

# A Comprehensive Introduction to Different Types of Convolutions in Deep Learning

JX-Ma

2023/10/6

## 1 卷积与互相关

深度学习中的卷积本质上是信号/图像处理中的互相关。

区别

- 卷积被定义为两个函数的乘积在一个函数反转并移位后的积分，卷积核反转并沿着水平轴滑动，与另一个函数交集的面积就是该特定位置的卷积值。
- 互相关中卷积核没有反转，两个函数的之间的交集面积就是互相关。

在深度学习中，卷积中的卷积核不是颠倒的。严格来说，这是互相关。但在深度学习中，习惯上将其称为卷积。因为卷积核的权重是在训练期间学习的。如果上例中的反转函数  $g$  是正确的函数，那么训练后学习的卷积核将看起来像反转函数  $g$ 。因此，不需要像真正的卷积那样在训练之前先反转卷积核。

## 2 深度学习中的卷积（单通道，多通道）

卷积的目的：从输入中提取有用的特征。

每种类型的卷积核有助于从输入图像中提取不同的方面或特征，例如水平/垂直/对角线边缘。

在卷积神经网络中，使用卷积核通过卷积提取不同的特征，卷积核的权重在训练期间自动学习。然后将所有这些提取的特征进行“组合”以做出决策。

卷积还考虑了像素的空间关系。这些可能非常有帮助，尤其是在许多计算机视觉任务中，因为这些任务通常涉及识别某些组件与其他组件具有一定空间关系的对象（例如，狗的身体通常与头、四条腿和尾巴相连）。

### 2.1 单通道卷积

卷积核的只有一个通道，与通道为 1 的图像卷积后生成一个矩阵。

### 2.2 多通道卷积

多通道卷积的例子是 RGB 图像，通道维度为 3，另一个例子就是卷积神经网络中的层。

层，通道，特征图，卷积核，内核

层次角度：层和卷积核概念处于同一级别，通道和内核处于下一级，通道和特征图是同一件事。

一个层中可以含有多个通道，如果输入的是 RGB 图像，输入层就有 3 个通道，通道通常来描述层的结构，内核用来描述卷积核的结构，“内核”是指权重的二维数组。术语“卷积核”是指堆叠在一起的多个内核的 3D 结构。对于 2D 卷积核，卷积核与内核相同。但对于 3D 卷积核和深度学习中的大多数卷积来说，卷积核是内核的集合。每个内核都是独一无二的，强调输入通道的不同方面。

多通道卷积：例如输入层是  $5 \times 5 \times 3$ ，卷积核（卷积核） $3 \times 3 \times 3$ ，卷积核中的每个内核分别对输入层中的三个通道做卷积运算，最后得到 3 个  $3 \times 3$  的通道，在这三个通道按元素相加得到结果为  $3 \times 3 \times 1$  的通道。

也可以将此过程比作在输入层滑动 3D 过滤矩阵，输入层的通道数必须与 3D 过滤矩阵中的内核数相等。只在输入层的高和宽两个方向滑动，滑动完的结果按照通道方向堆叠在一起最后形成一层输出层。

### 3 3D 卷积

当输入层的通道数大于卷积核中的内核数时，就可以将卷积核在输入层中的 3 个维度上进行滑动，这个时候就能得到具有 3 个维度的卷积层，这种卷积被称为 3D 卷积。

### 4 $1 \times 1$ 卷积

输入层为  $H \times W \times D$ ，卷积核尺寸  $1 \times 1 \times D$ ，卷积核变为  $H \times W \times 1$ 。（处理卷积核中的深度）

$1 \times 1$  卷积的优点：

- 降维
- 高效的低维嵌入
- 卷积后应用非线性运算

### 5 卷积运算

步长 s, padding x

### 6 转置卷积（反卷积）

反卷积：与正常卷积相反方向的转换，如生成高分辨率图像，将低维的特征图映射到高维空间。

如：在  $2 \times 2$  输入层上应用具有  $3 \times 3$  内核的转置卷积，使用了单位步长填充了  $2 \times 2$  零边框，采样输出后的结构为  $4 \times 4$ 。

将转置卷积应用于相同的  $2 \times 5 \times 2$  输入（在输入之间插入 1 个零），并使用单位步长填充  $2 \times 5 \times 2$  零边框。现在输出的大小为  $5 \times 5$ 。

转置卷积：卷积操作使用大的图像通过卷积核压缩为一个小的图像，例如一个  $4 \times 4$  的输入，通过  $3 \times 3$  的卷积核生成了  $2 \times 2$  的输入。运算过程将  $4 \times 4$  的矩阵转化为  $16 \times 1$  的矩阵， $3 \times 3$  的卷积核变为

$4 \times 16$  的矩阵，将变换后的矩阵做卷积运算最后得到  $4 \times 1$  的输出矩阵，输出矩阵在转化为  $2 \times 2$  的输出。

反过来，将变换后的卷积核矩阵转置变为  $16 \times 4$  的矩阵，乘以输出矩阵  $4 \times 1$ ，最后得到  $16 \times 1$  转换后的输入矩阵。这就是转置卷积的由来，并且也实现了小图像变为大图像的目标。

## 6.1 棋盘效应

人们在使用转置卷积时观察到的一种令人不快的行为是所谓的棋盘伪影。

棋盘伪像是由转置卷积的“不均匀重叠”造成的。这种重叠在某些地方比其他地方增添了更多的隐喻色彩。

在应用转置卷积时，可以做两件事来减少此类伪影。首先，确保您使用的卷积核大小除以步幅，避免重叠问题。其次，可以使用步幅 = 1 的转置卷积，这有助于减少棋盘效应。

## 7 扩张卷积

扩张卷积通过在内核元素之间插入空格来“膨胀”内核。

扩张卷积用于在不增加内核大小的情况下廉价地增加输出单元的感受野，这在多个扩张卷积依次堆叠时尤其有效。

## 8 可分离卷积

可分离卷积用于一些神经网络架构，可以执行空间可分离卷积（空间可分离卷积）或深度可分离卷积（深度可分离卷积）。

### 8.1 空间可分离卷积

将卷积核的分解为两个低维的向量相乘，这样的好处有首先需要的数据比未分离前少，而且运算次数也比未分离少。

具体运算次数： $N \times N$  的输入和  $M \times M$  的卷积核，未分离的运算次数  $(N - 2)^2 \times M^2$ ，分离后： $(2N - 2) \times (N - 2) \times M$

当  $N$  远大于  $M$  时，空间可分离卷积是标准卷积的  $\frac{2}{M}$  虽然空间可分离卷积节省了成本，但在深度学习中很少使用。主要原因之一是并非所有内核都可以分为两个更小的内核。如果我们用空间可分离卷积替换所有传统卷积，

我们就会限制自己在训练期间搜索所有可能的内核。训练结果可能不是最优的。卷积核矩阵的秩为 1 的时候才可以将矩阵分解。

### 8.2 深度可分离卷积

对于输入张量  $H \times W \times D$ ，我们用卷积核大小  $h \times h \times D$  个数为  $N_c$  进行卷积，在设置步长为 1，不填充边距的情况下，进行卷积生成  $(H - h + 1) \times (H - h + 1) \times N_c$  输出张量

采用 2D 卷积, 直接用  $N_c$  个  $h \times h \times D$  对输入张量进行卷积, 此时总乘量

$$N_c \times h \times h \times D \times (H - h + 1) \times (W - h + 1)$$

采用深度分离的卷积, 将卷积核分解为  $N_c$  个  $1 \times 1 \times D$  和  $D$  个  $h \times h \times 1$  矩阵, 然后让输入先和后者卷积在和前者卷积, 得出输出。此时总乘法:  $(h \times h + N_c) \times \text{深度} \times (\text{高} - \text{高} + 1) \times (\text{宽} - \text{高} + 1)$

深度分离的卷积对于大多数现代架构来说大大减少了运算次数。缺点是深度可分离卷积减少了卷积中的参数数量。因此, 对于小模型, 如果将 2D 卷积替换为深度可分离卷积, 则模型容量可能会显著降低。

## 9 扁平化卷积

如果卷积核中的秩为 1, 就可以将 3 维的卷积核分解为 3 个一维的卷积核做叉乘。但这是一个强条件, 并且标准卷积核的内在等级在实践中高于一。深度网络中的学习卷积核具有分布式特征值, 直接将分离应用于卷积核会导致显着的信息丢失。

## 10 分组卷积

提出: 是允许在内存有限的两个 GPU 上进行网络训练。

将卷积核按照深度一半分开, 各自与对应位置的输入张量进行卷积, 讲得出的结果组合起来就是最终的输出张量。

优点:

- 第一个优势是高效的培训。由于卷积被分为多个路径, 每个路径可以由不同的 GPU 单独处理。此过程允许以并行方式在多个 GPU 上进行模型训练。与使用一个 GPU 进行所有训练相比, 这种多 GPU 上的模型并行化允许每一步将更多图像输入到网络中。模型并行化被认为比数据并行化更好。后者将数据集分成批次, 然后我们对每个批次进行训练。然而, 当批量大小变得太小时, 我们本质上是在进行随机而不是批量梯度下降。这会导致收敛速度变慢, 有时甚至较差。
- 第二个优点是模型更加高效, 即模型参数随着卷积核组数量的增加而减少。
- 第三个优点是分组卷积可以提供比标称二维卷积更好的模型。

## 11 混洗分组卷积

混洗分组卷积涉及分组卷积和通道混洗。

通道混洗的想法是我们想要混合来自不同卷积核组的信息。

## 12 逐点分组卷积

逐点分组卷积, 顾名思义, 对  $1 \times 1$  卷积执行分组运算。该操作与分组卷积相同, 只有一项修改 - 在  $1 \times 1$  卷积核而不是  $N \times N$  卷积核 ( $N > 1$ ) 上执行。

## 13 心得

这篇文章学到了在卷积学习中不同类型卷积的理解. 其中包括一些卷积的概念, 和不同卷积的介绍, 以及如何去利用不同的卷积去优化算法。

首先是卷积和互相关的概念, 深度学习中的卷积本质上是信号/图像处理中的互相关。他们之间的区别是: 卷积被定义为两个函数的乘积在一个函数反转并移位后的积分, 卷积核反转并沿着水平轴滑动, 与另一个函数交集的面积就是该特定位置的卷积值。而互相关中的卷积核函数没有翻转, 他们的交集就是互相关。严格来说深度学习的卷积就是互相关, 因为在深度学习中的卷积核的权重是在训练期间学习的, 学习的卷积核将看起来像反转函数  $g$ 。因此, 不需要像真正的卷积那样在训练之前先反转卷积核。

第二点是深度学习中的卷积, 首先卷积的目的是从输入中提取有用的特征。在卷积神经网络中, 使用卷积核通过卷积提取不同的特征, 卷积核的权重在训练期间自动学习。然后将所有这些提取的特征进行“组合”以做出决策。接下来学习了单通道卷积和多通道卷积, 他们之间的区别是输入张量中第三维度大小的区别, 拿图片来举例子, 灰色图片颜色通道为 1, 此时使用的是单通道卷积, 而 RGB 图像表示色彩的有 3 个变量, 也就是说它的通道维度为 3, 这个时候就可以进行多通道卷积。最后在这点中, 了解了层, 通道, 特征图, 卷积核以及内核之间的区别。

第三点是 3D 卷积, 我的理解是就是卷积核作为窗口, 在输入张量中三个维度上进行滑动运算。也就是当输入层的通道数大于卷积核中的内核数时, 就可以将卷积核在输入层中的 3 个维度上进行滑动, 这个时候就能得到具有 3 个维度的卷积层, 这种卷积被称为 3D 卷积。第四个点是  $1 \times 1$  卷积, 使用宽度和高度为 1 深度和输入层深度相同的卷积核对输入层做卷积得到一个  $H \times W \times D$  的矩阵, 这样做的好处首先就是降维, 上周读过的文章中就提到过降维可以很好避免维度诅咒这个问题, 然后就是卷积后的矩阵可以更好的应用非线性计算。

第五个点是卷积运算, 这里介绍了卷积运算的一些概念, 这些在第二周的任务了解卷积运算中也学到过, 因此这次就相当于复习了一下。

第六个点是转置卷积也叫做反转积, 反转积的定义是与正常卷积相反方向的转换, 如生成高分辨率图像, 将低维的特征图映射到高维空间, 正常卷积的过程是将输入层的数据进行压缩。在  $2 \times 2$  输入层上应用具有  $3 \times 3$  内核的转置卷积, 使用了单位步长填充了  $2 \times 2$  零边框, 采样输出后的结构为  $4 \times 4$ 。原始的  $2 \times 2$  矩阵就变成了  $4 \times 4$  的矩阵。至于会叫转置卷积的原因是将卷积核变换的矩阵转置, 在和输出张量做卷积最后会得到输入张量。在进行转置卷积时可能遇到的情况是棋盘伪影。棋盘伪影是由转置卷积的“不均匀重叠”造成的。这种重叠在某些地方比其他地方增添了更多的隐喻色彩。在应用转置卷积时, 可以做两件事来减少此类伪影。首先, 确保您使用的卷积核大小除以步幅, 避免重叠问题。其次, 可以使用步幅 = 1 的转置卷积, 这有助于减少棋盘效应。

第七个点是扩张卷积, 扩张卷积就是通过在内核元素之间插入空格来“膨胀”内核。它的作用在于在不增加内核大小的情况下廉价地增加输出单元的感受野, 这在多个扩张卷积依次堆叠时尤其有效。

第八个点是可分离卷积可分离卷积可以分为空间可分离卷积和深度可分离卷积, 空间可分离卷积指的是可以将卷积核中的内核分解为两个一维的向量叉乘, 卷积运算时, 依次与这两个一维向量做卷积, 这样可以使运算次数大大减少, 但是使用空间可分离时内核矩阵需要满足矩阵的秩为 1。另一个深度可分离卷积深度可分离卷积对卷积核的处理是将卷积核按照深度维度分解, 例如  $N_c$  个  $h \times h \times D$  的卷积核可以分解为  $N_c$  个  $1 \times 1 \times D$  和  $D$  个  $h \times h \times 1$  矩阵, 在依次进行卷积。这种方法对比 2D 卷积的话也

能减少很多运算次数，但是深度可分离卷积也存在缺点，缺点是深度可分离卷积减少了卷积中的参数数量。因此，对于小模型，如果将 2D 卷积替换为深度可分离卷积，则模型容量可能会显著降低

第九个点是扁平化卷积，就是将 3 维的卷积核分解为 3 个一维的向量做叉乘，但是需要卷积核的秩为 1，而且深度网络中的学习卷积核具有分布式特征值，直接将分离应用于卷积核会导致显著的信息丢失。

最后三个分别是分组卷积，混洗分组卷积和逐点分组卷积，分组卷积就是将卷积核按照 GPU 数量分开，各自与对应位置的输入张量进行卷积，讲得出的结果组合起来就是最终的输出张量。分组卷积可以提供比标称二维卷积更好的模型。混洗分组卷积涉及分组卷积和通道混洗。通道混洗的想法是我们想要混合来自不同卷积核组的信息。逐点分组卷积，顾名思义，对  $1 \times 1$  卷积执行分组运算。该操作与分组卷积相同，只有一项修改 - 在  $1 \times 1$  卷积核而不是  $N \times N$  卷积核 ( $N > 1$ ) 上执行。