# 深度学习

深度学习的核心是**特征学习**，旨在通过**分层网络获取分层次的特征信息**，从而解决以往需要人工设计特征的重要难题。

**深度学习是一个框架**，包含多个重要算法:

Convolutional Neural Networks(CNN) 卷积神经网络

AutoEncoder自动编码器

Sparse Coding稀疏编码

Restricted Boltzmann Machine(RBM)限制波尔兹曼机

Deep Belief Networks(DBN)深信度网络

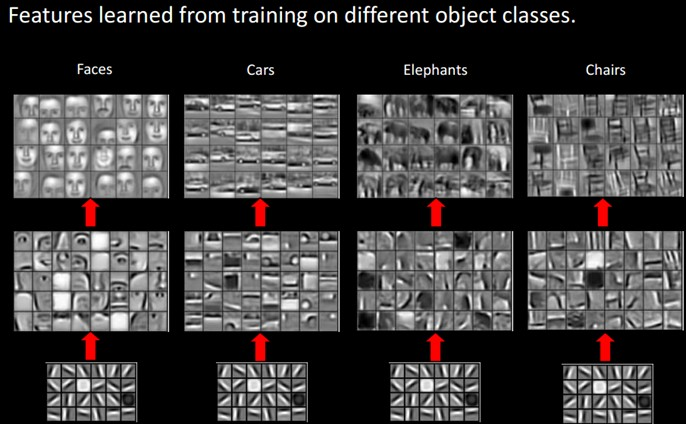
Recurrent neural Network(RNN)多层反馈循环神经网络神经网络

对于不同问题(图像，语音，文本)，需要选用**不同网络模型**才能达到更好效果。

最近几年**增强学习**(Reinforcement Learning)与深度学习的结合也创造了许多了不起的成果，AlphaGo就是其中之一。

David Hubel（出生于加拿大的美国神经生物学家） 和TorstenWiesel，以及 Roger Sperry，“发现了视觉系统的信息处理”，可视皮层是分级的。

对于不同的物体，人类视觉也是通过这样逐层分级，来进行认知的：

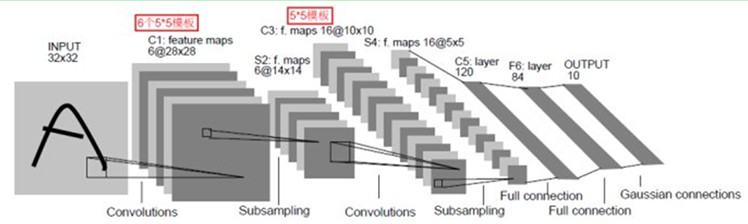


在**最底层特征**基本上是类似的，就是各种边缘，越往上，越能提取出此类物体的一些特征（轮子、眼睛、躯干等），到最上层，不同的高级特征最终组合成相应的图像，从而能够让人类准确的区分不同的物体。

模仿人类大脑的这个特点，**构造多层的神经网络**，较低层的识别初级的图像特征，若干底层特征组成更上一层特征，最终通过多个层级的组合，最终在顶层做出分类呢？答案是肯定的，这也是许多深度学习算法（包括CNN）的灵感来源。

卷积神经网络(CNN)是一种多层神经网络，擅长处理图像特别是大图像的相关机器学习问题。

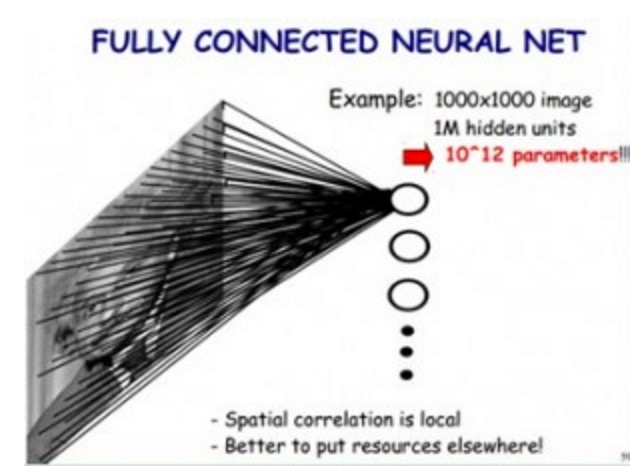
卷积网络通过一系列方法，成功将**数据量庞大的图像识别问题不断降维**，最终使其能够被训练。CNN最早由Yann LeCun提出并应用在手写字体识别上（MINST）。LeCun提出的网络称为LeNet，其网络结构如下：

 卷积 下采样 卷积 下采样 全连接 全连接 高斯连接

这是一个最典型的卷积网络，**由卷积层、池化层、全连接层**组成。其中卷积层与池化层配合，组成多个卷积组，逐层提取特征，最终通过若干个全连接层完成分类。

卷积层完成的操作，可以认为是受局部感受野概念的启发，而池化层，主要是为了降低数据维度。

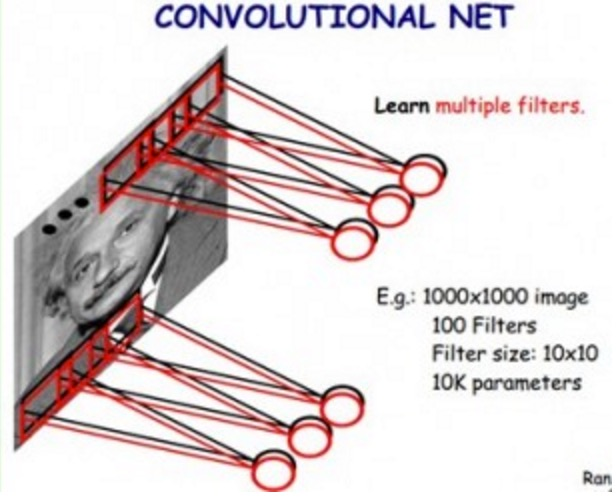
CNN通过卷积来模拟特征区分，并且通过卷积的权值共享及池化，来降低网络参数的数量级，最后通过传统神经网络完成分类等任务。

 使用**传统神经网络**方式，对一张图片进行分类，那么，我们把图片的每个像素都连接到隐藏层节点上，那么对于一张1000x1000像素的图片，如果我们有1M隐藏层单元，那么一共有10^12个参数，这显然是不能接受的。

在CNN里，可以大大减少参数个数，我们基于以下两个假设：

1）最**底层特征都是局部性**的，也就是说，我们用10x10这样大小的过滤器就能表示边缘等底层特征

2）图像上不同小片段，以及不同图像上的**小片段的特征是类似**的，也就是说，我们能用同样的一组**分类器**来描述各种各样不同的图像



我们用100个10x10的小过滤器，就能够描述整幅图片上的底层特征。

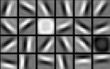
卷积（convolution)

有一个5x5的图像，我们用一个3x3的卷积核, 对图像进行卷积操作（可以理解为有一个滑动窗口，把卷积核与对应的图像像素做乘积然后求和），得到了3x3的卷积结果。

个过程我们可以理解为我们使用一个过滤器（卷积核）来过滤图像的各个小区域，从而得到这些小区域的特征值。

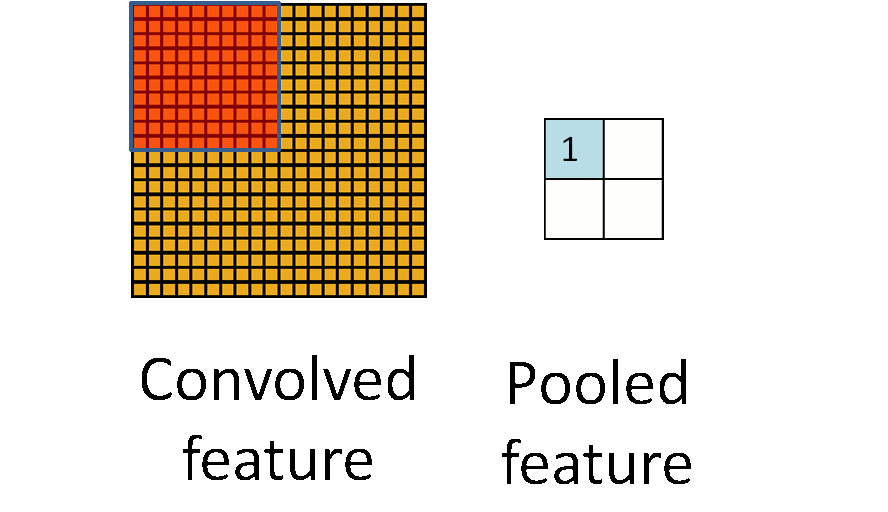
在实际训练过程中，卷积核的值是在学习过程中学到的。

在具体应用中，往往**有多个卷积核**，可以认为，**每个卷积核代表了一种图像模式**，**如果某个图像块与此卷积核卷积出的值大，则认为此图像块十分接近于此卷积核**。如果我们设计了6个卷积核，可以理解：我们认为这个图像上有6种底层纹理模式，也就是我们用6中基础模式就能描绘出一副图像。以下就是24种不同的卷积核的示例：



池化（poling)

简单的说就是下采样。



原始图片是20x20的，我们对其进行下采样，采样窗口为10x10，最终将其下采样成为一个2x2大小的特征图。

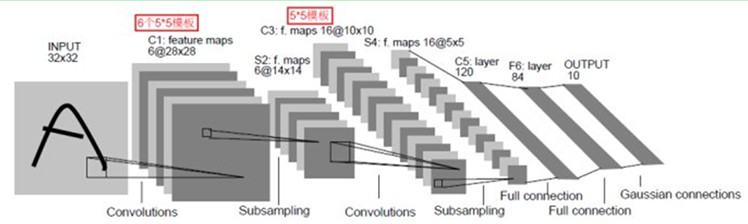
之所以这么做的原因，是因为即使做完了卷积，图像仍然很大（因为卷积核比较小），所以为了降低数据维度，就进行下采样。

**卷积 → 池化 （下采样）**

因为即使减少了许多数据，**特征的统计属性**仍能够描述图像，而且由于降低了数据维度，有效地避免了过拟合。

池化根据下采样的方法，分为最大值下采样（Max-Pooling）与平均值下采样（Mean-Pooling）。

**LeNet网络结构**

****

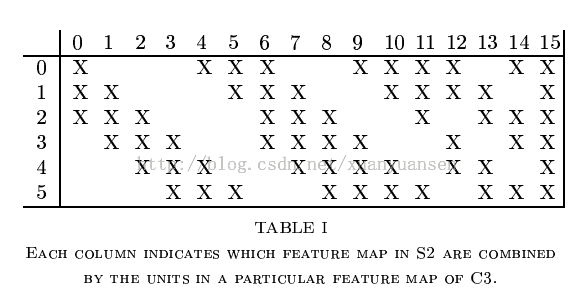
卷积层 （6个5x5核）下采样 卷积 下采样

原始图像进来以后，先进入一个**卷积层**C1，由**6个5x5的卷积核组成**，卷积出6个 28x28的图像，然后**下采样**到6个 14x14（S2）。

再进一个卷积层C3，由**16个5x5**的卷积核组成，之后再下采样到5x5（S4）。

1个 32x32 +（6个 5x5卷积核） →6个28x28 + 下采样 →6个14x14

6个14x14 + （16个 5x5卷积核）→96个10x10 + 下采样 → 96个 5x5

其中行代表S2层的某个节点，列代表C3层的某个节点。

我们可以看出，C3-0跟S2-0,1,2连接，C3-1跟S2-1,2,3连接，后面依次类推，仔细观察可以发现，其实就是排列组合：

0 0 0 1 1 1

0 0 1 1 1 0

0 1 1 1 0 0

...

1 1 1 1 1 1

即用不同特征的底层组合，可以得到进一步的高级特征，

通过全连接层C5、F6得到10个输出，对应10个数字的概率。

### 训练过程

卷积神经网络的训练过程与传统神经网络类似，也是参照了反向传播算法。

第一阶段，向前传播阶段：

a）从样本集中取一个样本(X,Yp)，将X输入网络；

b）计算相应的实际输出Op。

      在此阶段，信息从输入层经过逐级的变换，传送到输出层。这个过程也是网络在完成训练后正常运行时执行的过程。在此过程中，网络执行的是计算（实际上就是输入与每层的权值矩阵相点乘，得到最后的输出结果）：

          Op=Fn（…（F2（F1（XpW（1））W（2））…）W（n））

第二阶段，向后传播阶段

a）算实际输出Op与相应的理想输出Yp的差；

b）按极小化误差的方法反向传播调整权矩阵。

以上内容摘自其他博客，由于我也没有仔细了解这一块，建议直接参考[原博客](http://blog.csdn.net/zouxy09/article/details/8781543)。