如,照片搜索)对图像进行聚类,和/或使用卷积在图像内执行对象识别。如本文中所使用的,卷积被定义为

通过积分从两个给定的函数派生的函数,该积分表示函数之一的形状如何通过另一个函数的形状来修改。因此,CNN可通过在输入图像上传递与图像特征(例如,水平线、二维(2-D)形状等)相对应的一个或多个滤波器的输出以识别输入图像内图像特征的匹配项,来用于识别输入图像中包含的面孔、人物、路牌、动物等。

[0014] CNN获得需要在计算中存储或使用以执行一个或多个功能的向量(例如,从多维阵列分解的)。因此,CNN可以接收包括与一个或多个图像相对应的数据的多维阵列(例如,张量或向量行)。多维阵列分解为向量。这样的向量可以包括数千个元素。每个这样的元素可以包括大量的位。具有10,000个16位元素的向量对应于160,000位信息。存储此类向量需要大量存储器。但是,这样的向量可以包括值为零的大量元素。因此,一些CNN或其他处理引擎可以将这样的向量分解为稀疏向量和稀疏度图向量(sparsity map vector)(例如,位图向量)。

[0015] 如本文所定义,稀疏向量是以与密集向量的相同的次序包括向量的所有非零元素的向量,但其不包括所有零元素。如本文所定义,密集向量是包括零元素和非零元素两者的输入向量。这样,密集向量[0,0,5,0,18,0,4,0]对应于稀疏向量为[5,18,4]。如本文所定义,稀疏度图是包括一位元素的向量,其标识密集向量的相应元素是零还是非零。因此,稀疏度图可以将密集向量的非零值映射到“1”,并且可以将密集向量的零值映射到“0”。对于[0,0,5,0,18,0,4,0]的以上密集向量,稀疏度图可以是[0,0,1,0,1,0,1,0](例如,因为密集向量的第三个、第五个、第七个和八个元素不为零)。稀疏向量和稀疏度图的组合表示密集向量(例如,可以基于对应的稀疏向量和稀疏度图来生成/重构密集向量)。因此,CNN引擎可以基于对应的稀疏向量和稀疏度图来生成/确定密集向量,而不将密集向量存储在存储器中。

[0016] 在存储器中存储稀疏向量和稀疏度图而不是密集向量可以节省存储器资源和处理资源(例如,在密集向量中有足够的零的情况下)。例如,如果上述密集向量(例如,[0,0,5,0,18,0,4,0])的每个元素是16位信息,则存储密集向量所需的存储器量为128位(例如8个元素 $\times$ 16位)。但是,存储相应的稀疏向量(例如,[5,18,4])和稀疏度图(例如,0,0,1,0,1,0,1,0)所需的存储器量为64位(例如,(稀疏向量的3个元素 $\times$ 16位)+(稀疏度图的8个元素 $\times$ 1位))。因此,存储稀疏向量和稀疏度图而不是对应的密集向量减少了存储这种向量所需的存储器量。此外,利用稀疏向量和稀疏度图改善了带宽需求,因为您减少了被传递到计算引擎的数据量,从而提高了向计算引擎的传递速度。

[0017] 一些程序或应用可能需要两个输入向量(例如,密集向量)之间的点积/标量积。在一些情况下,输入向量可能已经作为与输入向量相对应的稀疏向量和稀疏度图存储在存储器中。常规地,当点积/标量积被调用时,处理器从存储器访问相应的稀疏向量和稀疏度图以重新生成相应的密集向量。然后将密集向量写入本地存储器中。因此,常规技术在计算点积之前将密集向量作为输入向量存储到本地存储器中。一旦被访问,该过程将执行常规的点积计算,其中一个密集向量的每个元素都将与另一个密集向量的相应元素相乘,并且将这些乘积求和。例如,如果第一个密集向量是[5,412,0,0,0,4,192],并且第二个密集向量是[2,0,0,432,52,4,0],则常规点积技术执行七个乘法计算并将七个乘积相加(例如,(5 $\cdot$ 2)+(412 $\cdot$ 0)+(0 $\cdot$ 0)+(0 $\cdot$ 432)+(0 $\cdot$ 52)+(4 $\cdot$ 4)+(192 $\cdot$ 0))。但是,这样的常规技术需要X数目的乘法计算,其中X对应于任一输入向量中的元素数。乘法计算很复杂,执行速度很慢,

并且需要大量资源来执行。如下面相对于背景技术所解释的,与传统的点积技术相比,本文公开的示例节省了存储器,提高了点积计算速度,并且需要更少的处理资源。

[0018] 本文公开的示例执行点积/标量积计算而不对任何为零的元素执行任何乘法计算(例如,由于任何数字和零的乘积为零)。相反,本文公开的示例基于稀疏度图执行更简单的、计算上轻的计算。以这种方式,减少了复杂的计算繁重的乘法计算的数量。结果,减少了执行点积计算所需的时间量。例如,如上所述,如果第一密集向量为[5,412,0,0,0,4,192],并且第二密集向量为[2,0,0,432,52,3,0],则与传统技术所需的七次乘法计算相反,此处公开的示例的复杂乘法计算的数量减少为两个(例如,(5•2)和(4•3))。本文公开的示例执行简单的计算(例如,逻辑与、减一、向量的二进制计数等),以标识哪些元素需要相乘,从而减少确定点积所需的处理资源,并提高进行这样的点积计算的速度。另外,因为本文公开的示例基于稀疏度图执行点积/标量积计算,所以减少了计算两个向量之间的点积所需的本地存储器量,并且增加了这种计算的速度(例如,通过消除涉及乘以零的琐碎的计算)。

[0019] 图1是示例CNN引擎100(例如,卷积神经网络引擎)的框图。CNN引擎100包括稀疏向量转换器102、示例存储器接口104、示例存储器105和示例点积计算器106。示例CNN引擎100接收密集向量或输入并输出示例点积结果108。

[0020] 图1的示例稀疏向量转换器102接收密集向量并将密集向量转换为稀疏向量(例如,仅包括接收到的向量的非零值的向量)和稀疏度图(例如,标识密集向量的哪些元素为零以及向量的哪些元素为非零的位图)。例如,如果稀疏向量转换器102接收到八乘一的密集向量[0;0;532;0;1432;4;0;0;1],则稀疏向量转换器102将八乘一维的密集向量转换成四乘一维的稀疏向量(例如,[532;1432;4;1]),其包括密集向量的非零值并消除了密集向量的零值。稀疏向量转换器102还生成八乘一维的稀疏度图(例如,[0;0;1;0;1;1;0;0;1]),该图表示接收到的密集向量的每个元素具有单个标识相应元素是零还是非零的位(例如,当向量的相应元素为“0”时为“0”,并且当向量的相应元素为非零时为“1”)。可以使用稀疏度图(例如,密集向量的位图向量)从稀疏向量(例如,仅具有密集向量的非零值的向量)重建密集向量(例如,具有零值和非零值的向量)。但是,当密集向量包含足够的零值元素时,与存储稀疏向量相比,存储稀疏向量和稀疏度图所需的存储器更少。例如,如果上述密集向量的每个元素对应于16位,则存储密集向量所需的位数为72位(例如,9个元素X 8位=72位)。但是,由于相应的稀疏向量仅包含4个元素,并且相应的稀疏度图每个元素仅需要一位,因此存储相应的稀疏向量和稀疏度图需要41位(例如,(4个元素X 8位)+(9个元素X1位)=41位)。在一些示例中,密集向量是从另一个处理器获得的。在一些示例中,密集向量是通过用户接口从用户获得的。示例稀疏向量转换器102将与密集向量相对应的所生成的稀疏向量和稀疏度图发送至示例存储器接口104。

[0021] 图1的示例存储器接口104与示例存储器105接合,以将所生成的稀疏向量和稀疏度图以及访问信息存储在示例存储器105中。例如,当存储器接口104接收到与密集向量对应的稀疏向量和稀疏度图时,存储器接口104将稀疏向量和稀疏度图存储在示例存储器105中。当点积/标量积函数被调用时,点积计算器106指示存储器接口104访问与要在点积/标量积计算中使用的稀疏向量的元素的值相对应的一个或多个稀疏度图和/或一个或多个存储器地址。存储器接口104访问是来自示例存储器105的信息,并将所请求的信息(例如,稀疏向量值)返回到示例点积计算器106。

[0022] 图1的示例存储器105存储稀疏向量和相应的稀疏度图。例如,存储器105将稀疏向量的每个元素存储在存储器105中的一个或多个地址中。以这种方式,稀疏向量的每个元素对应于一个或多个存储器地址。因此,当存储器接口104接收到指令以访问与稀疏向量内的位置相对应的稀疏向量的元素时,存储器接口104可以基于存储该元素的地址来访问稀疏向量内的所请求位置处的元素。

[0023] 图1的示例点积计算器106使用稀疏向量以及与两个稀疏向量相对应的稀疏度图来计算两个向量之间的点积/标量积。例如,代替使用需要将整个密集向量存储在本地存储器中并执行逐个元素的乘法计算的常规技术来计算两个向量之间的点积,示例点积计算器106基于稀疏度图标识稀疏向量中的用于乘法的元素来计算点积。与传统技术相比,此方法占用的存储器更少,并且需要的乘法运算更少。因为示例点积计算器106以较不复杂的计算来执行点积计算,所以点积计算器106能够比常规点积技术使用更少的存储器、更少的处理资源以及更高的速度来确定点积。一旦计算出,示例点积计算器106就输出示例点积结果108。示例点积结果108可以被输出给用户和/或可以被输出给另一处理器、应用,和/或在CNN引擎100的后续过程中使用或进入CNN引擎100。下面结合图2和图3进一步描述示例点积计算器106的示例实现方式。

[0024] 图2是图1的点积计算器106的示例实现方式的框图。图2的示例点积计算器106包括示例接口200、示例迭代控制向量生成器202、示例逻辑门204、示例减法器206、示例尾随二进制计数器208、示例掩码生成器210、示例元素位置确定器212、示例乘法器214、示例求和器216和示例结果存储装置218。

[0025] 图2的示例接口200接收点积指令。例如,用户、应用和/或程序可以发送指令以对两个向量(例如,向量A和向量B)执行点积。另外,接口200与图1的存储器接口104通信以访问输入向量A和B的稀疏位图和/或值。另外,一旦计算出(例如,确定了)结果,示例接口200就输出示例点积结果108。

[0026] 图2的示例迭代控制向量生成器202生成控制向量(例如,向量C),并利用点积协议的每次迭代来更新控制向量。最初,迭代控制向量生成器202基于输入向量A和B的稀疏度图(例如,稀疏度图 A_M 和 B_M)来生成控制向量。例如,迭代控制向量生成器202的示例逻辑门204通过执行稀疏度图 A_M 和 B_M 的逻辑与功能/运算来生成初始控制向量C。迭代控制向量生成器202生成控制向量C(例如,基于 A_M 和 B_M),以隔离需要相乘的密集向量的非零元素。在点积协议的后续迭代期间,示例迭代控制向量生成器202通过基于控制向量C和差向量C-1(例如,对应于控制向量C的值减去一)执行逻辑与功能来更新控制向量C,从而消除尾随的一以进行后继迭代。以这种方式,后继迭代将从密集向量中隔离出不同的元素,直到不再有尾随的元素要被隔离为止。迭代控制向量生成器202的减法器206通过从控制向量C的每个值中减去位值一来计算差向量(例如,C-1)。例如,如果控制向量C为[1,0,0,0](例如,十进制为8),减法器206将向量C-1生成为[0,1,1,1](例如,十进制为7)。一旦减法器206计算出差向量(例如,C-1),则示例逻辑门204对向量C和向量C-1执行逻辑与功能,以生成新的/更新后的控制向量。示例迭代控制向量生成器202通过用新的控制向量(例如, $C=C \text{ AND } (C-1)$)替换控制向量C来更新用于后续迭代的控制向量。当新控制向量的元素都是相同的二进制值(例如,每个元素为0)时,示例迭代控制向量生成器202确定点积协议已完成(例如,没有更多的迭代要运行)。

[0027] 图2的示例尾随二进制计数器208对向量的尾随零的数目(例如,在向量中出现非零值之前、与零相对应的最低有效位的数目)进行计数。例如,对于向量[0,1,0,1,0,0,0],尾随二进制计数器208确定尾随零的数量为三,因为在向量中出现非零值一之前向量的三个最低有效位(例如,向量中最靠右的位)为零。在另一示例中,对于向量[0,0,0,1],尾随二进制计数器208确定尾随零的数目为零,因为向量中不存在尾随零(例如,最低有效位为一)。控制向量上对应的尾随零计数对应于密集向量中元素的位置。为了将这些元素定位在稀疏向量中,生成掩码向量,以将这些位与稀疏度图隔离开来,并使用结果中的一的计数。在点积协议中,在迭代控制向量生成器202生成或更新控制向量之后,尾随二进制计数器208确定控制向量中的尾随零的数目。在其他示例中,尾随二进制计数器208可以对向量的一值的数目进行计数(例如,如果向量被反转的)。附加地或可替代地,示例逻辑门204可以针对点协议的其他部分执行逻辑功能。举例来说,逻辑门204对稀疏度图 A_M/B_M 和掩码(例如,由掩码生成器210生成)执行逻辑与功能。以这种方式,该示例的元素位置确定器212确定点积协议所需的存储器105中存储的稀疏向量中的值的元素位置,如下文进一步描述的。

[0028] 图2的示例掩码生成器210基于尾随二进制计数和输入向量中元素的数量生成掩码向量。举例来说,如果尾随零计数为四且每一输入向量中的值的数目为7,则掩码生成器210将生成具有与输入向量相同维度的掩码向量,其中四个最低有效位为“1”且其余位为“0”(例如[0,0,0,1,1,1,1])。在另一示例中,如果尾随零计数为零并且每个输入向量中的值的数目为4,则掩码生成器210将生成具有与输入向量相同维度的掩码向量,其中所有位都不为“1”(例如,因为尾随零计数为零),而其余位为“0”(例如,[0,0,0,0])。掩码隔离了您想要跳过的元素,并将您感兴趣的元素之外的任何元素归零。一旦示例掩码生成器210生成了掩码向量,示例逻辑门204就在稀疏度图 A_M 和掩码向量上执行逻辑与功能以生成第一结果,并且逻辑门204对稀疏度图 B_M 和掩码向量执行逻辑与功能以生成第二结果。尽管生成了掩码向量以隔离感兴趣的元素,但是可以存在其他方式来隔离感兴趣的元素(例如,要相乘的元素)。例如,示例掩码生成器210可以通过将控制向量减去1(例如,C-1),枚举控制向量C的逆来生成掩码向量,并且示例逻辑门204可以对控制向量减1和其本身以及控制向量C的逆执行逻辑与功能。

[0029] 图2的示例元素位置确定器212包括基于第一结果的一的计数来确定元素在稀疏向量 A_S 中的第一位置,并且基于第二结果的一的计数来确定元素在稀疏向量 B_S 中的第二位置。例如,如果第一结果(例如, A_M AND 掩码)产生具有五个“1”的向量,则元素位置确定器212确定点积协议所需的值是稀疏向量 A_S 的第五位置。在这样的示例中,如果第二结果(例如, B_M AND 掩码)产生具有零个“1”的向量,则元素位置确定器212确定点积协议所需的存储值是稀疏向量 B_S 的第零个位置。另外,元素位置确定器212指示接口200从存储在示例存储器105中的各个稀疏向量访问存储在确定的位置中的值。

[0030] 图2的示例乘法器214将由示例接口200访问的值(例如,对应于由元素位置确定器212确定的位置)相乘。一旦相乘,示例求和器216就将乘积与存储在结果存储装置218中的先前结果求和。最初,存储在结果存储装置218中的值是零,并且在每次点积协议迭代之后进行更新。以这种方式,在初始迭代期间,乘法器214将由接口200访问的值相乘,并将乘积存储在结果存储装置218中。在随后的迭代期间,乘法器214将由接口200访问的值相乘,求和器216将乘积与先前存储的结果求和(例如,来自先前迭代的乘积之和)。一旦示例迭代控

制向量生成器202确定点积协议已完成(例如,由于新的控制向量仅包括零值,因此没有要执行的迭代),接口200访问结果存储装置218中的结果并输出该结果作为点积结果108。下面结合图4进一步描述具有两个示例向量的点积协议的示例。

[0031] 图2的示例CNN引擎100可以部分地由执行指令的处理器来实现。图3是图1的点积计算器106的另一示例实施方式的图。在图3的示例中,点积计算器106由硬件(例如,在专用电路中)实现。在一些示例中,图3的硬件被集成在处理器内部(例如,在处理器封装、片上系统的部分中,等等)。图3的示例点积计算器106包括示例AND逻辑门300、306、314、316、322,示例多路复用器(MUX)302、332,示例减法器304,示例寄存器308、328、334,示例比较器310,示例非门312,示例一计数器318、324,以及示例求和器320、326、330。在一些示例中,示例组件300、302、304、306、308、310可以用于实现示例迭代控制向量生成器202,示例逻辑与门316、322可用于实现逻辑门204,示例组件304可用于实现减法器206,示例一计数器318、324可用于实现尾随二进制计数器208,示例组件312、314可以用于实现示例掩码生成器210,示例求和器320、326可以用于实现示例元素位置确定器212,示例乘法器328可以用于实现示例乘法器214,示例求和器330可以用于实现示例求和器216,并且示例寄存器334可以用于实现图2的示例结果存储装置218。

[0032] 当用户、应用和/或设备(例如,另一个处理器)发送指令以基于两个稀疏向量(例如,As和Bs)确定点积/标量积时,两个对应的稀疏度图(例如,Am和Bm)由示例AND逻辑门300经由示例存储器接口104从示例存储器105获得。示例AND逻辑门300执行AND逻辑功能以生成控制向量C。AND逻辑门300输出控制向量C到示例MUX 302。示例MUX 302接收开始信号,以识别点积/标量积计算何时开始。如下面进一步描述的,结合图4的示例,稀疏度图Am、Bm最初用于确定控制向量。然而,随后的迭代在不利用稀疏度图Am和Bm的情况下更新了控制向量。因此,在示例AND逻辑门300确定了第一控制向量C之后,示例MUX 302不再转发初始控制向量以进行进一步的计算。而是,MUX 302输出后续控制向量(例如,由示例组件304、306、308生成,如下文进一步描述)。

[0033] 图3的示例MUX 302,当由开始信号使能时,将示例AND逻辑门300的输出输出到示例减法器304(例如,初始控制向量)。当没有被开始信号使能时,示例MUX 302输出示例寄存器308的输出(例如,随后的控制向量)。示例减法器304将控制向量减去值一(例如,C-1),并且示例AND逻辑门306对控制向量(C)和控制向量减一(C-1)执行逻辑与功能,从而为存储在示例寄存器308中的后续迭代生成后续控制向量。示例比较器310确定后续控制向量是否等于零。如果后续控制向量等于零,则点积过程完成,并且比较器310输出指示过程结束的触发电压。如果后续控制向量不等于零,则过程以后续控制向量继续。

[0034] 虽然图3的示例组件306,308计算了用于后续迭代的后续控制向量,但是示例组件304、312、314生成用于当前迭代的掩蔽向量。在图3的示例中,基于控制向量减去一(C-1)和控制向量的逆的逻辑与功能来生成掩蔽向量(例如,与控制向量的尾随零计数相对应的向量)。因此,示例减法器304生成控制向量减一,而示例逻辑非门312(例如,反相器)计算控制向量的逆。示例逻辑与门314执行控制向量的逆和控制向量减一的逻辑与功能,从而得到掩码向量。附加地或可替代地,可能存在不同的硬件组件来生成掩码向量。

[0035] 一旦计算出掩码向量,图3的示例逻辑与门316用第一稀疏度图Am执行逻辑与功能,并且示例逻辑与门322用第二稀疏度图Bm执行逻辑与功能。示例一计数器318计算示例

逻辑与门316的输出中的一的总数(例如,ones_count(Am AND掩码)),并且示例一计数器324计算示例逻辑与门322的输出中的一的总数(例如,ones_count(Bm AND掩码))。示例求和器320将示例一计数器318的一计数加到稀疏向量的基地址。求和器326将示例一计数器324的1计数加到稀疏向量Bs的基地址。相应地,求和器320的输出对应于对于当前迭代应当相乘的稀疏向量As的元素的地址,并且求和器326的输出对应于在当前迭代期间应当相乘的稀疏向量Bs的元素的地址。

[0036] 稀疏向量As、Bs(例如,A_addr和B_addr)的地址被发送到示例存储器接口104,以从图1的示例存储器105获得存储在该地址中的值。一旦获得,示例存储器接口104就将相应的值(例如,A和B)发送到示例乘法器328以将这些值相乘。示例乘法器328将乘积输出到示例求和器330,以将乘积与先前迭代的乘积相加。如果没有先前的迭代,则示例求和器330将零与乘积相加,如下文进一步描述的。示例求和器330的输出存储在示例寄存器334中。寄存器334存储先前迭代的乘积之和。当点积/标量积计算完成时(例如,当所有迭代完成时),寄存器334存储并输出点积。例如,寄存器334在接收到对应于计算完成的示例比较器310的输出(例如,完成信号)之后,输出最终的点积/标量积。

[0037] 图3的示例寄存器334输出当前存储的值作为示例MUX 332的第一输入。示例MUX 332还包括与零相对应的第二输入以及与开始信号相对应的选择输入。以这种方式,当点积计算被发起时,MUX 332将输出零。将零值提供给求和器330,以将其与初始迭代的乘积相加。然而,在第一次迭代之后,开始信号改变了,并且MUX 332将输出示例寄存器334的输出。如上所述,示例寄存器334的输出包括所有先前迭代的乘积之和。因此,求和器330将当前迭代的乘积与先前迭代的乘积之和相加,从而与所有迭代完成时的点积相对应。

[0038] 图4示出了由图2和/或图3的示例点积计算器106进行的点积协议操作的示例。图4包括示例密集向量400、402,示例位置标识符404,示例稀疏向量406、408,稀疏向量的示例元素位置410,示例稀疏度图412、414,示例控制向量416,示例尾随零计数418、430,示例掩码420、432,示例稀疏向量值422、424、434、436,示例乘积426、438,示例差向量427、442,示例更新控制向量428,以及示例点积结果440。

[0039] 图4的示例密集向量400、402对应于可以被识别以用于点积运算中的向量。示例位置标识符404对应于密集向量400、402内的元素的位置。如上所述,稀疏向量转换器102将密集向量400、402转换为示例稀疏向量406、408和示例稀疏度图412、414。稀疏向量406对应于示例密集向量400的非零值,这些非零值以与密集向量400中相同的次序列出。稀疏向量408对应于以与密集向量400中相同的次序列出的示例密集向量402的非零值。稀疏向量406、408的值由示例元素位置410(0-4)索引。稀疏度图412是与每个位置标识符404中的密集向量400的元素是对应于零值还是非零值相对应的位图向量。例如,因为密集向量400的第二、第三、第五和第六位置对应于非零值,所以稀疏度图412在第二、第三、第五和第六位置包括“1”。稀疏度图414同样对应于密集向量402的位图向量。

[0040] 当接口200接收对密集向量400和402执行点积的指令时,点积计算器106访问稀疏度图412、414,并且迭代控制向量生成器202通过对示例稀疏度图412和示例稀疏度图414执行逻辑与功能来生成示例控制向量416。在第一次迭代(例如,迭代0)期间,尾随二进制计数器208确定示例尾随零计数418为二,因为在控制向量中存在两个尾随零(例如,在控制向量中有1之前,控制向量中的两个最低有效位为零)。因此,示例掩码生成器210基于尾随零计

数418生成示例掩码向量420。例如,掩码生成器210生成掩码向量420以使两个最低有效位(例如,等于尾随零计数为2的两个位)为“1”,并且其余位为“0”。尽管生成了示例掩码向量420来隔离感兴趣的元素,但是可能存在其他方式隔离感兴趣的元素(例如,要相乘的元素)。例如,示例掩码生成器210可以通过将控制向量减去1(例如,C-1),枚举控制向量C的逆来生成示例掩码向量420,并且示例逻辑门204可以对控制向量减1及其本身与控制向量C的逆执行逻辑与功能。

[0041] 一旦掩码生成器210生成掩码向量420,逻辑门204就对掩码420和第一稀疏度图412执行逻辑与功能。元素位置确定器212基于来自逻辑门204的结果的一计数来确定位置。例如,在图4中,元素位置确定器212将位置确定为零,因为在 A_M AND掩码的结果中不存在一。接口200从存储器105访问在第0位置的稀疏向量406的值,以返回为8的值422(例如,稀疏向量406的第0值)。同样地,掩码生成器210生成掩码向量420,逻辑门204对掩码和第二稀疏度图414执行逻辑与功能。元素位置确定器212基于来自逻辑门204的结果的一计数来确定元素位置。例如,在图4中,元素位置确定器212将位置确定为二,因为在 B_M AND掩码的结果中有两个一。接口200从存储器105访问稀疏向量408的第二位置中的元素/值,以返回为61的值424(例如,稀疏向量408的第二值)。

[0042] 一旦接口200访问了对应的值422、424(例如,8和61),示例乘法器214就将对应的值422、424相乘以生成第一示例乘积426(例如, $8 \times 61 = 488$)。第一示例乘积426被存储到示例结果存储装置218中。在第一迭代完成之后,示例迭代控制向量生成器202将控制向量416减去1(例如,C-1)以生成示例差向量427。控制向量生成器202对控制向量416和差向量427执行逻辑与运算。因为结果包括非零值(例如, $[0, 0, 1, 0, 0, 0, 0, 0]$),迭代控制向量生成器202确定需要后续迭代,并将控制向量416替换为示例新控制向量428。

[0043] 在第二次迭代(例如,迭代1)期间,尾随二进制计数器208确定示例尾随零计数430为5,因为控制向量428中有五个尾随零(例如,在控制向量中有一个一之前控制向量的五个最低有效位为零)。因此,示例掩码生成器210基于尾随零计数430来生成示例掩码向量432。例如,掩码生成器210生成掩码向量432以使五个最低有效位为“1”并且其余位为“0”。

[0044] 一旦掩码生成器210生成了掩码向量432,逻辑门204就对掩码和第一稀疏度图412执行逻辑与功能。元素位置确定器212基于来自逻辑门204的结果的一计数来确定元素位置。例如,在图4中,元素位置确定器212将位置确定为二,因为在 A_M AND掩码的结果中存在两个一。接口200从存储器105访问存储在稀疏向量406的第二位置中的元素/值,以返回为4的值434(例如,稀疏向量406的第二值)。同样,逻辑门204对掩码432和稀疏度图414执行逻辑与功能。元素位置确定器212基于来自逻辑门204的结果的一计数来确定元素位置。在图4中,元素位置确定器212将元素位置确定为三,因为在 B_M AND掩码的结果中存在三个一。接口200在与稀疏向量408的第三位置相对应的地址处访问存储在存储器105中的元素/值,以返回值为6的值436(例如,稀疏向量408的第三值)。

[0045] 一旦接口200访问了相应的值434、436(例如,3和6),示例乘法器214就将相应的值434、436相乘以生成第一示例乘积438(例如,18)。因为这不是第一次迭代,所以示例求和器216将先前存储的乘积426与当前乘积438求和(例如, $488 + 18 = 506$)以生成当前结果440。示例结果存储装置218存储当前结果。在第二次迭代完成之后,示例迭代控制向量生成器202将控制向量428减1(例如,C-1),如图4的上下文中所示,以生成示例差向量442并对控制向

量428和差向量442执行逻辑与功能。因为结果仅包括零值,如图4的上下文所示,迭代控制向量生成器202确定不需要后续迭代并且点积协议已完成。因此,示例接口200访问存储在示例结果存储装置218中的结果,以将其输出为点积结果108。

[0046] 虽然在图2中示出了实现图1的示例点积计算器106的示例方式,但图2中示出的元件、过程和/或设备中的一个或多个可以以任何其他方式来组合、划分、重新布置、省略、消除和/或实施。此外,示例接口200、示例迭代控制向量生成器202、示例逻辑门204、示例减法器206、示例尾随二进制计数器208、示例掩码生成器210、示例元素位置确定器212、示例乘法器214、示例求和器216、示例结果存储装置218和/或更一般地,图2的示例点积计算器106可以通过硬件、软件、固件和/或硬件、软件和/或固件的任何组合来实现。因此,例如,示例接口200、示例迭代控制向量生成器202、示例逻辑门204、示例减法器206、示例尾随二进制计数器208、示例掩码生成器210、示例元素位置确定器212、示例乘法器214、示例求和器216、示例结果存储装置218和/或更一般而言,图2的示例点积计算器106中的任何一个可以通过一个或多个模拟或数字电路、逻辑电路、可编程处理器、可编程控制器、图形处理单元(GPU)、数字信号处理器(DSP)、专用集成电路(ASIC)、可编程逻辑器件(PLD)和/或现场可编程逻辑器件(FPLD)来实现。当阅读该专利的任何装置或系统权利要求以涵盖纯软件和/或固件实现方式时,示例接口200、示例迭代控制向量生成器202、示例逻辑门204、示例减法器206、示例尾随二进制计数器208、示例掩码生成器210、示例元素位置确定器212、示例乘法器214、示例求和器216、示例结果存储装置218和/或更一般而言,图2的示例点积计算器106中的至少一个特此明确地被定义为包括非暂时性计算机可读存储设备或存储磁盘,例如存储器、数字通用磁盘(DVD)、光盘(CD)、蓝光磁盘等,其包括软件和/或固件。更进一步地,图2的示例点积计算器106可以包括:附加于或代替图2中所示的元件、过程和/或设备的一个或多个元件、过程和/或设备,和/或可以包括所图示的元件、过程和设备中的任何一个或全部中的一个以上。如本文中所使用的,短语“在通信中”,包括其变体,涵盖通过一个或多个中间组件的直接通信和/或间接通信,并且不需要直接的物理(例如,有线)通信和/或恒定的通信,而是此外,它还包括以定期间隔、计划间隔、非周期性间隔和/或一次性事件的选择性通信。

[0047] 在图5中示出了流程图,其表示用于实现图1和/或图2的示例点积计算器106的示例硬件逻辑、机器可读指令、硬件实现的状态机和/或其任何组合。机器可读指令可以是用于由计算机处理器(例如,下面结合图6讨论的示例处理器平台600中示出的处理器612)执行的可执行程序或可执行程序的一部分。程序可以体现在存储在非暂时性计算机可读存储介质(例如,CD-ROM、软盘、硬盘驱动器、DVD、蓝光磁盘或与处理器612相关联的存储器)中的软件中,但是整个程序和/或其部分可以可替代地由除处理器612之外的设备执行和/或以固件或专用硬件来体现。此外,尽管参考图5中所示的流程图描述了示例程序,但是,可以可替代地使用实现图2的示例点积计算器106的许多其他方法。例如,可以改变框的执行次序,和/或可以改变、消除或组合所描述的一些框。附加地或可替代地,任何或所有模块可以由构造成在不执行软件或固件的情况下执行相应的操作的一个或多个硬件电路(例如,分立和/或集成的模拟和/或数字电路、FPGA、ASIC、比较器、运算放大器(op-amp)、逻辑电路等)来实现。

[0048] 如上所述,图5的示例过程可以使用存储在非暂时性计算机和/或机器可读介质

(例如,硬盘驱动器、闪存、只读存储器、压缩盘、数字通用磁盘、高速缓存、随机存取存储器 和/或任何其他存储设备或存储盘,其中以任何持续时间(例如,延长的时间段、永久地、短 暂地、用于临时缓冲和/或用于信息的高速缓存)存储信息)上的可执行指令(例如,计算机 和/或机器可读指令)来实现。如本文所使用的,术语“非暂时性计算机可读介质”被明确定 义为包括任何类型的计算机可读存储设备和/或存储磁盘,并且排除传播信号并排除传输 介质。

[0049] “包括”和“包含”(及其所有形式和时态)在本文中用作开放式术语。因此,每当权 利要求采用任何形式的“包括”或“包含”(例如,包含、包括、包含有、包括有、具有等)作为序 言或在任何种类的权利要求记载中时,应理解为在不超出相应权利要求或记载的范围的情 况下,可以存在其他要素、术语等。如本文中所使用的,当短语“至少”在例如权利要求的序 言中用作过渡术语时,其以与术语“包含”和“包含”为开放式相同的方式是开放式的。当例 如以诸如A,B和/或C之类的形式使用时,术语“和/或”是指A、B、C的任何组合或子集,例如 (1)仅A,(2)仅B(3)仅C,(4)A与B,(5)A与C,(6)B与C,以及(7)A与B和C。

[0050] 图5是表示可以由图2和/或图3的点积计算器106的示例实现方式来执行以使用对 应的稀疏向量和稀疏度图来执行两个密集向量的点积计算的示例机器可读指令的示例流 程图500。尽管图5的流程图500是结合图2和/或图3的示例点积计算器106描述的,可以替代 地使用其他类型的点积计算器和/或其他类型的处理器。

[0051] 在框502处,示例点积计算器106确定是否已在接口200处接收到(例如,获得了)指 令以对两个向量执行点积。如果尚未接收到用于执行点积的指令(框502:否),则过程返回 至框502,直到接收到指令为止。如果在接口200处已经接收到用于对两个向量(A和B)执行 点积的指令(框502:是),则示例点积计算器106从示例存储器105访问与两个向量(A和B)相 对应的稀疏度图(A_M 和 B_M) (框504)。例如,接口200与图5的存储器接口104通信以访问与来自 示例存储器105的两个向量相对应的稀疏度图。

[0052] 在框506处,示例迭代控制向量生成器202通过对两个稀疏度图(A_M 和 B_M)执行逻辑 与功能来生成控制向量(C)。例如,逻辑门204执行逻辑与功能以生成控制向量(例如, $C=A_M$ AND B_M)。在一些示例中,如果控制向量C包括全零,则点积完成,并且接口200输出零(例如, 预存储在结果存储装置218中)作为点积结果108。如果控制向量不包括非零值(框507:否), 则点积的结果为零(例如,两个密集向量之间没有公共非零元素),并且过程继续到框534, 以在示例结果存储装置218中输出结果(例如,其被初始化为零)。如果控制向量包括非零值 (框507:是),则示例尾随二进制计数器208确定控制向量(C)的尾随二进制计数(例如,尾随 零计数或尾随一计数)(框508)。举例来说,图2的尾随二进制计数器208确定在控制向量中 出现一之前,多少个最低有效位为零。(在其他示例中,尾随位计数器208确定在控制向量中 出现零之前,多少个最低有效位是一。)

[0053] 在框510处,示例掩码生成器210基于尾随零/二进制计数来生成掩码向量。举例来 说,掩码生成器210可生成具有与输入向量(A和B)相同维度的向量,其中掩码向量的前X个 (例如,X为尾随零计数)最低有效位为“1”,并且掩码向量的其余位为“0”。(在其他示例中, 掩码生成器210可以生成维度与输入向量相同的向量,其中掩码向量的前X个最低有效位为 “0”,并且掩码向量的其余位为“1”。)在框512处,示例逻辑门204通过对掩码和第一稀疏度 图 A_M 执行逻辑与功能(例如,掩码AND A_M),生成第一结果。通过对掩码和第二稀疏度图 B_M 执

行逻辑与功能(例如,掩码AND B_M),生成第二结果。

[0054] 在框514处,示例元素位置确定器212基于第一结果的一计数来确定与第一向量(A)相对应的第一稀疏向量(A_S)的第一存储器位置。例如,元素位置确定器212对第一结果中的一(例如,二进制值)的数量进行计数,并基于一(例如,二进制值)的数量来确定稀疏向量的位置。在框516处,示例元素位置确定器212基于第二结果的一计数来确定与第二向量(B)相对应的第二稀疏向量(B_S)的第二存储器位置。例如,元素位置确定器212对第二结果中的一的数量进行计数,并基于一的数量来确定稀疏向量的位置。

[0055] 在框518处,示例接口200访问存储在稀疏向量(A_S 和 B_S)的第一位置和第二位置中的值。例如,如果第一稀疏向量 A_S 为[5;316;935;17]并且第一存储器位置为2,则接口200会从示例存储器105中存储的稀疏向量中访问为935的值(例如,对应于 A_S 的第二位置,其中5处于第0位置,316处于第1位置,935处于第2位置,并且17处于第3位置)。在框520处,示例乘法器214将来自对应的稀疏向量的访问值相乘以获得乘积。例如,如果从第一稀疏向量 A_S 访问的值为935,并且从第二稀疏向量 B_S 访问的值为5,则乘法器214将值935和5相乘以生成乘积5,675。

[0056] 在框522处,求和器216将乘积与结果存储装置218中存储的值(例如,存储的结果)求和。在框524处,结果存储装置218基于该总和来更新所存储的结果。在第一次迭代期间,存储在结果存储装置218中的值为零。因此,在一些示例中,在第一协议期间,可以跳过框522,并且结果存储装置218可以将乘积作为存储的结果存储在结果存储装置218中。在框526处,示例减法器206从与控制向量C的对应值减去1以生成C-1向量(例如,差向量)。例如,如果控制向量C是[1,0,0,0](例如,十进制的8),则减法器206将向量C-1生成为[0,1,1,1](例如,十进制的7)。

[0057] 在框528处,示例迭代控制向量生成器202通过使用逻辑门204以对控制向量(C)和差向量(C-1)执行逻辑与功能来生成更新后的控制向量。在框530处,示例迭代控制向量生成器202确定更新后的控制向量的元素是否全部对应于相同的二进制值(例如,确定更新后的控制向量的每个元素是否为零)。如果示例迭代控制向量生成器202确定更新后的控制向量元素未全部都对应于相同的二进制值(框530:否),则迭代控制向量生成器202将控制向量替换为更新后的控制向量(框532),并且过程返回框508以执行后续迭代。如果示例迭代控制向量生成器202确定更新后的控制向量的所有元素都对应于相同的二进制值(例如,全零)(方框530:是),则示例接口200访问结果存储装置218中存储的结果,并输出存储的结果作为点积结果108(框534)。

[0058] 图6是被构造为执行图5的指令来实现图2的示例点积计算器106的示例处理器平台1000的框图。处理器平台600可以是例如服务器、个人计算机、工作站、自学习机(例如,神经网络)、移动设备(例如,手机、智能电话、平板电脑(例如,iPadTM))或其他任何类型的计算设备。

[0059] 所示示例的处理器平台600包括处理器612。所示示例的处理器612是硬件。例如,处理器612可以由来自任何期望的系列或制造商的一个或多个集成电路、逻辑电路、微处理器、GPU、DSP或控制器来实现。硬件处理器可以是基于半导体的(例如,基于硅的)设备。在此示例中,处理器实现示例接口200、示例迭代控制向量生成器202、示例逻辑门204、示例减法器206、示例尾随二进制计数器208、示例掩码生成器210、示例元素位置确定器212、示例乘

法器214和示例求和器216。

[0060] 所示示例的处理器612包括本地存储器613(例如,高速缓存)。在一些示例中,本地存储器613实现示例结果存储装置218。所示示例的处理器612通过总线618与包括易失性存储器614和非易失性存储器616的主存储器进行通信。主存储器实现示例存储器105。易失性存储器614可以由同步动态随机存取存储器(SDRAM)、动态随机存取存储器(DRAM)、**RAMBUS®**动态随机存取存储器(**RDRAM®**)和/或任何其他类型的随机存取存储器设备来实现。非易失性存储器616可以由闪存和/或任何其他期望类型的存储器设备来实现。对主存储器614、616的访问由存储器控制器控制。

[0061] 所示示例的处理器平台600还包括接口电路620。接口电路620可以通过任何类型的接口标准来实现,例如以太网接口、通用串行总线(USB)、**蓝牙®**接口、近场通信(NFC)接口和/或PCI Express接口。

[0062] 在示出的示例中,一个或多个输入设备622连接到接口电路620。输入设备622允许用户向处理器612中输入数据和/或命令。输入设备可以通过例如音频传感器、麦克风、照相机(静止或视频)、键盘、按钮、鼠标、触摸屏、触控板、轨迹球、等位点(isopoint)和/或语音识别系统来实现。

[0063] 一个或多个输出设备624也连接到所示示例的接口电路620。输出设备624可以例如由显示设备(例如,发光二极管(LED)、有机发光二极管(OLED)、液晶显示器(LCD)、阴极射线管显示器(CRT)、就地切换(in-place switching, IPS)显示器、触摸屏等)、触觉输出设备、打印机和/或扬声器来实现。因此,所示示例的接口电路620通常包括图形驱动器卡、图形驱动器芯片和/或图形驱动器处理器。

[0064] 所示示例的接口电路620还包括通信设备,例如发射机、接收机、收发机、调制解调器、住宅网关、无线接入点和/或网络接口,以促进通过网络626与外部机器(例如,任何种类的计算机设备)的数据交换。通信可以经由例如以太网连接、数字用户线(DSL)连接、电话线连接、同轴电缆系统、卫星系统、站点线无线系统、蜂窝电话系统等。

[0065] 所示示例的处理器平台600还包括一个或多个用于存储软件和/或数据的大容量存储设备628。这种大容量存储设备628的示例包括软盘驱动器、硬盘驱动器、光盘驱动器、蓝光驱动器、独立磁盘冗余阵列(RAID)系统和数字多功能盘(DVD)驱动器。

[0066] 图5的机器可执行指令632可以存储在大容量存储设备628中,易失性存储器614中,非易失性存储器616中,和/或可移动非暂时性计算机可读存储介质(例如,CD或DVD)上。

[0067] 根据前述内容,将意识到,已经公开了使用稀疏向量执行点积计算的示例方法、装置和制品。所公开的方法、装置和制品通过存储较小的稀疏向量(例如,仅包括非零值的向量)以及与大的密集向量相对应的稀疏度图/向量(例如,密集向量的位图)减少存储大的密集向量(例如,包括零值和非零值二者的向量)所需的存储器量,从而提高了计算设备的效率。另外,本文公开的示例使用稀疏度图来执行点积,以(与现有技术相关地)减少了执行点积所需的本地存储器的量并减少了执行点积所需的复杂乘法运算的量。因此,本文公开的示例通过减少执行点积计算所需的处理器资源量(例如,需要较少的处理器周期来执行相同的计算)来提高计算设备的效率,从而提高了计算点积计算的速度。因此,所公开的方法、装置和制品针对计算机功能上的一项或多项改进。

[0068] 示例1包括点积计算器,该点积计算器包括:用于确定控制向量的尾随二进制计数

的计数器,该控制向量对应于对第一稀疏向量的第一位图和第二稀疏向量的第二位图的第一逻辑与运算的第一结果;掩码生成器,其用于基于所述尾随二进制计数来生成掩码向量;接口,其用于:基于对第一位图和掩码向量的第二逻辑与运算的第二结果,访问第一稀疏向量的第一值;以及基于对第二位图和掩码向量的第三逻辑与运算的第三结果,访问第二稀疏向量的第二值;以及乘法器,其用于将所述第一值与所述第二值相乘以生成乘积。

[0069] 示例2包括示例1的点积计算器,其中,第一位图用于标识第一向量的第一元素分别对应于零值还是非零值,并且第二位图用于标识所述第二向量的第二元素分别对应于零值还是非零值,并且第一稀疏向量对应于第一密集向量的非零值,并且第二稀疏向量对应于第二密集向量的非零值。

[0070] 示例3包括示例1的点积计算器,还包括逻辑门,所述逻辑门基于以第一位图和第二位图为输入的所述第一逻辑与运算来生成所述控制向量。

[0071] 示例4包括示例1的点积计算器,其中,所述掩码生成器用于生成所述掩码向量以在所述最低有效位中包括一定数量的第一二进制值,所述数量对应于尾随二进制计数,所述掩码生成器用于生成与第一向量具有相同维度的所述掩码向量。

[0072] 示例5包括示例1的点积计算器,还包括:元素位置确定器,其用于:确定所述第二结果中的第一数量的二进制值,所述接口用于基于与第一数量的二进制值相对应的第一地址访问所述第一值;以及确定所述第三结果中的第二数量的二进制值,所述接口用于基于与第二数量的二进制值相对应的第二地址来访问第二值。

[0073] 示例6包括示例1的点乘积计算器,还包括用于存储乘积的存储装置。

[0074] 示例7包括示例1的点积计算器,还包括:减法器,其用于通过从与控制向量的二进制位相对应的值中减去一来生成差向量;以及迭代控制向量生成器,其用于:生成与在控制向量和差向量上进行逻辑与运算相对应的更新后的控制向量;以及确定所述更新后的控制向量的所有元素是否对应于相同的二进制值。

[0075] 示例8包括示例7的点积计算器,其中,当迭代控制向量生成器确定所述更新后的控制向量的所有元素对应于相同的二进制值时,接口用于输出所述乘积作为点积结果。

[0076] 示例9包括示例7的点积计算器,其中,所述尾随二进制计数是第一尾随二进制计数,所述掩码向量是第一掩码向量,所述乘积是第一乘积,并且当所述迭代控制向量生成器确定更新后的控制向量的所有元素不对应于相同的二进制值时:所述计数器用于确定更新后的控制向量的第二尾随二进制计数;所述掩码生成器用于生成与第二尾随二进制计数相对应的第二掩码向量;所述接口用于:基于对所述第一位图和所述第二掩码向量的第四逻辑与运算的第四结果,访问所述第一稀疏向量的第三值;基于对所述第二位图和所述第二掩码向量的第五逻辑与运算的第五结果,访问所述第二稀疏向量的第四值;以及所述乘法器用于将所述第三值与所述第四值相乘以生成第二乘积,所述装置还包括:求和器,其用于将所述第一乘积与所述第二乘积求和。

[0077] 示例10包括示例9的点积计算器,其中,所述差向量是第一差向量,所述更新后的控制向量是第一更新后的控制向量,并且所述元素是第一元素,并且:所述减法器用于通过从更新后的控制向量的二进制值的值转换中减去一来生成第二差向量;以及所述迭代控制向量生成器用于生成与对更新后的控制向量和第二差向量的逻辑与运算相对应的第二更新后的控制向量;并且当第二更新后的控制向量的所有第二元素对应于相同的二进制值

时,所述接口用于输出所述第一乘积与所述第二乘积之和作为点积结果。

[0078] 示例11包括至少一个非暂时性计算机可读存储介质,其包括指令,这些指令在被执行时使机器至少:确定控制向量的尾随二进制计数,该控制向量对应于第一稀疏向量的第一位图和第二稀疏向量的第二位图上的第一逻辑与运算的第一结果;基于所述尾随二进制计数生成掩码向量;以及将(A)基于对第一位图和掩码向量的第二逻辑与运算的第二结果的第一稀疏向量的第一值与(B)基于对第二位图和掩码向量的第三逻辑与运算的第三结果的对应于第二向量的第二稀疏向量的第二值相乘。

[0079] 示例12包括示例11的计算机可读存储介质,其中,第一位图用于分别标识第一向量的零值或非零值,并且第二位图分别用于标识第二向量的零值或非零值;并且第一稀疏向量对应于第一密集向量的非零值,并且第二稀疏向量对应于第二密集向量的非零值。

[0080] 示例13包括示例11的计算机可读存储介质,其中,指令使所述机器基于对作为输入的第一位图和所述第二位图的第一逻辑与运算来生成所述控制向量。

[0081] 示例14包括示例11的计算机可读存储介质,其中,指令使所述机器生成所述掩码向量,以在所述最低有效位中包括一定数量的第一二进制值,所述数量对应于尾随二进制计数,所述掩码向量具有与第一向量相同的维度。

[0082] 示例15包括示例11的计算机可读存储介质,其中指令使机器:确定第二结果中的第一数量的二进制值;基于与所述第一数量的二进制值相对应的第一地址访问所述第一值;确定第三结果中的第二数量的二进制值;以及基于与所述第二数量的二进制值相对应的第二地址访问第二值。

[0083] 示例16包括示例11的计算机可读存储介质,其中,指令使所述机器将所述第一值和所述第二值的乘积存储在本地存储器中。

[0084] 示例17包括示例11的计算机可读存储介质,其中,指令使机器:通过从与控制向量的二进制位相对应的值中减去一来生成差向量;生成与对控制向量和差向量的逻辑与运算相对应的更新后的控制向量;以及确定更新后的控制向量的所有元素是否对应于相同的二进制值。

[0085] 示例18包括示例17的计算机可读存储介质,其中,当所述更新后的控制向量的所有元素对应于相同的二进制值时,所述指令使所述机器输出所述第一值和所述第二值的乘积作为点积结果。

[0086] 示例19包括示例17的计算机可读存储介质,其中,尾随二进制计数是第一尾随二进制计数,所述掩码向量是第一掩码向量,所述乘积是第一乘积,并且当更新后的控制向量的所有元素都不对应于相同的二进制值时,所述指令使所述机器:确定更新后的控制向量的第二尾随二进制计数;生成与第二尾随二进制计数相对应的第二掩码向量;基于对第一位图和第二掩码向量的第四逻辑与运算的第四结果,访问第一稀疏向量的第三值;基于对第二位图和第二掩码向量的第五逻辑与运算的第五结果,访问第二稀疏向量的第四值;以及将第三值与第四值相乘以生成第二乘积;以及将所述第一乘积与所述第二乘积求和。

[0087] 示例20包括示例19的计算机可读存储介质,其中,差向量是第一差向量,所述更新后的控制向量是第一更新后的控制向量,并且所述元素是第一元素,并且,所述指令使所述机器执行以下操作:通过从更新后的控制向量的二进制值的值转换中减去一来生成第二差向量;生成与更新后的控制向量与第二差向量之间的逻辑与功能相对应的第二更新后的控

制向量;以及当第二更新后的控制向量的所有第二元素对应于相同的二进制值时,输出第一乘积与第二乘积之和作为点积结果。

[0088] 示例21包括一种用于确定两个向量之间的点积的方法,该方法包括:利用至少一个逻辑电路确定控制向量的尾随二进制计数,所述控制向量对应于对第一向量的第一位图和第二向量的第二位图的第一逻辑与运算的第一结果;利用所述至少一个逻辑电路生成与所述尾随二进制计数相对应的掩码向量;基于对所述第一位图和所述掩码向量的第二逻辑与运算的第二结果,访问与所述第一向量相对应的第一稀疏向量的第一值;以及基于对所述第二位图和所述掩码向量的第三逻辑与运算的第三结果,访问与第二向量相对应的第二稀疏向量的第二值;以及利用所述至少一个逻辑电路将第一值与第二值相乘以生成乘积。

[0089] 示例22包括示例21的方法,其中,所述第一位图对应于第一向量的第一元素是对应于零值还是非零值,并且所述第二位图对应于第二向量的第二元素是对应于零值还是非零值;并且所述第一稀疏向量对应于所述第一向量的非零值,并且所述第二稀疏向量对应于所述第二向量的非零值。

[0090] 示例23包括示例21的方法,还包括基于以第一位图和第二位图作为输入的所述第一逻辑与运算来生成控制向量。

[0091] 示例24包括示例21的方法,还包括生成所述掩码向量以在所述最低有效位中包含一定数量的第一二进制值,所述数量对应于所述尾随二进制计数,所述掩码向量具有与所述第一向量相同的维度。

[0092] 示例25包括示例21的方法,该方法还包括:确定所述第二结果中的第一数量的二进制值;基于与所述第一数量的二进制值相对应的第一地址访问所述第一值;确定第三结果中的第二数量的二进制值;以及基于与所述第二数量的二进制值相对应的第二地址访问所述第二值。

[0093] 示例26包括示例21的方法,还包括:将所述第一值和所述第二值的乘积存储在本地存储器中。

[0094] 示例27包括示例21的方法,还包括:通过从与控制向量的二进制位相对应的值中减去一来生成差向量;生成与对控制向量和差向量的逻辑与运算相对应的更新后的控制向量;以及确定更新后的控制向量的所有元素是否对应于相同的二进制值。

[0095] 示例28包括示例27的方法,还包括:当更新后的控制向量的所有元素对应于相同的二进制值时,输出第一值和第二值的乘积作为点积结果。

[0096] 示例29包括示例27的方法,其中,所述尾随二进制计数是第一尾随二进制计数,所述掩码向量是第一掩码向量,所述乘积是第一乘积,并且还包括,当所述更新后的控制向量的所有元素不对应于相同的二进制值时:确定更新后的控制向量的第二尾随二进制计数;生成与第二尾随二进制计数相对应的第二掩码向量;基于对第一位图和第二掩码向量的第四逻辑与运算的第四结果,访问第一稀疏向量的第三值;基于对第二位图和第二掩码向量的第五逻辑与运算的第五结果,访问第二稀疏向量的第四值;将第三值与第四值相乘以生成第二乘积;以及将所述第一乘积与所述第二乘积求和。

[0097] 示例30包括示例29的方法,其中,差向量是第一差向量,更新后的控制向量是第一更新后的控制向量,并且所述元素是第一元素,并且还包括:通过从更新后的控制向量的二进制值的值转换中减去一来生成第二差向量;生成与对更新后的控制向量和第二差向量的

逻辑与运算相对应的第二更新后的控制向量;以及当第二更新后的控制向量的所有第二元素对应于相同的二进制值时,输出第一乘积与第二乘积之和作为点积结果。

[0098] 示例31包括一种点积计算器,包括:用于确定控制向量的尾随二进制计数的第一单元,该控制向量对应于对第一稀疏向量的第一位图和第二稀疏向量的第二位图的第一逻辑与运算的第一结果;第二单元,用于基于尾随二进制计数生成掩码向量;第三单元,用于:基于对第一位图和掩码向量的第二逻辑与运算的第二结果,访问第一稀疏向量的第一值;以及基于对第二位图和掩码向量的第三逻辑与运算的第三结果,访问第二稀疏向量的第二值;以及第四单元,用于将第一值与第二值相乘以生成乘积。

[0099] 示例32包括示例31的点积计算器,其中,第一位图用于标识第一向量的第一元素分别对应于零值还是非零值,并且所述第二位图用于标识第二向量的第二元素分别对应于零值还是非零值;以及第一稀疏向量对应于第一密集向量的非零值,并且所述第二稀疏向量对应于第二密集向量的非零值。

[0100] 示例33包括示例31的点积计算器,还包括第五单元,用于基于以所述第一位图和所述第二位图为输入的所述第一逻辑与运算来生成所述控制向量。

[0101] 示例34包括示例31的点积计算器,其中,第二单元包括用于生成所述掩码向量以在最低有效位中包括一定数量的第一二进制值的单元,所述数量对应于尾随二进制计数,所述第二单元包括用于生成掩码向量以具有与第一向量相同的维度的单元。

[0102] 示例35包括示例31的点积计算器,还包括第六单元,用于:确定第二结果中的第一数量的二进制值,所述第三单元包括用于基于与第一数量的二进制值相对应的第一地址访问第一值的单元;以及确定第三结果中的第二数量的二进制值,第三单元包括用于基于与第二数量的第二值相对应的第二地址访问第二值的单元。

[0103] 示例36包括示例31的点乘积计算器,还包括用于存储所述乘积的第七单元。

[0104] 示例37包括示例31的点积计算器,还包括:第八单元,用于通过从与控制向量的二进制位相对应的值中减去一来生成差向量;以及第九单元,用于:生成与对控制向量和差向量的逻辑与运算相对应的更新后的控制向量;以及确定所述更新后的控制向量的所有元素是否对应于相同的二进制值。

[0105] 示例38包括示例37的点积计算器,其中,第三单元包括用于在所述迭代控制向量生成器确定更新后的控制向量的所有元素对应于相同的二进制值时,将所述乘积作为点积结果输出的单元。

[0106] 示例39包括示例37的点积计算器,其中,尾随二进制计数是第一尾随二进制计数,所述掩码向量是第一掩码向量,所述乘积是第一乘积,并且当所述迭代控制向量生成器确定更新后的控制向量的所有元素不对应于相同的二进制值时:所述第一单元包括用于确定所述更新后的控制向量的第二尾随二进制计数的单元;所述第二单元包括用于生成与所述第二尾随二进制计数相对应的第二掩码向量的单元;第三单元,包括用于执行以下操作的单元:基于对第一位图和第二掩码向量的第四逻辑与运算的第四结果,访问第一稀疏向量的第三值;以及基于对第二位图和第二掩码向量的第五逻辑与运算的第五结果,访问第二稀疏向量的第四值;以及第四单元,包括用于将第三值与第四值相乘以生成第二乘积的单元,该装置还包括:求和器,其用于将第一乘积与第二乘积求和。

[0107] 示例40包括示例39的点积计算器,其中,差向量是第一差向量,所述更新后的控制

向量是第一更新后的控制向量,并且所述元素是第一元素,并且第八单元包括用于通过从更新后的控制向量的二进制值的值转换中减去一来生成第二差向量的单元;以及第九单元包括用于生成与对更新后的控制向量和第二差向量的逻辑与运算相对应的第二更新后的控制向量的单元;以及第三单元包括用于当第二更新后的控制向量的所有第二元素对应于相同的二进制值时,输出第一乘积与第二乘积之和作为点积结果的单元。

[0108] 尽管本文已经公开了某些示例方法、装置和制品,但是本专利的覆盖范围不限于此。相反,本专利涵盖了合理落入本专利的权利要求范围内的所有方法、装置和制品。

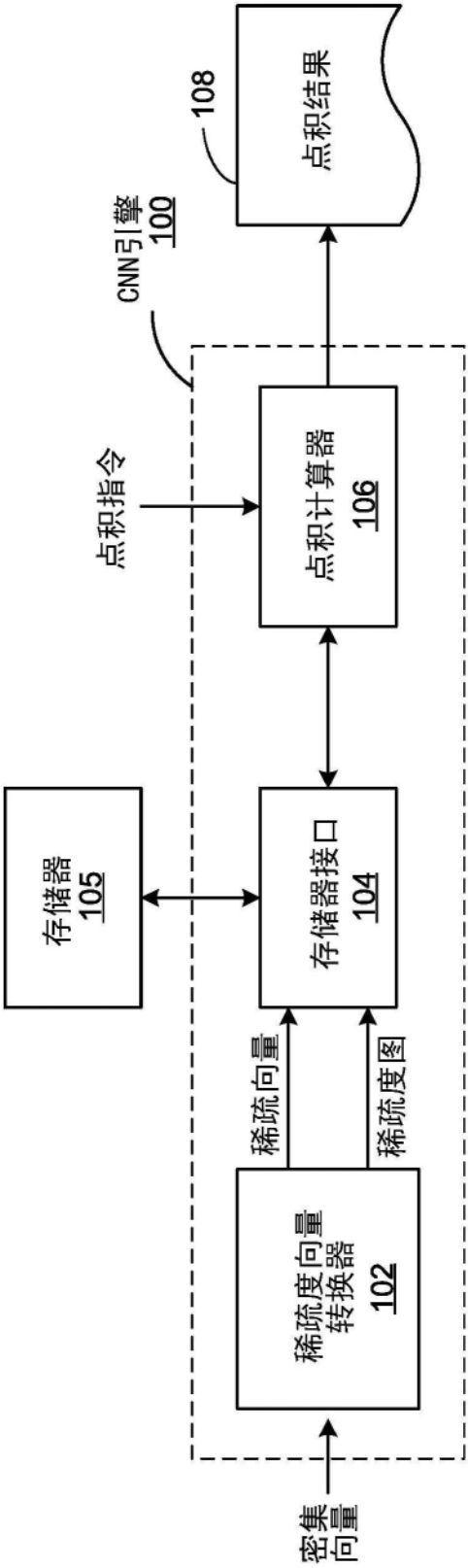


图1

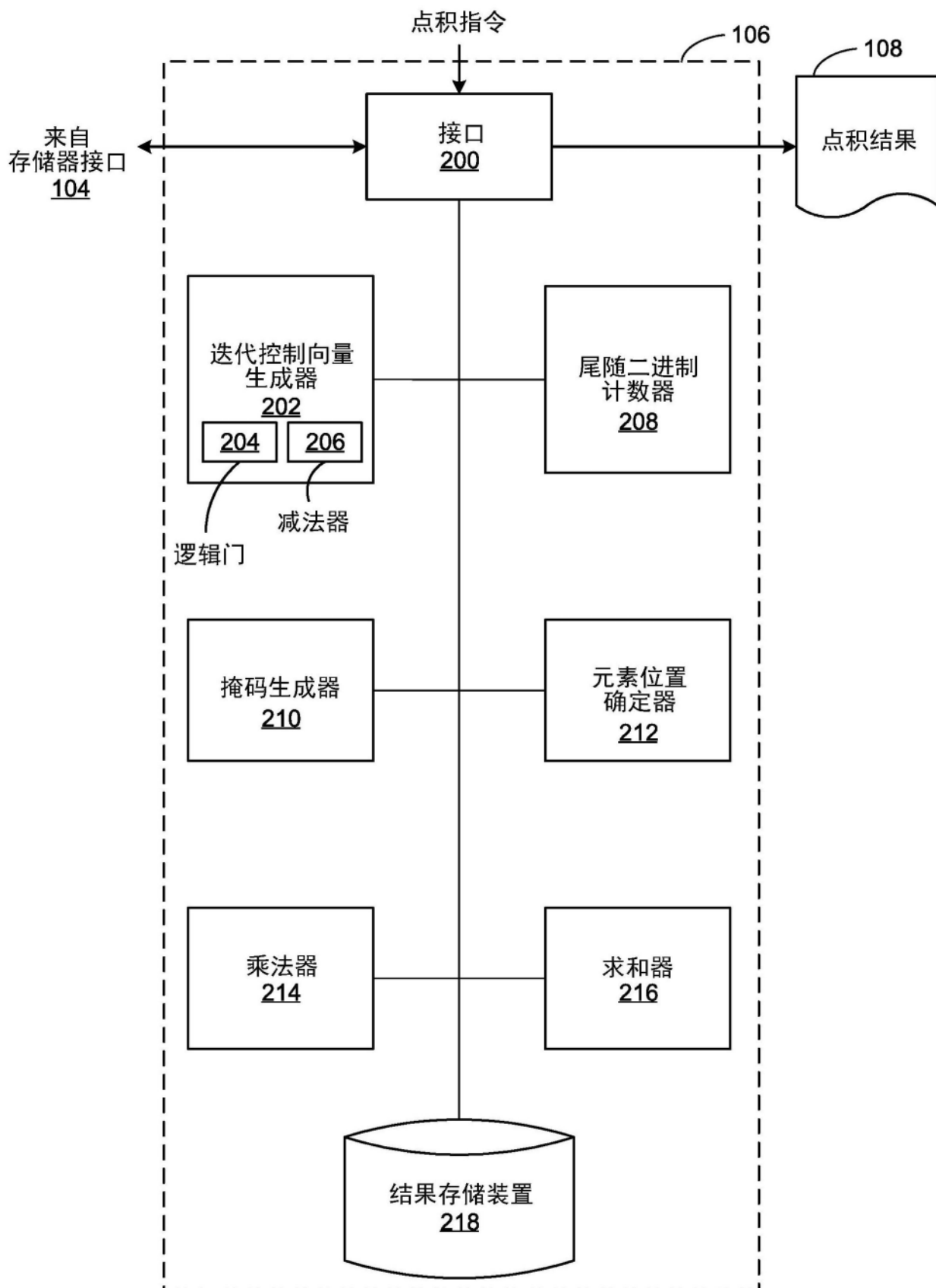


图2

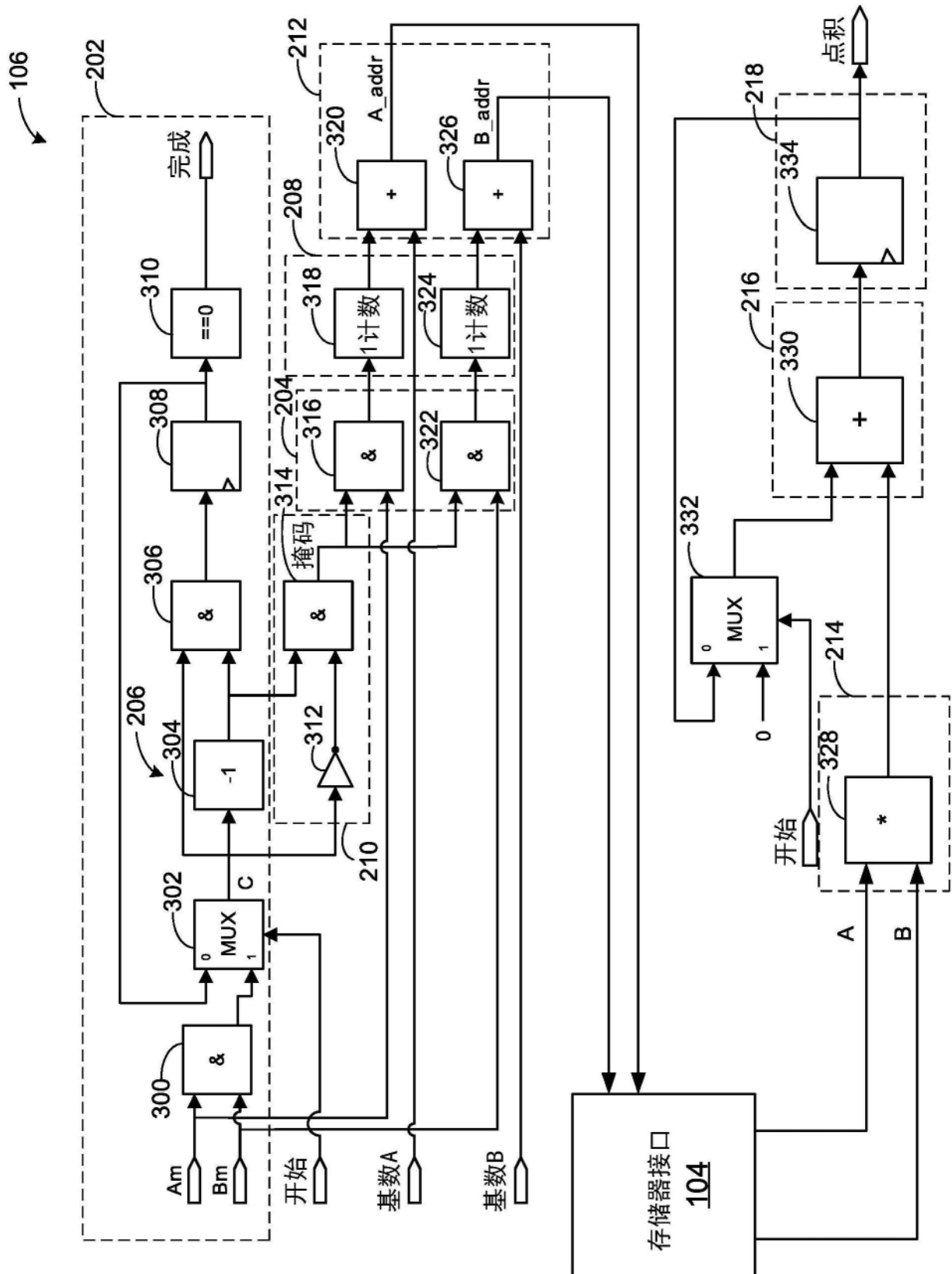


图3

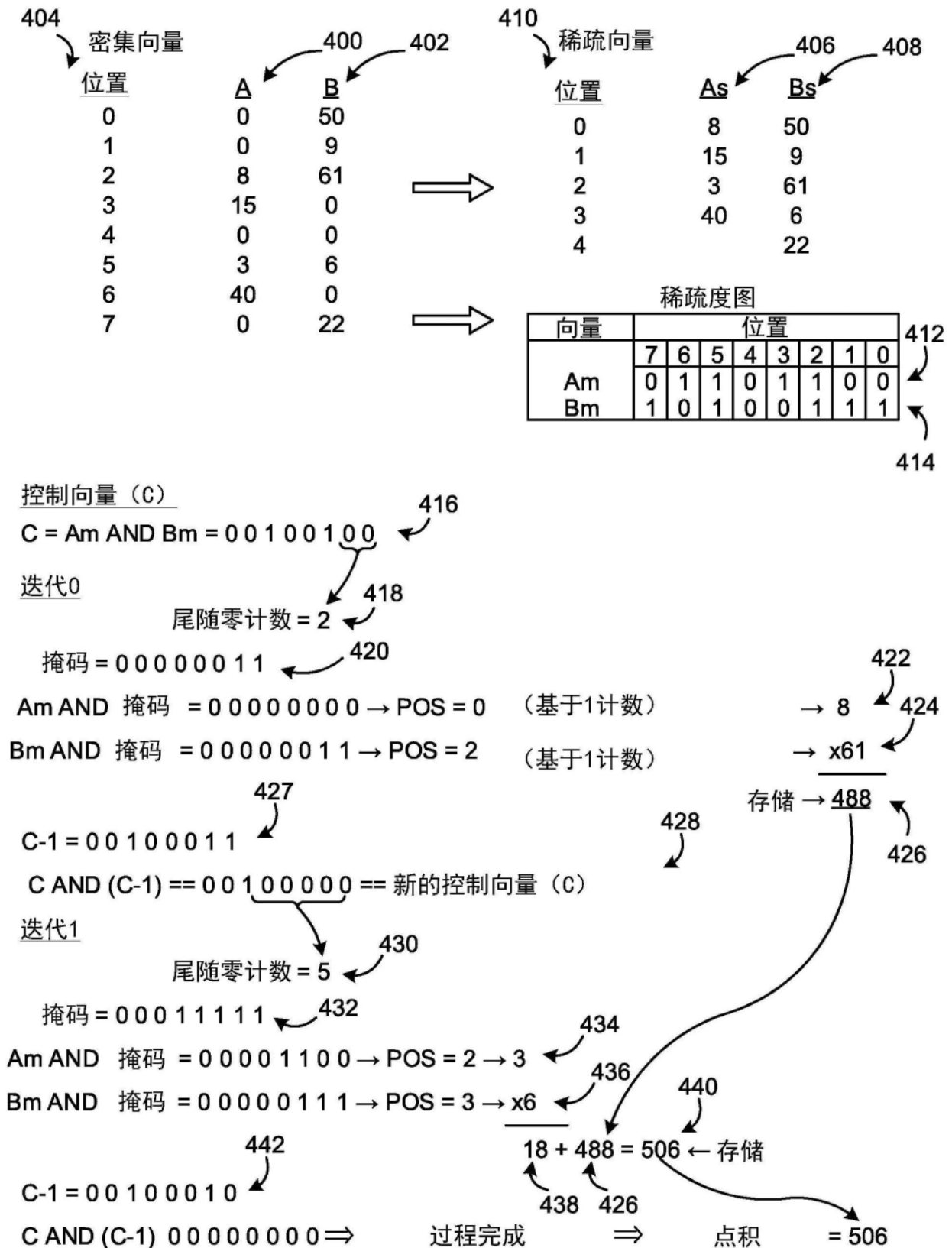


图4

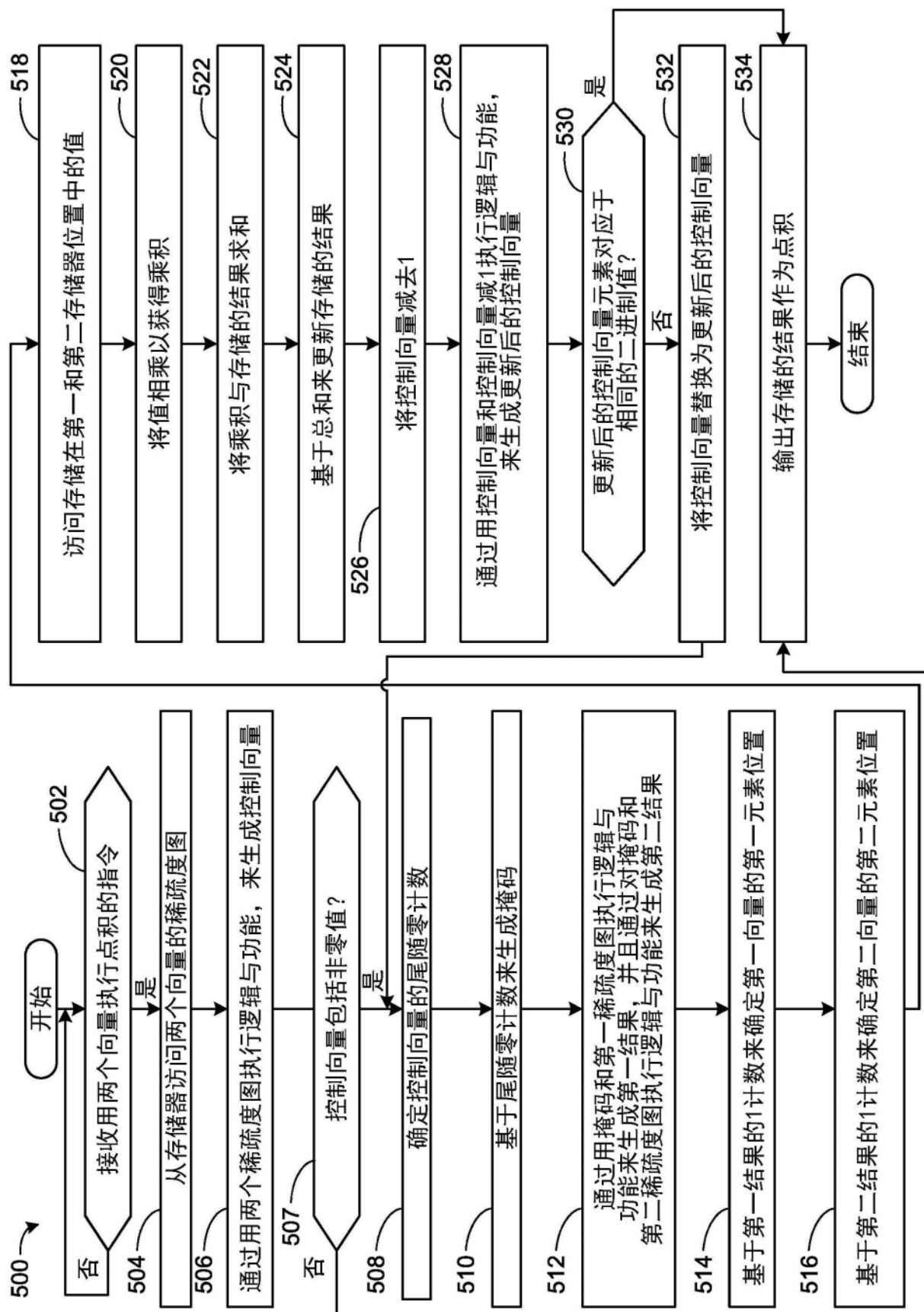


图5

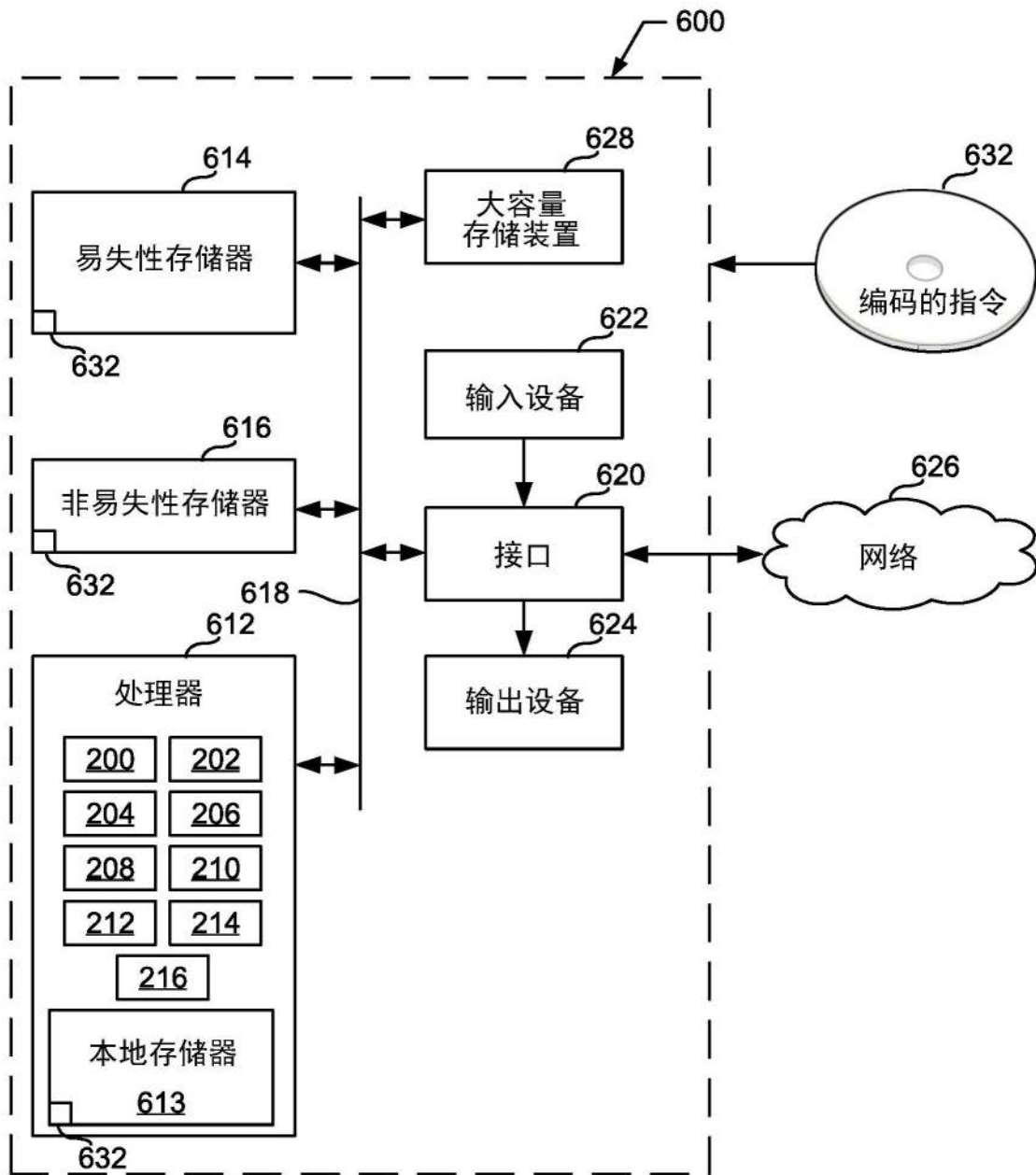


图6