

复习

- 简述常见的激活函数 (≥ 3 种) , 列出其表达式。
- 试述前馈神经网络 (全连接神经网络) 的结构特点。
- 一个前馈神经网络模型:
 - 该模型包含1个隐藏层
 - 输入层神经元数量为2
 - 隐藏层神经元数量为3
 - 输出层神经元数量为1请计算该神经网络模型的参数量 (W 、 b 的元素总数) 。
- 简述通用近似定理的主要推论。

复习

- 简述机器学习的四个要素。
- 以下哪个损失函数主要应用在回归问题？
A. 0-1损失函数 B. 交叉熵损失函数 C. 平方损失函数 D. 合页损失函数
- 在梯度下降算法中，学习率过大、过小各会导致什么问题？
- 简述随机梯度下降（SGD）算法的流程。
- 如何避免过拟合？列举几种常用的策略。
- 提前停止（Early stopping）策略为何有效？简述其原理。

复习

■ 对于反向传播的算法流程，请按顺序排列以下各选项：

- ①反向传播计算每一层的误差
- ②采用梯度下降方法更新参数
- ③计算每一层参数的导数值
- ④前馈计算，得到每层的净输入和激活值

■ 试比较静态计算图与动态计算图。

《神经网络与深度学习》



PyTorch搭建前馈神经网络

<https://nndl.github.io/>

PyTorch搭建前馈神经网络

PyTorch

天池实验室



阿里云 | TIANCHI 天池



kaggle™



PyTorch搭建前馈神经网络



《神经网络与深度学习》



卷积神经网络

<https://nndl.github.io/>

从全连接到卷积

▶使用相机采集图片（12M像素）

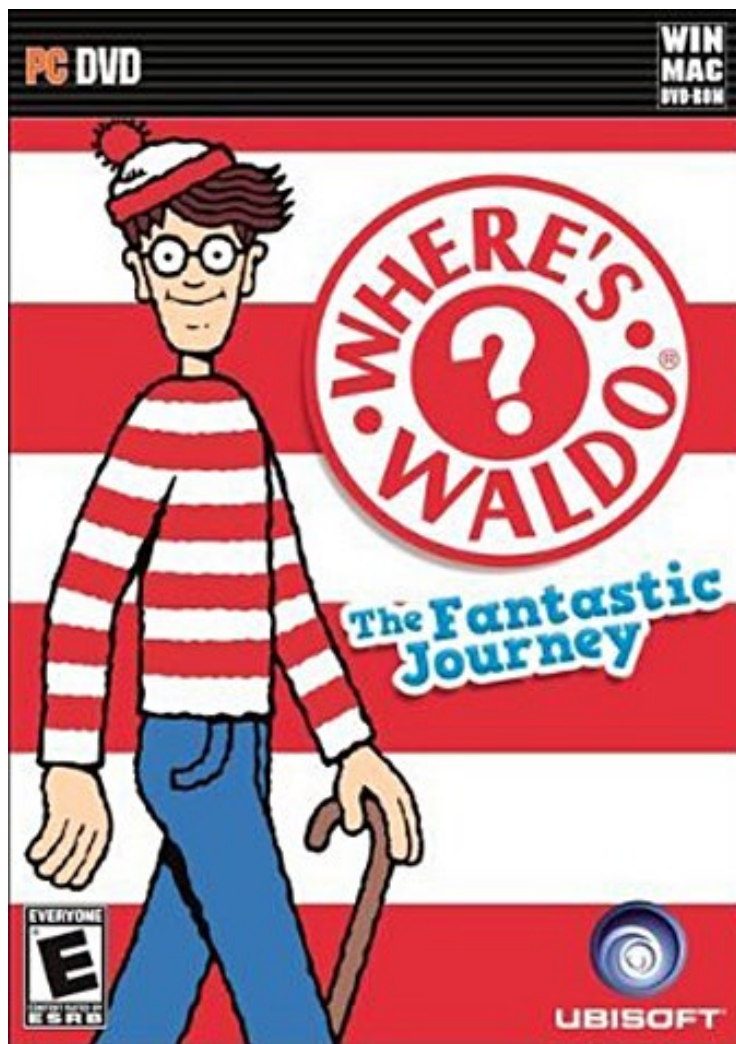


▶RGB图片有36M元素

▶使用规模为100的单隐藏层MLP，模型有3.6B元素

▶远多于世界上所有猫、狗的总数（900M狗，600M猫）

从全连接到卷积



从全连接到卷积

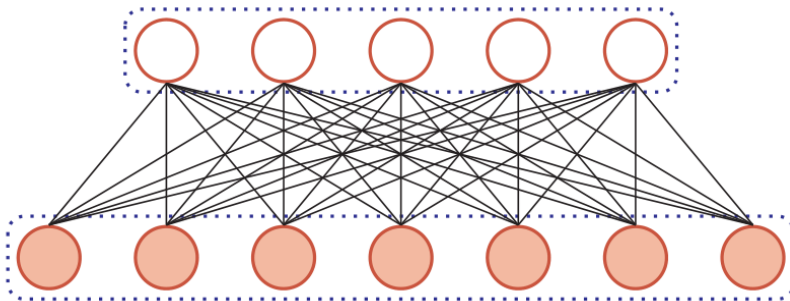


从全连接到卷积



从全连接到卷积

► 全连接神经网络，权重矩阵的参数非常多



思考：参数过多导致什么问题？

- 神经网络的训练困难；
- 过拟合。

思考：全连接的形式导致什么问题？

- 对输入的细微改变敏感，难以提取局部不变性特征。

► 局部不变性特征

- 自然图像中的物体都具有局部不变性特征
 - 尺度缩放、平移、旋转等操作不影响其语义信息。
- 全连接前馈网络很难提取这些局部不变特征

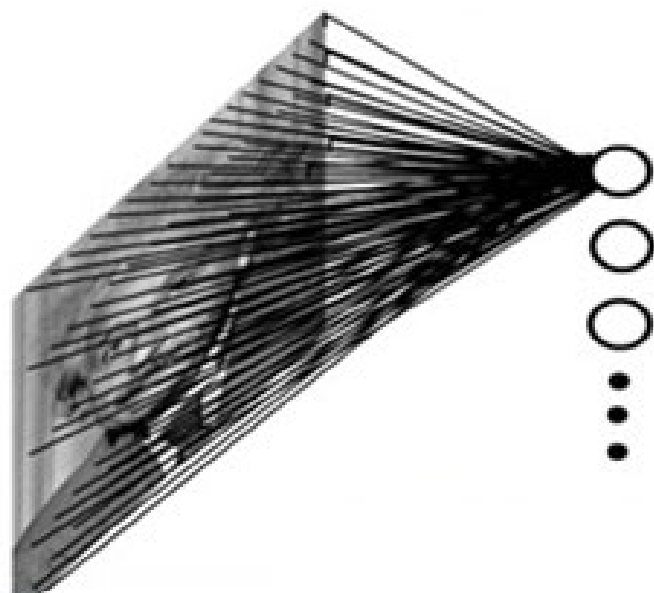
卷积神经网络

► 卷积神经网络 (Convolutional Neural Networks, CNN)

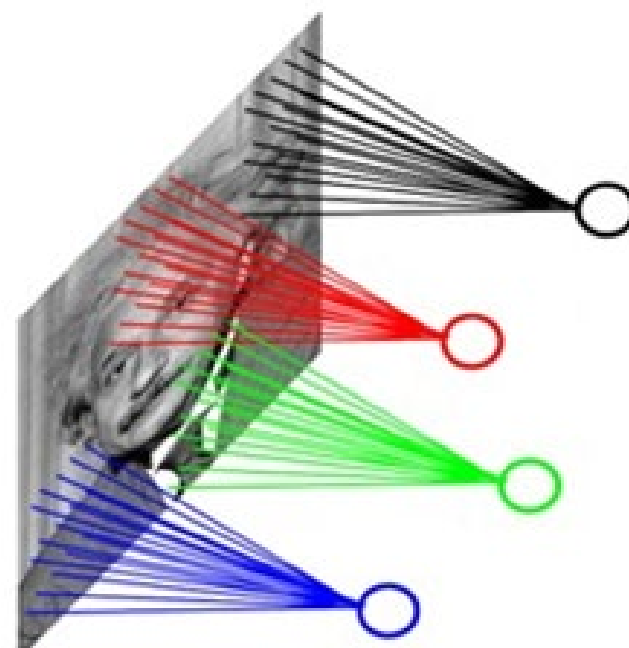
- 一种前馈神经网络
- 受生物学上感受野 (Receptive Field) 的机制而提出的
 - 在视觉神经系统中，一个神经元的感受野是指视网膜上的特定区域，只有这个区域内的刺激才能够激活该神经元。
- 卷积神经网络有三个结构上的特性：
 - 局部连接
 - 权重共享
 - 空间或时间上的次采样

从全连接到卷积

全连接模式（经典神经网络）



局部连接模式（卷积神经网络）



局部感受野

卷积

- ▶ 卷积经常用在信号处理中，用于计算信号的延迟累积。
- ▶ 假设一个信号发生器每个时刻 t 产生一个信号 x_t ，其信息的衰减率为 w_k ，即在 $k-1$ 个时间步长后，信息为原来的 w_k 倍
 - ▶ 假设 $w_1 = 1, w_2 = 1/2, w_3 = 1/4$
- ▶ 时刻 t 收到的信号 y_t 为当前时刻产生的信息和以前时刻延迟信息的叠加。

卷积

- ▶ 卷积经常用在信号处理中，用于计算信号的延迟累积。
- ▶ 假设一个信号发生器每个时刻 t 产生一个信号 x_t ，其信息的衰减率为 w_k ，即在 $k-1$ 个时间步长后，信息为原来的 w_k 倍
 - ▶ 假设 $w_1 = 1, w_2 = 1/2, w_3 = 1/4$
- ▶ 时刻 t 收到的信号 y_t 为当前时刻产生的信息和以前时刻延迟信息的叠加

$$\begin{aligned}y_t &= 1 \times x_t + 1/2 \times x_{t-1} + 1/4 \times x_{t-2} \\&= w_1 \times x_t + w_2 \times x_{t-1} + w_3 \times x_{t-2} \\&= \sum_{k=1}^3 w_k \cdot x_{t-k+1}.\end{aligned}$$

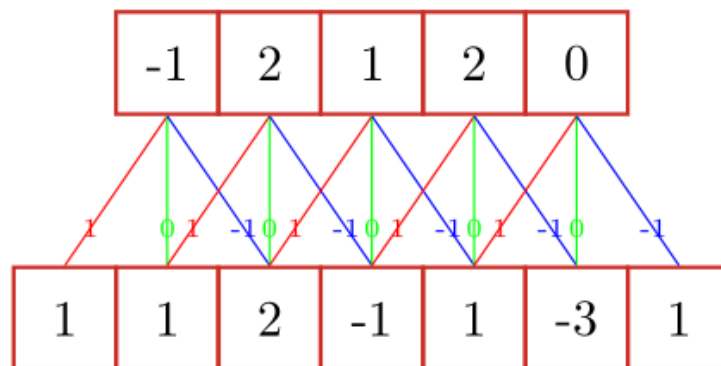
滤波器 (filter) 或卷积核 (convolution kernel)

卷积

► 给定一个输入信号序列 x 和滤波器 w ,卷积的输出为:

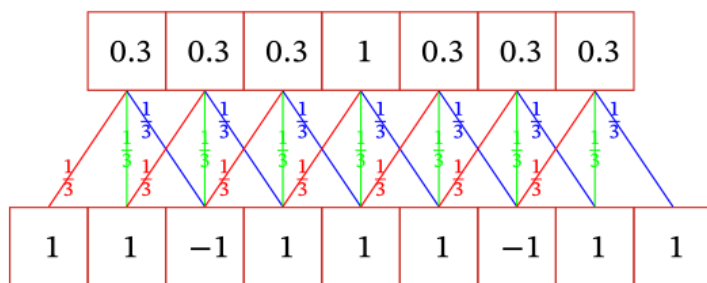
$$y_t = \sum_{k=1}^K w_k x_{t-k+1}$$

Filter: [-1,0,1]



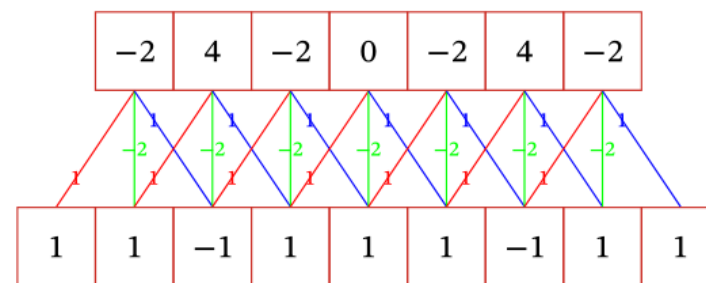
卷积

► 不同的滤波器来提取信号序列中的不同特征



(a) 滤波器 $[1/3, 1/3, 1/3]$

低频信息

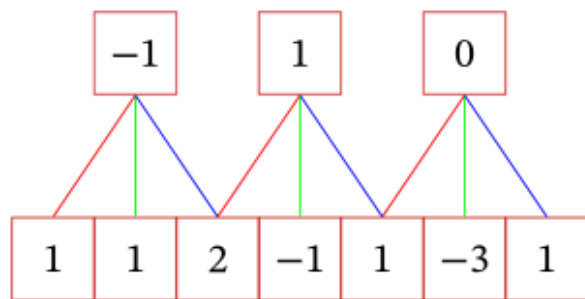


(b) 滤波器 $[1, -2, 1]$

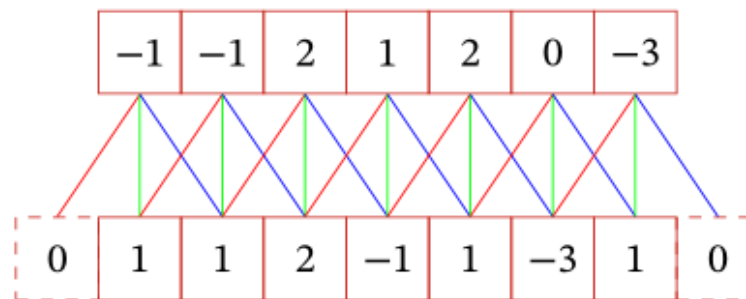
高频信息

卷积扩展

- 卷积中的两个重要参数：步长（Stride），填充（Padding）
- 引入滤波器的滑动步长 S 和零填充 P



(a) 步长 $S = 2$



(b) 零填充 $P = 1$

卷积扩展

►思考：假设输入长度为M，步长为S，卷积核大小为K，零填充为P，那么输出长度是多少？

$$\left\lfloor \frac{M + 2P - K}{S} \right\rfloor + 1$$

卷积类型

► 卷积的结果按输出长度不同可以分为三类：

► 窄卷积：步长 $T = 1$ ，两端不补零 $P = 0$ ，卷积后输出长度为 $M - K + 1$

► 宽卷积：步长 $T = 1$ ，两端补零 $P = K - 1$ ，卷积后输出长度 $M + K - 1$

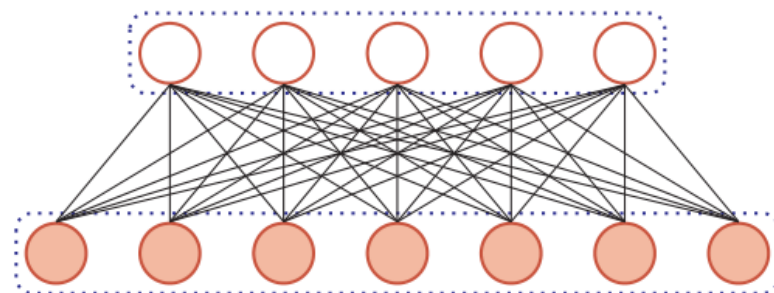
► 等宽卷积：步长 $T = 1$ ，两端补零 $P = (K - 1)/2$ ，卷积后输出长度 M

► 在早期的文献中，卷积一般默认为窄卷积。

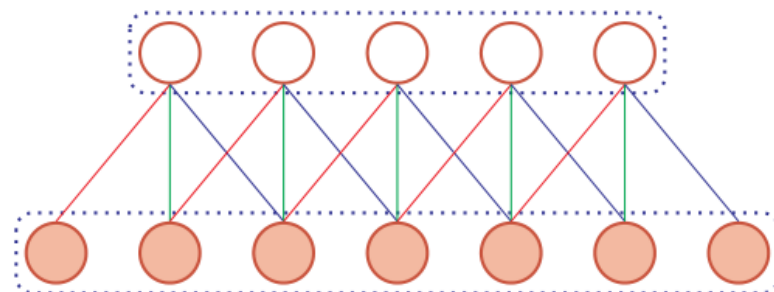
► 而目前的文献中，卷积一般默认为等宽卷积。

卷积

► 用卷积层代替全连接层



(a) 全连接层



(b) 卷积层

二维卷积

- 在图像处理中，图像是以二维矩阵的形式输入到神经网络中，因此我们需要二维卷积。

一个输入信息 \mathbf{X} 和滤波器 \mathbf{W} 的二维卷积定义为

$$\mathbf{Y} = \mathbf{W} * \mathbf{X},$$

$$y_{ij} = \sum_{u=1}^U \sum_{v=1}^V w_{uv} x_{i-u+1, j-v+1}.$$

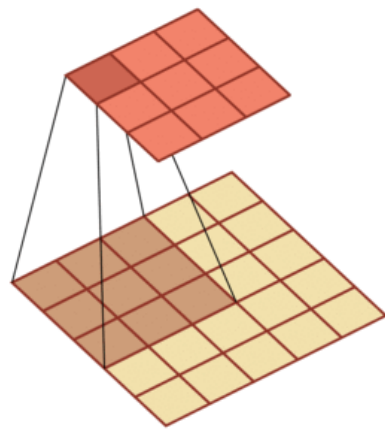
1	1	1	1	1
-1	0	-3	0	1
2	1	1	-1	0
0	-1	1	2	1
1	2	1	1	1

*

1	0	0
0	0	0
0	0	-1

=

0	-2	-1
2	2	4
-1	0	0



互相关（交叉相关）

- ▶ 计算卷积需要进行卷积核翻转。
- ▶ 卷积操作的目标：提取特征。

翻转是不必要的！

- ▶ 互相关（Cross-Correlation）

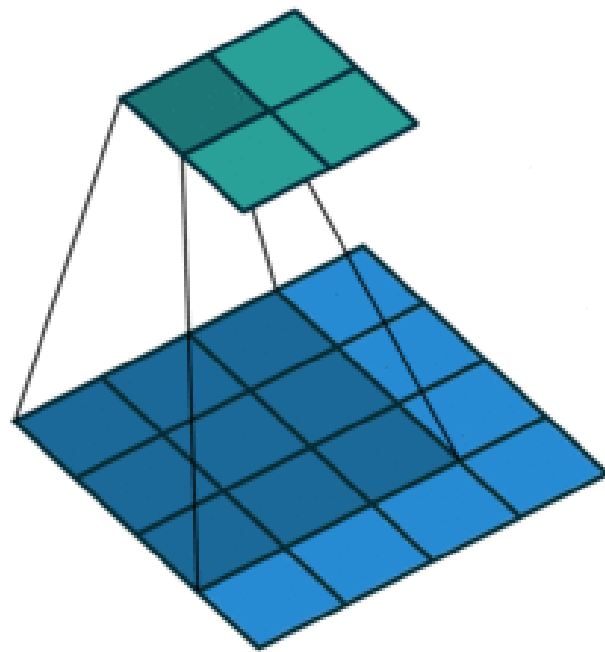
$$y_{ij} = \sum_{u=1}^m \sum_{v=1}^n w_{uv} \cdot x_{i+u-1, j+v-1}$$

除非特别声明，卷积一般指“互相关”。

二维卷积

Input		Kernel		Output																	
<table><tr><td>0</td><td>1</td><td>2</td></tr><tr><td>3</td><td>4</td><td>5</td></tr><tr><td>6</td><td>7</td><td>8</td></tr></table>	0	1	2	3	4	5	6	7	8	*	<table><tr><td>0</td><td>1</td></tr><tr><td>2</td><td>3</td></tr></table>	0	1	2	3	=	<table><tr><td>19</td><td>25</td></tr><tr><td>37</td><td>43</td></tr></table>	19	25	37	43
0	1	2																			
3	4	5																			
6	7	8																			
0	1																				
2	3																				
19	25																				
37	43																				

$$\begin{aligned}0 \times 0 + 1 \times 1 + 3 \times 2 + 4 \times 3 &= 19, \\1 \times 0 + 2 \times 1 + 4 \times 2 + 5 \times 3 &= 25, \\3 \times 0 + 4 \times 1 + 6 \times 2 + 7 \times 3 &= 37, \\4 \times 0 + 5 \times 1 + 7 \times 2 + 8 \times 3 &= 43.\end{aligned}$$



两维卷积

输入

3	0	8	4
6	5	1	0
0	7	0	3
9	1	2	1

卷积核W

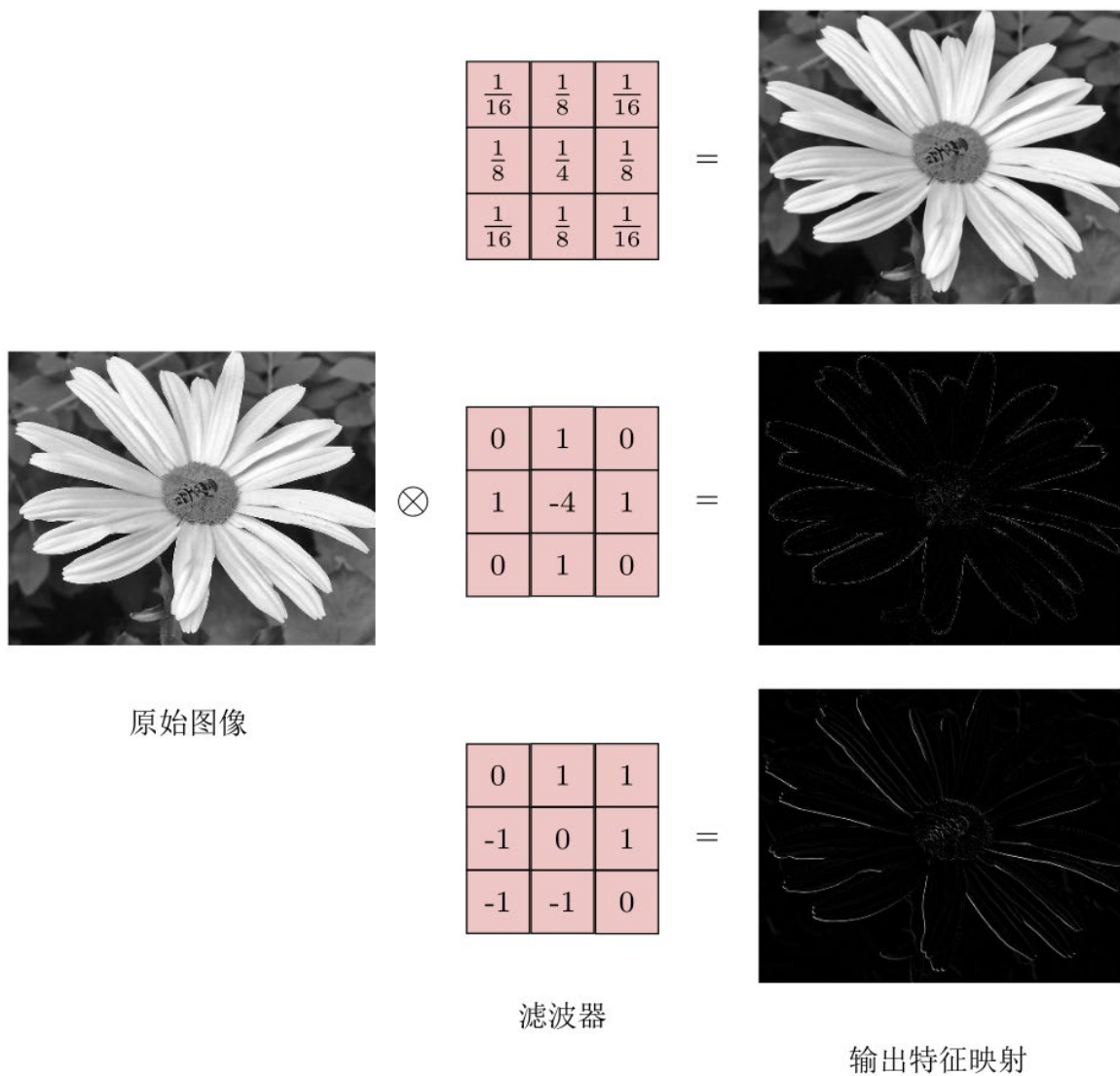
0	2	3
3	1	0
2	0	1

偏置b=1

$*$ $=$?

同时，取P=0, S=1（不进行填充补0，步长为1）

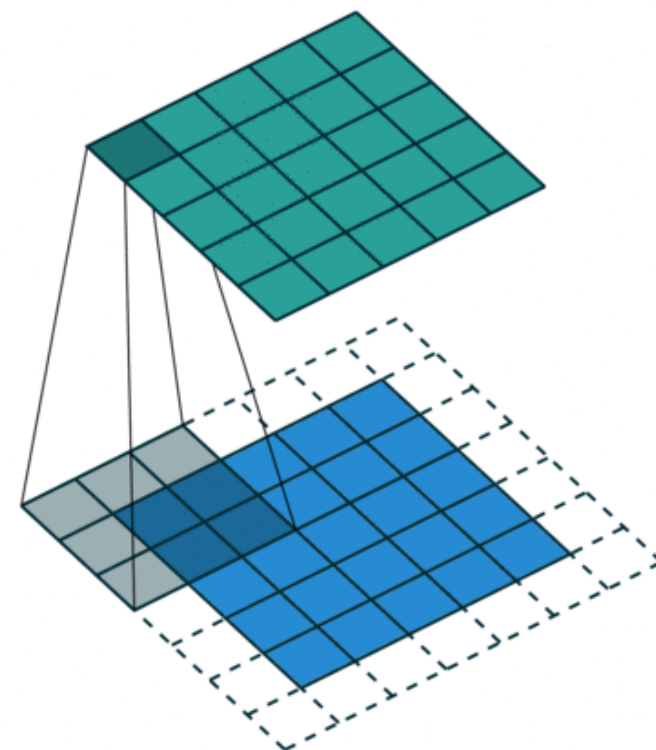
卷积作为特征提取器



二维卷积中的填充、步长

► 假设对图像进行2D卷积：

- 给定大小为 32×32 的输入图像；
- 应用 5×5 大小的卷积核；
- 则第1层输出大小为 28×28 ，第7层输出大小为 4×4 ；
- 卷积能够快速减小输出大小——难以增加神经网络层数！
- 填充（Padding）：在输入周围添加行/列
- 思考：如何决定填充的大小？

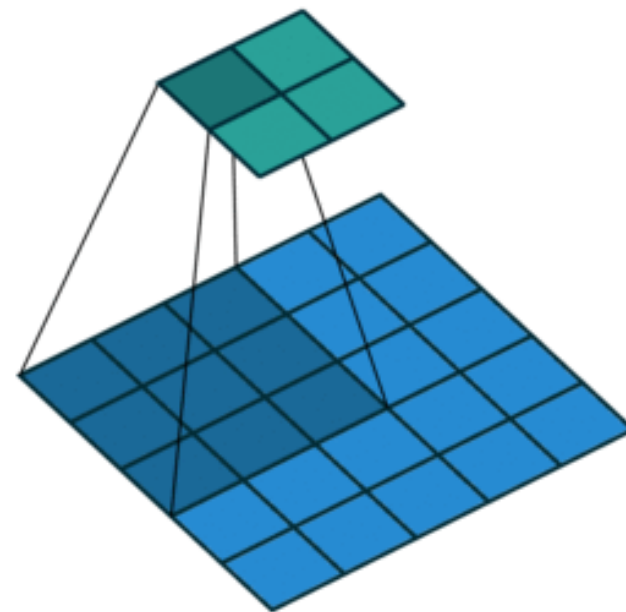


二维卷积中的填充、步长

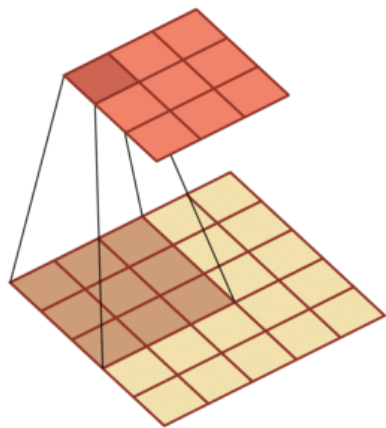
- ▶ 当输入图像的尺寸很大时，如何快速得到较小的输出？
- ▶ 步长 (Stride)：卷积核在输入上滑动的行/列数量

▶ 总结：

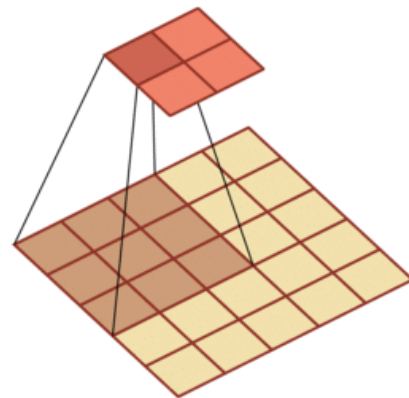
- ▶ 填充、步长是卷积层的超参数；
- ▶ 填充在输入周围添加额外的行/列，控制输出形状的减小量；
- ▶ 步长是每次滑动核窗口时的行/列步长，可以成倍减小输出形状。



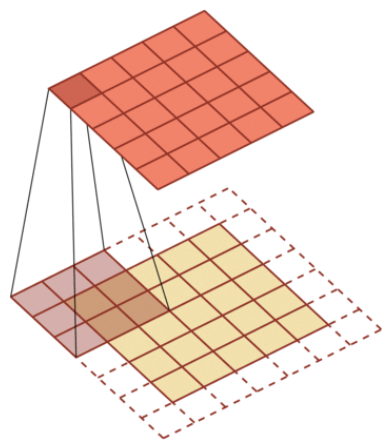
二维卷积



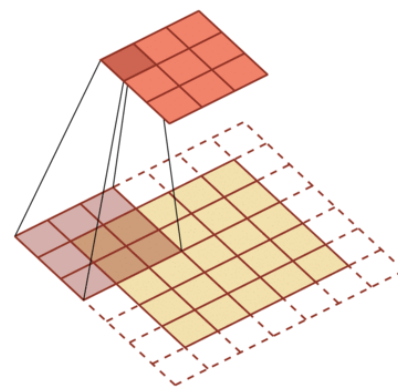
步长1，零填充0



步长2，零填充0



步长1，零填充1



步长2，零填充1