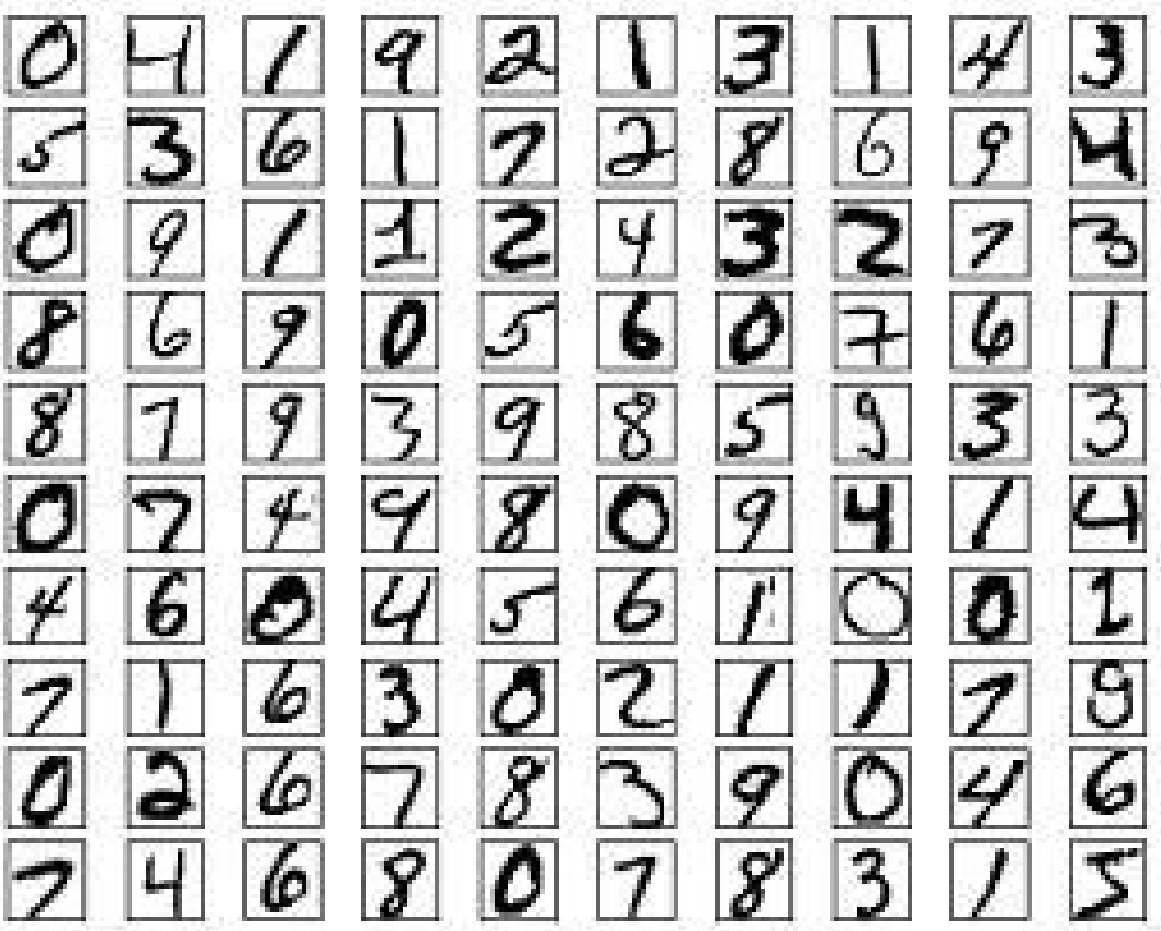
Lenet-5

一、Lenet-5简介

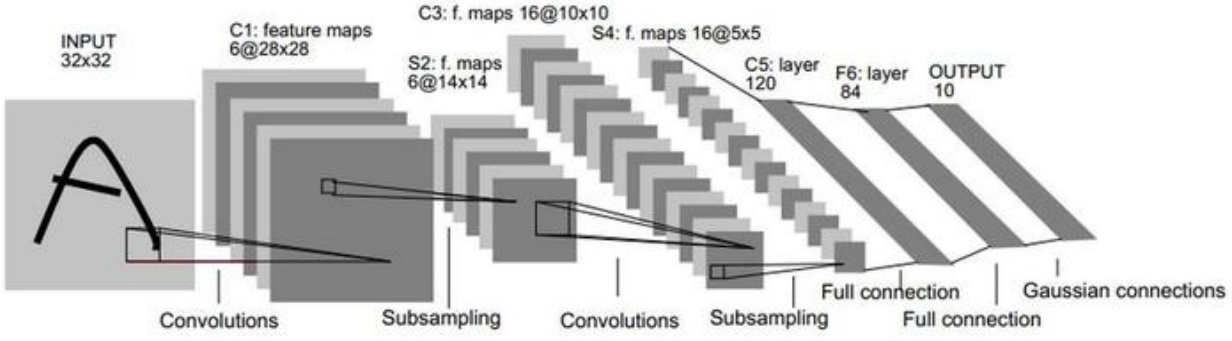
Lenet-5是1998年，Yann LeCun在《Gradient-Based Learning Applied to Document Recognition》一文中提出的一种卷积神经网络，是一种用于手写体字符识别非常高效的卷积神经网络。

MNIST数据集：



二、Lenet-5的结构

Lenet-5除了输入层之外一共有7层，如下图所示，C1是一个卷积层，S2池化层，C3卷积层，S4池化层，C5卷积层，F6全连接层，输出层（全连接）。（C-convolutions，S-subsampling，F-full connection）。



卷积（convolution）：

$$s(t) = \int_{-\infty}^{\infty} f(a) * g(t-a)da$$

$$s(t) = f(t) \times g(t) = \sum_{a=-\infty}^{\infty} f(a)g(t-a)$$

在2015中国计算机大会特邀报告上，在中国人工智能学会理事长李德毅院士的主题报告中，李院士便提到了卷积的理解问题，非常有意思。他讲到，什么叫卷积呢？

举例来说，在一根铁丝某处不停地弯曲，假设发热函数是f(t)，散热函数是 g(t)，此时此刻的温度就是 f(t) 跟g(t)的卷积。

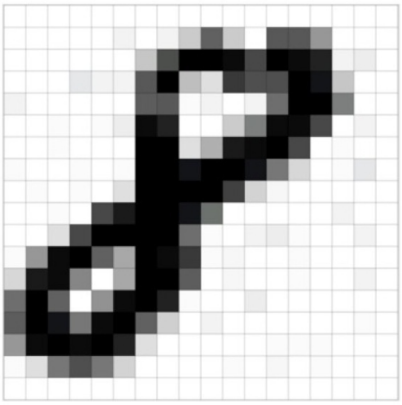
在一个特定环境下，发声体的声源函数是f(t)，该环境下对声源的反射效应函数是 g(t)，那么这个环境下的接受到声音就是 f(t) 和 g(t) 的卷积。

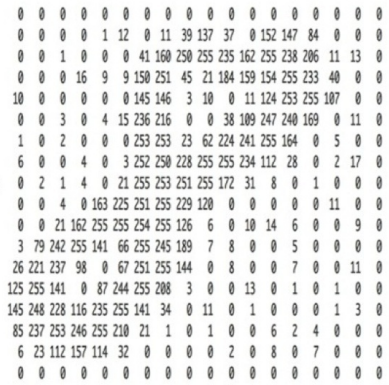
类似地，记忆其实也是一种卷积的结果。假设认知函数是 f(t)，它代表对已有事物的理解和消化，随时间流逝而产生的遗忘函数是g(t)，那么人脑中记忆函数 h(t) 就是函数是 f(t) 跟g(t) 的卷积。

$$h_{记忆}(t)$$

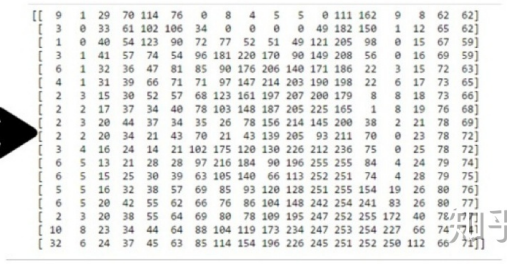
$$= f_{认知}(t) * g_{遗忘}(t)$$

$$= \int_{0}^{+\infty} f_{认知}(\tau) g_{遗忘}(t-\tau) d\tau$$





8

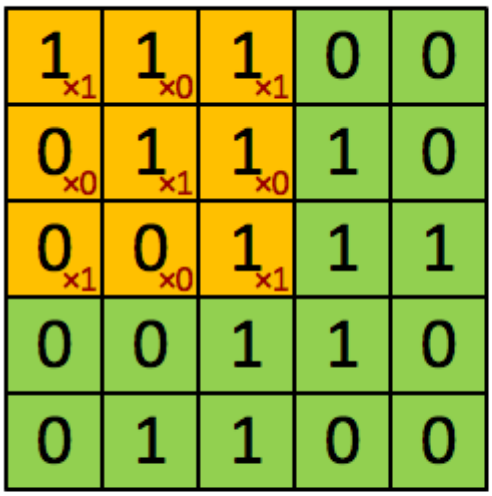


猫

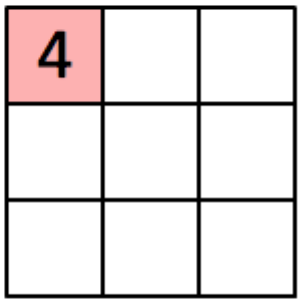
$$s(t) = \int_{-\infty}^{\infty} f(a) * g(t-a)da$$

放在图像分析里，f(x) 可以理解为原始像素点，所有的原始像素点叠加起来，就是原始图了。g(x)可以称为作用点，所有作用点合起来我们称为卷积核，卷积核上所有作用点依次作用于原始像素点后（即乘起来），线性叠加的输出结果，即是最终卷积的输出，也是我们想要的结果。

在图像处理中应用卷积操作，主要目的就是从图像中提取特征。卷积可以很方便地通过从输入的一小块数据矩阵（也就是一小块图像）中学到图像的特征，并能保留像素间的空间关系。

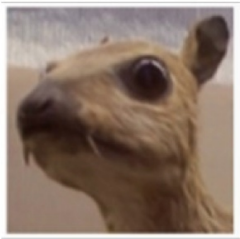


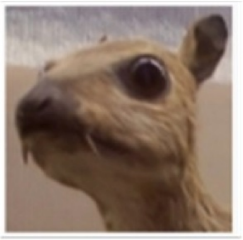
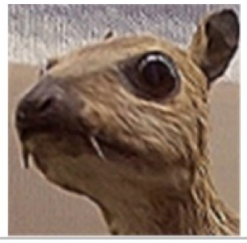
Image



Convolved Feature

原始图片



操作	卷积核（滤波器）	卷积后图像
同一化 (Identity)	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
边界检测 (Edge detection)	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
锐化 (Sharpen)	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
均值模糊化 (Box blur/Averageing)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	

（1）同一化核（Identity）。从图13-6可见，这个滤波器什么也没有做，卷积后得到的图像和原图一样。因为这个核只有中心点的值是1。邻域点的权值都是0，所以对滤波后的取值没有任何影响。

（2）边缘检测核（Edge Detection），也称为高斯-拉普拉斯算子。需要注意的是，这个核矩阵的元素总和为0（即中间元素为8，而周围8个元素之和为-8），所以滤波后的图像会很暗，而只有边缘位置是有亮度的。

（3）图像锐化核（Sharpness Filter）。图像的锐化和边缘检测比较相似。首先找到边缘，然后再把边缘加到原来的图像上面，如此一来，就强化了图像的边缘，使得图像看起来更加锐利。

（4）均值模糊（Box Blur /Averaging）。这个核矩阵的每个元素值都是1，它将当前像素和它的四邻域的像素一起取平均，然后再除以9。均值模糊比较简单，但图像处理得不够平滑。因此，还可以采用高斯模糊核（Gaussian Blur），这个核被广泛用在图像降噪上。

下采样（subsampling）：

缩小图像使得图像符合显示区域的大小，生成对应图像的缩略图。

采样层是使用 pooling的相关技术来实现的，目的就是用来降低特征的维度并保留有效信息，一定程度上避免过拟合。但是pooling的目的不仅仅是这些，他的目的是保持旋转、平移、伸缩不变形等。

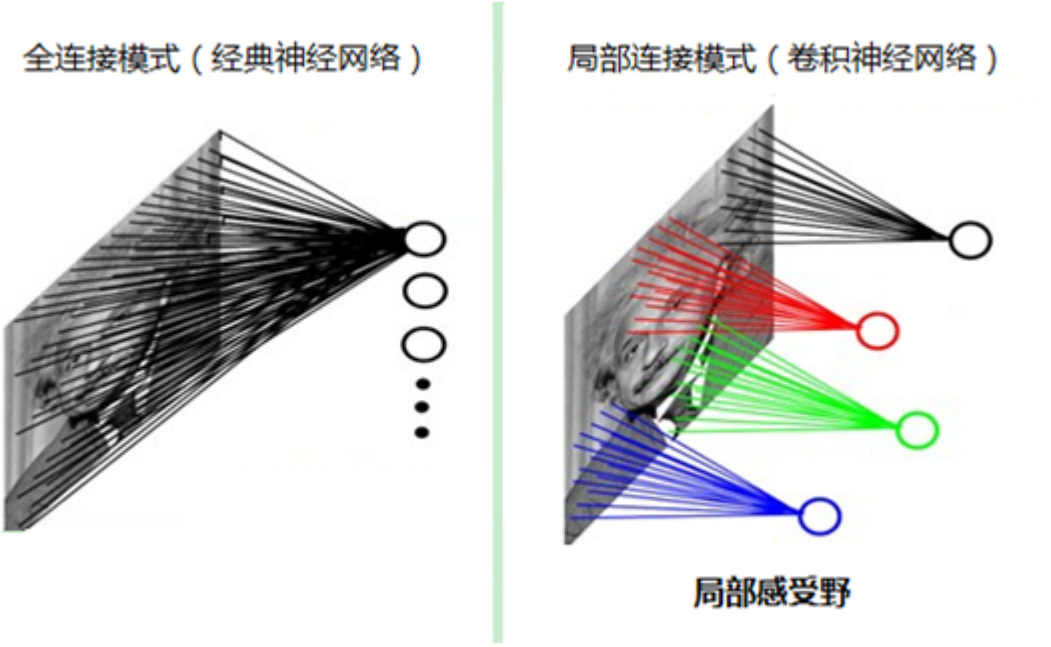
采样有最大值采样，平均值采样，求和区域采样和随机区域采样等。池化也是这样的，比如最大值池化，平均值池化，随机池化，求和区域池化等。

1. mean-pooling，即对邻域内特征点只求平均。
2. max-pooling，即对邻域内特征点取最大。

三、Lenet-5的特点（CNN的特点）

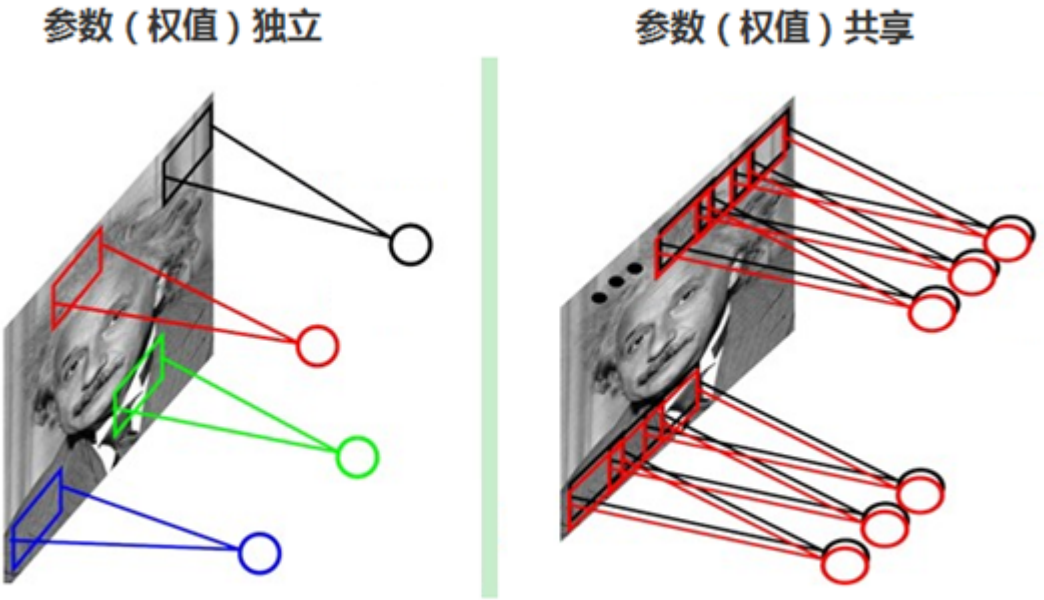
1、局部感知

人类对外界的认知一般是从局部到全局、从片面到全面，类似的，在机器识别图像时也没有必要把整张图像按像素全部都连接到神经网络中，在图像中也是局部周边的像素联系比较紧密，而距离较远的像素则相关性较弱，因此可以采用局部连接的模式（将图像分块连接，这样能大大减少模型的参数），如下图所示：



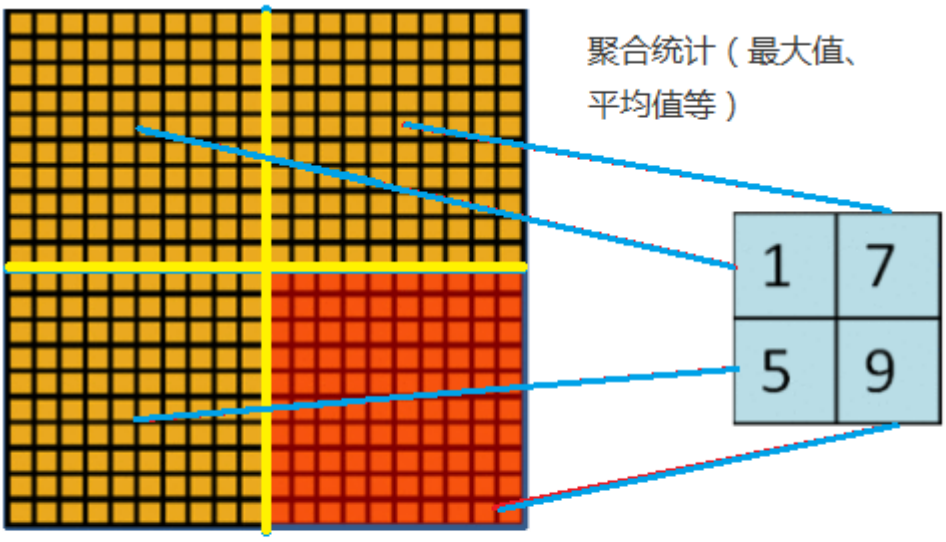
2、参数（权值）共享

每张自然图像（人物、山水、建筑等）都有其固有特性，也就是说，图像其中一部分的统计特性与其它部分是接近的。这也意味着这一部分学习的特征也能用在另一部分上，能使用同样的学习特征。因此，在局部连接中隐藏层的每一个神经元连接的局部图像的权值参数（例如5×5），将这些权值参数共享给其它剩下的神经元使用，那么此时不管隐藏层有多少个神经元，需要训练的参数就是这个局部图像的权限参数（例如5×5），也就是卷积核的大小，这样大大减少了训练参数。如下图。



3、池化

随着模型网络不断加深，卷积核越来越多，要训练的参数还是很多，而且直接拿卷积核提取的特征直接训练也容易出现过拟合的现象。回想一下，之所以对图像使用卷积提取特征是因为图像具有一种“静态性”的属性，因此，一个很自然的想法就是对不同位置区域提取出有代表性的特征（进行聚合统计，例如最大值、平均值等），这种聚合的操作就叫做池化，池化的过程通常也被称为特征映射的过程（特征降维），如下图：



四、详细描述

1、INPUT层-输入层

首先是数据 INPUT 层，输入图像的尺寸为32*32。

2、C1层-卷积层

输入图片：32*32

卷积核大小：5*5

卷积核种类：6

输出featuremap大小：28*28 （32-5+1）=28

神经元数量：28*28*6

可训练参数：（5*5+1）* 6（每个滤波器5*5=25个unit参数和一个bias参数，一共6个滤波器）

连接数：（5*5+1）*6*28*28=122304

详细说明：对输入图像进行第一次卷积运算（使用 6 个大小为 5*5 的卷积核），得到6个C1特征图（6个大小为28*28的 feature maps, 32-5+1=28）。我们再来看看需要多少个参数，卷积核的大小为5*5，总共就有6*（5*5+1）=156个参数，其中+1是表示一个核有一个bias。对于卷积层C1，C1内的每个像素都与输入图像中的5*5个像素和1个bias有连接，所以总共有156*28*28=122304个连接（connection）。有122304个连接，但是我们只需要学习156个参数，主要是通过权值共享实现的。

3、S2层-池化层（下采样层）

输入：28*28

采样区域：2*2

采样方式：4个输入相加，乘以一个可训练参数，再加上一个可训练偏置。结果通过sigmoid。

采样种类：6

输出featureMap大小：14*14（28/2）

神经元数量：14*14*6

连接数：（2*2+1）*6*14*14

S2中每个特征图的大小是C1中特征图大小的1/4。

详细说明：第一次卷积之后紧接着就是池化运算，使用 2*2核 进行池化，于是得到了S2，6个14*14的 特征图（28/2=14）。S2这个pooling层是对C1中的2*2区域内的像素求和乘以一个权值系数再加上一个偏置，然后将这个结果再做一次映射。同时有5x14x14x6=5880个连接。

4、C3层-卷积层

输入：S2中所有6个或者几个特征map组合

卷积核大小：5*5

卷积核种类：16

输出featureMap大小：10*10 (14-5+1)=10

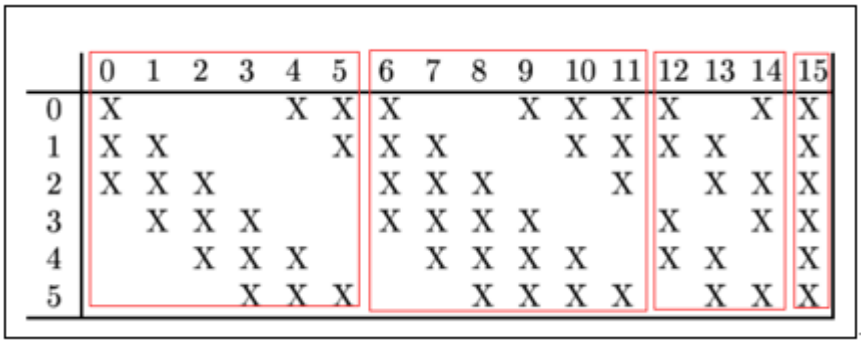
C3中的每个特征map是连接到S2中的所有6个或者几个特征map的，表示本层的特征map是上一层提取到的特征map的不同组合

存在的一个方式是：C3的前6个特征图以S2中3个相邻的特征图子集为输入。接下来6个特征图以S2中4个相邻特征图子集为输入。然后的3个以不相邻的4个特征图子集为输入。最后一个将S2中所有特征图为输入。

则：可训练参数：6*(3*5*5+1)+6*(4*5*5+1)+3*(4*5*5+1)+1*(6*5*5+1)=1516

连接数：10*10*1516=151600

详细说明：第一次池化之后是第二次卷积，第二次卷积的输出是C3，16个10x10的特征图，卷积核大小是 5*5. 我们知道S2 有6个 14*14 的特征图，怎么从6 个特征图得到 16个特征图了？ 这里是通过对S2 的特征图特殊组合计算得到的16个特征图。具体如下：



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	X				X	X	X			X	X	X	X		X	X
1	X	X				X	X	X			X	X	X	X		X
2	X	X	X				X	X	X			X		X	X	X
3		X	X	X			X	X	X	X			X		X	X
4			X	X	X			X	X	X	X		X	X		X
5				X	X	X			X	X	X	X		X	X	X

C3的前6个feature map（对应上图第一个红框的6列）与S2层相连的3个feature map相连接（上图第一个红框），后面6个feature map与S2层相连的4个feature map相连接（上图第二个红框），后面3个feature map与S2层部分不相连的4个feature map相连接，最后一个与S2层的所有feature map相连。卷积核大小依然为5*5，所以总共有6*(3*5*5+1)+6*(4*5*5+1)+3*(4*5*5+1)+1*(6*5*5+1)=1516个参数。而图像大小为10*10，所以共有151600个连接。

为什么采用上述这样的组合了？论文中说有两个原因：1）减少参数，2）这种不对称的组合连接的方式有利于提取多种组合特征。

5、S4层-池化层（下采样层）

输入：10*10

采样区域：2*2

采样方式：4个输入相加，乘以一个可训练参数，再加上一个可训练偏置。结果通过sigmoid

采样种类：16

输出featureMap大小：5*5（10/2）

神经元数量：5*5*16=400

连接数：16*（2*2+1）*5*5=2000

S4中每个特征图的大小是C3中特征图大小的1/4

详细说明：S4是pooling层，窗口大小仍然是2*2，共计16个feature map，C3层的16个10x10的图分别进行以2x2为单位的池化得到16个5x5的特征图。有5x5x5x16=2000个连接。连接的方式与S2层类似。

6、C5层-卷积层

输入：S4层的全部16个单元特征map（与s4全相连）

卷积核大小：5*5

卷积核种类：120

输出featureMap大小：1*1（5-5+1）

可训练参数/连接：120*（16*5*5+1）=48120

详细说明：C5层是一个卷积层。由于S4层的16个图的大小为5x5，与卷积核的大小相同，所以卷积后形成的图的大小为1x1。这里形成120个卷积结果。每个都与上一层的16个图相连。所以共有(5x5x16+1)x120 = 48120个参数，同样有48120个连接。

7、F6层-全连接层

输入：c5 120维向量

计算方式：计算输入向量和权重向量之间的点积，再加上一个偏置，结果通过sigmoid函数输出。

可训练参数:84*(120+1)=10164

详细说明：6层是全连接层。F6层有84个节点，对应于一个7x12的比特图，-1表示白色，1表示黑色，因为在计算机中字符的编码是ASCII,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。

ASCII编码(American Standard Code for Information Interchange)：美国信息交换标准代码主要用于显示现代英语和其他西欧语言）。

8、Output层-全连接层

Output层也是全连接层，共有10个节点，分别代表数字0到9，且如果节点i的值为0，则网络识别的结果是数字i。采用的是径向基函数（RBF）的网络连接方式。假设x是上一层的输入，y是RBF的输出，则RBF输出的计算方式是：

$$y_i = \sum_j (x_j - w_{ij})^2$$

上式w_ij 的值由i的比特图编码确定，i从0到9，j取值从0到7*12-1。RBF输出的值越接近于0，则越接近于i，即越接近于i的ASCII编码图，表示当前网络输入的识别结果是字符i。该层有84x10=840个参数和连接。