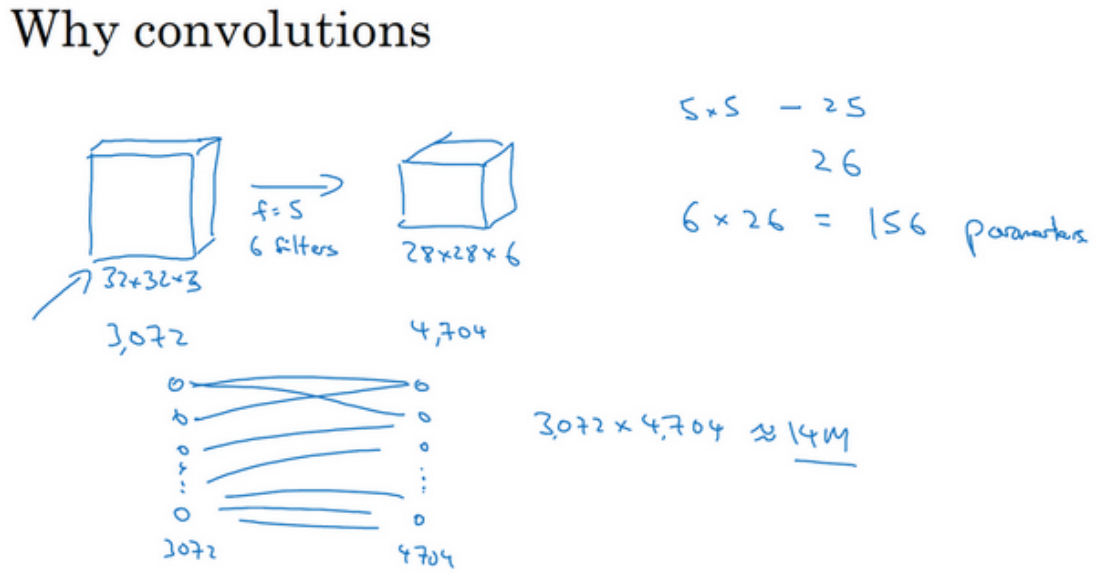
1.11 为什么使用卷积？（Why convolutions?）

这是本周最后一节课，我们来分析一下卷积在神经网络中如此受用的原因，然后对如何整合这些卷积，如何通过一个标注过的训练集训练卷积神经网络做个简单概括。和只用全连接层相比，卷积层的两个主要优势在于参数共享和稀疏连接，举例说明一下。

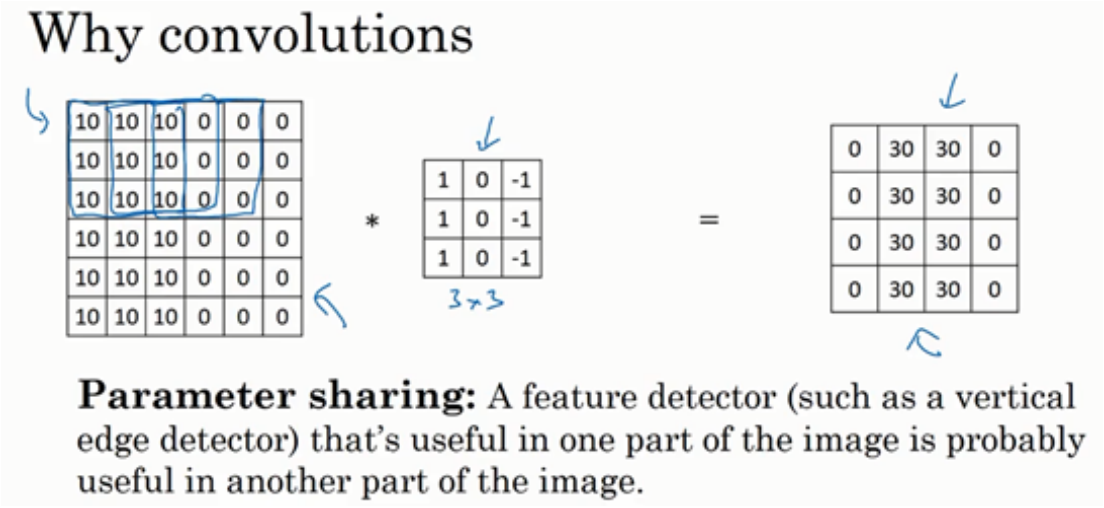


假设有一张 32×32×3 维度的图片，这是上节课的示例，假设用了 6 个大小为 5×5 的过滤器，输出维度为 28×28×6。32×32×3=3072，28×28×6=4704。我们构建一个神经网络，其中一层含有 3072 个单元，下一层含有 4074 个单元，两层中的每个神经元彼此相连，然后计算权重矩阵，它等于 4074×3072≈1400 万，所以要训练的参数很多。虽然以现在的技术，我们可以用 1400 多万个参数来训练网络，因为这张 32×32×3 的图片非常小，训练这么多参数没有问题。如果这是一张 1000×1000 的图片，权重矩阵会变得非常大。我们看看这个卷积层的参数数量，每个过滤器都是 5×5，一个过滤器有 25 个参数，再加上偏差参数，那么每个过滤器就有 26 个参数，一共有 6 个过滤器，所以参数共计 156 个，参数数量还是很少。

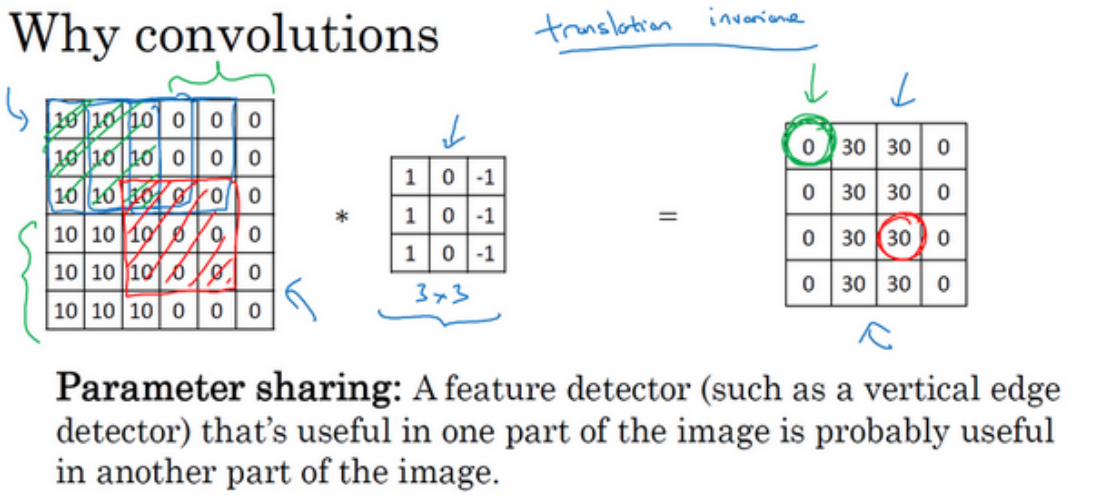
卷积网络映射这么少参数有两个原因：

一是参数共享。观察发现，特征检测如垂直边缘检测如果适用于图片的某个区域，那么它也可能适用于图片的其他区域。也就是说，如果你用一个 3×3 的过滤器检测垂直边缘，那么图片的左上角区域，以及旁边的各个区域（左边矩阵中蓝色方框标记的部分）都可以使用这个 3×3 的过滤器。每个特征检测器以及输出都可以在输入图片的不同区域中使用同样的参数，以便提取垂直边缘或其它特征。它不仅适用于边缘特征这样的低阶特征，同样适用于高阶特征，例如提取脸上的眼睛，猫或者其他特征对象。即使减少参数个数，这 9 个参数同

样能计算出 16 个输出。直观感觉是，一个特征检测器，如垂直边缘检测器用于检测图片左上角区域的特征，这个特征很可能也适用于图片的右下角区域。因此在计算图片左上角和右下角区域时，你不需要添加其它特征检测器。假如有一个这样的数据集，其左上角和右下角可能有不同分布，也有可能稍有不同，但很相似，整张图片共享特征检测器，提取效果也很好。



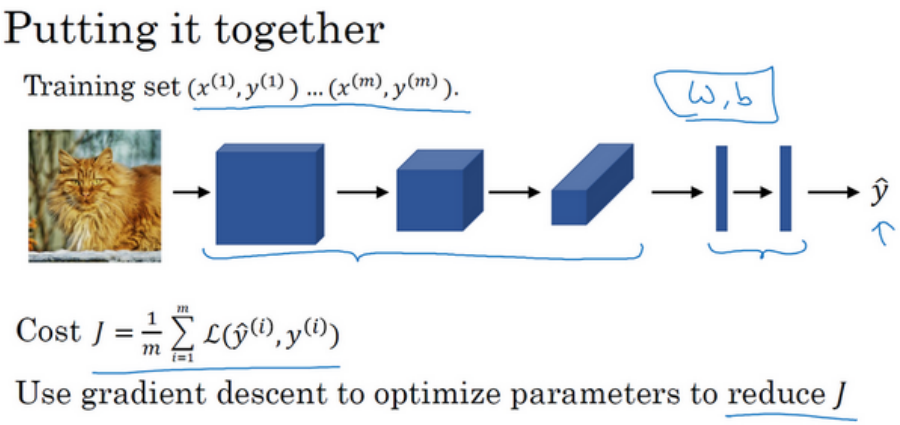
第二个方法是使用稀疏连接，我来解释下。这个 0 是通过 3×3 的卷积计算得到的，它只依赖于这个 3×3 的输入的单元格，右边这个输出单元（元素 0）仅与 36 个输入特征中 9 个相连接。而且其它像素值都不会对输出产生任影响，这就是稀疏连接的概念。



再举一个例子，这个输出（右边矩阵中红色标记的元素 30）仅仅依赖于这 9 个特征（左边矩阵红色方框标记的区域），看上去只有这 9 个输入特征与输出相连接，其它像素对输出没有任何影响。

神经网络可以通过这两种机制减少参数，以便我们用更小的训练集来训练它，从而预防过度拟合。你们也可能听过，卷积神经网络善于捕捉平移不变。通过观察可以发现，向右移动两个像素，图片中的猫依然清晰可见，因为神经网络的卷积结构使得即使移动几个像素，这张图片依然具有非常相似的特征，应该属于同样的输出标记。实际上，我们用同一个过滤器生成各层中，图片的所有像素值，希望网络通过自动学习变得更加健壮，以便更好地取得所期望的平移不变属性。

这就是卷积或卷积网络在计算机视觉任务中表现良好的原因。



最后，我们把这些层整合起来，看看如何训练这些网络。比如我们要构建一个猫咪检测器，我们有下面这个标记训练集，x表示一张图片，$\hat{y}$是二进制标记或某个重要标记。我们选定了一个卷积神经网络，输入图片，增加卷积层和池化层，然后添加全连接层，最后输出一个 **softmax**，即$\hat{y}$。卷积层和全连接层有不同的参数w和偏差b，我们可以用任何参数集合来定义代价函数。一个类似于我们之前讲过的那种代价函数，并随机初始化其参数w和b，代价函数J等于神经网络对整个训练集的预测的损失总和再除以m（即$\text{Cost}\, J = \frac{1}{m}\sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)})$）。所以训练神经网络，你要做的就是使用梯度下降法，或其它算法，例如 **Momentum** 梯度下降法，含 **RMSProp** 或其它因子的梯度下降来优化神经网络中所有参数，以减少代价函数J的值。通过上述操作你可以构建一个高效的猫咪检测器或其它检测器。