# 第九章 卷积网络

## 1. 卷积网络基本概念

**卷积网络**是一种专门用来处理具有类似网络结构的数据的神经网络。例如时间序列数据和图像数据。卷积网络指至少在网络的一层中使用卷积运算来替代一般的矩阵乘法运算的神经网络。

## 2. 卷积运算

在通常形式中，卷积是对两个实变函数的一种数学运算。

**卷积**（convolution）运算：

设$x$与$w$是 R 上的两个可积函数，对于积分：

$$s(t) = \int x(a)w(t-a)da$$

称该运算为卷积。卷积运算常用星号表示：

$$s(t) = (x * w)(t)$$

在卷积网络的术语中，卷积的第一个参数（函数$x$）通常叫做**输入**（input），第二个参数（函数$w$）叫做**核函数**（kernel function）。输出有时被称为**特征映射**（feature map）。

定义离散形式的卷积运算：

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a)$$

在机器学习的应用中，输入通常是多维数组的数据，而核通常是由学习算法优化得到的多维数组的参数。我们把这些多维数组叫做张量。因为在输入与核中的每一个元素都必须明确地分开存储，我们通常假设在存储量数值的有限点集以外，这些函数的值都为零，意味着在实际操作中，我们可以通过对有限个数组元素的求和来实现无限求和。

可以在多个维度上进行卷积运算：

例如在二维空间中，设输入为$I$，二维核函数为$K$，二维卷积运算：

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m,j-n)$$

卷积是可以交换（commutative）的，上式可等价写作：

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i-m,j-n)K(m,n)$$

卷积运算可交换性出现是因为将核相对输入进行了翻转（flip）。我们将核翻转的唯一目

的是实现可交换性。许多神经网络库会实现一个相关的函数，称为**互相关函数**（cross-correlation），和卷积运算几乎一样但并没有对核进行翻转：

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(i+m, j+n)K(m,n)$$

在机器学习中，学习算法会在核合适的位置学得恰当的值，所以一个基于核翻转的卷积运算的学习算法所学习到的核，是对未进行翻转的算法学得的核的翻转。

在机器学习中，卷积经常和其他函数一起使用，无论卷积运算是否对它的核进行翻转，这些函数的组合通常是不可交换的。

例：一个在 2 维张量上的卷积运算（没有对核进行翻转）：

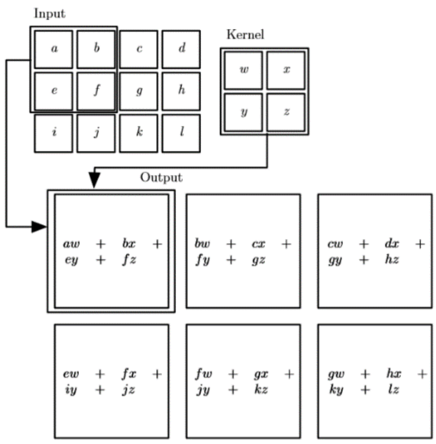



图 9.1: 一个 2 维卷积的例子（没有对核进行翻转）。我们限制只对核完全处在图像中的位置进行输出，在一些上下文中称为“有效”卷积。我们用画有箭头的盒子来说明输出张量的左上角元素是如何通过对输入张量相应的左上角区域应用核进行卷积得到的。

离散卷积可以看作矩阵的乘法，然而这个矩阵的一些元素被限制为必须和另外一些元素相等。例如单变量离散卷积中，矩阵每一行元素与上一行对应平移一个单位的元素相同。这种矩阵叫做 **Toeplitz 矩阵**（Toeplitz matrix）。对于二维情况，卷积对应着一个**双重分块循环矩阵**（doubly block circulant matrix）除了这些元素相等的限制以外，卷积通常对应着一个非常稀疏的矩阵（一个几乎所有元素都为 0 的矩阵），因为核的大小通常要远小于输入图像的大小。

## 3. 动机

卷积运算通过三个重要的思想来帮助改进机器学习系统：

**稀疏交互（sparse interactions）**

**参数共享（parameter sharing）**

**等变表示（equivariant representations）**

**稀疏交互**（sparse interactions）（也叫做稀疏连接（sparse connectivity）或者稀疏权重（sparse weights））的特征。

这是使核的大小远小于输入的大小来达到的。这意味着我们需要存储的参数更少，不仅

减少了模型的存储需求，而且提高了它的统计效率。
例：稀疏连接：

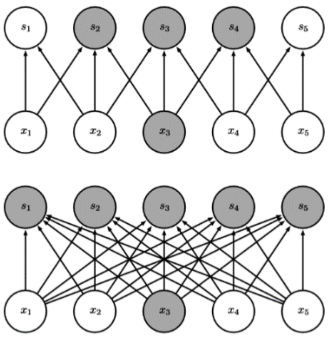


图 9.2: 稀疏连接，对每幅图从下往上看。我们强调了一个输入单元 $x_3$ 以及在 $s$ 中受该单元影响的输出单元。*(上)* 当 $s$ 是由核宽度为 3 的卷积产生时，只有三个输出受到 $x$ 的影响[2]。*(下)* 当 $s$ 是由矩阵乘法产生时，连接不再是稀疏的，所以所有的输出都会受到 $x_3$ 的影响。

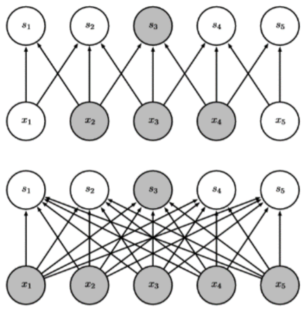
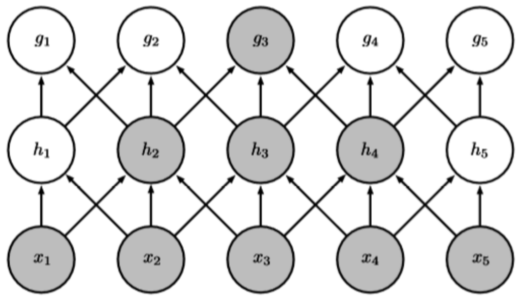


图 9.3: 稀疏连接，对每幅图从上往下看。我们强调了一个输出单元 $s_3$ 以及 $x$ 中影响该单元的输入单元。这些单元被称为 $s_3$ 的 **接受域**（receptive field）[3]。*(上)* 当 $s$ 是由核宽度为 3 的卷积产生时，只有三个输入影响 $s_3$。*(下)* 当 $s$ 是由矩阵乘法产生时，连接不再是稀疏的，所以所有的输入都会影响 $s_3$。

在很多实际应用中著需要保持 k 比 m 小几个数量级就能在机器学习中取得好的表现。
在深度卷积网络中，处在网络深层的单元坑内与绝大部分输入是间接交互的。
例：网络可以通过只描述稀疏交互的基石来高效地描述多个变量的复杂交互。

> 
> 
>
> 图 9.4: 处于卷积网络更深的层中的单元，它们的接受域要比处在浅层的单元的接受域更大。如果网络还包含类似步幅卷积（图 9.12）或者池化（第 9.3 节）之类的结构特征，这种效应会加强。这意味着在卷积网络中尽管直接连接都是很稀疏的，但处在更深的层中的单元可以间接地连接到全部或者大部分输入图像。

**参数共享**（parameter sharing）是指一个模型的多个函数中使用相同的参数。
即一个网络含有绑定的权重（tied weights）。在卷积神经网络中核的每一个元素都作用在输入的每一位置上。卷积运算中的参数共享保证了我们只需要学习一个参数集合，而不是对于每一位置都需要学习一个单独的参数集合。
例：参数共享的实现、提高边缘检测的效率：

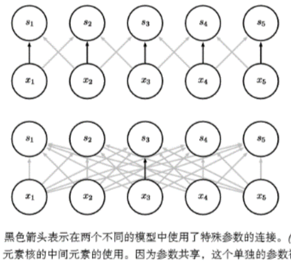


图 9.5: 参数共享。黑色箭头表示在两个不同的模型中使用了特殊参数的连接。*(上)* 黑色箭头表示在卷积模型中对 3 元素核的中间元素的使用。因为参数共享，这个单独的参数被用于所有的输入位置。*(下)* 这个单独的黑色箭头表示在全连接模型中对权重矩阵的中间元素的使用。这个模型没有使用参数共享，所以参数只使用了一次。

图 9.6: 边缘检测的效率。右边的图像是通过先获得原始图像中的每个像素，然后减去左边相邻像素的值而形成的。这个操作给出了输入图像中所有垂直方向上的边缘的强度，对目标检测来说是有用的。两个图像的高度均为 280 个像素。输入图像的宽度为 320 个像素，而输出图像的宽度为 319 个像素。这个变换可以通过包含两个元素的卷积核来描述，使用卷积需要 $319 \times 280 \times 3 = 267,960$ 次浮点运算（每个输出像素需要两次乘法和一次加法）。为了用矩阵乘法描述相同的变换，需要一个包含 $320 \times 280 \times 319 \times 280$ 个或者说超过 80 亿个元素的矩阵，这使得卷积对于表示这种变换更有效 40 亿倍。直接运行矩阵乘法的算法将执行超过 160 亿次浮点运算，这使得卷积在计算上大约有 60,000 倍的效率。当然，矩阵的大多数元素将为零。如果我们只存储矩阵的非零元，则矩阵乘法和卷积都需要相同数量的浮点运算来计算。矩阵仍然需要包含 $2 \times 319 \times 280 = 178,640$ 个元素。将小的局部区域上的相同线性变换应用到整个输入上，卷积是描述这种变换的极其有效的方法。照片来源：Paula Goodfellow。

对于卷积，参数共享的特殊形式使得神经网络层具有对平移**等变**（equivariance）的性质。如果一个函数满足输入改变，输出也以同样的方式改变这一性质，就说它是等变的。特别地，若函数$f(x)$与$g(x)$满足$f(g(x)) = g(f(x))$就说$f(x)$对$g(x)$具有等变性。
对于卷积来说，如果令 g 是输入的任意平移函数，则卷积函数对 g 具有等变性。
卷积对其他的一些变换并不是天然等变的，例如对于图像的放缩或旋转变换，需要其他的机制来处理这些变换。

## 4. 池化

卷积网络一个典型层包含三级。第一级，并行地计算多个卷积产生一组线性激活响应。第二级，每一个线性激活响应将会通过一个非线性的激活函数，例如整流线性激活函数，这级有时也被称为**探测级**（detector stage）。第三级中，使用**池化函数**（pooling function）来进一步调整这一层的输出。

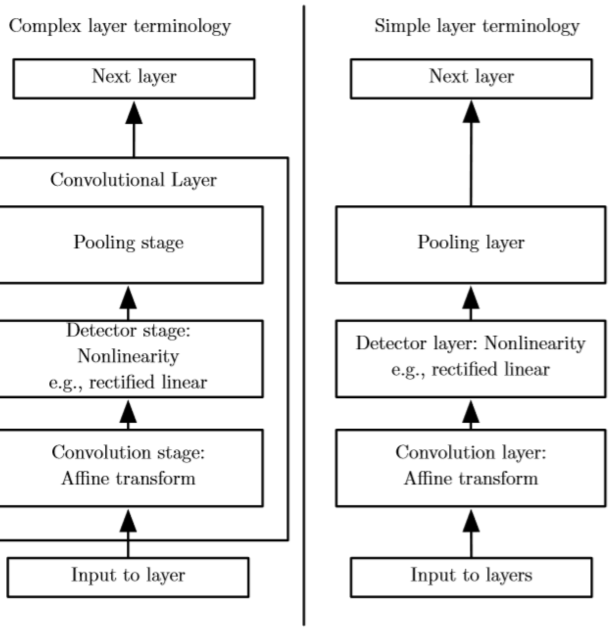



图 9.7: 一个典型卷积神经网络层的组件。有两组常用的术语用于描述这些层。*(左)* 在这组术语中，卷积网络被视为少量相对复杂的层，每层具有许多 "级"。在这组术语中，核张量与网络层之间存在一一对应关系。在本书中，我们通常使用这组术语。*(右)* 在这组术语中，卷积网络被视为更多数量的简单层；每一个处理步骤都被认为是一个独立的层。这意味着不是每一 "层" 都有参数。

池化函数使用某一位置的相邻输出的总体统计特征来代替网络在该位置的输出。**最大池化**（max pooling）函数给出相邻矩形区域内的最大值。其他常用的池化函数包括相邻矩形区域内的平均值、$L^2$范数以及基于据中心像素距离的加权平均函数。
无论怎样的池化函数，当输入做少量平移时，池化能够帮助输入的表示近似**不变**（invariant）。(即平移不变性，当输入进行少量平移时，经过池化函数后大多数输出并不会发生改变)。局部平移不变性是一个很有用的性质，尤其是当我们关心某个特征是否出现而不关心它出现的具体位置时。

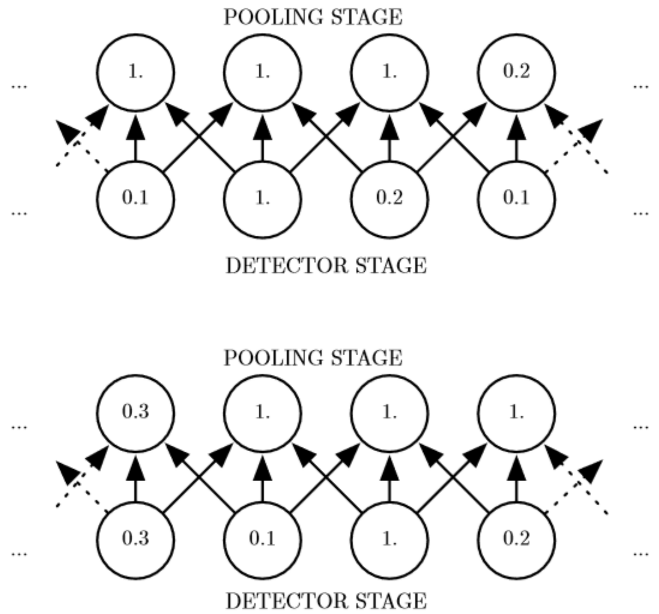



图 9.8: 最大池化引入了不变性。*(上)* 卷积层中间输出的视图。下面一行显示非线性的输出。上面一行显示最大池化的输出，每个池的宽度为三个像素并且池化区域的步幅为一个像素。*(下)* 相同网络的视图，不过对输入右移了一个像素。下面一行的所有值都发生了改变，但上面一行只有一半的值发生了改变，这是因为最大池化单元只对周围的最大值比较敏感，而不是对精确的位置。

使用池化可以看作是增加了一个无限强的先验：这一层学得的函数必须具有对少量平移的不变性。当这个假设成立时，池化可以极大地提高网络的统计效率。
对空间区域进行池化产生了平移不变性，但当我们对分离参数的卷积的输出进行池化时，特征能够学得应该对于那种变换具有不变性。

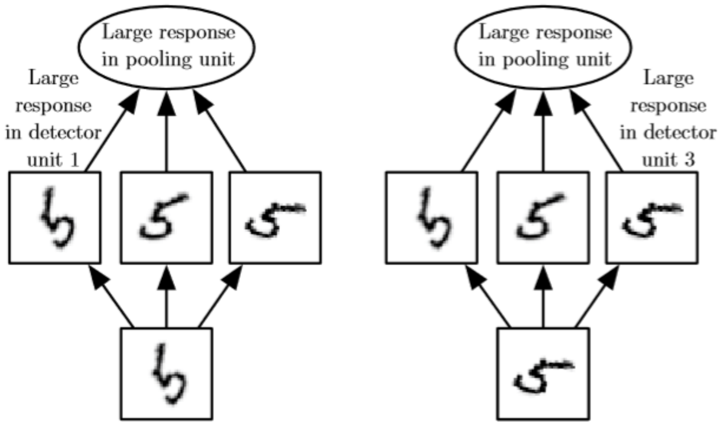



图 9.9: 学习不变性的示例。使用分离的参数学得多个特征，再使用池化单元进行池化，可以学得对输入的某些变换的不变性。这里我们展示了用三个学得的过滤器和一个最大池化单元可以学得对旋转变换的不变性。这三个过滤器都旨在检测手写的数字 5。每个过滤器尝试匹配稍微不同方向的 5。当输入中出现 5 时，相应的过滤器会匹配它并且在探测单元中引起大的激活。然后，无论哪个探测单元被激活，最大池化单元都具有大的激活。我们在这里演示了网络如何处理两个不同的输入，这导致两个不同的探测单元被激活，然而对池化单元的影响大致相同。这个原则在 maxout 网络 (Goodfellow *et al.*, 2013b) 和其他卷积网络中更有影响。空间位置上的最大池化对于平移是天然不变的；这种多通道方法只在学习其他变换时是必要的。

池化综合了全部邻居的反馈，使得池化单元可以小于探测单元，我们可以通过综合池化区域的 k 个像素的统计特征，而不是单个像素来实现。

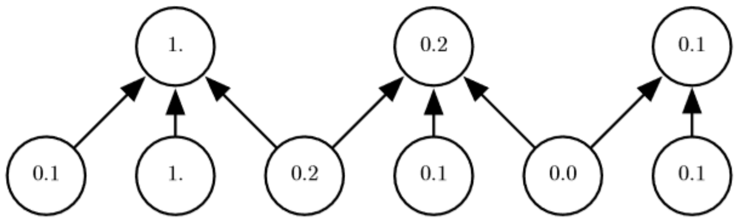



图 9.10: 带有降采样的池化。这里我们使用最大池化，池的宽度为三并且池之间的步幅为二。这使得表示的大小减少了一半，减轻了下一层的计算和统计负担。注意到最右边的池化区域尺寸较小，但如果我们不想忽略一些探测单元的话就必须包含这个区域。

在很多任务中，池化对于处理不同大小的输入具有重要作用。将特征一起动态地池化也是可行的。

例：使用卷积和池化操作的用于分类的完整卷积网络结构：

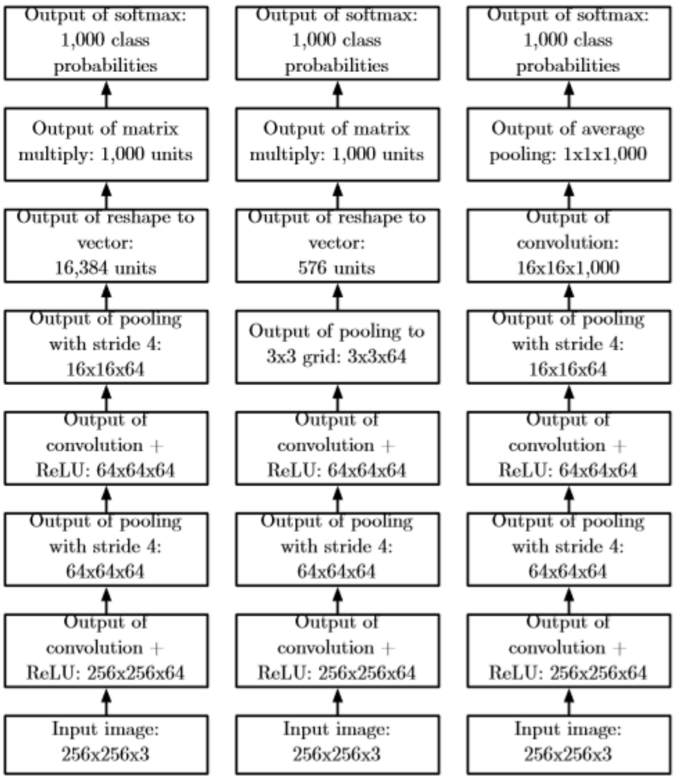



图 9.11: 卷积网络用于分类的结构示例。本图中使用的具体步幅和深度并不建议实际使用；它们被设计得非常浅以适合页面。实际的卷积网络还常常涉及大量的分支，不同于这里为简单起见所使用的链式结构。*(左)* 处理固定大小的图像的卷积网络。在卷积层和池化层几层交替之后，卷积特征映射的张量被重新变形以展平空间维度。网络的其余部分是一个普通的前馈网络分类器，如第六章所述。*(中)* 处理大小可变的图像的卷积网络，但仍保持全连接的部分。该网络使用具有可变大小但是数量固定的池的池化操作，以便向网络的全连接部分提供固定 576 个单位大小的向量。*(右)* 没有任何全连接权重层的卷积网络。相对的，最后的卷积层为每个类输出一个特征映射。该模型可能会用来学习每个类出现在每个空间位置的可能性的映射。将特征映射进行平均得到的单个值，提供了顶部 softmax 分类器的变量。

## 5. 卷积与池化作为一种无限强的先验

先验被认为是强或者弱取决于先验中概率密度的集中程度。弱先验具有较高的熵值,。这样的先验允许数据对于参数的改变具有或多或少的自由性。强先验具有较低的熵值,这样的先验在决定参数最终取值时起着更加积极的作用。一个无限强的先验需要对一些参数的概率置零并且完全禁止对这些参数赋值, 无论数据对于这些参数的值给出了多大的支持。

我们可以把卷积的使用当作是对网络中一层的参数引入了一个无限强的先验概率分布。这个先验说明了该层应该学得的函数只包含局部连接关系并且对平移具有等变性。类似的，使用池化也是一个无限强的先验：每一个单元都具有对少量平移的不变性。

把 CNN 网络当作一个具有无限强先验的全连接网络来实现会导致极大的计算浪费，但可以了解运行原理。

卷积和池化可能导致欠拟合。例如如果任务依赖于保存精确的空间信息，则在所有的特征上使用池化将会增大训练误差。

当比较卷积模型的统计学习表现时，只能以基准中的其他卷积模型作为比较的对象。

## 6. 基本卷积函数的变体

神经网络中的卷积通常指由多个并行卷积组成的运算, 因为具有单个核的卷积只能提取一种类型的特征，尽管它作用在多个空间位置上。我们通常希望网络的每一次能够在多个位置上提取多种类型的特征。

输入通常不仅仅是实值的网格，而是由一系列观测数据的向量构成的网络。

因为卷积网络通常使用多通道的卷积，所以即使使用了核翻转，也不一定保证网络的线性运算是可交换的。只有当其中的每个运算的输出和输入具有相同的通道数时，这些多通道的运算才是可交换的。

假设有一个 4 为核张量 K，它的每个元素是$K_{i,j,k,l}$，表示输出中处于通道$i$的一个单元和输入中通道$j$中的一个单元的连接强度，并且在输入单元之间有 k 行 l 列的偏置。V 为输入的观测数据，每个元素为$V_{i,j,k}$。设输入 Z 和输出 V 有相同的形式且不涉及翻转卷积核张量。则输出 Z:

$$Z_{i,j,k} = \sum_{l,m,n} V_{l,j+m-1,k+n-1} K_{i,l,m,n}$$

我们可以跳过核中的位置来降低计算开销，即看作对全卷积函数输出的下采样（downsampling）。在输出的每个方向上每间隔 s 个像素进行采样，可定义一个**下采样卷积函数**:

$$Z_{i,j,k} = c(\mathbf{K}, \mathbf{V}, s)_{i,j,k} = \sum_{l,m,n} [V_{l,(j-1)\times s+m,(k-1)\times s+n,} K_{i,l,m,n}]$$

将 s 称为下采样卷积的**步幅**（stirde）

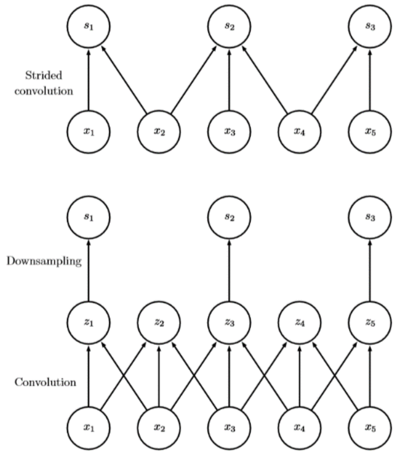



图 9.12: 带有步幅的卷积。在这个例子中，我们的步幅为二。*(上)* 在单个操作中实现的步幅为二的卷积。*(下)* 步幅大于一个像素的卷积在数学上等价于单位步幅的卷积随后降采样。显然，涉及降采样的两步法在计算上是浪费的，因为它计算了许多将被丢弃的值。

在任何卷积网络的实现中都有一个重要性质，即对输入 V 用 0 进行填充（pad）使之加宽。

有三种零填充设定的情况值得注意：

**有效**（valid）**卷积**：无论怎样都不使用零填充的情况，并且卷积核只允许访问那些图像中能够完全包含整个核的位置。输出的所有像素都是输入中相同数量像素的函数，使得输出像素的表示更加规范，而输出的大小在每一层都会缩减。

**相同**（same）**卷积**：只进行足够的零填充来保持输入和输出具有相同的大小。在这种情况下卷积运算不改变下一层的结构，而输入像素靠近边界的部分相比较于中间部分对于输出像素的影响更小，可能会导致边界像素存在一定的欠表示

**全**（full）**卷积**：进行足够多的零填充使得每个像素在每个方向上下好被访问了 k 次，最终输出图像的宽度为 m+k-1。这种情况下，输出像素中靠近边界的部分相比于中间的部分是更少像素的函数。导致学得一个在卷积特征映射的所有位置都表现不错的单核更为困难。

通常零填充的最优数量处于“有效卷积”与“相同卷积”之间。

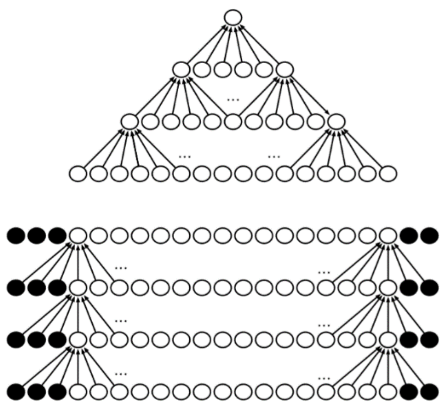


图 9.13: 零填充对网络大小的影响。考虑一个卷积网络，每层有一个宽度为六的核。在这个例子中，我们不使用任何池化，所以只有卷积操作本身缩小网络的大小。*(上)* 在这个卷积网络中，我们不使用任何隐含的零填充。这使得表示在每层缩小五个像素。从十六个像素的输入开始，我们只能有三个卷积层，并且最后一层不能移动核，所以可以说只有两层是真正的卷积层。可以通过使用较小的核来减缓收缩速率，但是较小的核表示能力不足，并且在这种结构中一些收缩是不可避免的。*(下)* 通过向每层添加五个隐含的零，我们防止了表示随深度收缩。这允许我们设计一个任意深的卷积网络。

**非共享卷积**（unshared convolution）：具有一个小核的离散卷积运算很像，但并不横跨位置来共享参数，是一些局部连接的网路层，此时多层感知机对应的邻接矩阵是相

同的，但每个连接都有它自己的权重。局部连接层的线性部分：

$$Z_{i,j,k} = \sum_{l,m,n} [V_{l,j+m-1,k+n-1} w_{i,j,k,l,m,n}]$$

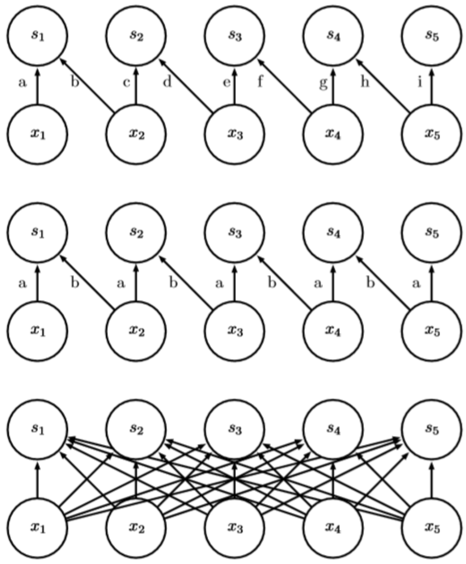



图 9.14: 局部连接，卷积和全连接的比较。(上) 每一小片（接受域）有两个像素的局部连接层。每条边用唯一的字母标记，来显示每条边都有自身的权重参数。(中) 核宽度为两个像素的卷积层。该模型与局部连接层具有完全相同的连接。区别不在于哪些单元相互交互，而在于如何共享参数。局部连接层没有参数共享。正如用于标记每条边的字母重复出现所指示的，卷积层在整个输入上重复使用相同的两个权重。(下) 全连接层类似于局部连接层，它的每条边都有其自身的参数（在该图中用字母明确标记的话就太多了）。然而，它不具有局部连接层的连接受限的特征。

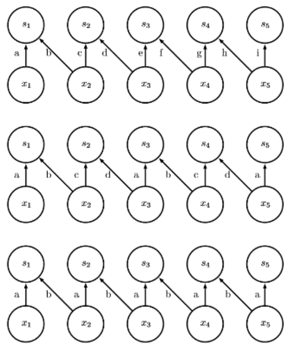
**平铺卷积**（tiled convolution）：学习一组核使得核在空间移动时可以循环利用，意味着在近邻的位置上拥有不同的过滤器，但对于这些参数的存储需求仅仅会增长常数倍，这个常数为核的集合的大小。注意输出的位置在每个方向上在 t 个不同的核组成的集合中进行循环。

$$Z_{i,j,k} = \sum_{l,m,n} V_{l,j+m-1,k+n-1} K_{i,l,m,n,j\%t+1,k\%t+1}$$

这里的%代表取模运算，性质包括 t%t=0，(t + 1)%t = 1 等。

局部连接层与平铺卷积层与最大池化存在关联：这些层的探测单元都是由不同的过滤器驱动的。卷积层对于平移具有内置的不变性。

例：局部连接层、平铺卷积、标准卷积比较：




**关于反向传播**：

需要训练一个卷积网络，其中包含步幅为 s 的步幅卷积，该卷积的核为 k，作用于多

通道的图像 V，定义为 c（K，V，s）。假设需要最小化某个损失函数 J(V, K)。在前向传播过程中，需要用 c 本身来输出 Z，然后 Z 传递到网络的其余部分并被用来计算损失函数 J。在反向传播过程中，会得到一个张量 G 满足$G_{i,j,k}=\frac{\partial}{\partial Z_{i,j,k}}J(\mathbf{V},\mathbf{K})$。

对核中的权重求导：

$$g(\mathbf{G},\mathbf{V},s)_{i,j,k,l}=\frac{\partial}{\partial K_{i,j,k,l}}J(\mathbf{V},\mathbf{K})=\sum_{m=n}G_{i,m,n}V_{j,(m-1)\times s+k,(n-1)\times s+l}$$

若此层不是网络的底层，需要对 V 求梯度来对误差进一步反向传播。可以使用：

$$h(\mathbf{K},\mathbf{G},s)_{i,j,k}=\frac{\partial}{\partial V_{i,j,k}}J(\mathbf{V},\mathbf{K})$$

$$=\sum_{\substack{l,m\\ \text{s.t.}\\ (l-1)\times s+m=j}}\ \sum_{\substack{n,p\\ \text{s.t.}\\ (n-1)\times s+p=k}}\ \sum_{q}K_{q,i,m,p}G_{q,l,n}$$

关于如何在偏置项中共享参数：

对于局部连接层，对每个单元都给定它特有的偏置，对于平铺卷积，用与核一样的平铺模式来共享参数。对于卷积层来说，通常在输出的每个通道上都设置一个偏置，这个派年至在每个卷积映射的所有位置上共享，若输入是已知的固定大小，亦可以在输出映射的每个位置学习一个单独的偏置。

## 7. 结构化输出

卷积神经网络可以用于输出高维的结构化对象，不仅是预测分类任务的类标签或回归任务的实数值。通常这个对象是一个 tensor，由标准卷积层产生。

经常出现的一个问题是输出平面可能比输入平面要小。用于对图像中单个对象分类的常用结构中，网络空间维数的最大减少来源于使用大步幅的池化层。

为了产生与输入大小相似的输出，可以避免把池化放在一起，或单纯地产生一张低分辨率的标签网格，原则上可以使用具有单位步幅的池化操作。

对图像逐个像素标记的一种策略是先产生图像标签的原始猜测，然后使用相邻像素之间的交互来修正该原始猜测。（形成一种特殊的循环神经网络）

## 8. 数据类型

卷积网络使用的数据通常包含多个通道，每个通道是时间上或空间中某一点的不同观测量。

| | 单通道 | 多通道 |
|---|---|---|
| 1 维 | 音频波形：卷积的轴对应于时间。我们将时间离散化并且在每个时间点测量一次波形的振幅。 | 骨架动画 (skeleton animation) 数据：计算机渲染的 3D 角色动画是通过随时间调整"骨架"的姿势而生成的。在每个时间点，角色的姿势通过骨架中的每个关节的角度来描述。我们输入到卷积模型的数据的每个通道，表示一个关节关于一个轴的角度。 |
| 2 维 | 已经使用傅立叶变换预处理过的音频数据：我们可以将音频波形变换成 2 维张量，不同的行对应不同的频率，不同的列对应不同的时间点。在时间轴上使用卷积使模型等效于在时间上移动。在频率轴上使用卷积使得模型等效于在频率上移动，这使得在不同八度音阶中播放的相同旋律产生相同的表示，但处于网络输出中的不同高度。 | 彩色图像数据：其中一个通道包含红色像素，另一个包含绿色像素，最后一个包含蓝色像素。在图像的水平轴和竖直轴上移动卷积核，赋予了两个方向上平移等变性。 |
| 3 维 | 体积数据：这种数据一般来源于医学成像技术，例如 CT 扫描等。 | 彩色视频数据：其中一个轴对应着时间，另一个轴对应着视频帧的高度，最后一个对应着视频帧的宽度。 |

表 9.1: 用于卷积网络的不同数据格式的示例。

卷积网络一个有点是可以处理具有可变空间尺度的输入。

注意：使用卷积处理可变尺寸的输入，仅对输入是因为包含对同种事物的不同量的观

察而导致的尺寸变化这种情况才有意义。若输入是因为它可以选择性地包括不同种类的观察而具有可变尺寸，使用卷积是不合理的。

## 9. 高效的卷积算法

卷积等效于使用傅里叶变换将输入与核都转换到频域、执行两个信号的住店相乘，再使用傅里叶逆变换转换回时域。

当一个 d 维的核可以表成 d 个向量（每一维一个向量）的外积时，该核称为**可分离**（separable）的。当核可分离，它等价于组合 d 个一维卷积，每个卷积使用这些向量中的一个。