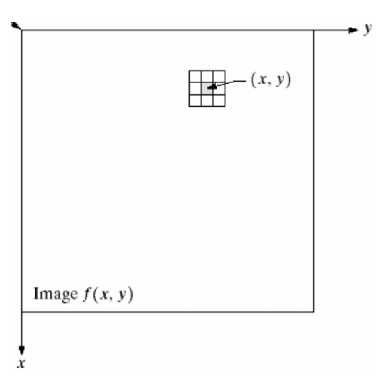
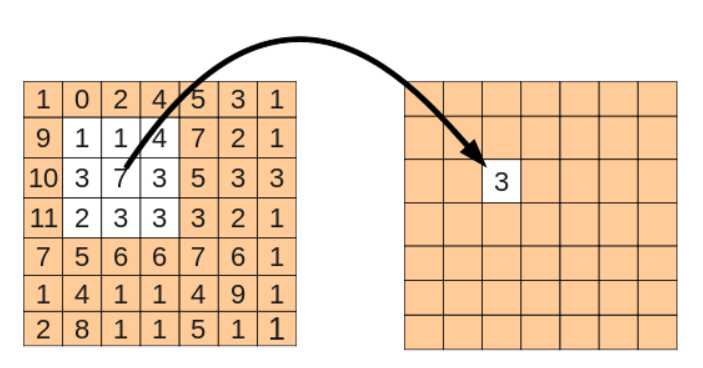
从最基础的开始

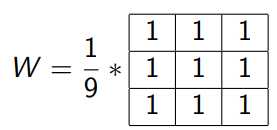
对二维数字信号（图像）的操作，可以写成矩阵形式。



比如对图像做平滑，一个典型的8领域平滑，其结果中的每个值都来源于原对应位置和其周边8个元素与一个3X3矩阵的乘积：

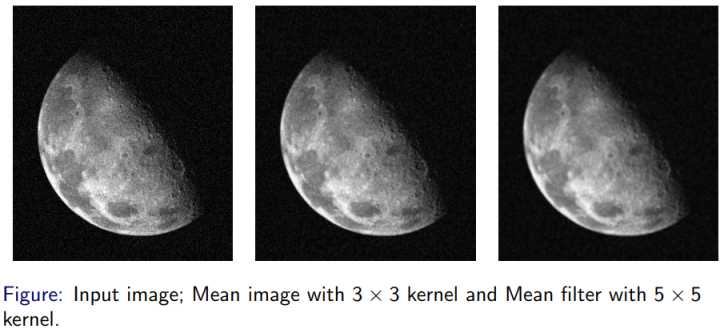


也就相当于对原矩阵，按照顺序将各区域元素与W矩阵相乘，W 矩阵为



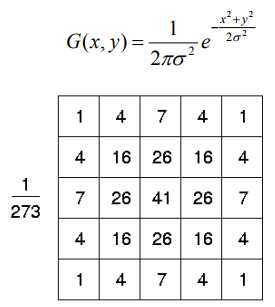
这也被称作核(Kernel, 3X3)

其处理效果如下：



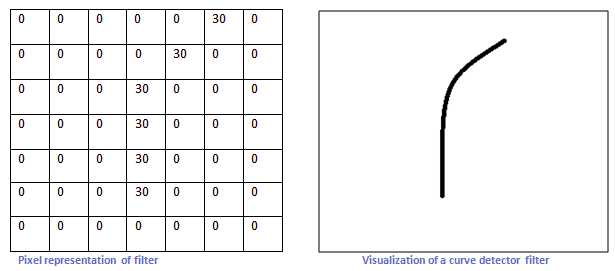
也就是，这个核对图像进行操作，相当于对图像进行了低通滤波。因此这个核也被称为滤波器，整个操作过程按照概念称为卷积。

扩展来讲，对二维图像的滤波操作可以写成卷积，比如常见的高斯滤波、拉普拉斯滤波（算子）等。

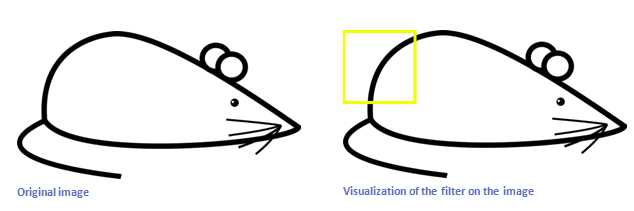


滤波器跟卷积神经网络有什么关系呢。不如我们预想一个识别问题：我们要识别图像中的某种特定曲线，也就是说，这个滤波器要对这种曲线有很高的输出，对其他形状则输出很低，这也就像是神经元的**激活**。

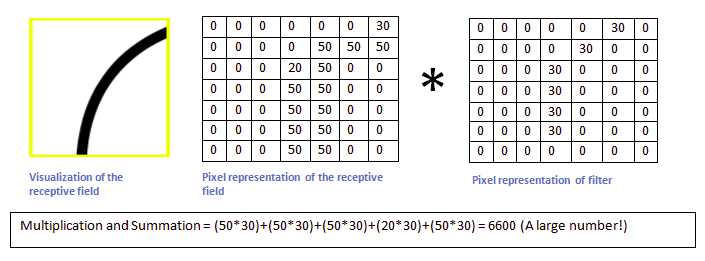
我们设计的滤波器和想要识别的曲线如下：



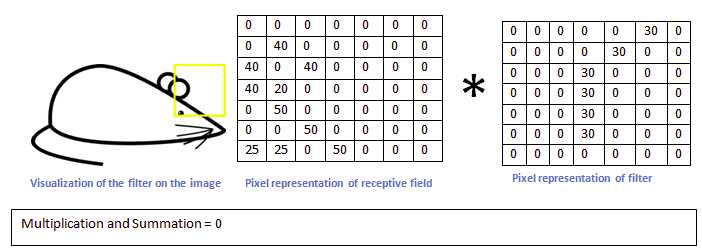
假设上面的核（滤波器）按照卷积顺序沿着下图移动：



那么当它移动到上面的位置时，按照矩阵操作，将这个区域的图像像素值与滤波器相乘，我们得到一个很大的值（6600）：

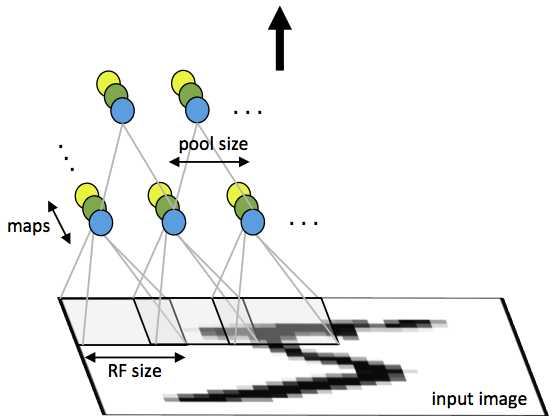


而当这个滤波器移动到其他区域时，我们得到一个相对很小的值：

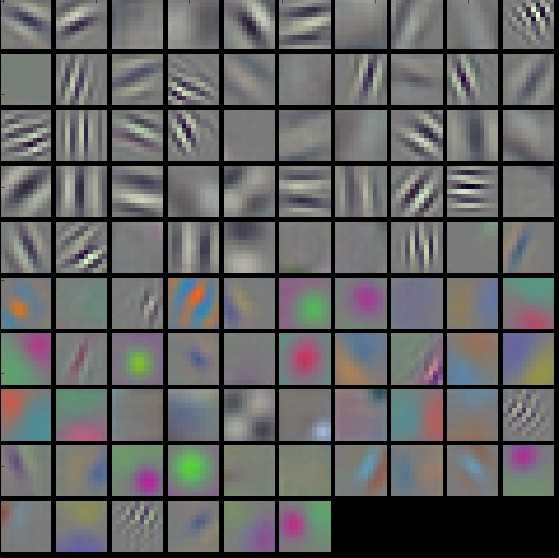


如此，我们对整个原图进行一次卷积，得到的结果中，在那个特定曲线和周边区域，值就很高，在其他区域，值相对低。这就是一张**激活图**。对应的高值区域就是我们所要检测曲线的位置。

在训练卷积审计网络（CNN）的某一个卷积层时，我们实际上是在训练一系列的滤波器(filter)。比如，对于一个32x32x3（宽32像素x高32像素xRGB三通道）的图像，如果我们在CNN的第一个卷积层定义训练12个滤波器，那就这一层的输出便是32X32X12.按照不同的任务，我们可以对这个输出做进一步的处理，这包括激活函数，池化，全连接等。



简单来说，训练CNN在相当意义上是在训练每一个卷积层的滤波器。让这些滤波器组对特定的模式有高的激活，以达到CNN网络的分类/检测等目的。



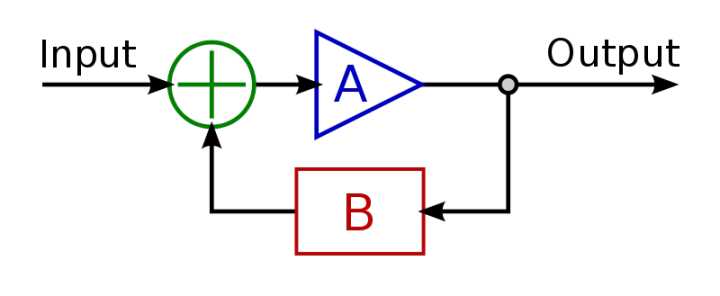
<Fig. 一个实际CNN（AlexNet）第一个卷积层的滤波器>

卷积神经网络的第一个卷积层的滤波器用来检测低阶特征，比如边、角、曲线等。随着卷积层的增加，对应滤波器检测的特征就更加复杂（理性情况下，也是我们想要的情况）。比如第二个卷积层的输入实际上是第一层的输出（滤波器激活图），这一层的滤波器便是用来检测低价特征的组合等情况（半圆、四边形等），如此累积，以检测越来越复杂的特征。实际上，我们的人类大脑的视觉信息处理也遵循这样的低阶特征到高阶特征的模式（[Owl of Minerva：为什么无彩色系（黑白灰色）在色彩搭配中可以和谐地与任何彩色搭配？](https://www.zhihu.com/question/27380522/answer/36794240)）。最后一层的滤波器按照训练CNN目的的不同，可能是在检测到人脸、手写字体等时候激活[1]。

所以，在相当程度上，构建卷积神经网络的任务就在于构建这些滤波器。也就是，将这些滤波器变成这样(改变滤波器矩阵的值，也就是Weight)的——能识别特定的特征。这个过程叫做**训练**。

在训练开始之时，卷积层的滤波器是完全随机的，它们不会对任何特征激活（不能检测任何特征）。这就像刚出生的孩子，TA不知道什么是人脸、什么是狗，什么是上下左右。TA需要学习才知道这些概念，也就是通过接触人脸、狗、上下左右，并被告知这些东西分别是人脸、狗、上下左右。然后TA才能在头脑中记住这些概念，并在之后的某一次见到之后能准确的给出结果。

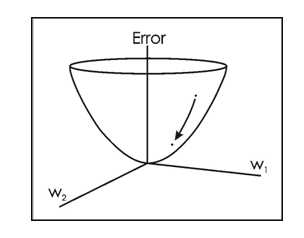
把一个空白的滤波其，修改其权重(weights)以使它能检测特定的模式，整个过程就如工程里面的反馈。



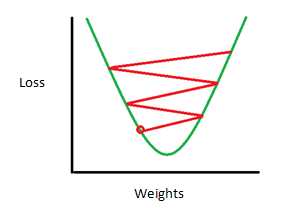
想想一下，如果有一只无意识的猴子，完全随机的修改一个5X5滤波器矩阵的25个值，那完全可能经过一定的轮次之后，这个滤波器能够检测棱角等特征。这是一种无反馈的训练情况。对神经网络的训练当然不能如此，我们不可能靠运气去做这件事情。

举个例子，我们要训练一个用于分类的神经网络，让它能判定输入图像中的物体最可能是十个类别的哪一类。那么，训练过程就是这样的：

第一次训练，输入一张图像，这个图像通过各层卷积处理输出量一组向量[1,1,1,1,1,1,1,1,1,1], 也就是，对于完全由随机滤波器构建的网络，其输出认为这张图等概率的是十个类别中的某一种。但是对于训练，我们有一个Gound Thuth, 也就是这张图中物体所属的类别：[0，0，1，0，0，0，0，0，0，0]，也就是属于第三类。这时候我们可以定义一个损失函数，比如常见的MSE（mean squared error）.我们假定L是这个损失函数的输出。这时候我们的目的就是，让L的值反馈(这种神经网络概念下称为 back propagation， 反向传输)给整个卷积神经网络，以修改各个滤波器的权重，使得损失值L最小。



这是一个典型的最优化问题。当然地，在工程上我们几乎不可能一次就把滤波器的权重W修改到使L最小的情况，而是需要多次训练和多次修改。



如果情况理想的话，权重修改的方向是使得L的变化收敛的。这也就是说很可能达到了我们训练这个神经网络的目的——让各个卷积层的滤波器能够组合起来最优化的检测特定的模式。