——国防科大2020年高性能评测与优化课程小组讨论

神经网络压缩与加速

组员：胡鼎煌 倪广森 肖欣怡
指导：龚春叶、甘新标、杨博

1 需求分析

为什么深度神经网络需要压缩和加速？

Table 1. The computation and parameters for state-of-art convolution neural networks

Method	Parameters			Computation		
	Size(M)	Conv(%)	Fc(%)	FLOPS(G)	Conv(%)	Fc(%)
AlexNet	61	3.8	96.2	0.72	91.9	8.1
VGG-S	103	6.3	93.7	2.6	96.3	3.7
VGG16	138	10.6	89.4	15.5	99.2	0.8
NIN	7.6	100	0	1.1	100	0
GoogLeNet	6.9	85.1	14.9	1.6	99.9	0.1
ResNet-18	5.6	100	0	1.8	100	0
ResNet-50	12.2	100	0	3.8	100	0
ResNet-101	21.2	100	0	7.6	100	0

- 深度学习迅速发展，神经网络层数越来越多，计算复杂度越来越高。
- 训练网络对硬件的要求也越来越高，大量的计算需要消耗大量电源、占用大量运行内存。

2

具体方法分析

2 网络剪枝

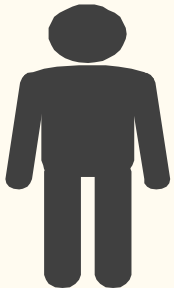
主要思想：将权重矩阵中相对“不重要”的权值剔除，然后再重新进行微调。

细粒度剪枝(Fine-grained Pruning)

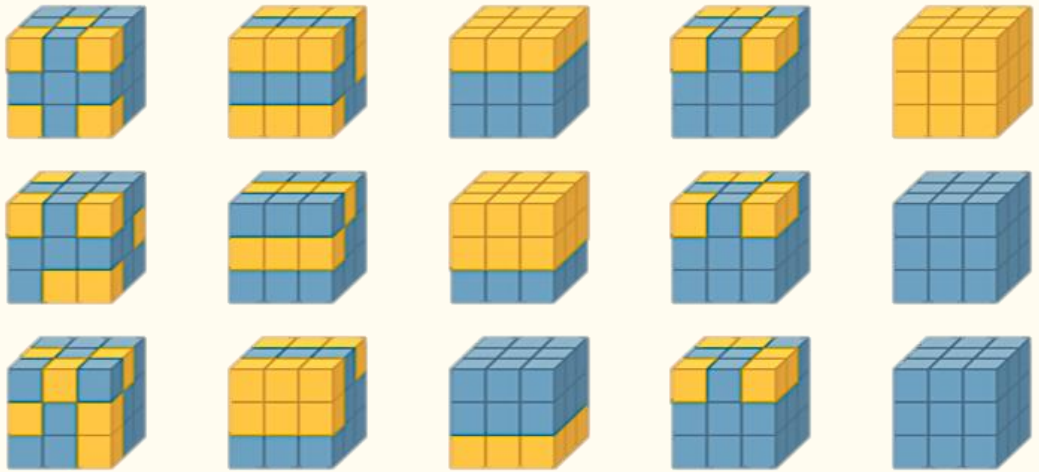
向量级和内核级剪枝(Vector-level and Kernel-level Pruning)

组剪枝(Group-level Pruning)

滤波器剪枝(Filter-level Pruning)



2 网络剪枝



Fine-grained Vector-level Kernel-level Group-level Filter-level

2 网络剪枝

网络剪枝方法ThiNet效果

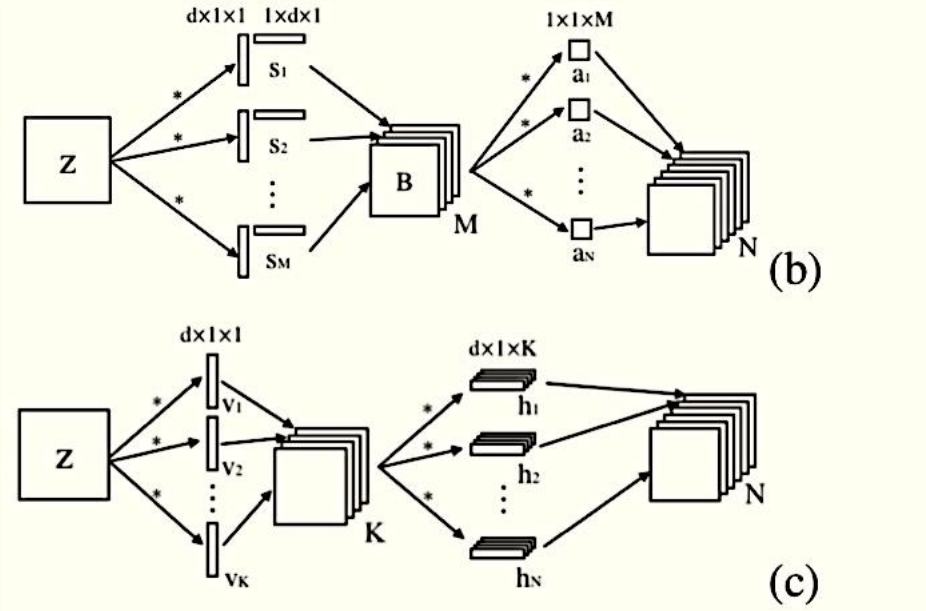
Table 1. Pruning results of VGG-16 on ImageNet using ThiNet. Here, M/B means million/billion (10^6/10^9), respectively; f./b. denotes the forward/backward timing in milliseconds tested on one M40 GPU with batch size 32.

Model	Top-1	Top-5	#Param.	#FLOPs[1]	f./b. (ms)
Original[2]	68.34%	88.44%	138.34M	30.94B	189.92/407.56
ThiNet-Conv	69.80%	89.53%	131.44M	9.58B	76.71/152.05
Train from scratch	67.00%	87.45%	131.44M	9.58B	76.71/152.05
ThiNet-GAP	67.34%	87.92%	8.32M	9.34B	71.73/145.51
ThiNet-Tiny	59.34%	81.97%	1.32M	2.01B	29.51/55.83
SqueezeNet[15]	57.67%	80.39%	1.24M	1.72B	37.30/68.62

3 低秩分解

- 低秩矩阵：其每行或每列都可以用其他的行或列线性表出，包含大量的冗余信息。利用这种冗余信息，可以对缺失数据进行恢复，也可以对数据进行特征提取。

Two-component Decomposition：将权重张量分为两部分，并且将卷积层替换为两个连续的层。



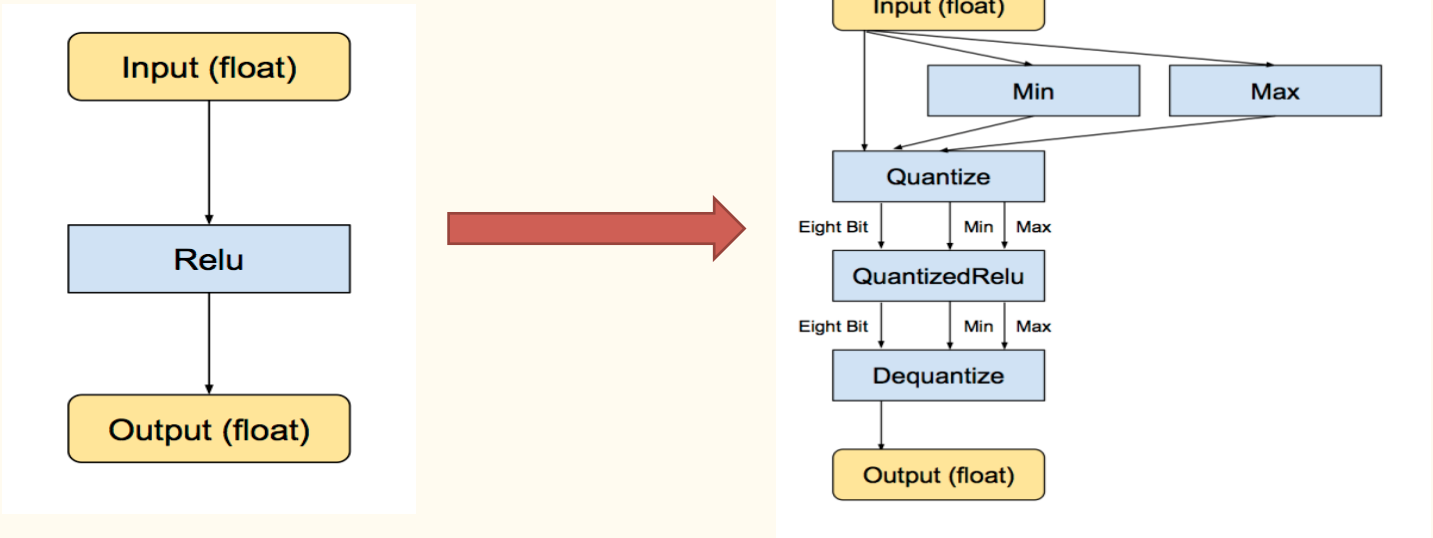
3 低秩分解

Three-component Decomposition：
可以通过两次连续的二元素分解来获得一种简单的三元素分解方法。

Four-component Decomposition：
从多个不同的方面进行分解，比如：输入、输出、空间维度。

4 网络量化

量化是许多压缩和加速应用程序的一种方法。对神经网络的量化，也就是将神经网络中大部分op的浮点权重值转换为定点整数表示，同时将op替换成可以执行类似的定点整数运算的op。



4 网络量化

（1）标量和矢量量化：

标量量化：整个动态范围被划分为若干小区间，每个小区间有一个代表值，落在该区间的信号值就用这个值代替。

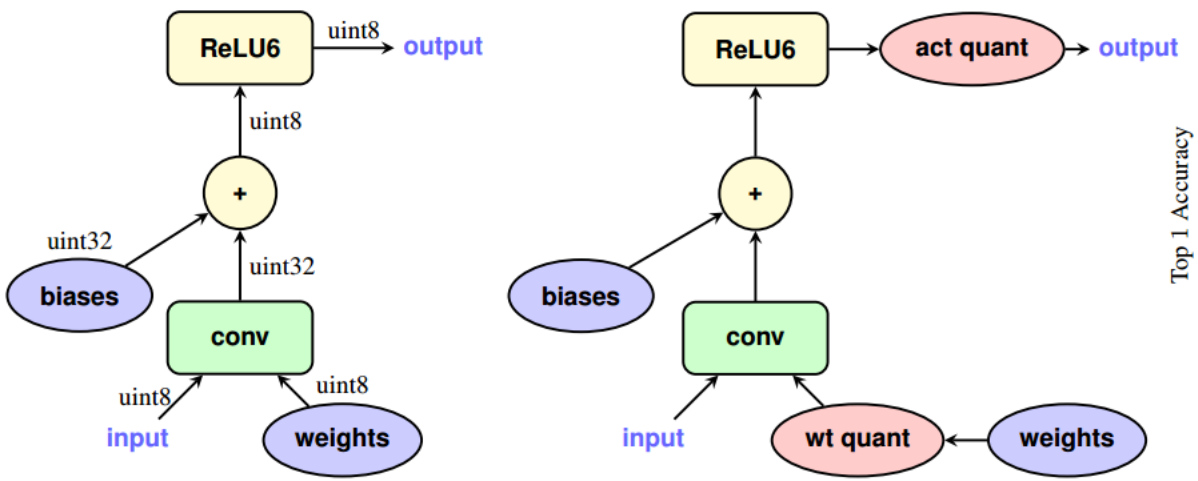
矢量量化：在标量量化基础上，对矢量进行量化，对每个区域内的矢量用一个矢量表示

（2）定点量化

对这个网络的权值进行一个定点的表示，有权重量化和激活量化，对网络不同部分进行定点量化的比较。

4 网络量化

CVPR谷歌推出的量化论文中Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference：



(a) Integer-arithmetic-only inference

(b) Training with simulated quantization

(c) ImageNet latency-vs-accuracy tradeoff

5 Teacher-student Network

师生网络：

使用一个教师网络来训练一个学生网络，学生网络可以有不同的网络结构。教师网络是一个大型的神经网络或神经网络的集合，而学生网络是一个紧凑、高效的神经网络。通过利用教师网络传送过来的信息，学生网络可以达到更高的准确率。

发展过程（性能逐渐增强）：

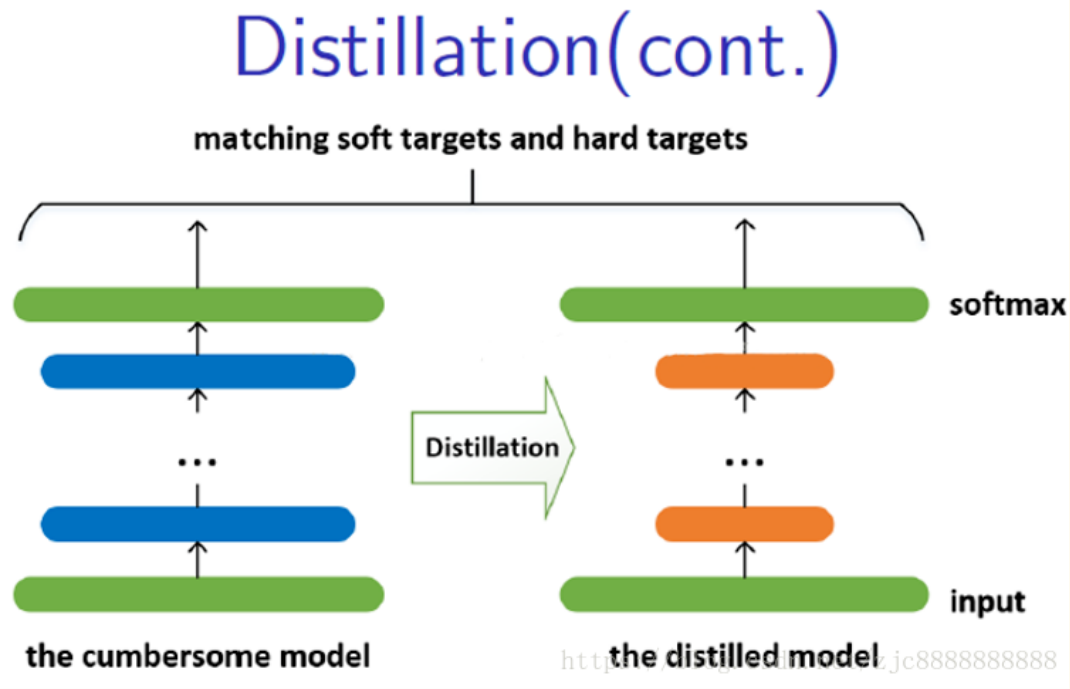
知识蒸馏(Knowledge Distillation) ->

FitNets->

模仿教师网络的注意力图来训练学生网络

5 师生网络

知识蒸馏：


Distillation(cont.)
matching soft targets and hard targets
softmax
Distillation
input
the cumbersome model
the distilled model

6 紧凑网络设计

紧凑网络设计：

网络加速和压缩的目标是优化给定深度神经网络的执行和存储框架。它的一个特性是网络架构没有改变。对网络加速和压缩的另一个并行研究是设计更高效但成本更低的网络体系结构本身。

常用方法：

- 去掉全连接层，使用全局平均池
- 分支(多组卷积)

6 紧凑网络设计

分组卷积即将输入的feature maps分成不同的组（沿channel维度进行分组），然后对不同的组分别进行卷积操作，即每一个卷积核至于输入的feature maps的其中一组进行连接，而普通的卷积操作是与所有的feature maps进行连接计算。**分组数k越多，卷积操作的总参数量和总计算量就越少（减少k倍）。然而分组卷积有一个致命的缺点就是不同分组的通道间减少了信息流通**，即输出的feature maps只考虑了输入特征的部分信息，因此在实际应用的时候会在分组卷积之后进行信息融合操作

6 紧凑网络设计

效果对比

名称	方法	效果
SqueezeNet	1×1 的卷积和分支策略	比 AlexNet 小 50 倍
MobileNet	分支的数量等于输入/输出通道的数量	比 VGG16 模型小 32 倍，快 27 倍
ShuffleNet	信道洗牌	比 AlexNet 快 13 倍

7 硬件加速器

硬件加速器：

近年来，越来越多的应用都属于嵌入式系统，需要在低能耗、轻重量的环境中提供高性能。使用基于CPU/GPU的解决方案已经不再合适，基于FPGA/ASIC的解决方案逐渐热门起来。

7 硬件加速器

通常，一个加速器由五部分组成:数据缓冲区、参数缓冲区、处理元素、全局控制器和片外传输管理器

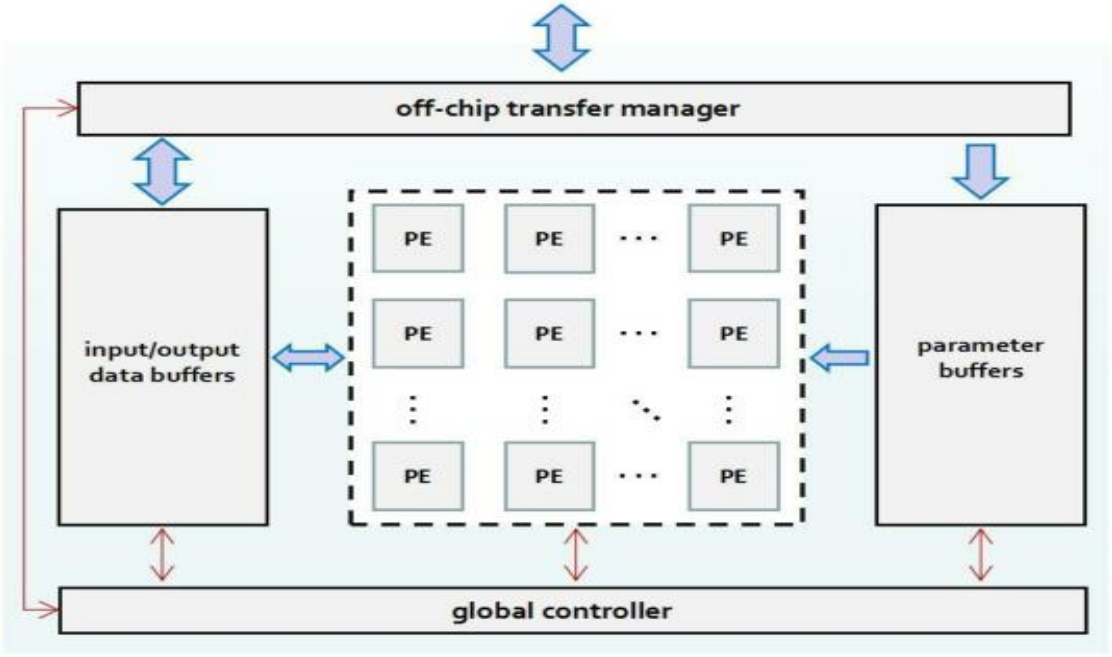


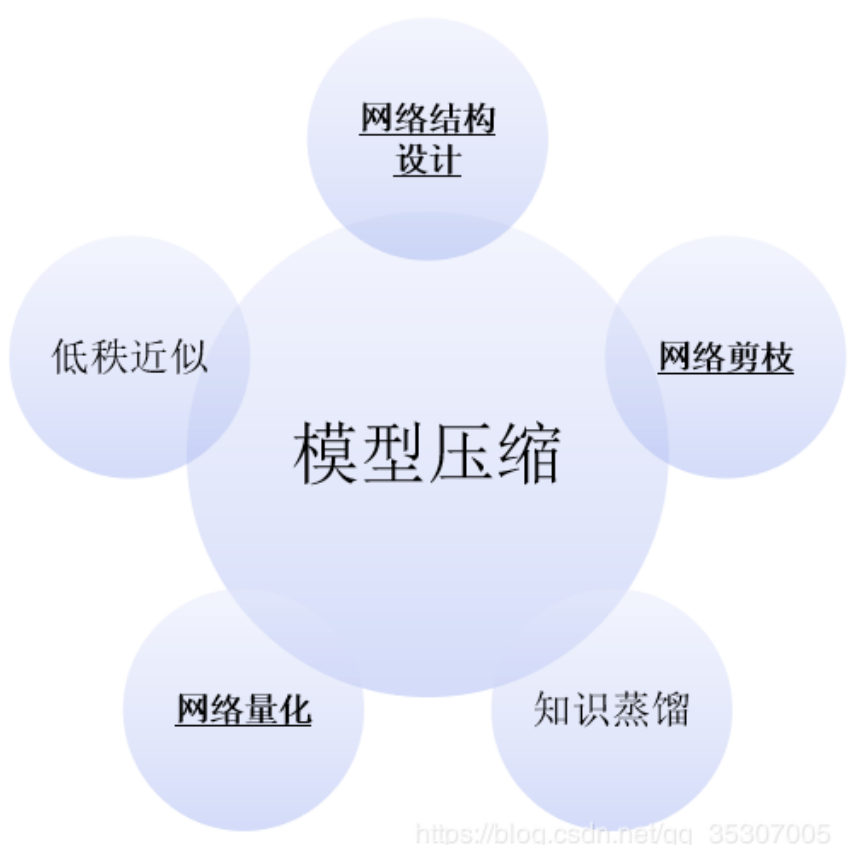
Figure 3. General architecture of an accelerator on dedicated hard-

7 硬件加速器

硬件加速的思路：

- 提高吞吐量
  - 循环优化是加速器设计中最常采用的技术之一，包括循环平铺、循环展开、循环交换等。
- 降低能耗
  - 利用稀疏性来降低能耗
  - 减少片外传输
- 自动化
  - 自动将深度神经网络映射到硬件上

8 Conclusion



软件
模型
压缩
方法

8 Conclusion

在这个领域未来可能的研究方向有：

- 无微调或不监督压缩
- 自适应压缩
- 为目标检测做网络加速
- 软硬件协同设计