# Basic CNN Components

## Convolutional layer

卷积层的目的是从输入中学习特征的表示，其有若干个用来计算各不相同的特征映射（Feature Map，可以理解为对原图像中某种特征的单独表示）的卷积核组成。每一个特征映射的神经元都连接着前一层一片相邻神经元的区域，这片区域被称作当前神经元对前一层的感受野（Receptive field）。新的特征映射可以通过输入与一个学习完毕的卷积核做卷积运算，再将结果通过一个element-wise的非线性激励函数计算而来，在此过程中，卷积核将被输入的每一个空间位置所共享。一般我们使用若干个卷积核来构建完整的特征映射。

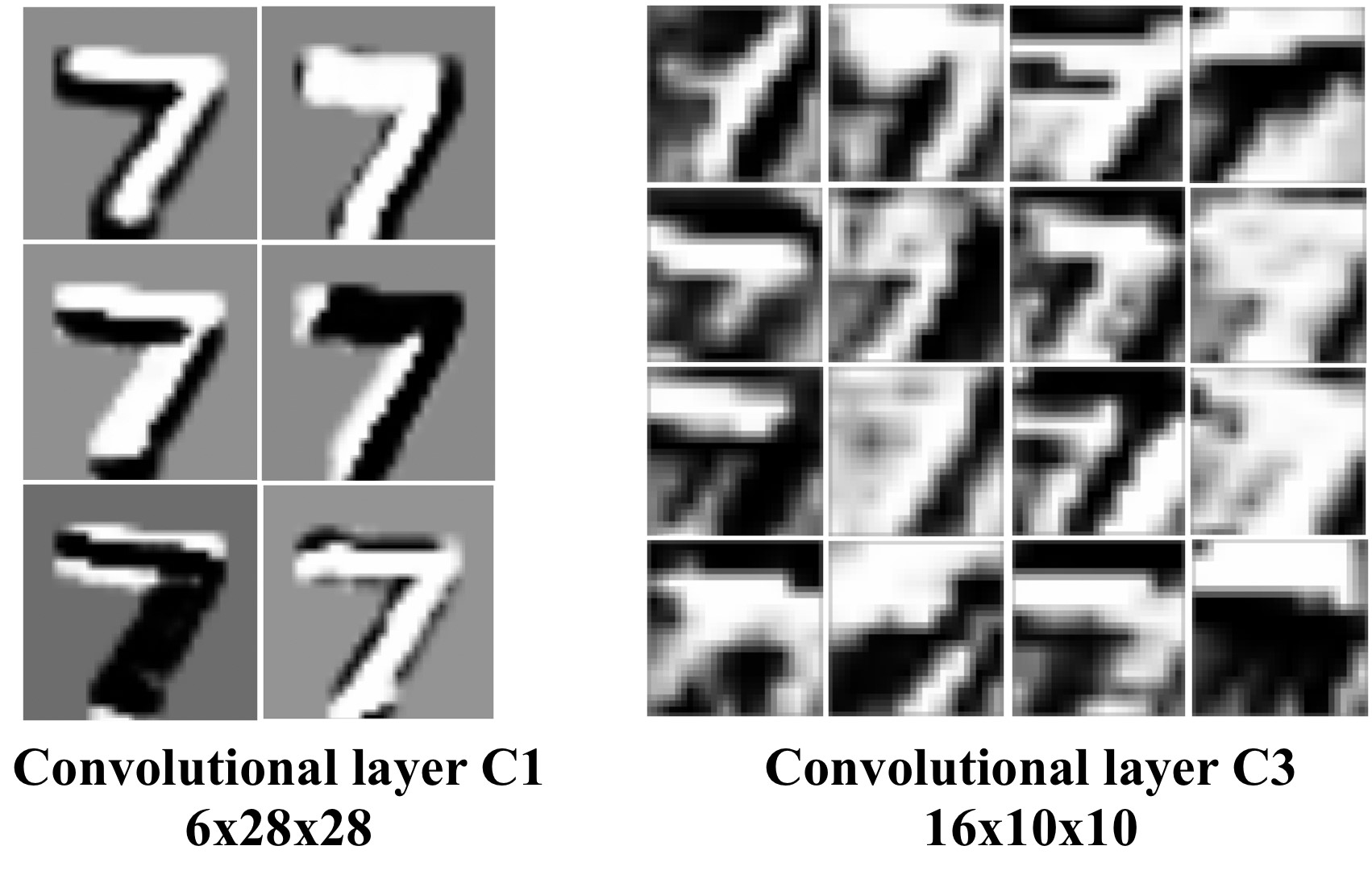
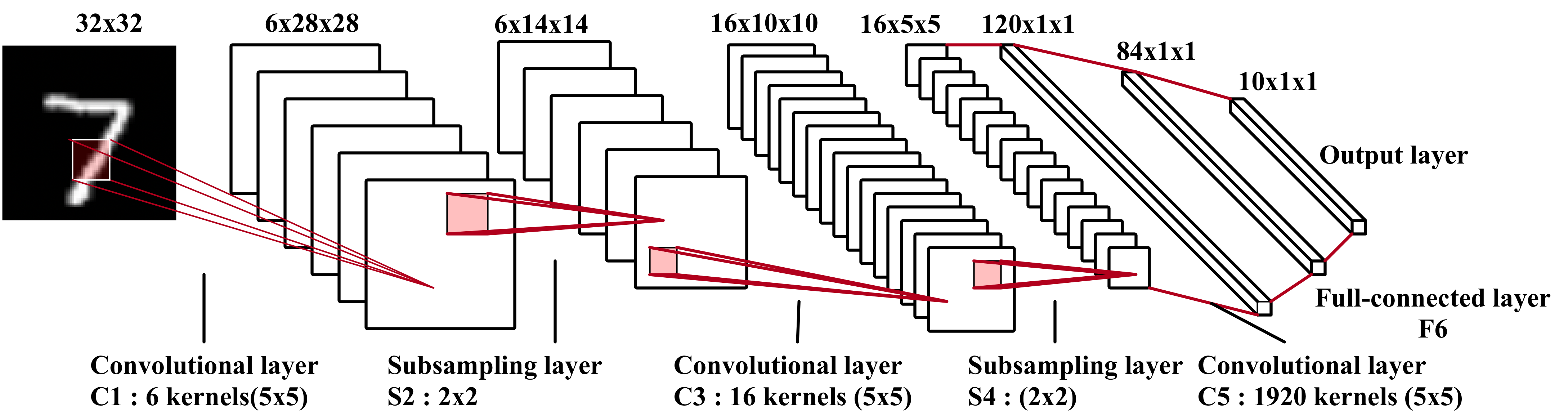
特征值的计算，我们可以用此公式表示：

表示的是位置的在第层的第个特征值。

表示的是计算第层的第个特征值所用的权值和偏置值，在原图像上滑动并计算每一个时，我们会共用，称作权值共享。

表示的是上一层中心位于的一个区块，其大小等于。

设计者发现，生物的神经元在感知的时候是从局部采集信息，逐步推广到全局的。在感知同一特征时无论其位置在何处，都是用同一规则去感知和理解它的。因此卷积层利用滑动窗口的方式来获取局部特征，再利用权值共享来大大降低模型复杂度和训练成本，并一定程度上抵抗仿射变化。



而随后激活函数可以将非线性引入CNN中，使多层网络在侦测非线性特征的表现更佳。

定义非线性激活函数，激活值。

常见的激活函数包括。

## Pooling layer

池化层的目的是通过降低特征映射的分辨率，从而实现平移不变性。其通常介于两个卷积层之间，池化层的每个特征映射和卷积层的特征映射是一一对应的。

定义池化函数，附近的区域

池化实际上是在对特征图进行压缩和提取主要特征，其分辨率会等比例下降但深度依旧不变，可以降低特征映射中的一些冗余，突出主要特征，并减少计算负担。

常用的方法有Mean Pooling和 Max Pooling等等，区别在于压缩时对一个区域采用何种方式采样。

在第一层卷积之后，提取到的还是一些low-level的比较具体的特征，而经过层叠卷积层和池化层，可以学习到更加high-level的更抽象的特征。

## Fully-connected layer

在若干个卷积层和池化层之后，通常有一个或若干个全连接层，其每一个神经元都和前一层的所有神经元相连接，用来执行high-level的推断并合成全局的语义信息。全连接层也不是必要的，其可以被一个的卷积层替代。

## Output layer

CNN的最后一层是一个输出层，用来得到我们需要的结果，分类任务通常采用softmax算法，或者SVM。

## Loss function

用表示CNN中所有的参数（权重和偏置），将样本表示为，表示网络的输出。Loss function可以表示为：

而最适宜的参数是通过降低此任务所适合的loss function的值而得到的。训练CNN的过程即是一个全局最优化问题，有一系列优化器可以用于计算最优解。