CNNs

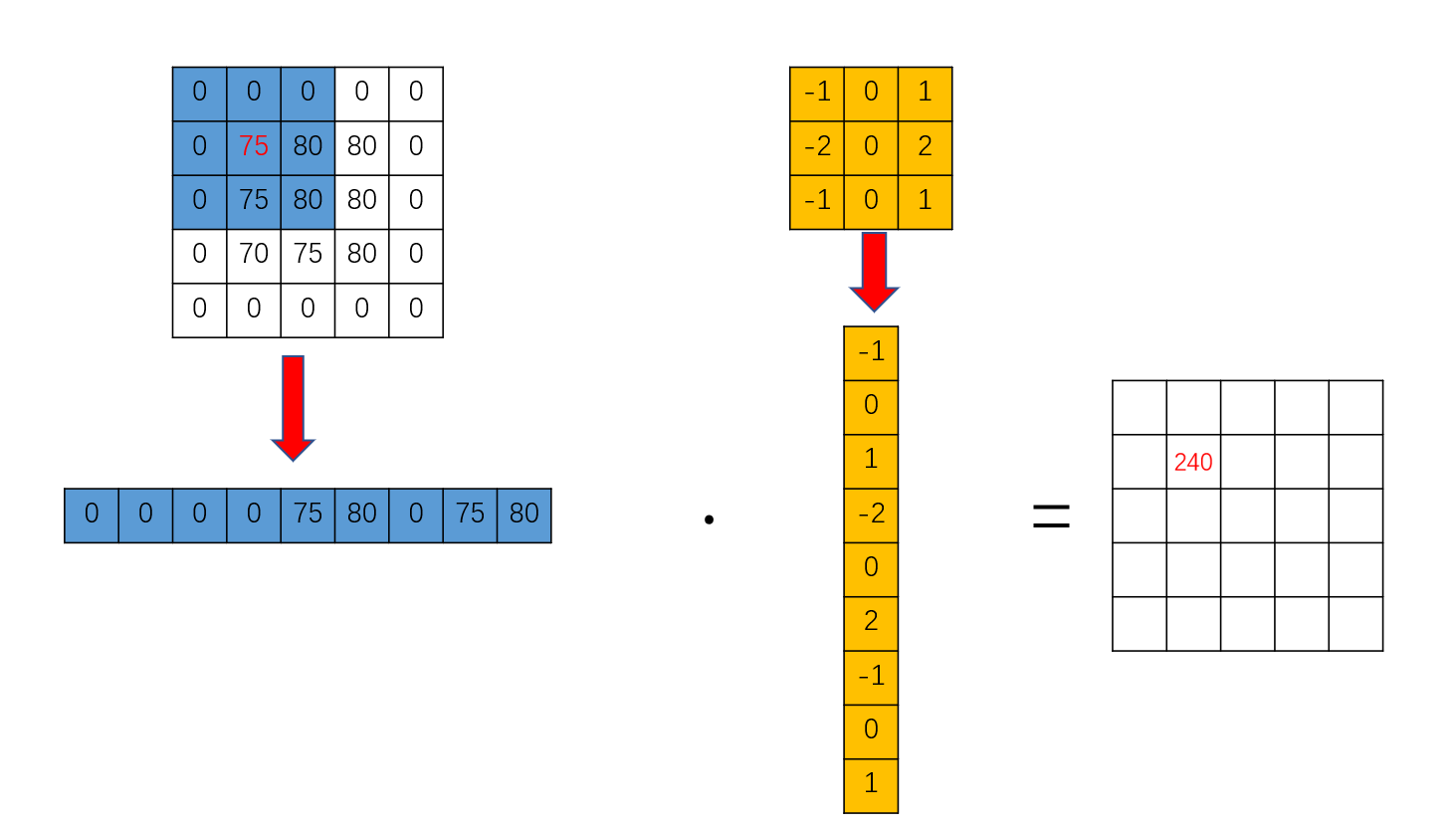
# 卷积的理解

## 1.1图像处理中的卷积

卷积在信号与系统方面的物理意义比较好理解，但是在数字图像处理中的解释就难免显得有点牵强，搜了一些文献，我发现大多数所谓的卷积核都只是描述成了一个权重矩阵，对两个矩阵进行数乘相加，硬要是和卷积扯上关系，私以为那也就是用到了卷积中的乘法(权重)和累加(积分)的性质了吧，至少我没看出来它哪里体现了“卷”这个性质，当然也有人解释说我们实际用到的卷积核是经过180°翻转的，这样理解倒也不是不可以，但觉得这样解释有点冗余。

## 1.2卷积的计算

一个卷积核和图像中对应区域(感受域，也被称为receptive field)的卷积的求解过程，可以理解为对这两个矩阵进行reshape，比如3\*3的区域与3\*3的卷积核相乘的结果，就是1\*9和9\*1向量做内积的结果而已，当然扩展到多维的也一样的，但是要注意运算顺序就可以了，比如8\*8\*3的区域和8\*8\*3的卷积核，扩展为1\*192和192\*1的向量内积就可以了。以下是第一个位置的计算过程(填充采用0填充)。



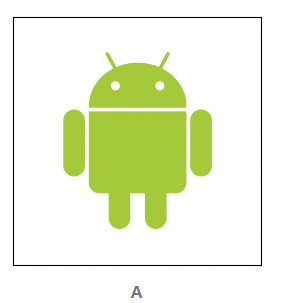
## 1.3卷积的应用

那么卷积核到底能表征什么呢？如果将其理解为权重矩阵，那么就比较好理解了，权重本身就可以为突出某些“特征”而设计。实际上卷积核作用于图像也就是为了得到某种特征，比如边缘特征，曲线特征等，更高层的卷积核表示的内容也往往更具体。

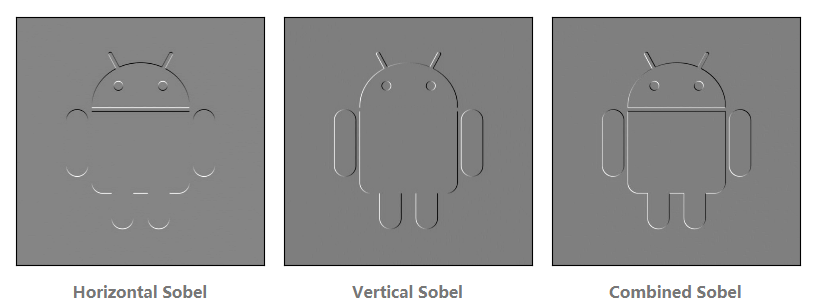
举例就是实际图像处理中边缘检测可能会用到如下两个卷积核(*Sobel*算子)，他们可以得到一些图像的水平和垂直特征。



如果我们对如下的图像应用上述两个卷积核:



就可以得到如下内容



可以看到，这两个卷积核可以对边缘进行一定的检测，但在我们的图片分类中，我们要分类的图像的特征我们实际可能不知道，即使知道，往往也不具备泛用性，因为我们如果有1000个类，那么我们不会去人工去寻找每个类的特征，然后得到对应的卷积核，所以确定卷积核的参数这件事，我们交给训练去做，也就是说卷积核中的参数(矩阵中各个位置的值)是我们要训练的内容，不是事先指定的。但是卷积核的大小以及步长则是我们事先指定的超参数。

# 二、CNN的结构

## 2.1整体结构

## 2.2详细结构

### 2.2.1输入层

输入层只有一个节点，节点内容就是一个带深度的矩阵，表示图像信息，比如32\*32\*3表示分辨率是32\*32，深度为3可能表示RGB的三个通道比如(255,255,255)

### 2.2.2卷积层

卷积层是在上一层矩阵的基础上，利用卷积核扫过得到的新矩阵，这里牵涉到了步长和填充问题，比如我们有5个4\*4\*3的卷积核，那么我们下一层可能就得到的一个深度为5的32\*32\*5的矩阵。

### 2.2.3非线性化层

非线性化层不是独立的一层，其在卷积层中出现，主要是对输出做一定的处理，比如把他们放缩到[-1,1]或者[0,1]之间。比如sigmod函数，tanh函数，以及relu函数等。

### 2.2.4池化层

池化层的操作其实有点类似卷积层，只不过卷积核的作用是加权求和，而池化层则是选出*max*值，而且步长是整个卷积核的大小。

### 2.2.5全连接层

### 2.2.6输出层

输出层的个数就是我们需要识别的类别数，比如我们要从1000张图片分出10类，那输出层就应该是一个1\*9的向量，每一个位置表示对应类别的得分，比如得分高者就是我们最后认为它属于的类别。

# 参考文献

1. https://mlnotebook.github.io/post/CNN1/ 介绍CNN的，里面卷积讲的不错
2. http://scs.ryerson.ca/~aharley/vis/conv/flat.html 一个0~9的CNN案例