Computer Organization & Design

—The Hardware/Software Interface

计算机组成原理课程群

中山大学

数据科学与计算机学院

无人系统研究所

主讲老师：陈刚

Email:cheng83@mail.sysu.edu.cn

https://sdcs.sysu.edu.cn/content/4547

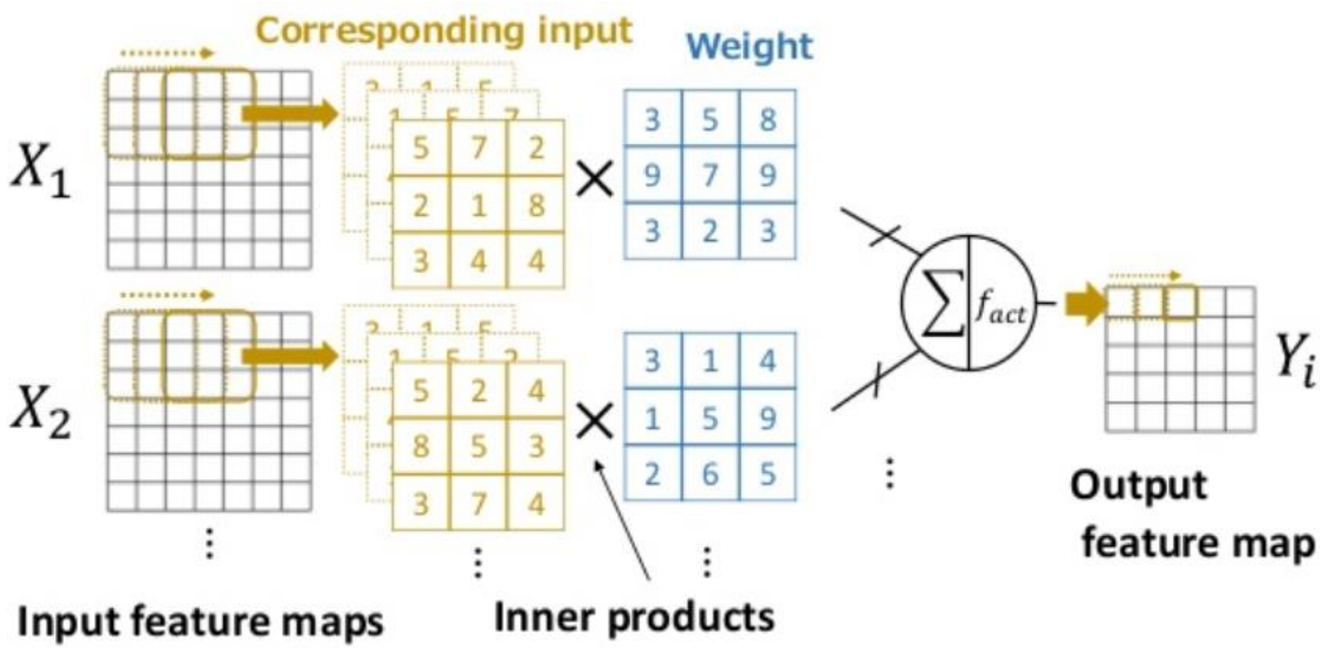
* Middle-Term Homework

了解卷积的过程

使用MIPS汇编和MIPS仿真器，设计并实现一个普通整数卷积计算算子，要求有完整的输入输出，输入为7*7*1格式的张量，对应的卷积核一个，尺寸为3*3*1，步长为1，输出为经过卷积计算后的对应的5*5*1张量，要求计算结果正确。

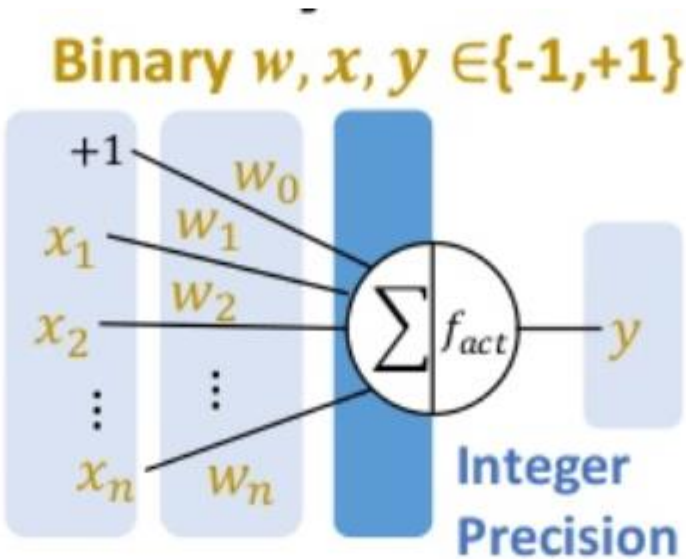
参考卷积操作图：

- 这里7*7*1代表图像，大家可以读入更大的图像（sys_call），这里只是方便大家调试，做了一些简化
- 思路：每个像素点（x,y）对应一个固定的mask, 取出窗口
- 代码结构：4层for循环



了解BNN的过程:卷积过程可以化为位操作

∗ Basic idea



Binary Neuron Model

	-1	+1
-1	+1	-1
+1	-1	+1

Binary mult.

	0	1
0	1	0
1	0	1

XNOR

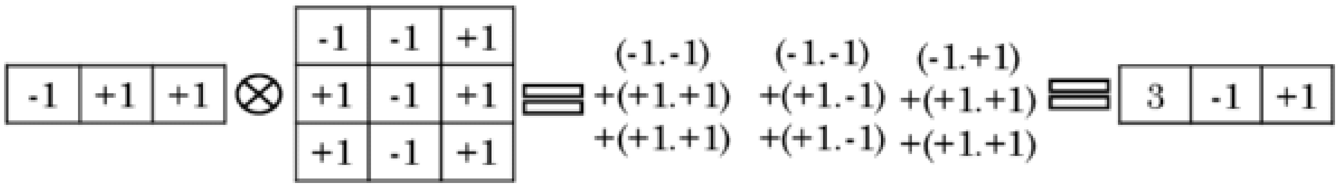
- ☐ Rounding activation value and weight to binary value {-1,+1}

$$\begin{bmatrix} -1 \\ +1 \\ \vdots \\ +1 \end{bmatrix} = f_{act}\left(\begin{bmatrix} -1 & +1 & \cdots & +1 \\ -1 & -1 & \cdots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & \cdots & +1 \end{bmatrix} \begin{bmatrix} +1 \\ +1 \\ \vdots \\ -1 \end{bmatrix} \right)$$

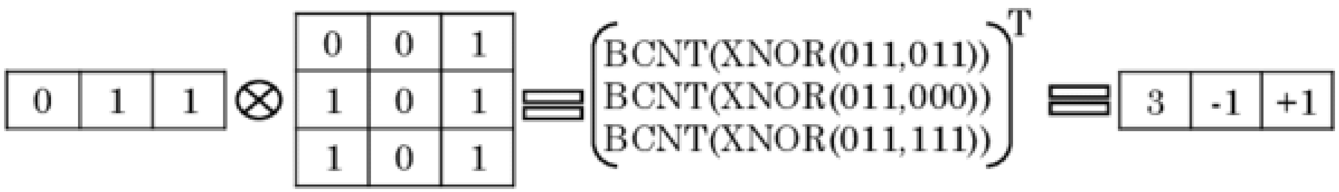
y W x

- ☐ Multiplication of the binary value has correspondence with XNOR logic operation
- ☐ Suitable for hardware
  - ☐ BCNT for multiplication
  - ☐ Memory efficient

了解BNN的过程：卷积过程可以化为位操作



(a) An example of binarized MM



(b) Binarized MM using XNOR and BCNT. -1 is represented using 0.

BCNT= OneCount-ZeroCount

IN	Computation	OUT
000	-1-1-1= -3	101
001	-1-1+1 = -1	111
010	-1+1-1= -1	111
011	-1+1+1= +1	001
100	+1-1-1= -1	111
101	+1-1+1= +1	001
110	+1+1-1= +1	001
111	+1+1+1= +3	011

(c) BCNT using a lookup table (OUT is in 2's complement form)

2*BCNT(XNOR(A*B))-N（Why？？？）

- For FPGA, XNOR gate can be implemented for BNN, avoiding float MM operation.
- **However, the GPU implementations of BNN is still in a proof-of concept stage.**

- a 和 b 分别是-1和+1的向量
- A和B分别是0和1的向量（0代表-1，1代表1）
- A=（a+1）/2；B=（b+1）/2
- a=2A-1; b=2B-1
- a*b=(2A-1)*(2B-1)=2(2AB-(A+B)+1)-1=2(AB+(1-A)(1-B))-1
- a*b=2(xnor(A,B))-1

Layer-flow integration

✓ Expensive computations such as division and multiplication operations can be avoided

$$x_1 = 2 \times popcount(\omega\ xnor\ x_0) - N$$

$$x_2 = \eta \cdot x_1 + b$$

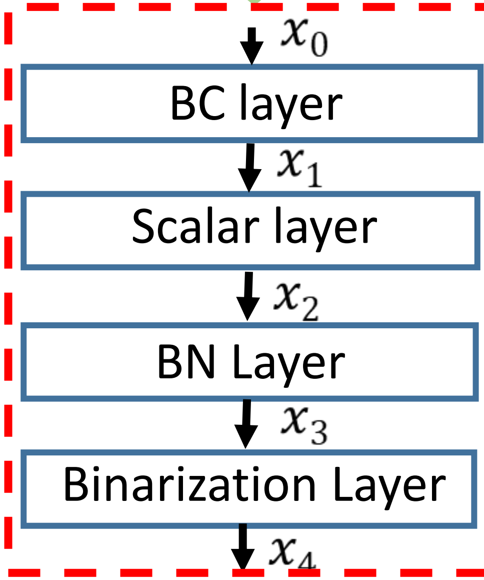
$$x_3 = \gamma \cdot \frac{x_2 - \mu}{\sigma} + \beta$$

$$x_4 = \begin{cases} 1 & if \quad x_3 > 0 \\ 0 & otherwise \end{cases}$$

$$x_4 = \begin{cases} x_1 \geq \xi_1 & if\ \gamma > 0 \\ x_1 \geq \xi_2 & if\ \gamma < 0 \\ sign(\beta) & if\ \gamma = 0 \end{cases}$$

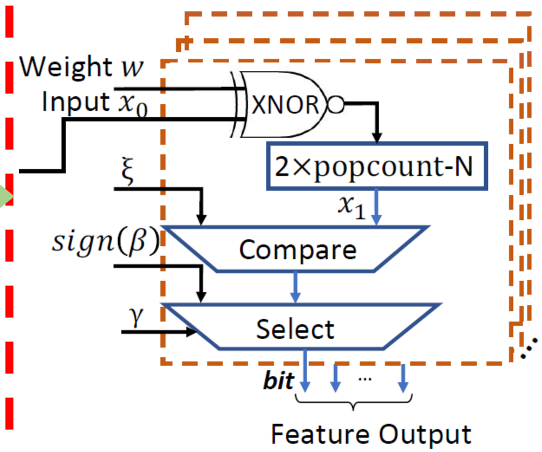
$$\xi_1 = \left\lceil -\frac{\beta \cdot \sigma}{\gamma \cdot \eta} + \frac{\mu}{\eta} - \frac{b}{\eta} \right\rceil$$

$$\xi_2 = \left\lfloor -\frac{\beta \cdot \sigma}{\gamma \cdot \eta} + \frac{\mu}{\eta} - \frac{b}{\eta} \right\rfloor$$



operations

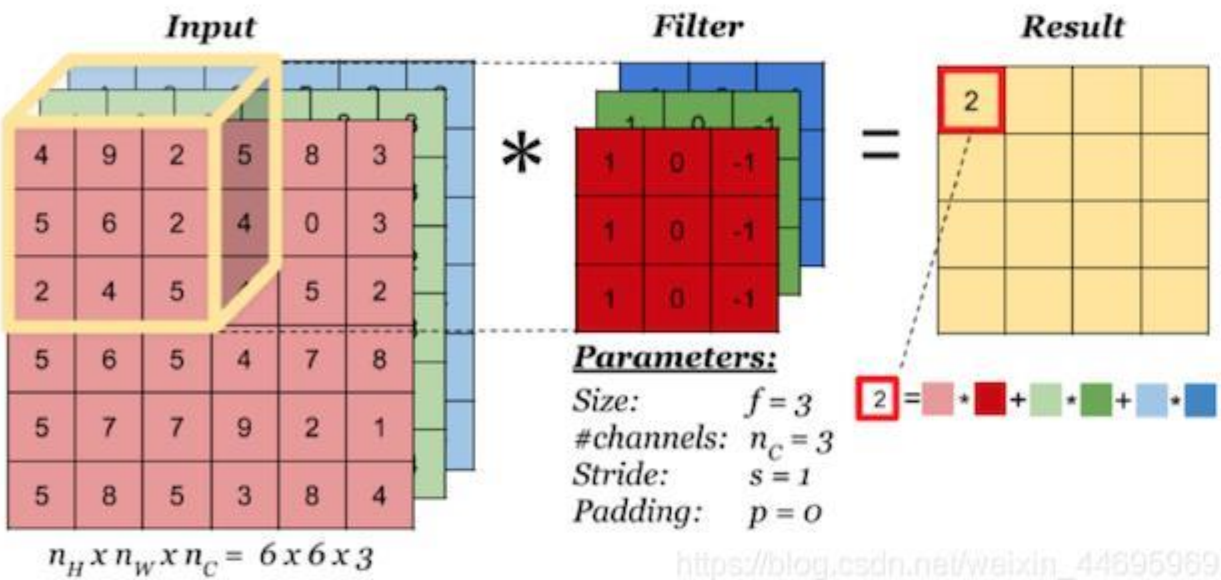
- ✓ Popcount
- ✓ Xnor
- ✓ Shifting
- ✓ Comparer
- ✓ Division
- ✓ Multiplication
- ✓ Adder
- ✓ Subtraction



operations

- ✓ Popcount
- ✓ Xnor
- ✓ Shifting
- ✓ Comparer

2. 设计并实现一个二值卷积计算算子，要求有完整的输入输出，输入为 7*7*16bit 格式的张量，对应的卷积核一个，尺寸为 3*3*16bit，输出为经过卷积计算后的对应的 5*5*1 的张量，要求计算结果正确。参考文献：论文 1、论文 2



(1) 7*7*16bit压缩成7*7*2 byte，对每一个byte(8bit) 构建一个查找表计算POPCNT

(2) 进一步优化，每次是可以取32bit进行访问，但是局限于POPCNT（ARM或者GPU提供非常快的POPCNT函数）