Lenet\_5实现手写数字识别及其实现

本文针对lenet-5各部分进行了详细的介绍。出自论文Gradient-Based Learning Applied to Document Recognition

**一、基础概念**

为更好地了解lenet5原理，本部分介绍卷积和池化。

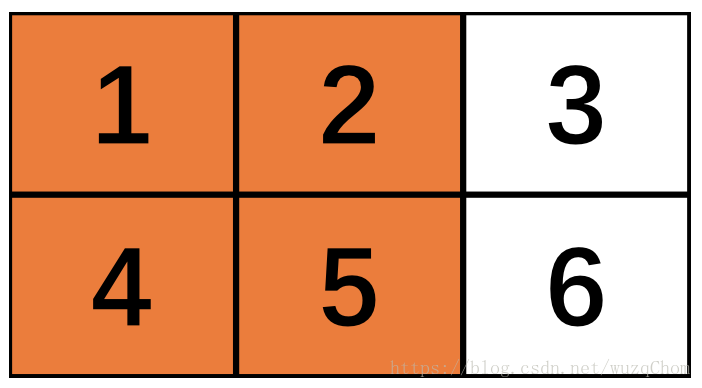
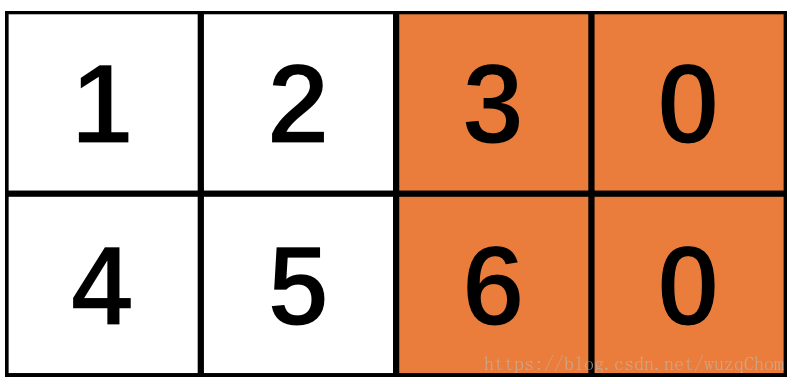
1、卷积

卷积层是卷积神经网络的核心，卷积核（亦称滤波器）能对原始矩阵进行不同特征的提取。

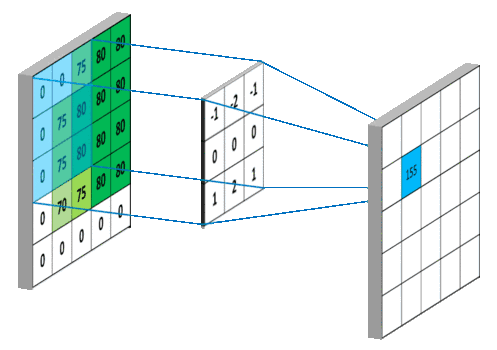
卷积的两种方式：

1. Padding=same

原始图片如图所示，若对原始图片用2\*2的卷积核，步长为2的方式进行卷积。第一次对（1，2，3，4）所在区域卷积，并无任何问题，但是当卷积窗口向右滑动两列时，发现余下的窗口无法完成卷积，而padding=same的方式，会进行填充，如图所示。

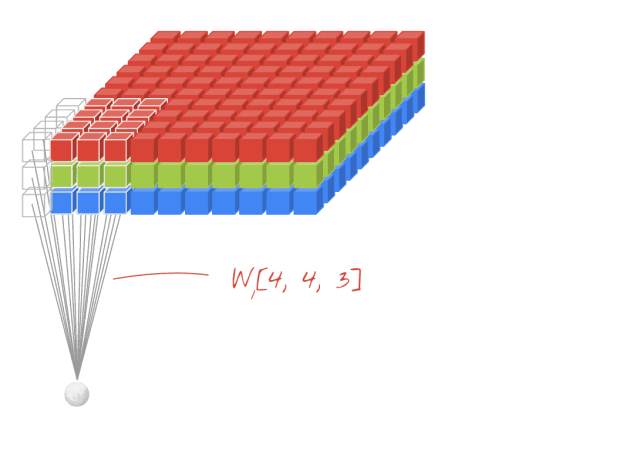
 

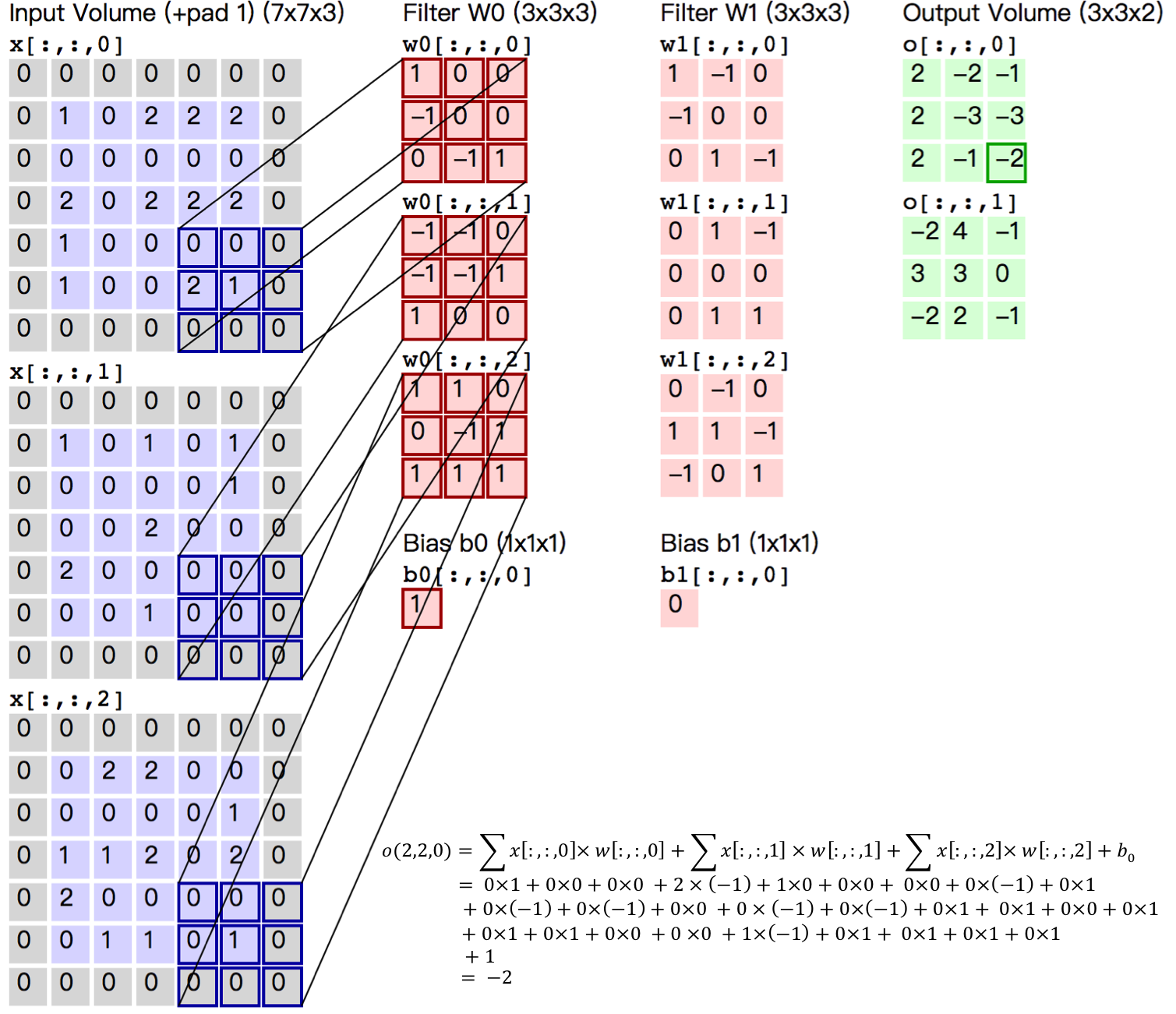
两维



三维

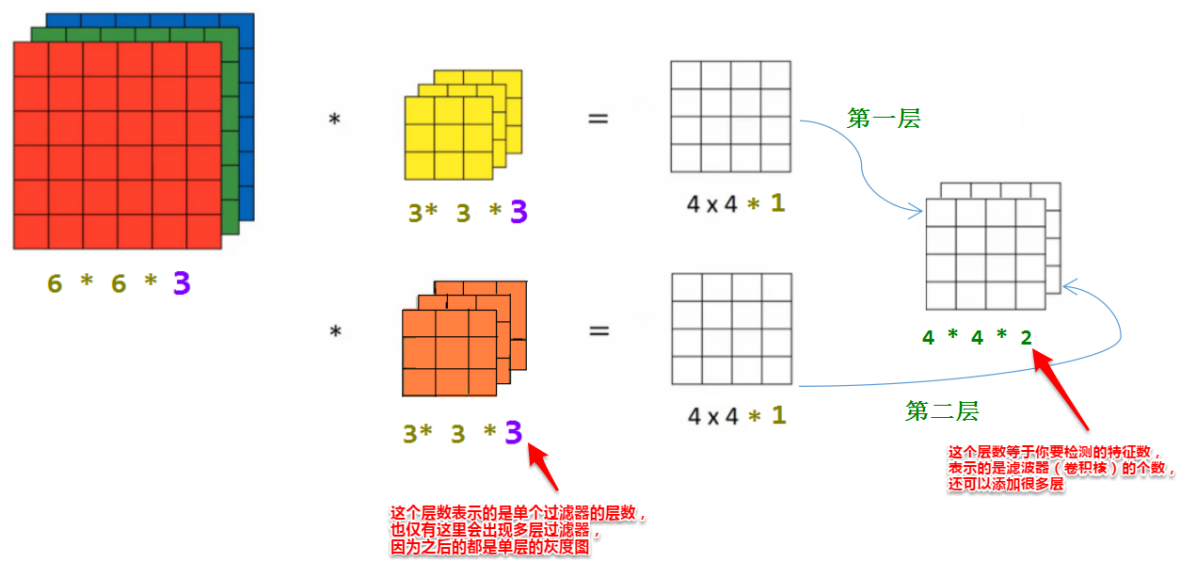
当输入的神经元有三通道时，这时的卷积核的大小不止用长宽来表示，还有深度，感受视野也对应有了深度。在图像卷积操作中，神经元在空间维度上是局部连接，但在深度上是全连接，下图给出了三通道神经元的卷积计算过程。





（2）Padding=valid

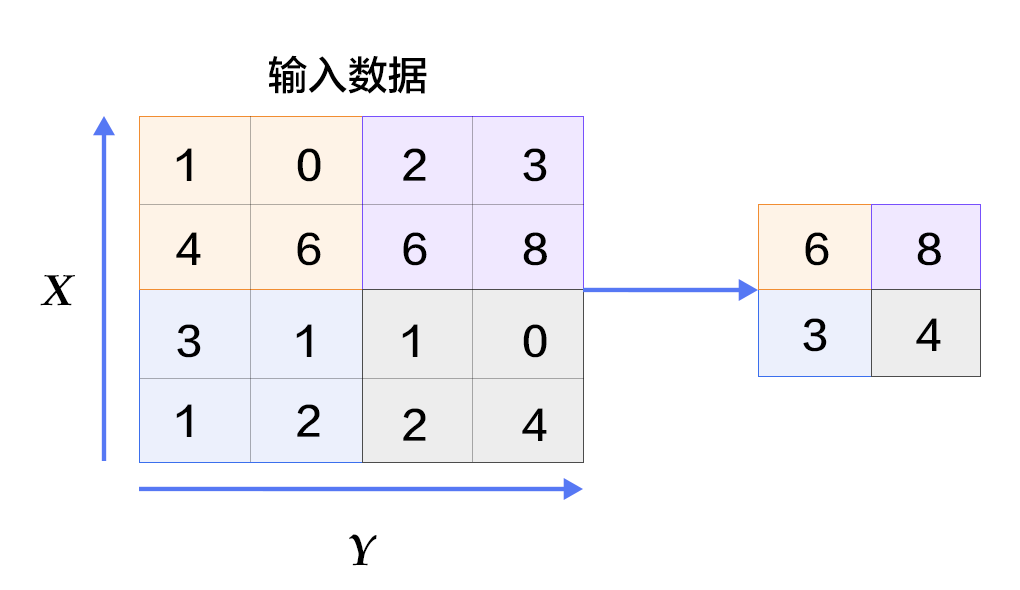
如（1）中的例子，对于padding=valid会直接将第三列丢弃。



2、池化层（亦称下采样）

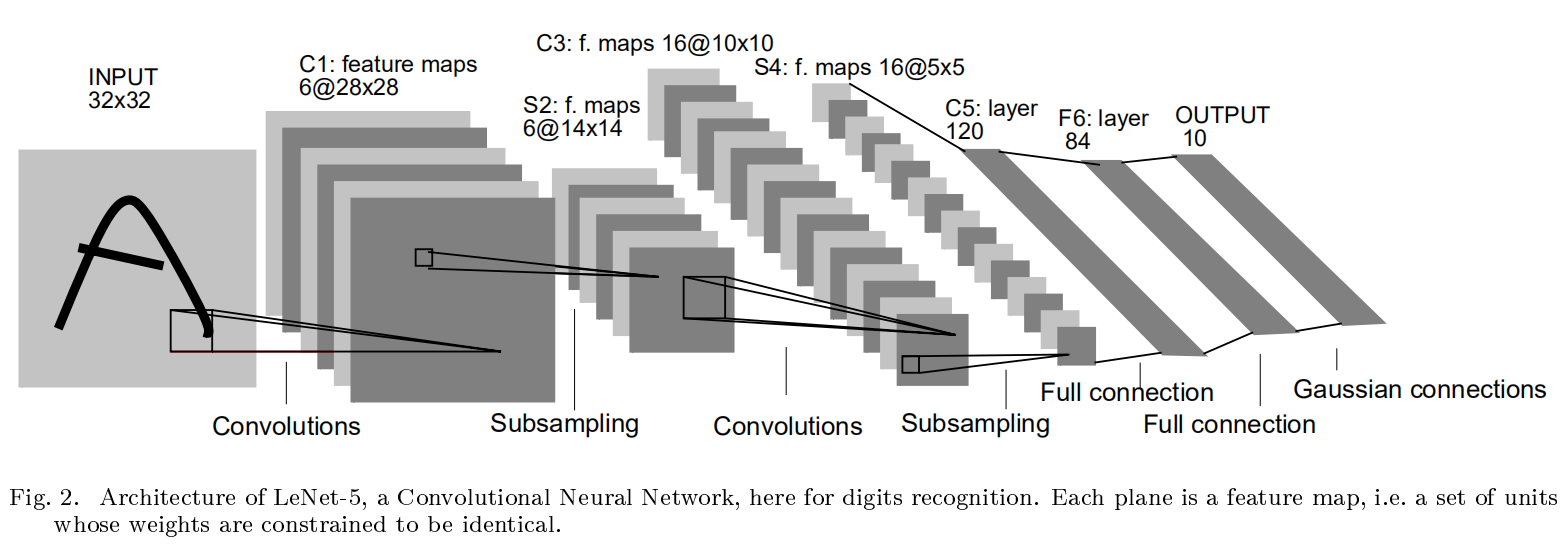
池化方式简介

池化是非线性下采样的一种形式，主要作用是通过减少网络的参数来减小计算量，并且能够在一定程度上控制过拟合。通常在卷积层的后面会加上一个池化层。池化包括最大池化、平均池化等。其中最大池化是用不重叠的矩形框将输入层分成不同的区域，对于每个矩形框的数取最大值作为输出层，



**二、Lenet-5**

1、提出



Lenet5共有七层，不包含输入输出层

1. 各层的属性表示

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 层 | 输入 | 核/采样区域 | 训练参数 | 神经元数量 | Padding | 连接数 | 输出 |
| 输入层，输出图像的尺寸归一化为32\*32 | | | | | | | |
| C1 | (32\*32)\*6 | 5\*5\*6 | (5\*5+1)\*6 | 28\*28\*6 | Valid | (5\*5+1)6\*28\*28 | (28\*28)\*6 |
| S2 | (28\*28)\*6 | 2\*2\*6 |  | 14\*14\*6 |  | (2\*2+1)6\*14\*14 | (14\*14)\*6 |
| C3 | (14\*14\*3)\*6 | (5\*5\*3)\*6 | (5\*5\*3+1)\*6 | 10\*10\*6 | valid | (5\*5\*3+1)\*  10\*10\*6 | (10\*10)\*6 |
| (14\*14\*4)\*6 | (5\*5\*4)\*6 | (5\*5\*4+1)\*6 | 10\*10\*6 | (5\*5\*4+1)\*  10\*10\*6 | (10\*10)\*6 |
| (14\*14\*4)\*3 | (5\*5\*4)\*3 | (5\*5\*4+1)\*3 | 10\*10\*3 | (5\*5\*4+1)\*  10\*10\*3 | (10\*10)\*3 |
| (14\*14\*6)\*1 | (5\*5\*6)\*1 | (5\*5\*6+1)\*1 | 10\*10\*1 | (5\*5\*6+1)\*  10\*10\*1 | (10\*10)\*1 |
| S4 | (10\*10)\*16 | 2\*2\*16 |  | (5\*5)\*16 |  | (2\*2+1)\*  5\*5\*16 | (5\*5)\*16 |
| C5 | (5\*5)\*16 | (5\*5)\*120 | (5\*5\*16+1)\*  120 | (1\*1)\*120 |  | (5\*5\*16+1)\*  1\*1\*120 | (1\*1)\*120 |
| F6 | 120 | (1\*1\*120)\*84 | (1\*1\*120+1)  \*84 | (1\*1)\*84 |  | (1\*1\*120+1)  1\*1\*84 | 84 |
|  | 为什么有84个节点：  解答：因为在计算机中字符的编码是ASCII编码，这些图是用7\*12大小的位图表示的，也就是高宽比为7:12，如下图，选择这个大小可以用于对每一个像素点的值进行估计。 | | | | | | |
| Output | 84 |  |  |  |  | 84\*10 | 10 |

神经元的个数=输出图片大小\*输出图片的数量

每个神经元的连接数=每次卷积涉及的参数

连接数=每个神经元连接数\*神经元的个数

1. **实现**

**本文参考了博客：**<https://cuijiahua.com/blog/2018/01/dl_3.html>