

# 卷积神经网络基础

## 目录

- ◆ 什么是卷积
- ◆ 卷积相关操作与参数

# 什么是卷积

## 什么是卷积

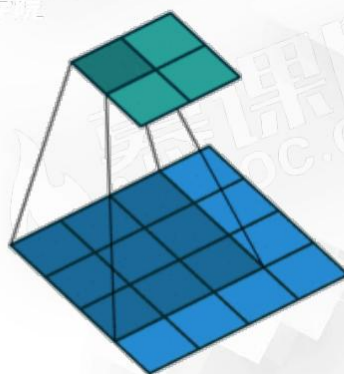
◆ Convolution, 输入信息与核函数（滤波器）的乘积

一维信号的时间卷积：输入 $x$ ，核函数 $w$ ，输出时一个连续时间段 $t$ 的加权平均结果

( ) ( ) ( )

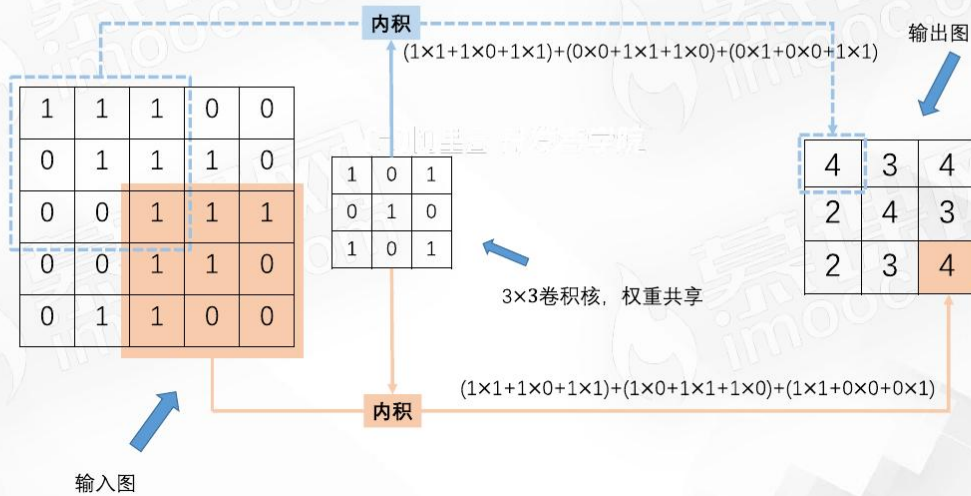
二维图像的空间卷积：输入图像 $I$ ，卷积核 $K$ ，输出图像 $O$

( ) ( )



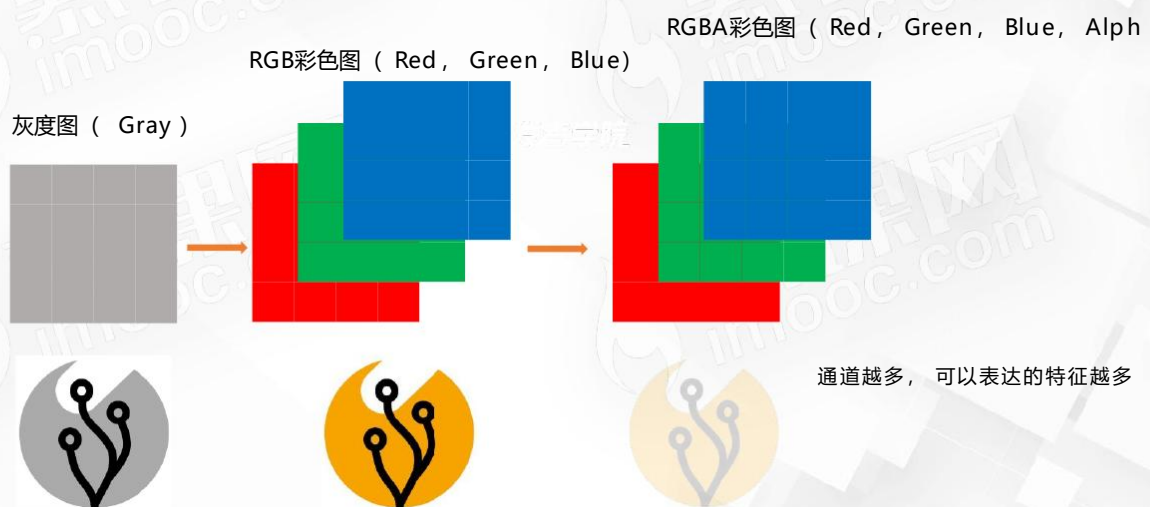
# 单个二维图片卷积

- ◆ 输入为单通道图像，输出为单通道图像



# 数字图像的表达

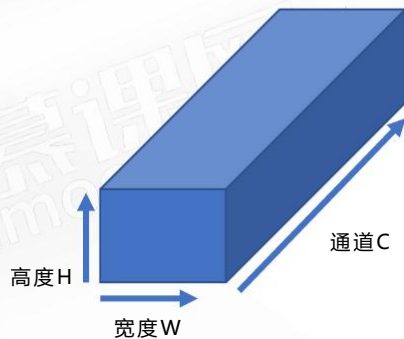
- ◆ 图像数据存储，多通道的二维矩阵



# 特征图

## ◆ 如何表达每一个网络层中高维的图像数据？

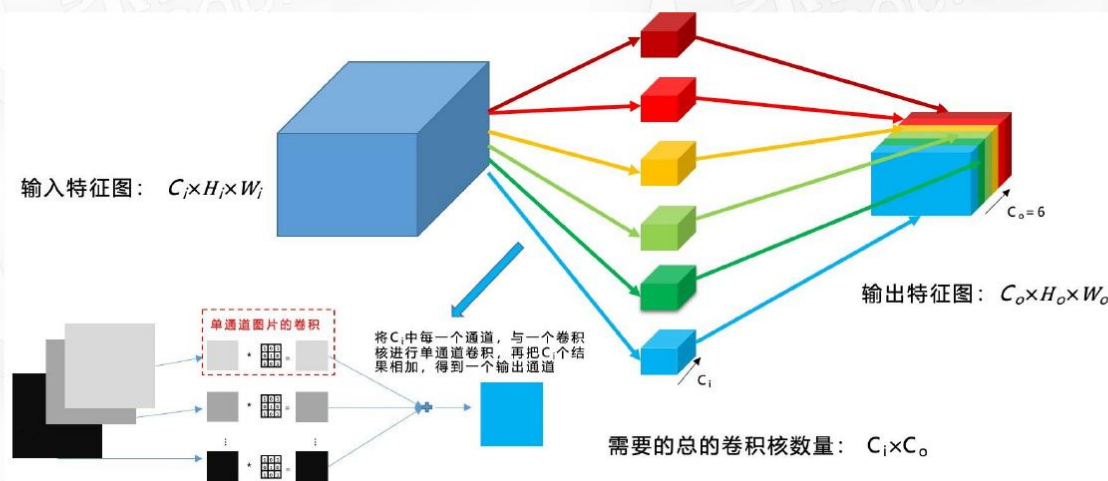
特征图 (Featuremap) : 通道 高度 宽度,  $C_i$



- 输入特征图:  $C_i$
- 输出特征图:  $C_o$
- 如何卷积? 需要多少卷积核?

## 多通道卷积

## ◆ 输出特征图的每一个通道，由输入图的所有通道和相同数量的卷积核先一一对应各自进行卷积计算，然后求和



## 卷积相关操作与参数

### 填充

◆ padding: 给卷积前的输入图像边界添加额外的行/列

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																				

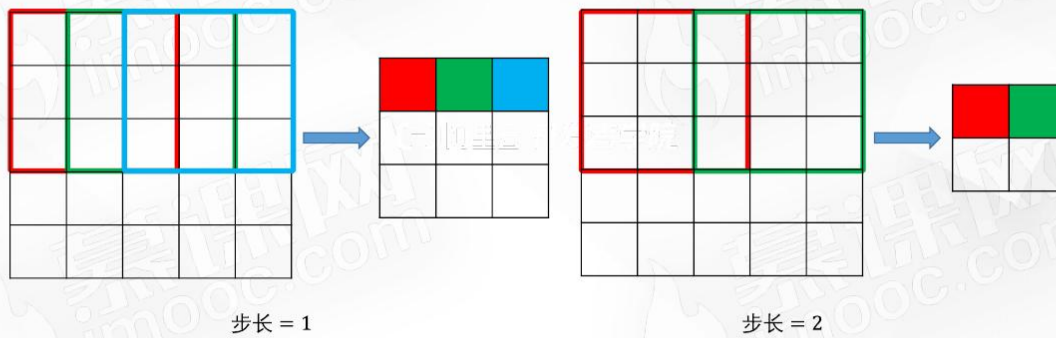
• 控制卷积后图像分辨率，方便计算特征图尺寸的变化

Input		Kernel		Output																																													
<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>2</td><td>0</td></tr><tr><td>0</td><td>3</td><td>4</td><td>5</td><td>0</td></tr><tr><td>0</td><td>6</td><td>7</td><td>8</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	1	2	0	0	3	4	5	0	0	6	7	8	0	0	0	0	0	0	*	<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>0</td><td>3</td><td>8</td><td>4</td></tr><tr><td>9</td><td>19</td><td>25</td><td>10</td></tr><tr><td>21</td><td>37</td><td>43</td><td>16</td></tr><tr><td>6</td><td>7</td><td>8</td><td>0</td></tr></table>	0	3	8	4	9	19	25	10	21	37	43	16	6	7	8	0
0	0	0	0	0																																													
0	0	1	2	0																																													
0	3	4	5	0																																													
0	6	7	8	0																																													
0	0	0	0	0																																													
0	1																																																
2	3																																																
0	3	8	4																																														
9	19	25	10																																														
21	37	43	16																																														
6	7	8	0																																														

• 弥补边界信息“丢失”

# 步长

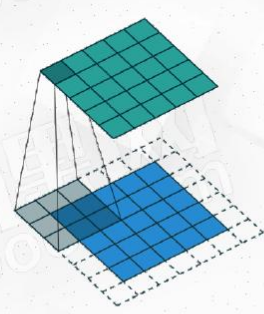
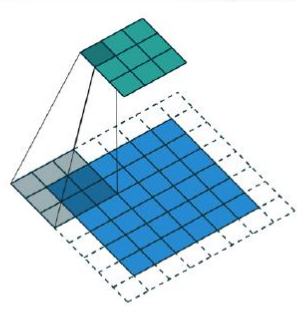
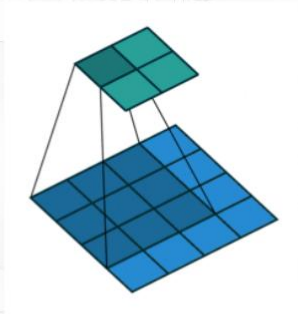
◆ 步长 (stride)：卷积核在图像上移动的步子



- stride太小，重复计算较多，计算量大，训练效率降低；
- stride太大，会造成信息遗漏，无法有效提炼数据背后的特征；

## 输入输出大小计算

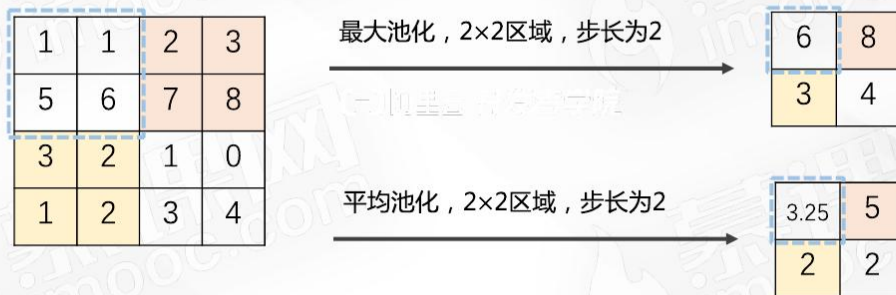
◆ 输入尺寸 ，填充大小为p，卷积核大小为k，步长为s，





# 池化

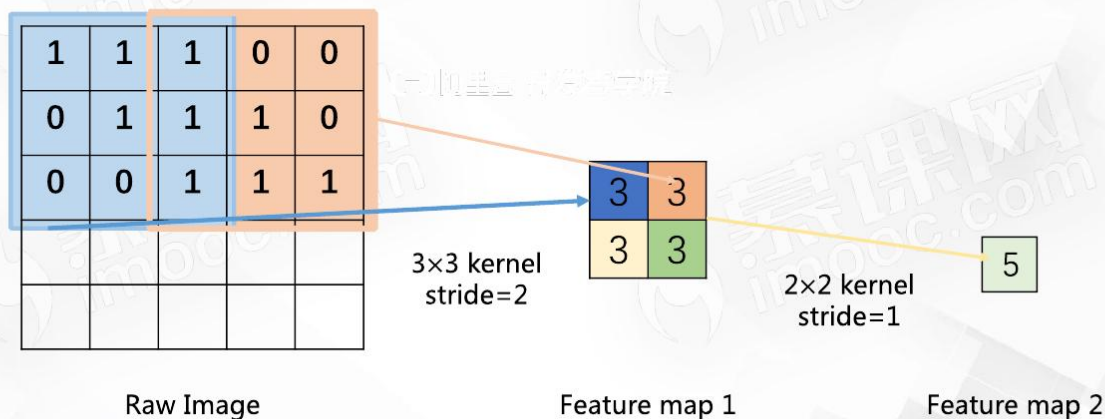
◆ 池化 (pooling)：对图像进行下采样，降低图像分辨率



- 使特征图变小, 简化网络计算复杂度;
- 压缩特征进行逐级抽象;

# 感受野

◆ 感受野 (Receptive Field)：某一层一个元素对应输入层的一个映射区域大小。感受野越大, 得到的全局信息越多



**下次预告：卷积与全连接的比较**