

4-1 卷积运算

王中雷

厦门大学王亚南经济研究院和经济学院, 2025

内容摘要

1. 一维卷积
2. 填充 (Padding)
3. 步幅 (Stride)
4. 二维卷积
5. 例子
6. 多通道卷积

卷积神经网络的直观

1. 当输入特征是图片时，为什么不考虑全连接神经网络

- 模型参数将变得非常巨大，容易造成过拟合
 - ▷ 为了解决这个问题，我们需要更多的训练样本、内存以及更快的算法
 - ▷ 实际上几乎不可能做到
- 忽略了图片中的空间结构
- 不具有空间平移不变性

2. 针对图片处理，我们需要考虑一个新的模型框架（卷积神经网络）

卷积神经网络

1. 卷积运算的优势：

- 卷积运算可独立地在图片的不同区域独立进行，该运算可并行
- 卷积运算在图片不同区域用同一个卷积核（kernel），所以模型参数更少
- 卷积运算考虑到了图片的局部信息，所以该计算具有一定的空间不变性

2. 涉及到卷积运算的神经网络称为卷积神经网络（CNN）

卷积运算

1. 卷积运算并不陌生

- 在统计学中，卷积运算涉及到两个随机变量加和的密度函数： $Z = X + Y$

$$f_Z(z) = \int f_X(x)f_Y(z-x)\mu(\mathrm{d}x),$$

▷ μ : 控制测度 (dominating measure)

▷ $f_\star(\cdot)$: 概率密度函数

▷ X, Y : 两个独立的随机变量

2. 简单起见，记 $f(\cdot)$ 和 $g(\cdot)$ 两个函数的卷积为

$$h(z) = (f * g)(z)$$

- 数学的意义上，只要运算有意义，函数 f 和 g 可以是任意函数

一维卷积

输入

3	6	5	4	8	9	1	7	9	10	10
---	---	---	---	---	---	---	---	---	----	----

卷积核

-1	0	1
----	---	---

计算

结果

2	-2	3	5	-7	-2	8	3	1
---	----	---	---	----	----	---	---	---

1. 我们通过翻转 $g(x)$ 来得到一个卷积核 (kernel), 我们可用 `np.sum` 进行卷积运算
2. x 的值不太重要

填充 (Padding)

1. 直观

- 通过卷积运算，图片的尺寸在减小
- 图片边缘的重要信息被低估

填充 (Padding)

1. 直观

- 通过卷积运算，图片的尺寸在减小
- 图片边缘的重要信息被低估

填充 (Padding)

输入

0	3	6	5	4	8	9	1	7	9	10	10	0
---	---	---	---	---	---	---	---	---	---	----	----	---

卷积核

-1	0	1
----	---	---

计算

结果

6	2	-2	3	5	-7	-2	8	3	1	-10
---	---	----	---	---	----	----	---	---	---	-----

填充 (Padding)

1. 两种不同的卷积核：

- **Valid**卷积：不施加填充
- **Same**卷积：施加填充使得输出和输入图片尺寸相同

2. 我们可以在图片边缘填充多层 0

3. 我们在进行卷积运算时，用大于 1 的步幅

- 降低输出特征的规模
- 平衡准确性和计算效率

步幅

输入

3	6	5	4	8	9	1	7	9	10	10
---	---	---	---	---	---	---	---	---	----	----

卷积核

-1	0	1
----	---	---

计算

结果

2	3	-7	8	1
---	---	----	---	---

1. 在上面的例子中，步幅（stride）为 2
2. 步幅可以是其他正整数
3. 填充（padding）和步幅（stride）可以在同一个卷积运算中同时用

空洞 (Dilated/Atrous) 卷积

输入

3	6	5	4	8	9	1	7	9	10	10
---	---	---	---	---	---	---	---	---	----	----

空洞卷积核

-1	0	1	0	1
----	---	---	---	---

计算

结果

10	7	4	12	2	8	18
----	---	---	----	---	---	----

1. 在上面的例子中，空洞率是 2
2. 空洞增加了卷积运算的感受野 (receptive field)，但不增加卷积核参数个数
3. 设置填充、步幅以及空洞率可以同时应用

二维卷积

原始序列

填充 (Padding = 1)

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

卷积核

-1	1
-1	1

结果

步幅 (Stride=2)

3	-1	4
17	11	6
12	2	6

均值核



$$\begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$



高斯核



$$\begin{bmatrix} 1/16 & 1/8 & 1/16 \\ 1/8 & 1/4 & 1/8 \\ 1/16 & 1/8 & 1/16 \end{bmatrix}$$



锐化卷积



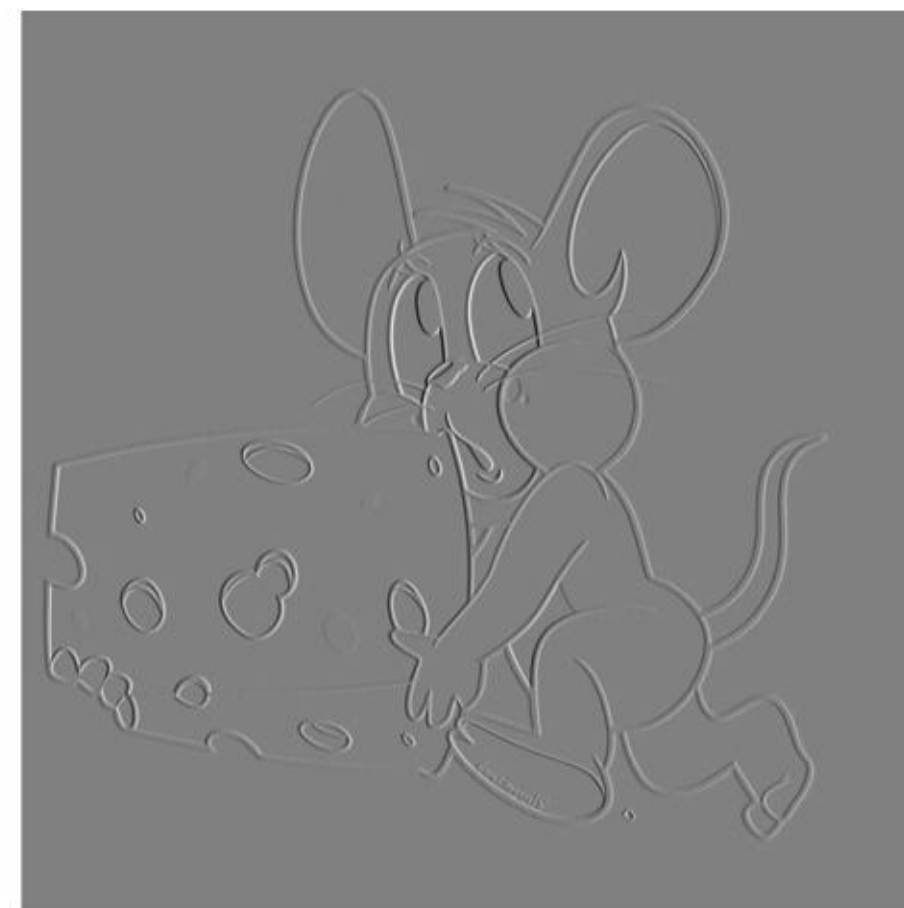
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$



竖直方向Sobel核



$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$



水平方向Sobel核

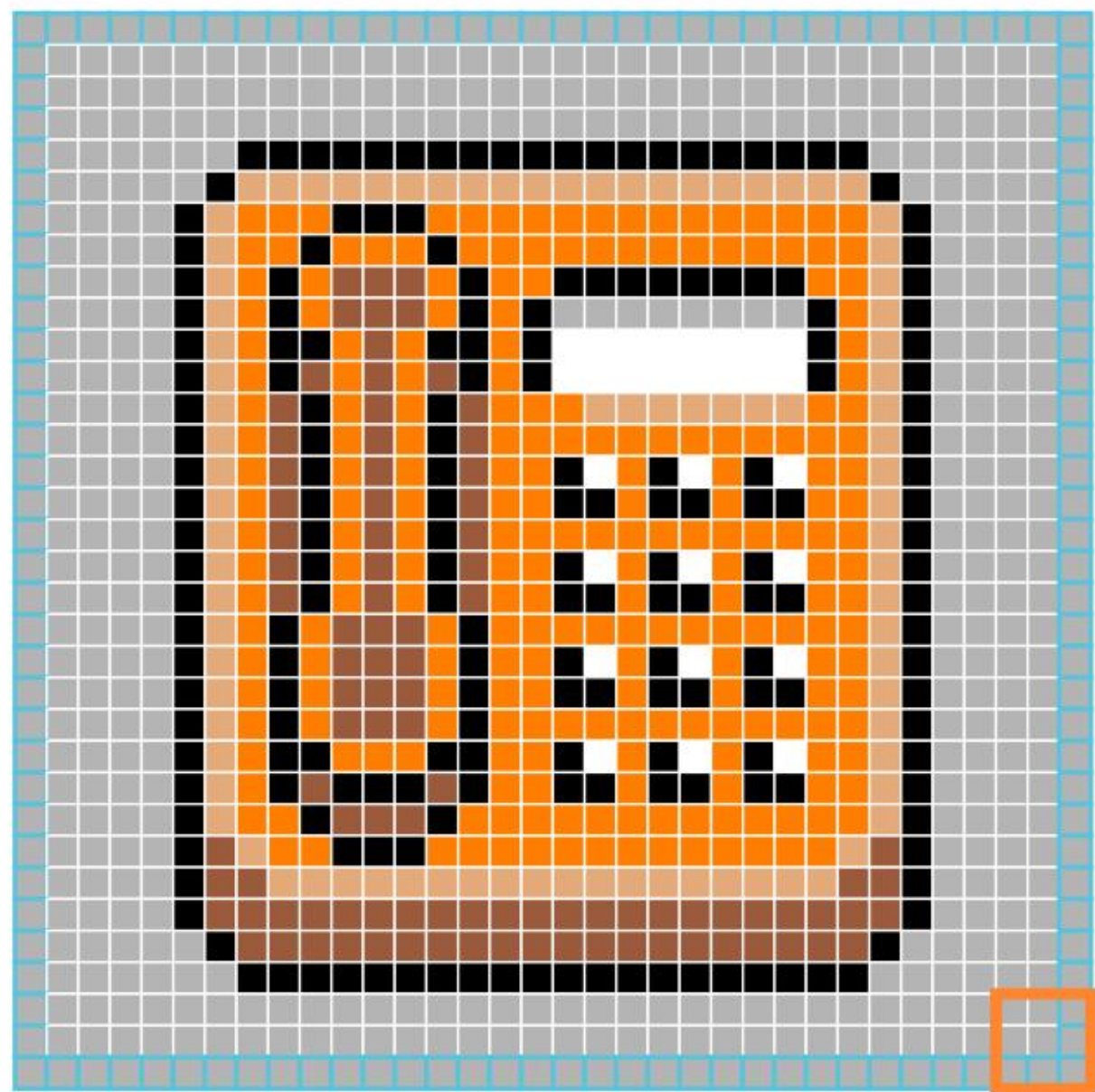


$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$



另一个例子

原始图片



卷积核

-1	-1	-1
-1	9	-1
-1	-1	-1

锐化

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

Sobel (竖直)

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

Sobel (水平)



多通道卷积

1. 通常来讲，卷积核的第三个维度与输入“图片”的第三个通道数相同

2. 输出图片尺寸的计算

- 输入图片尺寸: $d_H \times d_W \times d_C$
- 卷积核尺寸: $f \times f \times d_C$
- 填充 (Padding): p
- 步骤 (Stride): s
- 输出图像尺寸: $d'_H \times d'_W$

$$d'_H = \left\lfloor \frac{d_H + 2p - f}{s} + 1 \right\rfloor, \quad d'_W = \left\lfloor \frac{d_W + 2p - f}{s} + 1 \right\rfloor$$