深度学习

卷积神经网络（2）

Author:WangQi
Email:wangqikaixin@gmail.com

概览

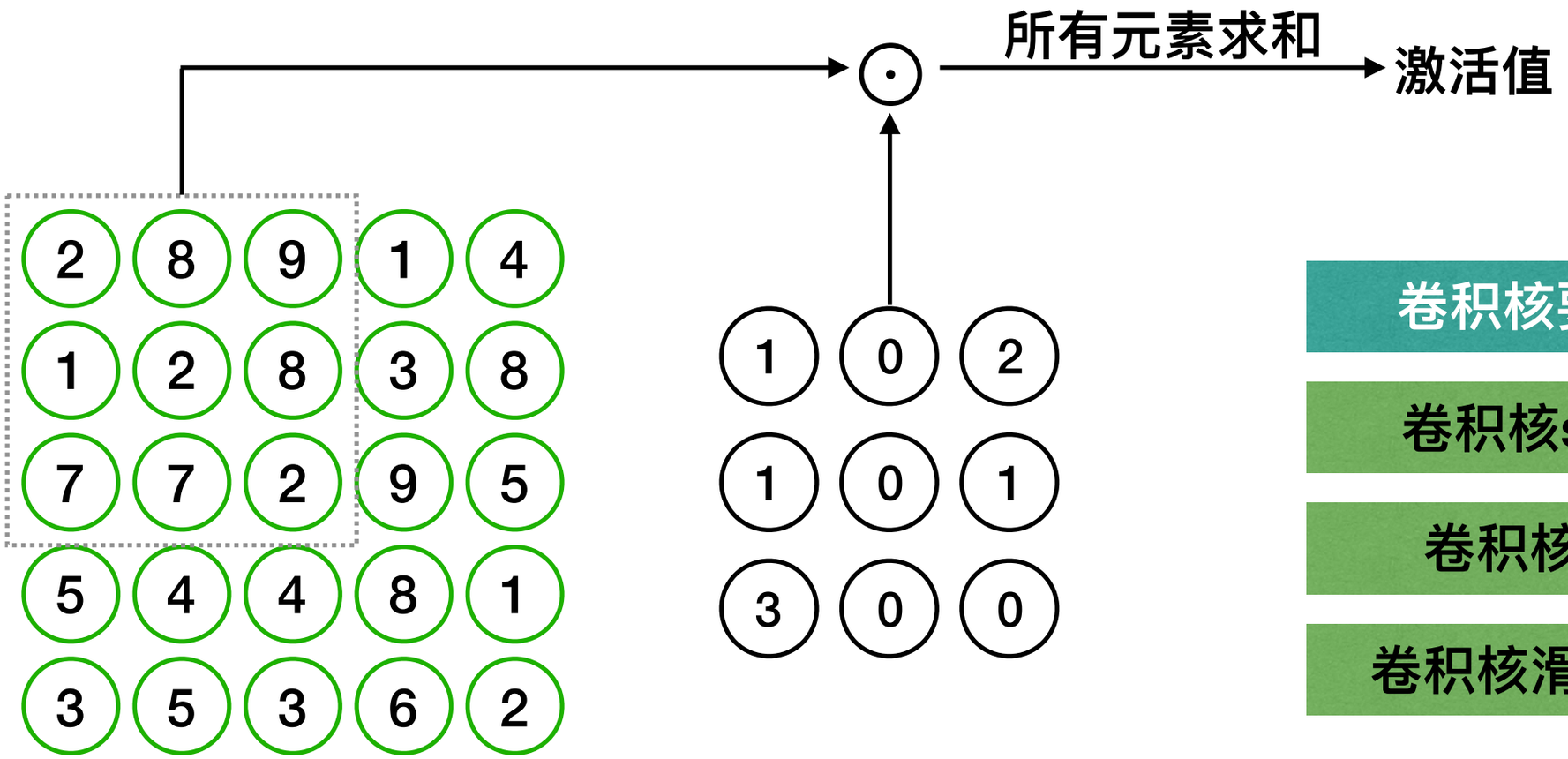
1. 卷积与池化的详细运算。
2. 对特征图卷积。
3. 多通道卷积。
    1. 多通道卷积
    2. 单通道与多通道卷积对比。
4. 卷积与池化的边界与步长。
    1. VALID边界处理。
    2. SAME边界处理。
    3. 示例
5. 其它

Author:WangQi
Email:wangqikaixin@gmail.com

1. 卷积与池化的详细运算

Author:WangQi
Email:wangqikaixin@gmail.com

卷积运算



一张图片的像素值

一个卷积核

卷积核要素：

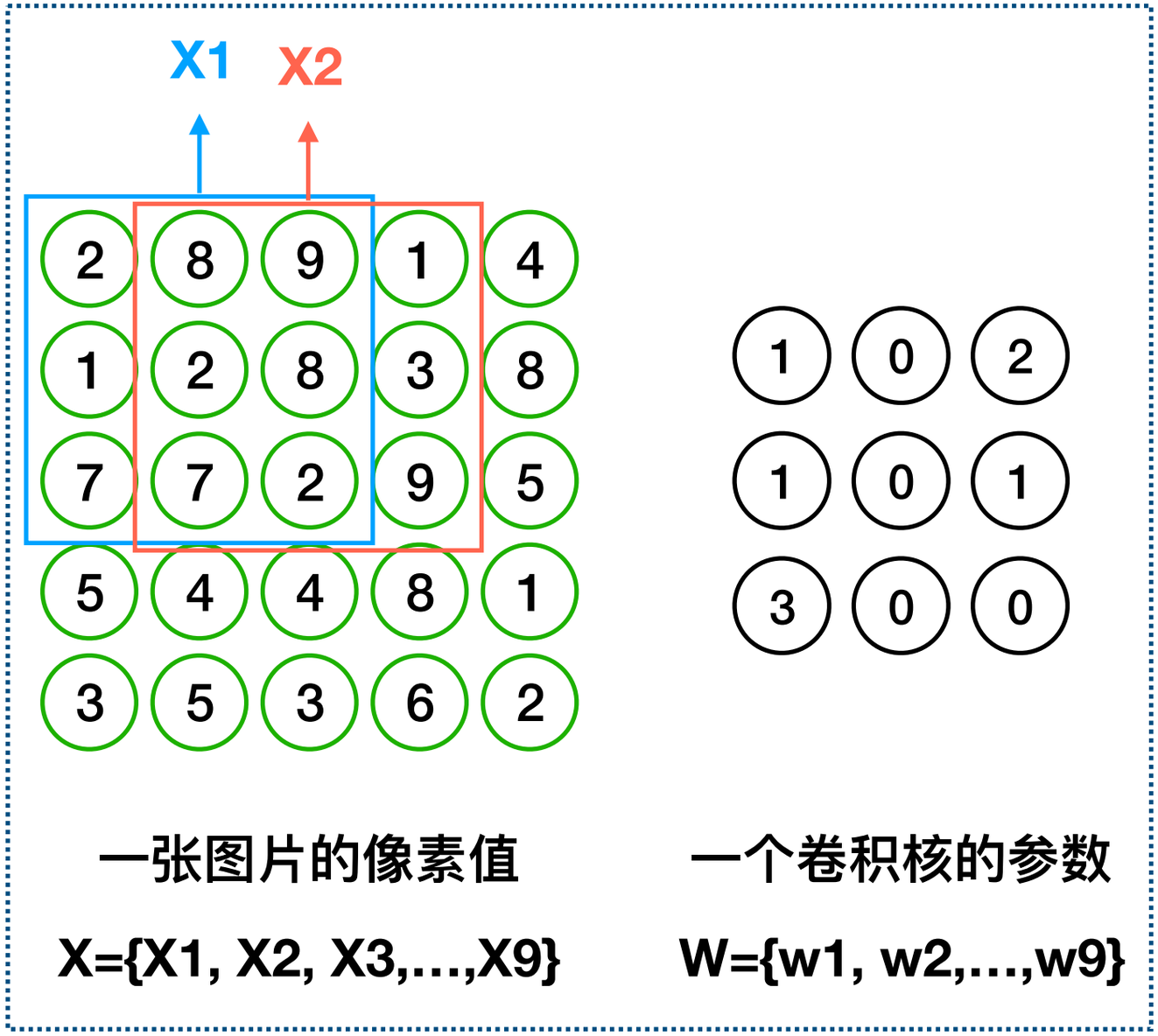
卷积核shape

卷积核的值

卷积核滑动步长

Author:WangQi
Email:wangqikaixin@gmail.com

卷积运算



一张图片的像素值

X={X1, X2, X3,…,X9}

一个卷积核的参数

W={w1, w2,…,w9}

此处我们规定a_n为第n个激活值

$$a_n = sum(X_n \odot W)$$

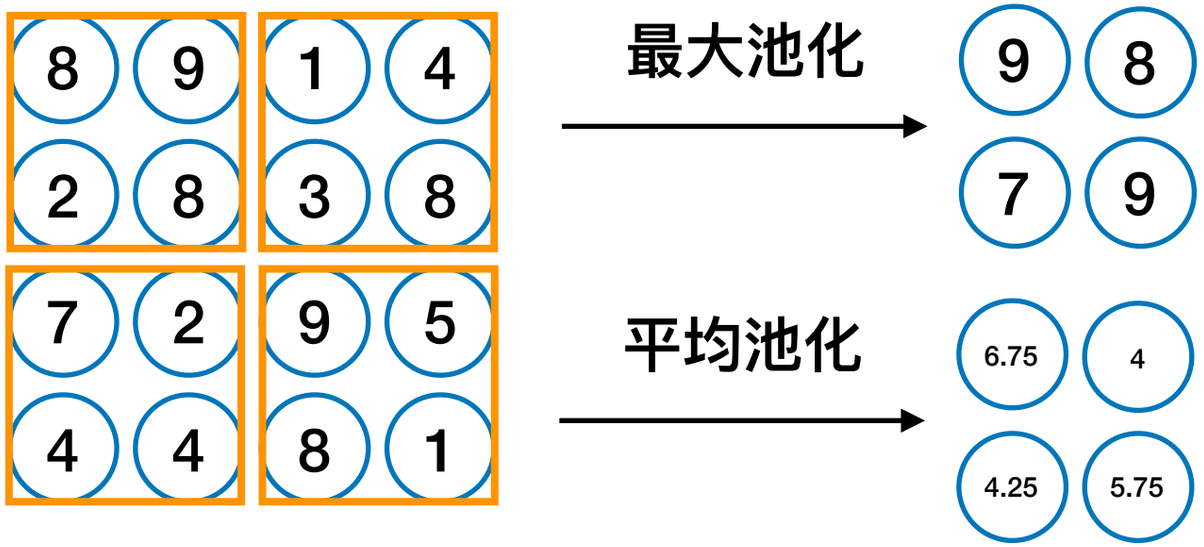
所以a可以看做是卷积后得到的特征图。

通常是非线性激活函数

带激活函数的卷积 $a_n = \text{activation\_function}(sum(X_n \odot W) + b_n)$

Author:WangQi
Email:wangqikaixin@gmail.com

池化运算



池化后的特征图

$$a_n = \max(z_n)$$

池化后的特征图

$$a_n = avg(z_n)$$

池化核要素：

- 池化核shape
- 滑动步长
- 池化方式

带激活函数的池化

$$a_n = \text{activation\_function}(\text{pooling}(z_n))$$

通常无激活函数或为线性激活函数

Author:WangQi
Email:wangqikaixin@gmail.com

卷积核的大小与卷积运算量

图片大小	卷积核大小	特征图大小	计算量（乘法次数）	是否合理
28px*28px	3*3	26*26	6084	是
28px*28px	5*5	24*24	14400	是
28px*28px	14*14	15*15	44100	否
28px*28px	20*20	9*9	32400	否

Author:WangQi
Email:wangqikaixin@gmail.com

2. 对特征图卷积

Author:WangQi
Email:wangqikaixin@gmail.com

对特征图进行卷积

思考：对特征图卷积的意义是什么？

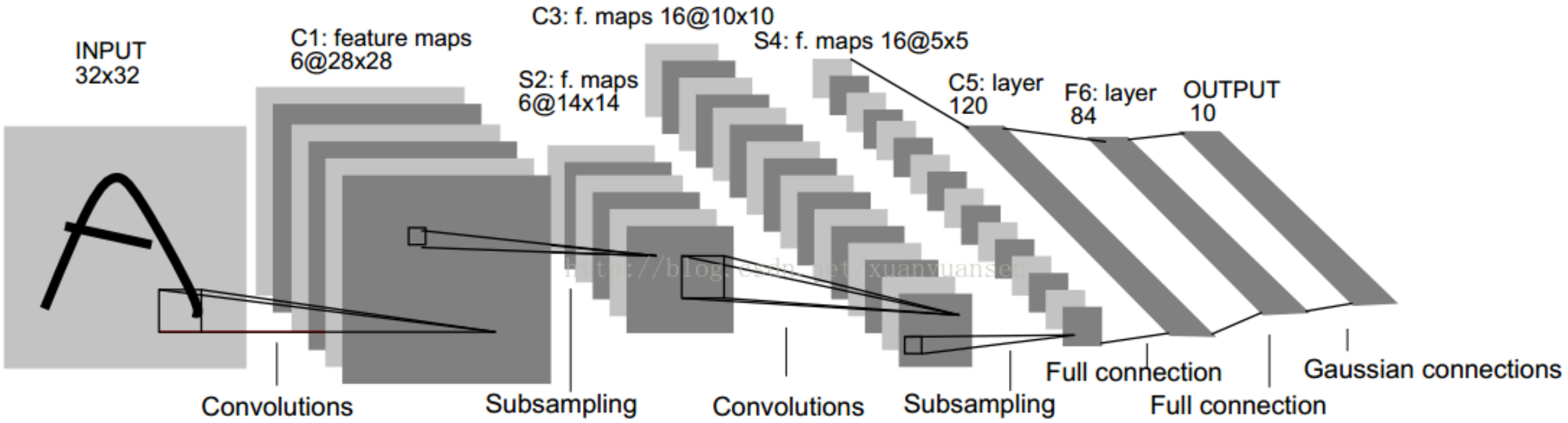


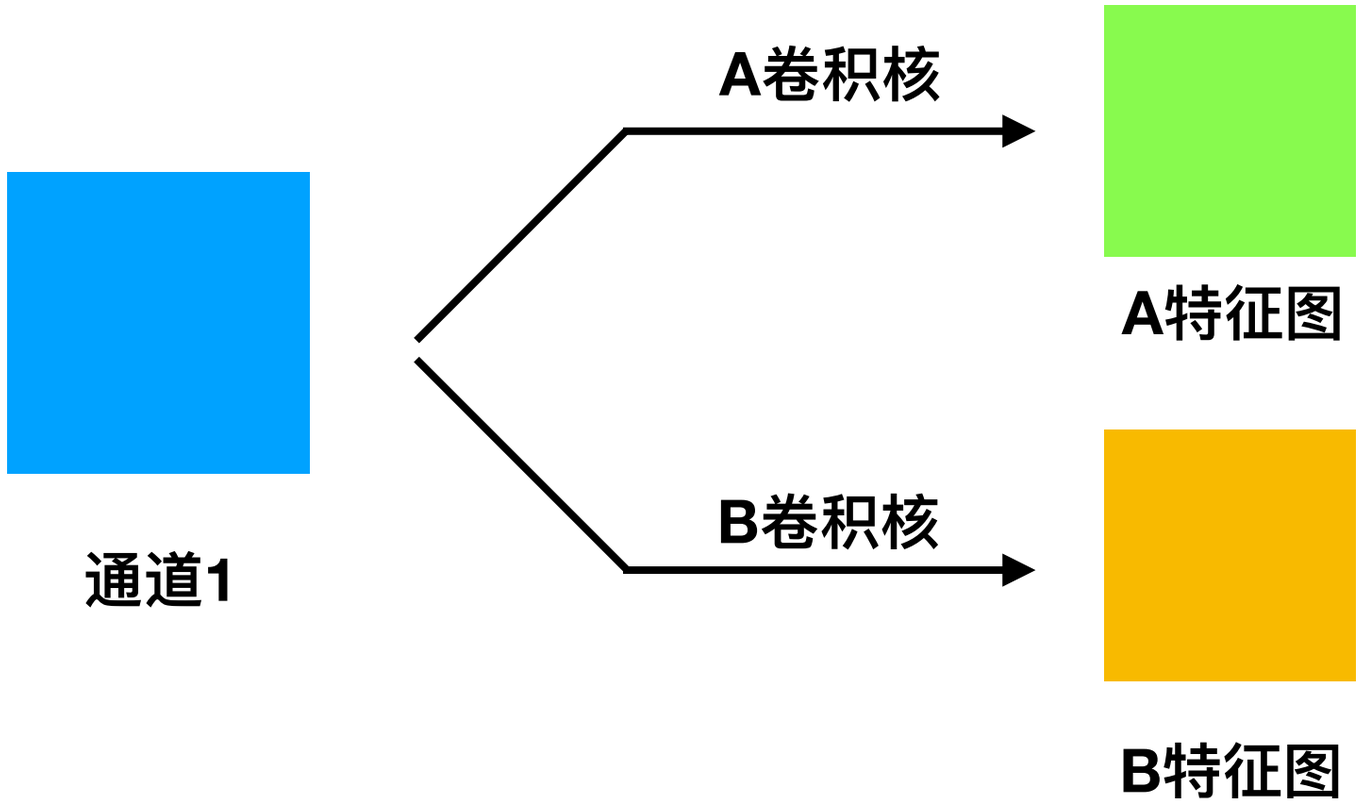
Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

特征提取：提取到了相对宏观的特征。

数据降维：多次卷积、池化使得数据的维度进一步降低。

Author:WangQi
Email:wangqikaixin@gmail.com

对特征图进行卷积



为了方便，我们将图片或者特征图统称为“通道”。

Author:WangQi
Email:wangqikaixin@gmail.com

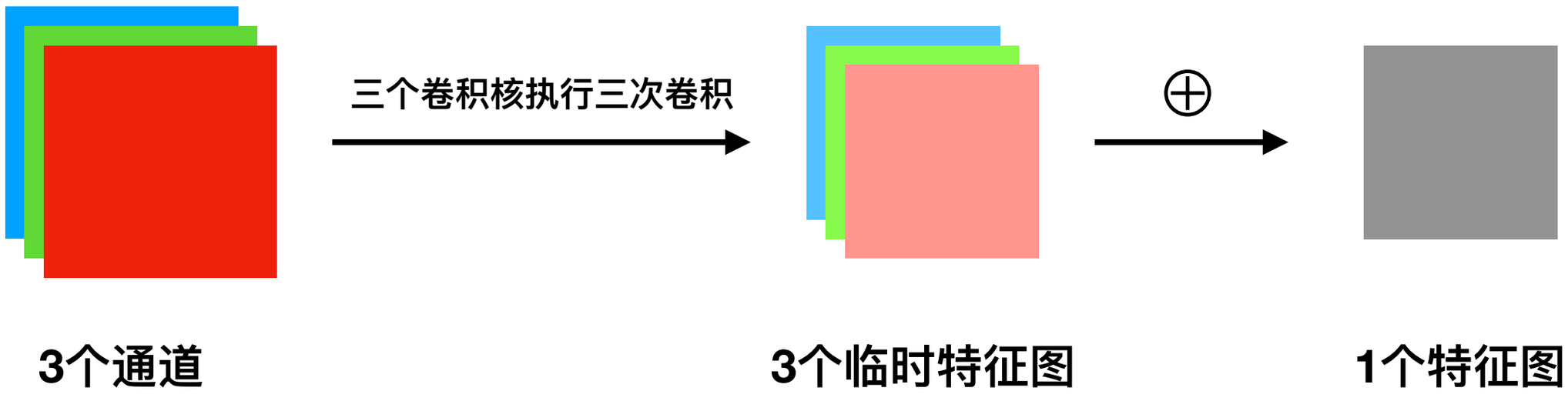
3. 多通道卷积

Author:WangQi
Email:wangqikaixin@gmail.com

3.1 多通道卷积

Author:WangQi
Email:wangqikaixin@gmail.com

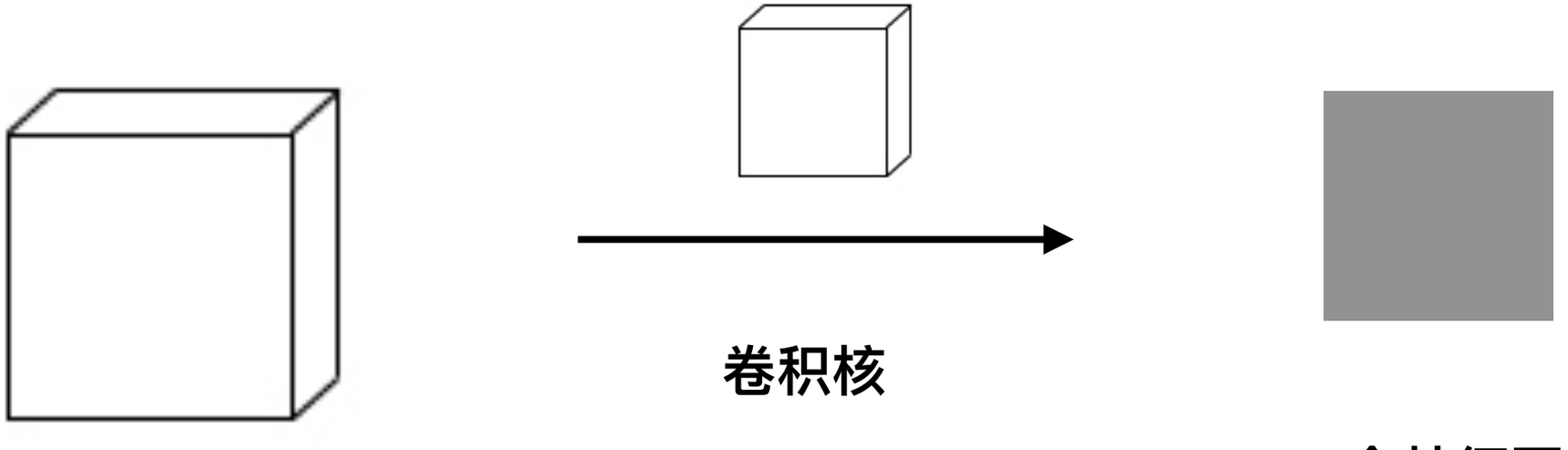
多通道“单核”卷积



注意：此处多通道“单核”卷积指的是每个通道分配一个卷积核。

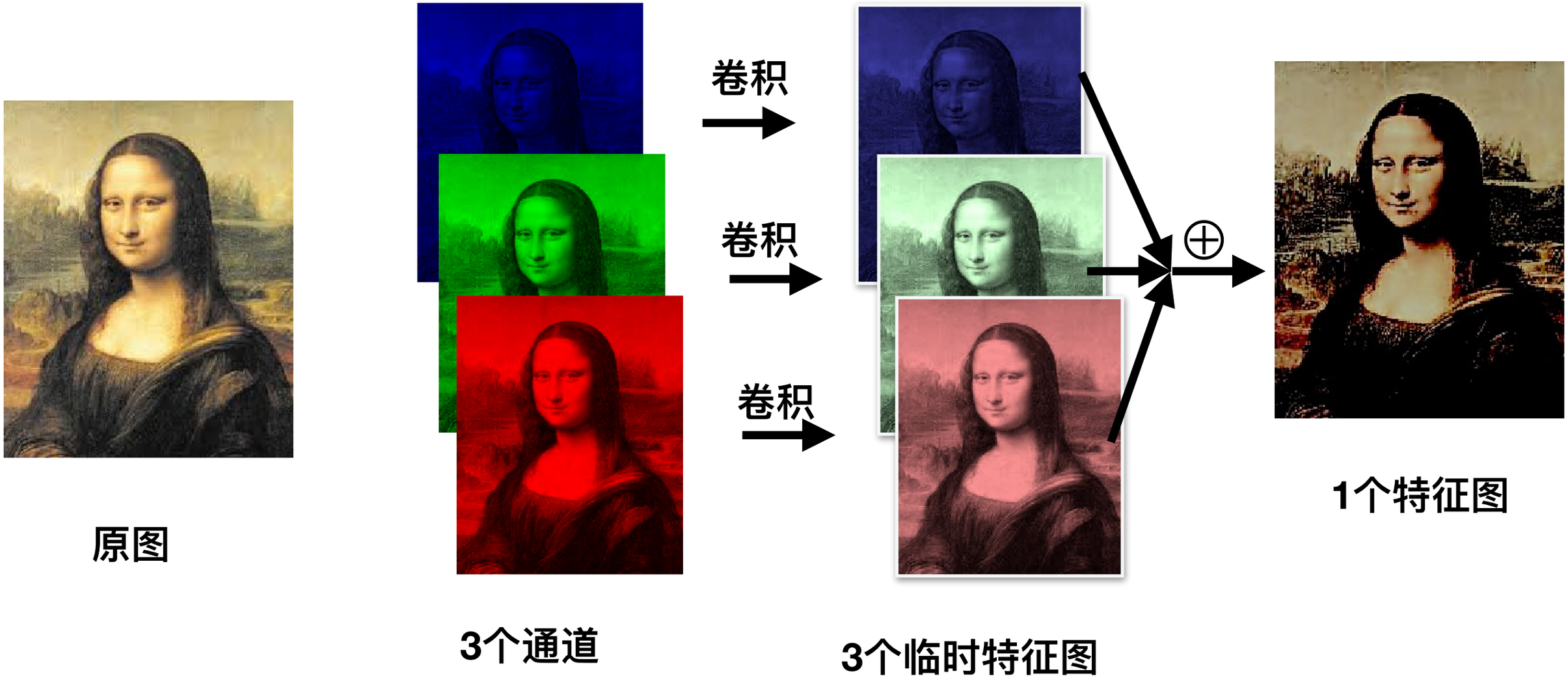
Author:WangQi
Email:wangqikaixin@gmail.com

多通道“单核”卷积



Author:WangQi
Email:wangqikaixin@gmail.com

多通道“单核”卷积示例



Author:WangQi
Email:wangqikaixin@gmail.com

思考：

1）多通道“单核”卷积中的卷积核数量与什么有关?

2）每个卷积核形状否相同?

3）每个卷积核值是否相同?

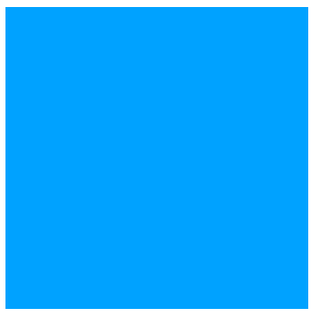
Author:WangQi
Email:wangqikaixin@gmail.com

3.2 单通道与多通道卷积对比

Author:WangQi
Email:wangqikaixin@gmail.com

单通道与多通道卷积对比

单通道



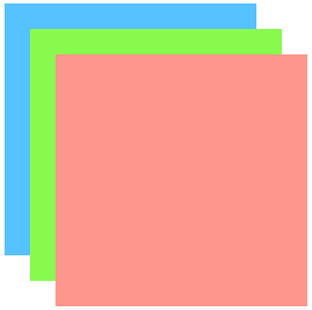
1个卷积核执行1次卷积

1个通道

1个特征图

多通道

三个卷积核执行三次卷积



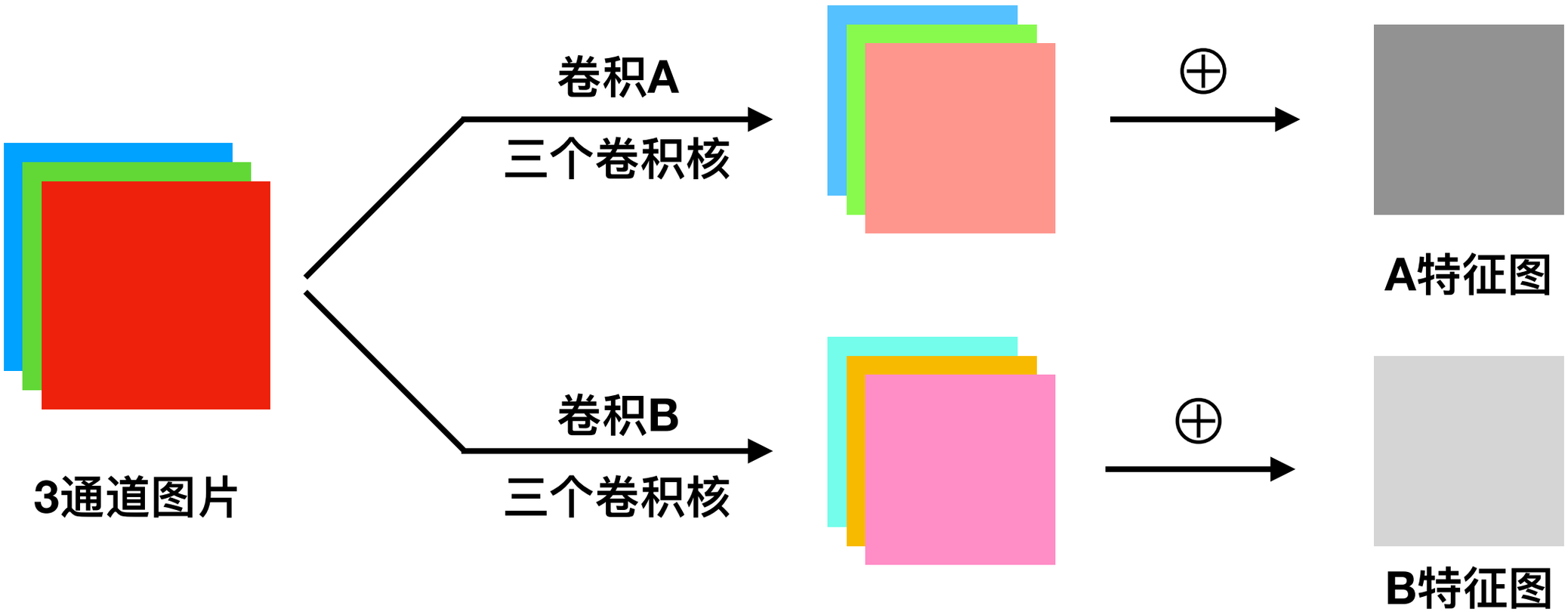
3个通道

3个临时特征图

1个特征图

Author:WangQi
Email:wangqikaixin@gmail.com

多通道“多核”卷积



Author:WangQi
Email:wangqikaixin@gmail.com

小练习

假如有一张大小为12px*12px大小的png图片，用2个3*3大小的卷积核处理。

问题1：得到了几个特征图?

问题2：每个特征图的大小分别是多少?

问题3：“每个卷积”对应多少个卷积核?

问题4：所有卷积核一共有多少个参数?

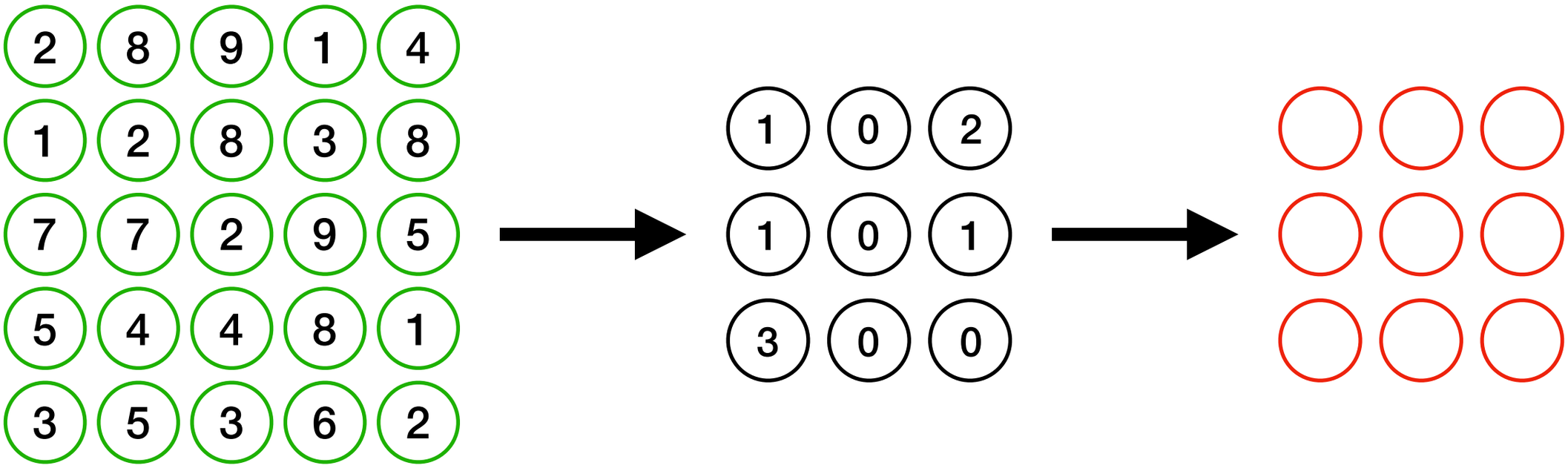
Author:WangQi
Email:wangqikaixin@gmail.com

4. 边界与步长

Author:WangQi
Email:wangqikaixin@gmail.com

卷积与池化的问题1

卷积中不同元素参与卷积计算的次数不同



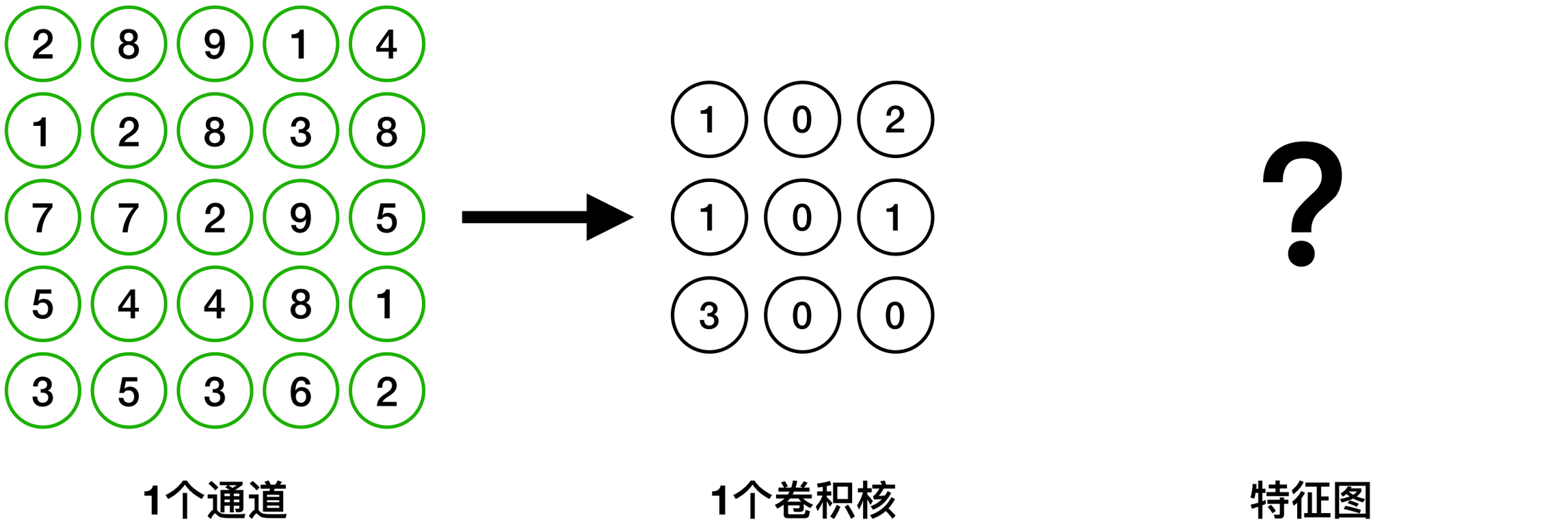
1个通道　　1个卷积核　　1个特征图

Author:WangQi
Email:wangqikaixin@gmail.com

卷积与池化的问题2

卷积与池化的步长不同时，边界处如何计算?

下图卷积步长为3时，特征图是怎样的?



Author:WangQi
Email:wangqikaixin@gmail.com

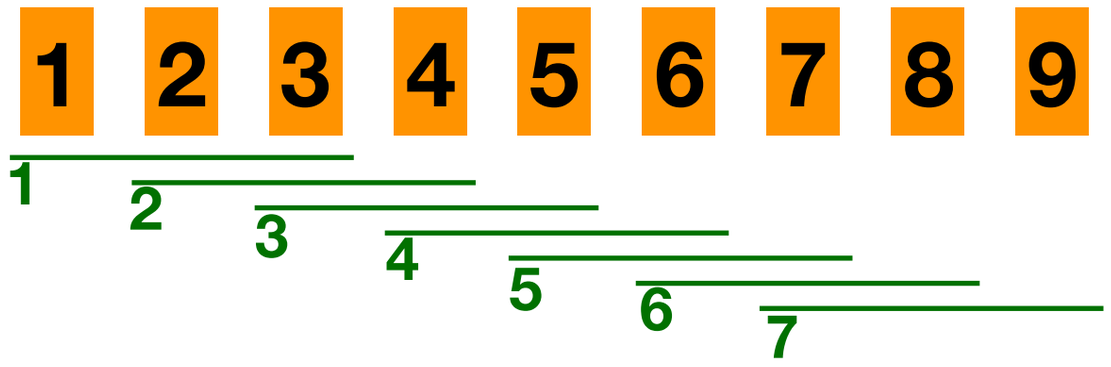
4.1 VALID边界

Author:WangQi
Email:wangqikaixin@gmail.com

VALID边界处理

padding="VALID"

width: 9
kernel: 3
strides: 1

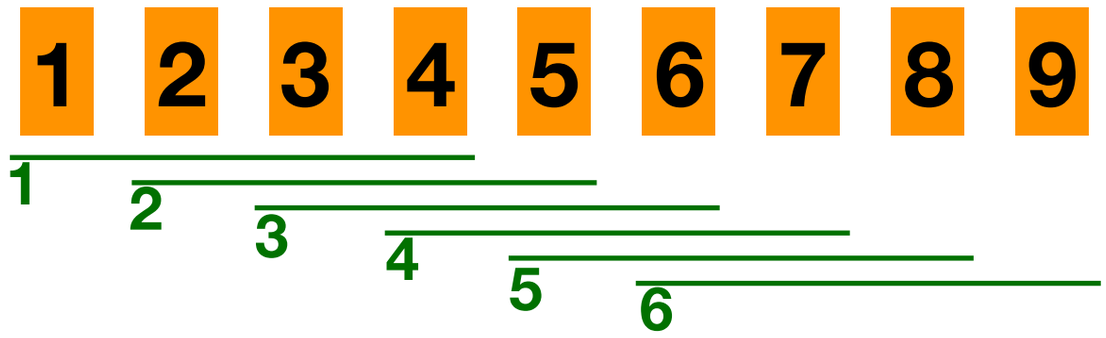


VALID：只利用有效数据。

Author:WangQi
Email:wangqikaixin@gmail.com

VALID边界处理

padding="VALID"



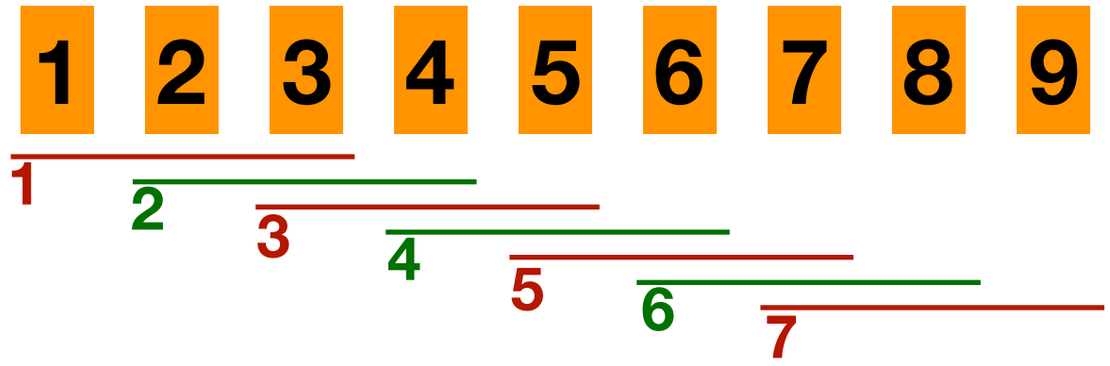
width: 9
kernel: 4
strides: 1

out_width = in_width - kernel_width + 1

Author:WangQi
Email:wangqikaixin@gmail.com

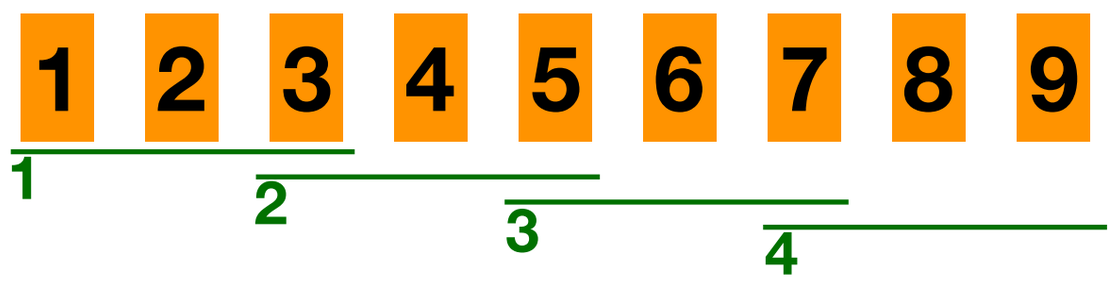
VALID边界处理

padding="VALID"



width: 9
kernel: 3
strides: 1

padding="VALID"

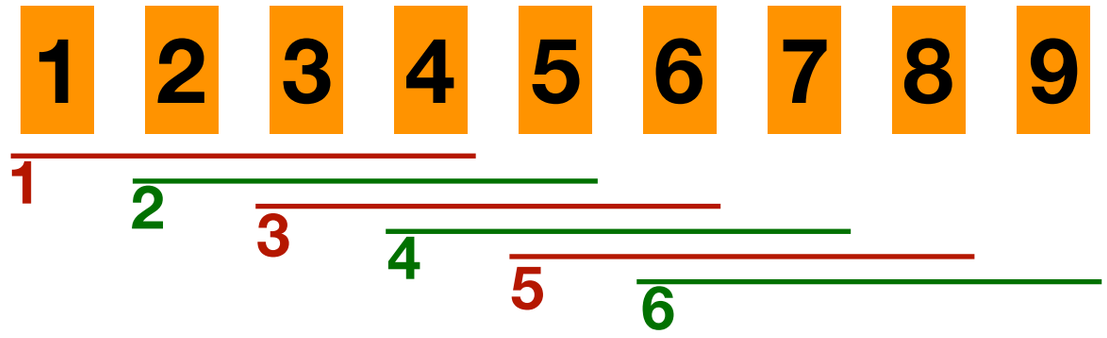


width: 9
kernel: 3
strides: 2

Author:WangQi
Email:wangqikaixin@gmail.com

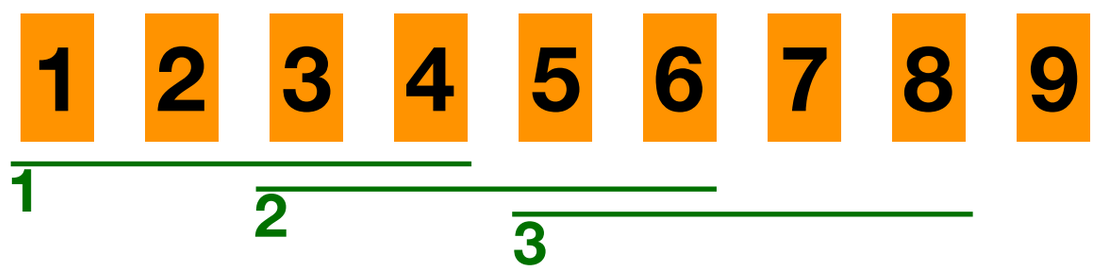
VALID边界处理

padding="VALID"



width: 9
kernel: 4
strides: 1

padding="VALID"

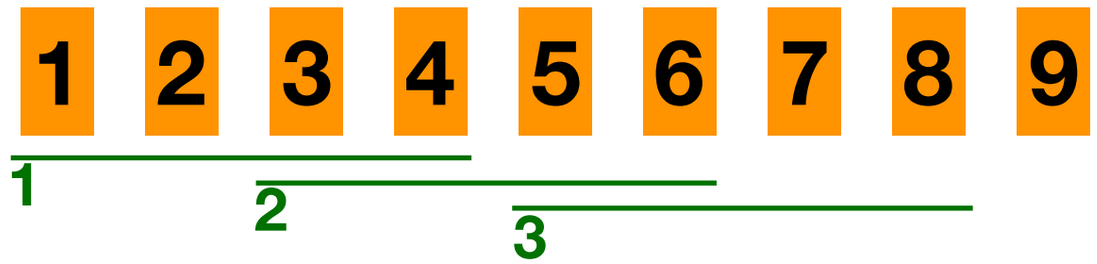


width: 9
kernel: 4
strides: 2

Author:WangQi
Email:wangqikaixin@gmail.com

VALID边界处理

padding="VALID"



width: 9
kernel: 4
strides: 2

大于1的步长的输出相当于对等于1的步长的输出做了一个平均采样

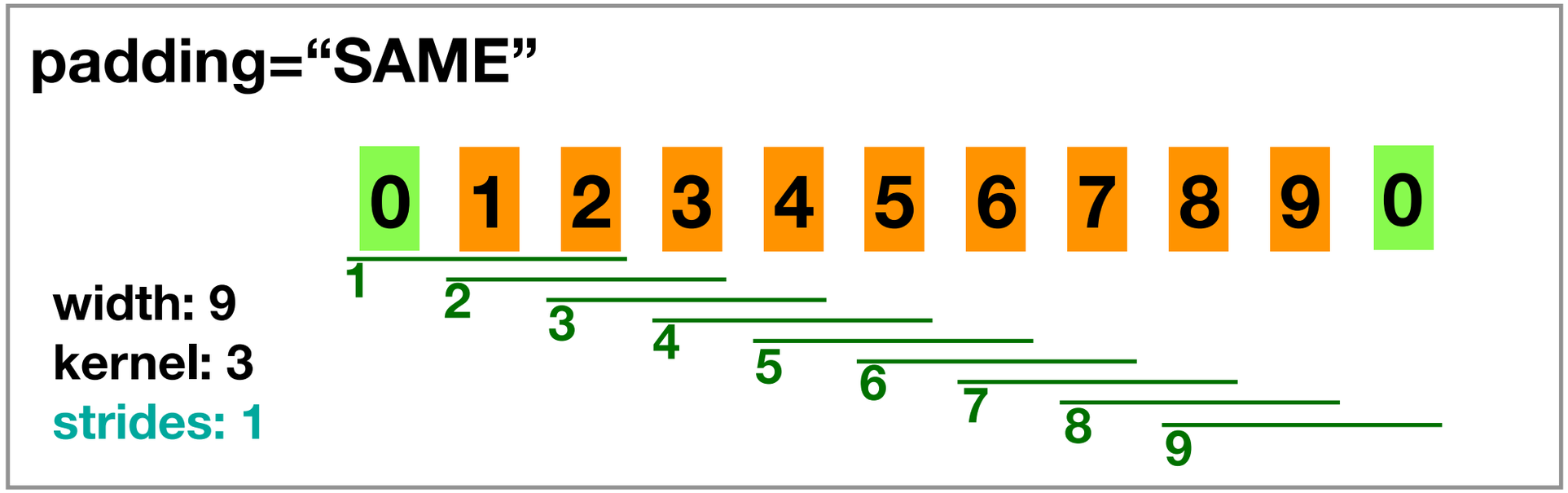
out_width = ceil(float(in_width - kernel_width + 1) / float(strides))

Author:WangQi
Email:wangqikaixin@gmail.com

4.2 SAME边界

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

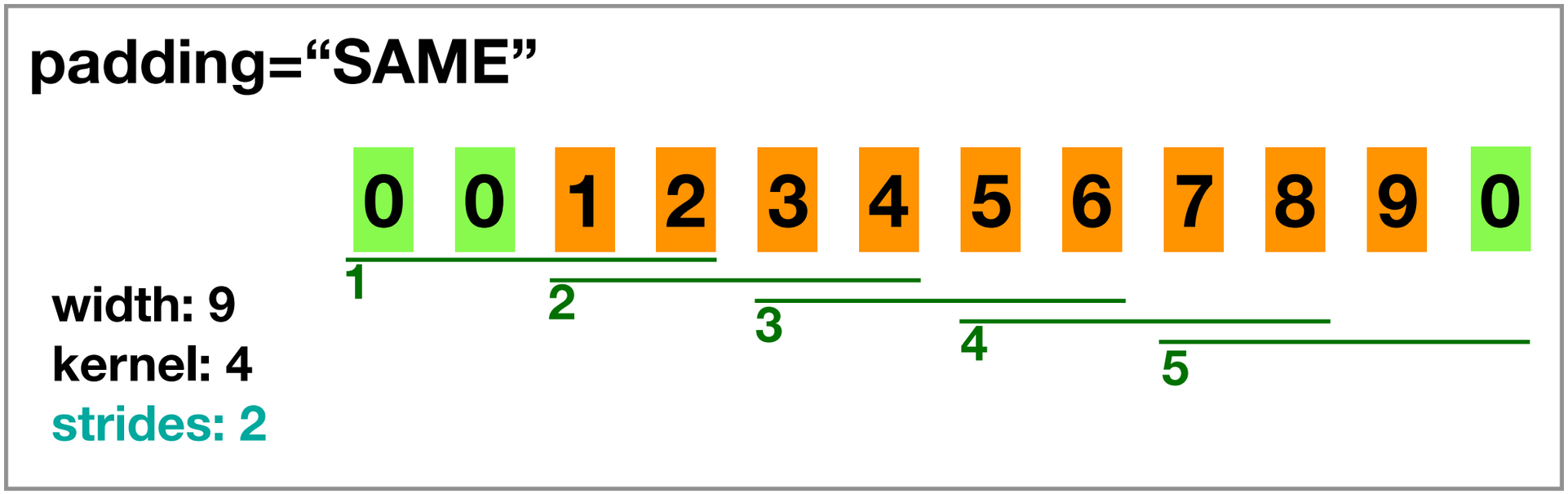


SAME：通常需要边界填充。

规定：out_width = ceil(float(in_width) / float(strides))

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理



规定：out_width = ceil(float(in_width) / float(strides))

pad：？

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

输入边界与输出边界的大小关系

out_width = ceil(float(in_width) / float(strides))

输入边界与输出边界的变换关系

out_width = ceil(float(in_width + pad - kernel_width + 1) / float(strides))

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

out_width = ceil(float(in_width + pad - kernel_width + 1) / float(strides))

o　　i　　p　　k　　s

$$o = \left\lceil \frac{i + p - k + 1}{s} \right\rceil$$

p?

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

$$o = \left\lceil \frac{i + p - k + 1}{s} \right\rceil$$

$$o - 1 < \frac{i + p - k + 1}{s} \leq o$$

$$s(o-1) - i + k - 1 < p \leq so - i + k - 1$$

p的取值是一个范围，即在这个范围内填充均可。 使用不同的p填充的意义是什么?

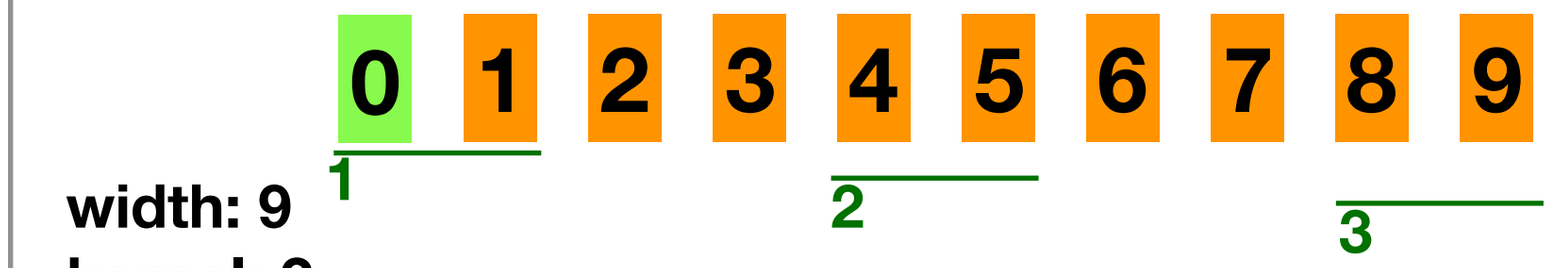
Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

$$s(o-1)-i+k-1<p\leq so-i+k-1$$

i=9, k=2, s=4, 则o=3, p=1, 2, 3, 4　当p=1时：

padding="SAME"



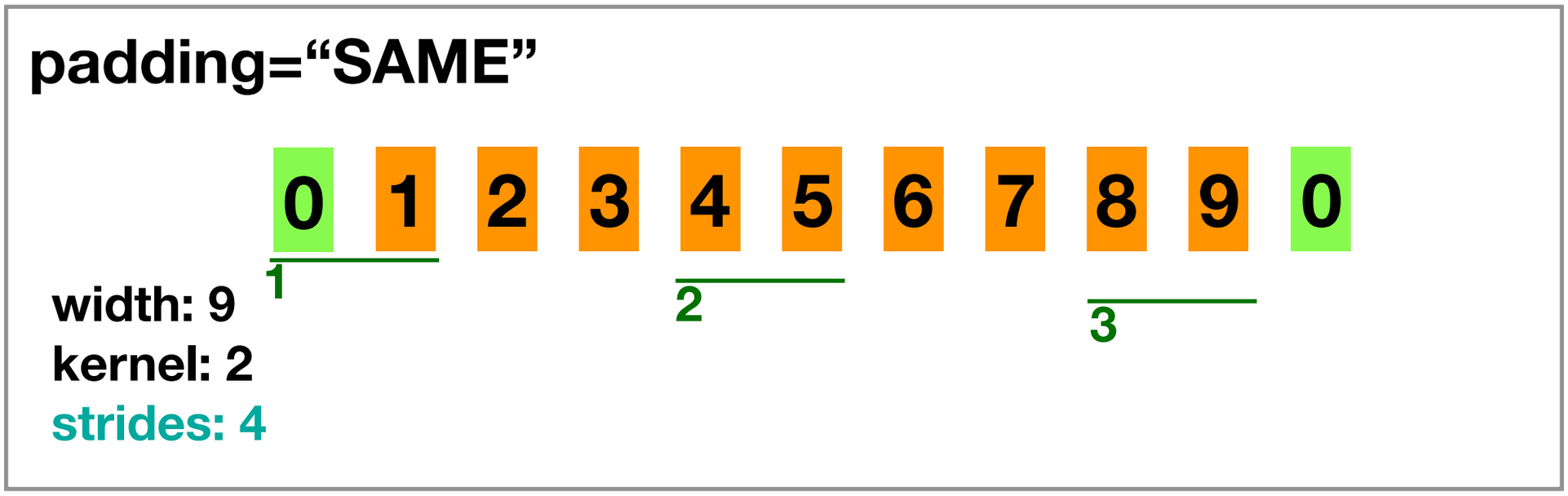
width: 9
kernel: 2
strides: 4

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

$$s(o-1)-i+k-1<p\leq so-i+k-1$$

i=9, k=2, s=4, 则o=3, p=1, 2, 3, 4　当p=2时：



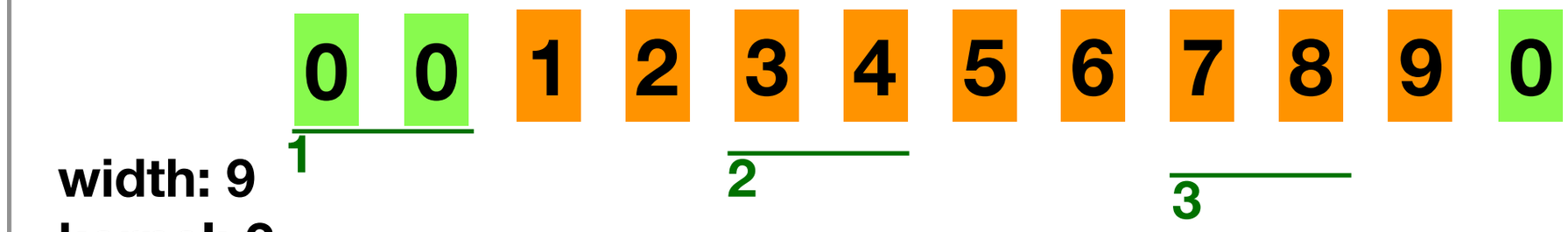
Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

$$s(o-1)-i+k-1<p\leq so-i+k-1$$

i=9, k=2, s=4, 则o=3, p=1, 2, 3, 4　当p=3时：

padding="SAME"

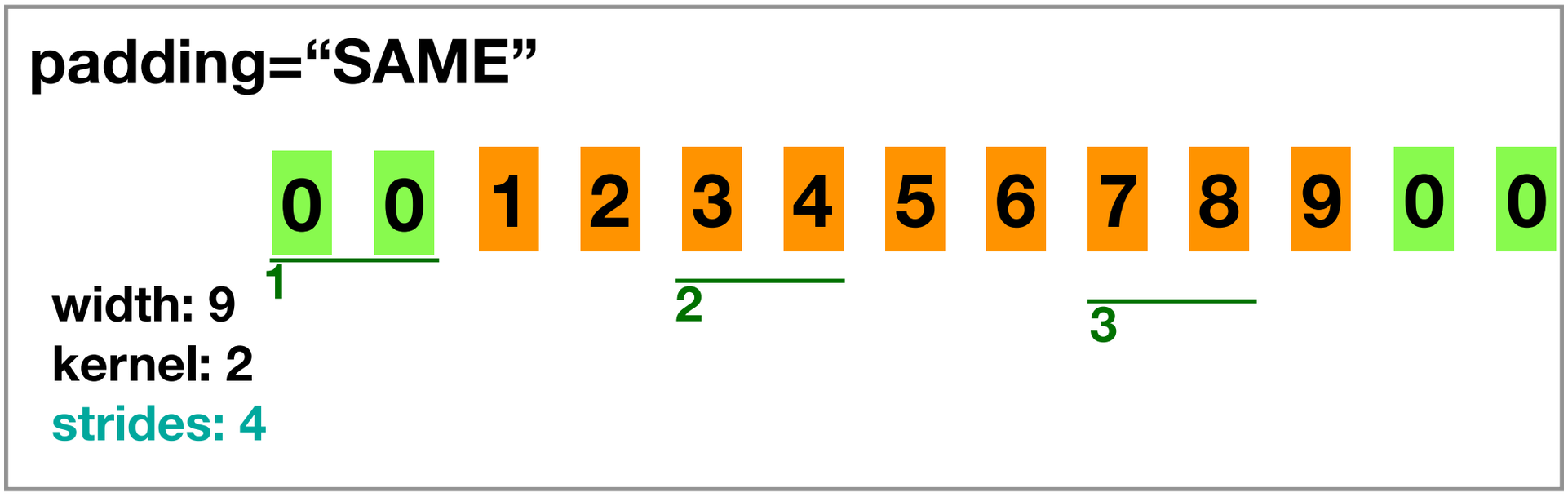


Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

$$s(o-1)-i+k-1<p\leq so-i+k-1$$

i=9, k=2, s=4, 则o=3, p=1, 2, 3, 4　当p=4时：



Author:WangQi
Email:wangqikaixin@gmail.com

思考：p可能是负数吗?

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

$$s(o-1)-i+k-1<p\leq so-i+k-1$$

$$p_{\min}=s(o-1)-i+k$$

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

$$s(o-1)-i+k-1<p\leq so-i+k-1$$

简化 $p_{\min}=s(o-1)-i+k$

$$if\,(i \bmod s=0):p=\max(k-s,0)$$

$$if\,(i \bmod s\neq 0):p=\max(k-(i \bmod s),0)$$

Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

$$if\,(i \bmod s = 0) : p = \max(k - s, 0)$$

$$if\,(i \bmod s \neq 0) : p = \max(k - (i \bmod s), 0)$$

思考：已知P的大小，填充的0可以放置在哪个位置合适?

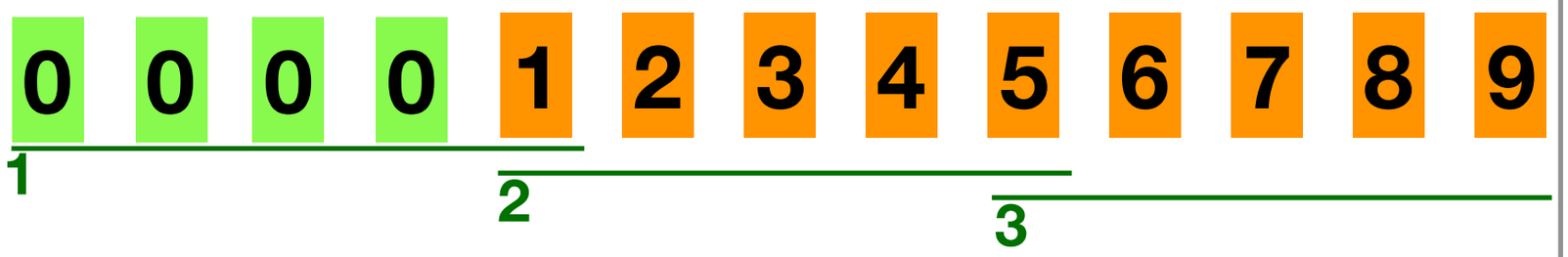
Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

width: 9
kernel: 5
strides: 4　　**已知pad=4**

填充法一　NO

padding="SAME"



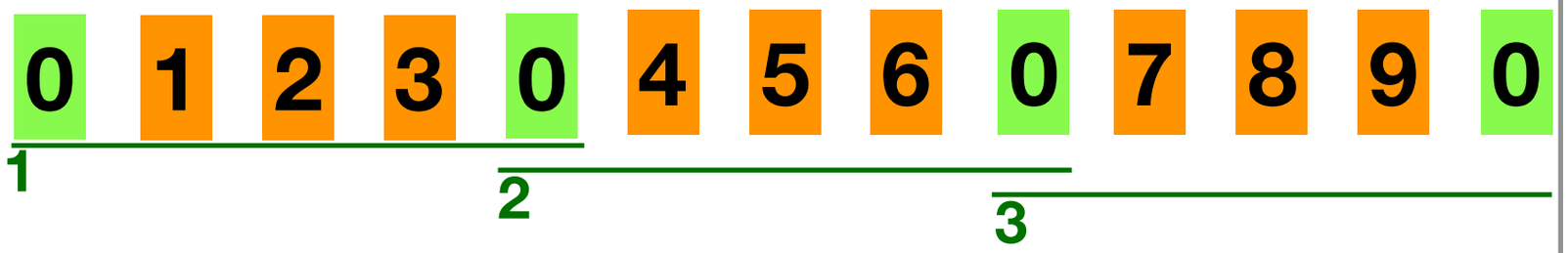
Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

width: 9
kernel: 5
strides: 4　　**已知pad=4**

填充法二　NO

padding="SAME"

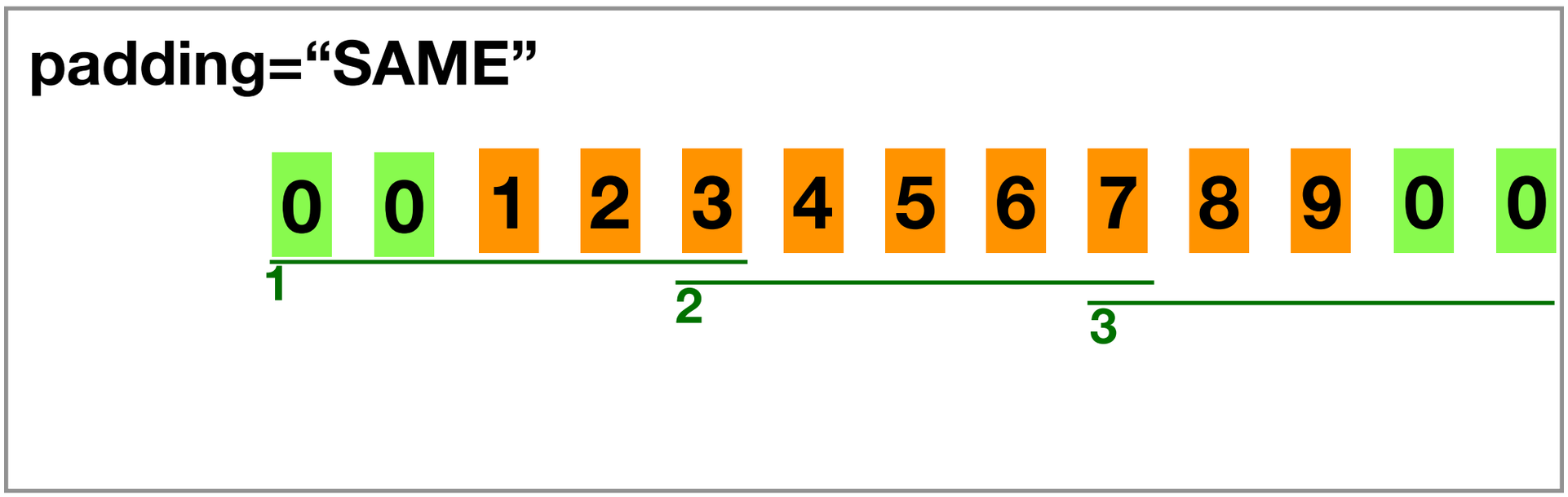


Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

width: 9
kernel: 5
strides: 4　　已知pad=4

填充法三　YES



Author:WangQi
Email:wangqikaixin@gmail.com

SAME边界处理

常用填充方法：

$$top\_pad = height\_pad // 2$$

$$bottom\_pad = height\_pad - top\_pad$$

$$left\_pad = width\_pad // 2$$

$$right\_pad = width\_pad - left\_pad$$

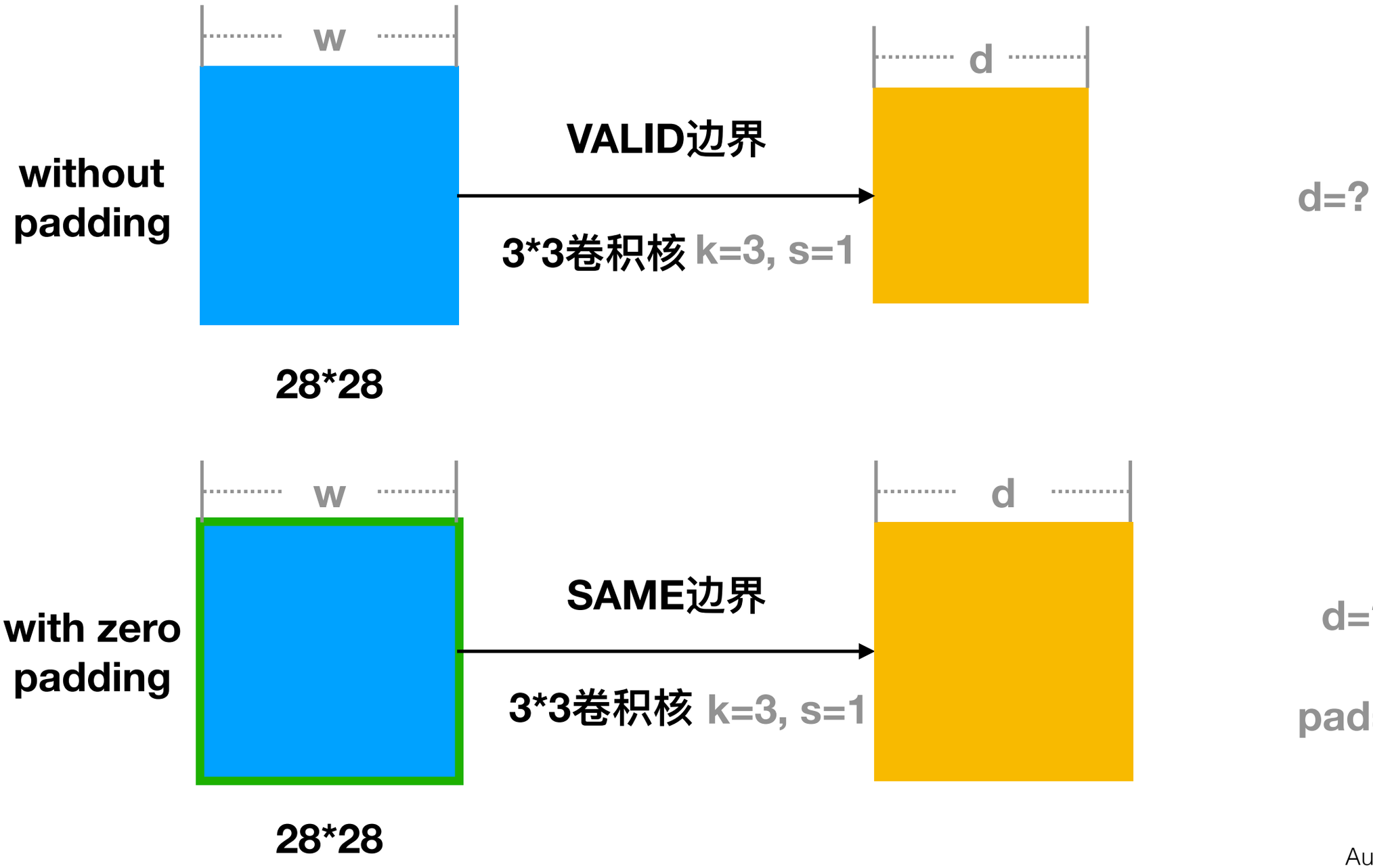
即：左奇右偶，上奇下偶。

Author:WangQi
Email:wangqikaixin@gmail.com

4.3 示例

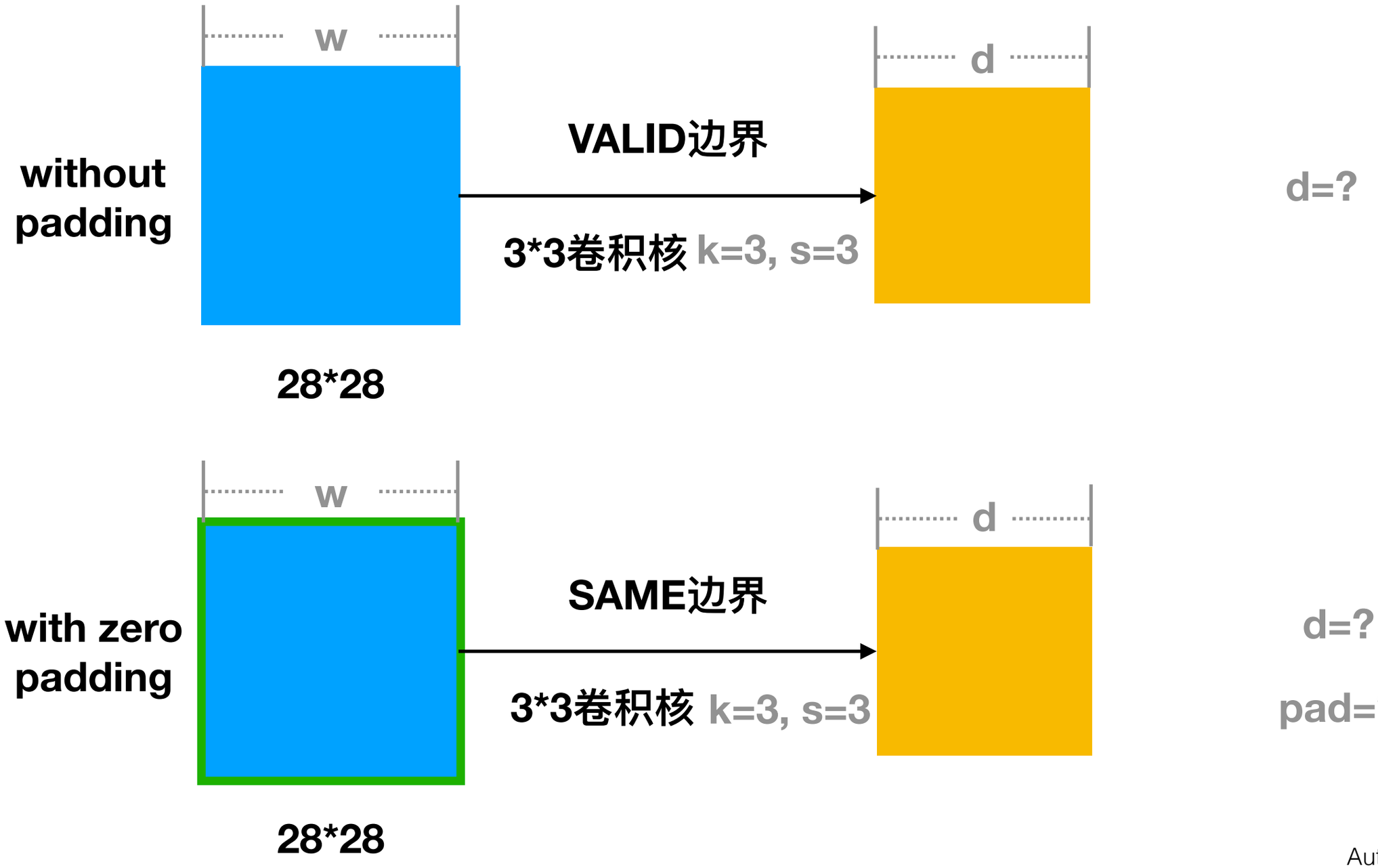
Author:WangQi
Email:wangqikaixin@gmail.com

卷积边界处理示例



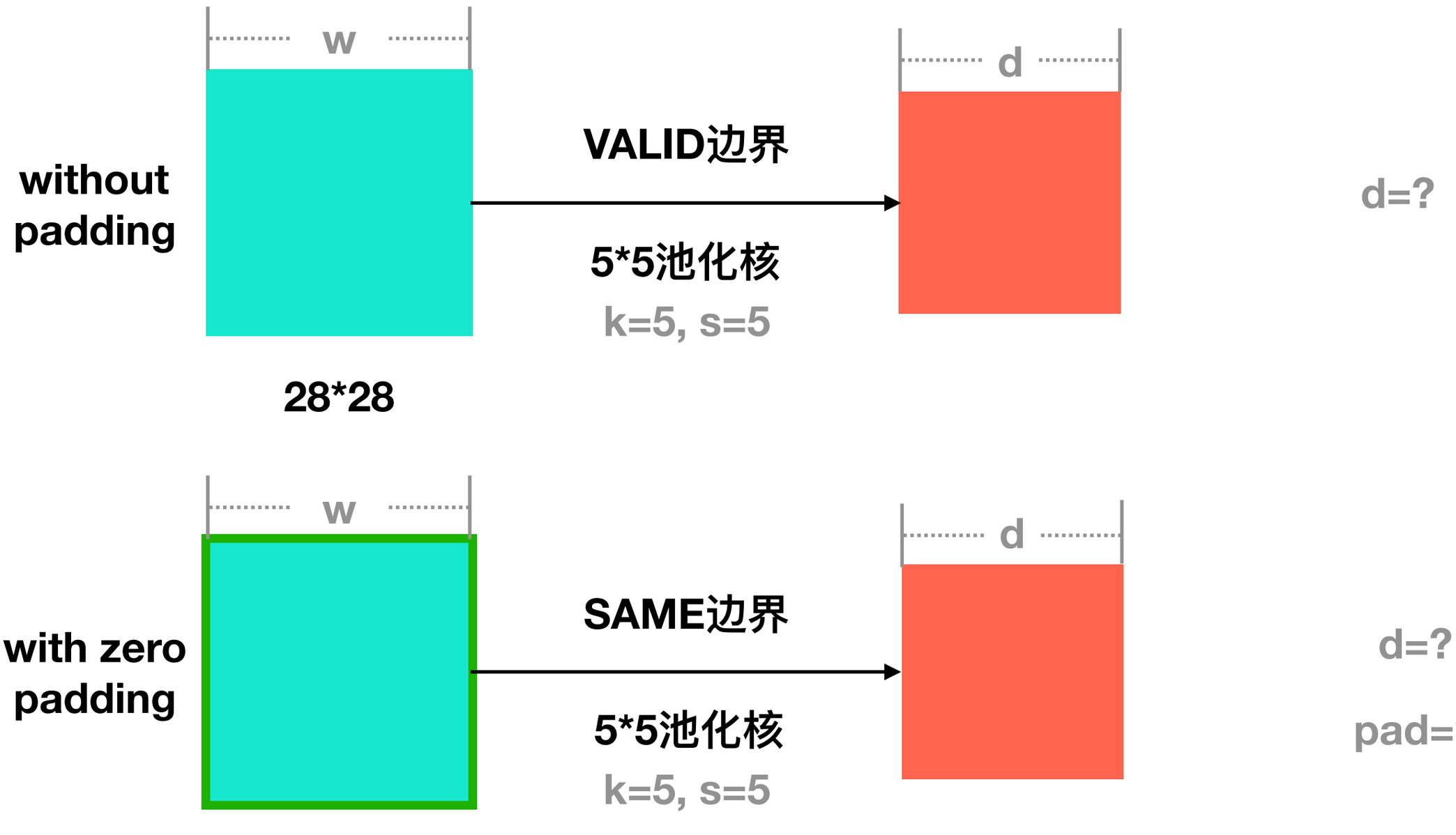
Author:WangQi
Email:wangqikaixin@gmail.com

卷积边界处理示例



Author:WangQi
Email:wangqikaixin@gmail.com

池化常用边界处理



Author:WangQi
Email:wangqikaixin@gmail.com

边界处理

思考：SAME边界，除了填充0还能填充什么？

Author:WangQi
Email:wangqikaixin@gmail.com

小练习

已知单通道的某图片大小为13*15，现使用6*4大小的卷积核以5*7的步长进行卷积，请问：

问题1：使用VALID的边界处理方式，feature map大小是多少?

问题2：如果SAME的边界处理方式，feature map大小是多少？补0的长度是多少?

	边长	ksize	strides	VALID	SAME	SAME pad
width	13	6	5	**2**	**3**	**3**
height	15	4	7	**2**	**3**	**3**

Author:WangQi
Email:wangqikaixin@gmail.com

5. 其它

Author:WangQi
Email:wangqikaixin@gmail.com

平移不变性

训练时使用的图片

预测时使用的图片

卷积与池化均拥有平移不变性，即更能适应图片的特征位置变化。

Author:WangQi
Email:wangqikaixin@gmail.com

平移不变性

卷积：卷积得到的map是微观特征的激活值，所以特征位置改变时，对应位置的最激活值位置也改变了，但特征之间的相对位置具有一定的不变性。而更深层的网络学习到的是微观特征的组合方式，只要特征的组合方式没有变化，CNN就能识别。

池化：池化操作会在一定范围内检测特征。以最大池化为例，特征位置改变了，但在池化滑动窗的范围内，最大值没有变。

Author:WangQi
Email:wangqikaixin@gmail.com

思考：存在1*1卷积核吗？如果存在，卷积的结果是什么？有什么意义？

1*1卷积核是存在的，如果1个特征图通过1*1卷积得到了一个新的特征图，则两个特征图的关系是线性的（卷积激活函数是线性的情况下），这时候1*1卷积意义不大。但如果有多个通道的特征图通过1*1卷积可以实现通道维度的改变，同时也实现了多个通道信息的汇聚。

Author:WangQi
Email:wangqikaixin@gmail.com

小结

- 卷积运算的过程与意义。
- 多核卷积可以得到多个特征图，其数量与核数量相同。
- 多通道多核卷积中的卷积方式。
- 卷积核通常设置比较小，这样运算量比较小，且容易提取到局部特征。
- 卷积与池化常用边界处理包括VALID与SAME两种。VALID可能丢弃一部分信息，SAME通常使用0填充边界。

Author:WangQi
Email:wangqikaixin@gmail.com

THANKS

Author:WangQi
Email:wangqikaixin@gmail.com