model acceleration学习

模型的运行速度一般是看推理时模型的运行速度，因此需要尽量的加快推理时的速度，如常见的dropout，其将在推理时相乘的比例移到训练时相除，减少推理时的运算量。

Batch Normalization和Convolution（2d）融合

对于二维卷积，其卷积核参数的shape为[output\_channel, input\_channel, kernel\_size, kernel\_size]。在前面学过Batch Normalization是对某一层（卷积或全连接）的输出数据的每一个Channel中的所有数据进行归一化，即如下面公式所示：

将其详细写出来即为如下公式：

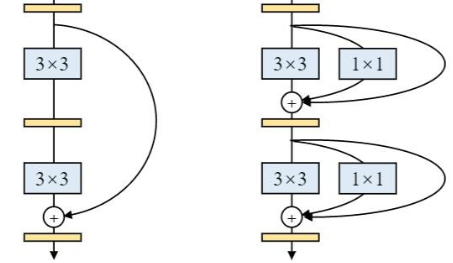
其中为第个Channel上的数据，由于训练好之后，所有的均为常数，且是由卷积的来，而卷积只是卷积核与特征图加权求和，因此可以直接对卷积核中的数值进行相乘，并对其偏置进行偏移，BN与卷积即可融合。由于在卷积层后加BN时，卷积核的偏置一般不设置（一直为0），因此融合后的卷积核参数如下：

其中为第个卷积核的权重。至此就可以使用新的卷积核参数只进行一次卷积，就把卷积和BN一起计算，实现了卷积与BN的融合。

3×3卷积、1×1卷积、shortcut融合（结构重参数化）

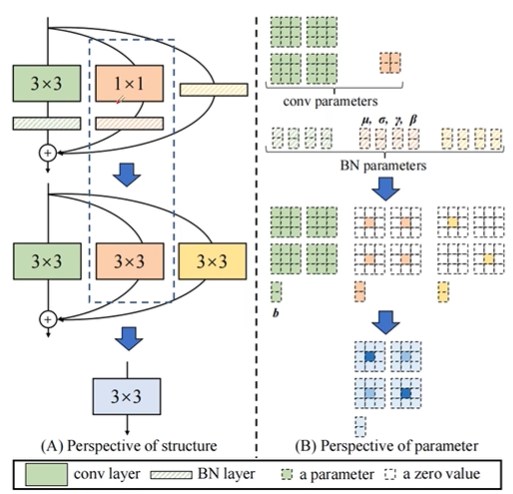
对于多分支结构，根据现有的一些经验来看，并行多个分支一般能够增加模型的表征能力。同时多分支结构能构增加一个神经单元中感受野变化，因而具有更强的多尺度表达能力。

目前比较常见的多分支结构如下图所示：



左图为ResNet的多分支结构，右图为RapVgg的多分支结构

对于RapVgg这样的多分支结构，是可以在推理时进行结构重参数化（即融合），其示意图如下图所示，将3×3卷积、1×1卷积、shortcut和各自的BN层一同融合。



关于卷积层与BN层的融合在上文已介绍过，之后将介绍如何融合3×3卷积、1×1卷积、shortcut为一个3×3卷积。

首先对于3×3卷积不需要做任何改变，对于1×1卷积，其卷积核参数的shape为[output\_channel, input\_channel, 1, 1]，由于要将其融合为3×3卷积，因此可以对shape进行扩展，将其扩展为[output\_channel, input\_channel, 3, 3]，即最中间那个值为原1×1卷积核参数中的值，其余值均为0。这样在变化为3×3卷积（padding=1）进行加权求和与原1×1卷积的结果是一样的。

对于shortcut连接，可以将其视为卷积核参数固定的1×1卷积，即期shape为[input \_channel, input\_channel, 1, 1]。对于第i个卷积核，其shape为[, input\_channel, 1, 1]，那么对于第i个卷积核的第i个通道，其值为1，其余值均为0。这样卷积的结果相当于恒等映射。同样的将其扩展为3×3卷积，这样就得到了三个3×3卷积。

由卷积的分配律，我们可以将三个3×3卷积融合为一个3×3卷积，即对三个卷积核的参数进行相加，偏置也进行相加。即如下公式所示：

其中表示卷积运算。