liumaolincycle的博客

（本论文是我在做本科毕设时翻译的，已经有两年了，但现在看来这篇论文依然非常经典，所以直接放上来了，水平有限，欢迎指正）

ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton

摘要

我们训练了一个大型的深度卷积神经网络，来将在ImageNet LSVRC-2010大赛中的120万张高清图像分为1000个不同的类别。对测试数据，我们得到了top-1误差率37.5%，以及top-5误差率17.0%，这个效果比之前最顶尖的都要好得多。该神经网络有6000万个参数和650,000个神经元，由五个卷积层，以及某些卷积层后跟着的max-pooling层，和三个全连接层，还有排在最后的1000-way的softmax层组成。为了使训练速度更快，我们使用了非饱和的神经元和一个非常高效的GPU关于卷积运算的工具。为了减少全连接层的过拟合，我们采用了最新开发的正则化方法，称为“dropout”，它已被证明是非常有效的。在ILSVRC-2012大赛中，我们又输入了该模型的一个变体，并依靠top-5测试误差率15.3%取得了胜利，相比较下，次优项的错误率是26.2%。

1 引言

当前物体识别的方法基本上都使用了机器学习方法。为了改善这些方法的性能，我们可以收集更大的数据集，学习更强有力的模型，并使用更好的技术，以防止过拟合。直到最近，标记图像的数据集都相当小——大约数万张图像（例如，NORB [16]，Caltech-101/256 [8, 9]，以及CIFAR-10/100 [12]）。简单的识别任务可以用这种规模的数据集解决得相当好，特别是当它们用标签-保留转换增强了的时候。例如，在MNIST数字识别任务中当前最好的误差率（<0.3%）接近于人类的表现[4]。但是现实环境中的物体表现出相当大的变化，因此要学习它们以对它们进行识别就必须使用更大的训练集。事实上，小规模图像数据集的缺陷已被广泛认同（例如，Pinto等人[21]），但是直到最近，收集有着上百万张图像的带标签数据集才成为可能。更大型的新数据集包括LabelMe [23]，它由几十万张完全分割图组成，还有ImageNet [6]，它由多于22,000个种类中超过1500万张带标签的高分辨率图像组成。

为了从几百万张图像中学习数以千计的物体，我们需要一个学习能力更强的模型。然而，物体识别任务的极大复杂性意味着这个问题不能被指定，即使是通过与ImageNet一样大的数据集，所以我们的模型中也应该有大量的先验知识，以补偿我们所没有的全部数据。卷积神经网络（CNN）构成了一个这种类型的模型[16, 11, 13, 18, 15, 22, 26]。它们的能力可以通过改变其深度与广度得到控制，它们也可作出有关图像性质的强壮且多半正确的假设（即，统计数据的稳定性和像素依赖关系的局部性）。因此，与层次规模相同的标准前馈神经网络相比，CNN的连接关系和参数更少，所以更易于训练，而其理论上的最佳性能可能只略差一点。

不论CNN的性质多有吸引力，也不论它们局部结构的相对效率有多高，将它们大规模地应用到高分辨率图像中仍然是极其昂贵的。幸运的是，目前的GPU搭配了一个高度优化的2D卷积工具，强大到足以促进大规模CNN的训练，而且最近的数据集像ImageNet包含足够的带标签的样例来训练这样的模型，还不会有严重的过拟合。

本文的具体贡献如下：我们在ILSVRC-2010和ILSVRC-2012大赛中使用过的ImageNet的子集上[2]，训练了迄今为止最大型的卷积神经网络之一，并取得了迄今为止在这些数据集上报告过的最好结果。我们写了一个高度优化的GPU二维卷积工具以及训练卷积神经网络过程中的所有其他操作，这些我们都提供了公开地址。我们的网络中包含一些既新鲜而又不同寻常的特征，它们提高了网络的性能，并减少了网络的训练时间，这些详见第3节。我们的网络中甚至有120万个带标签的训练样本，这么大的规模使得过拟合成为一个显著的问题，所以我们使用了几种有效的方法来防止过拟合，这些在第4节中给以描述。我们最终的网络包含五个卷积层和三个全连接层，且这种层次深度似乎是重要的：我们发现，移去任何卷积层（其中每一个包含的模型参数都不超过1%）都会导致性能变差。

最后，网络的规模主要受限于当前GPU的可用内存和我们愿意容忍的训练时间。我们的网络在两块GTX 580 3GB GPU上训练需要五到六天。我们所有的实验表明，等更快的GPU和更大的数据集可用以后，我们的结果就可以轻而易举地得到改进。

2 数据集

ImageNet是一个拥有超过1500万张带标签的高分辨率图像的数据集，这些图像分属于大概22,000个类别。这些图像是从网上收集，并使用Amazon Mechanical Turk群众外包工具来人工贴标签的。作为PASCAL视觉目标挑战赛的一部

分，一年一度的ImageNet大型视觉识别挑战赛（ILSVRC）从2010年开始就已经在举办了。ILSVRC使用ImageNet的一个子集，分为1000种类别，每种类别中都有大约1000张图像。总之，大约有120万张训练图像，50,000张验证图像和150,000张测试图像。

ILSVRC-2010是ILSVRC中能获得测试集标签的唯一版本，因此这也就是我们完成大部分实验的版本。由于我们也在ILSVRC-2012上输入了模型，在第6节中我们也会报告这个数据集版本上的结果，该版本上的测试集标签难以获取。在ImageNet上，习惯性地报告两个误差率：top-1和top-5，其中top-5误差率是指测试图像上正确标签不属于被模型认为是最有可能的五个标签的百分比。

ImageNet由各种分辨率的图像组成，而我们的系统需要一个恒定的输入维数。因此，我们下采样这些图像到固定的分辨率256×256。给定一张矩形图像，我们首先重新缩放图像，使得短边长度为256，然后从得到的图像中裁剪出中央256×256的一片。除了遍历训练集从每个像素中减去平均活跃度外，我们没有以任何其他方式预处理图像。所以我们用这些像素（中央那一片的）原始RGB值训练网络。

3 体系结构

图2总结了我们网络的体系结构。它包含八个学习层——五个卷积层和三个全连接层。下面，我们将介绍该网络体系结构的一些新颖独特的功能。3.1-3.4是根据我们对于其重要性的估计来排序的，最重要的排在最前面。

3.1 ReLU非线性

将神经元的输出f，作为其输入x的函数，对其建模的标准方法是用 $f(x) = \tanh(x)$ 或者 $f(x) = (1+e^{-x})^{-1}$。就梯度下降的训练时间而言，这些饱和非线性函数比不饱和非线性函数 $f(x) = \max(0, x)$ 要慢得多。我们跟随Nair和Hinton[20]称这种不饱和非线性的神经元为修正线性单元（ReLU）。训练带ReLUs的深度卷积神经网络比带tanh单元的同等网络要快好几倍。如图1所示，它显示出对于特定的四层卷积网络，在CIFAR-10数据集上达到25%的训练误差所需的迭代次数。此图显示，如果我们使用了传统的饱和神经元模型，就不能用如此大的神经网络来对该工作完成实验。

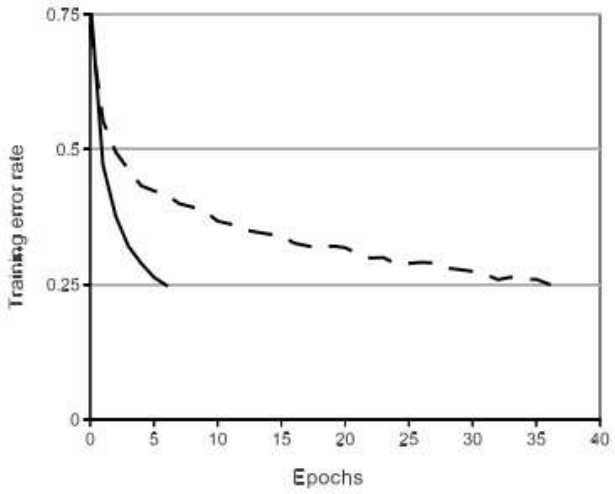

图1：带ReLU的四层卷积神经网络（实线）在CIFAR-10数据集上达到25%训练误差率要比带tanh神经元的同等网络（虚线）快六倍。每个网络的学习速率是独立选取的，以使得训练尽可能快。没有使用任何形式的正则化。这里演示的效果因网络结构的不同而不同，但带ReLU的网络学习始终比带饱和神经元的同等网络快好几倍。

我们不是第一个在CNN中考虑传统神经元模型的替代品的。例如，Jarrett等人[11]声称，非线性函数 $f(x) = |\tanh(x)|$ 由于其后随局部average pooling的对比度归一化的类型，它在Caltech-101数据集上工作得特别好。然而，在该数据集上的主要关注点是防止过拟合，所以他们正在观察的效果不同于我们报告的为拟合训练集使用ReLU时的加速能力。更快的学习对大型数据集上训练的大型模型的性能有很大影响。

3.2 在多个GPU上训练

单个GTX 580 GPU只有3GB内存，这限制了可以在其上训练的网络的最大规模。事实证明，120万个训练样本才足以训练网络，这网络太大了，不适合在一个GPU上训练。因此我们将网络分布在两个GPU上。目前的GPU特别适合跨GPU并行化，因为它们能够直接从另一个GPU的内存中读出和写入，不需要通过主机内存。我们采用的并行方案基本上是在每个GPU中放置一半核（或神经元），还有一个额外的技巧：GPU间的通讯只在某些层进行。这就是说，例如，第3层的核需要从第2层中所有核映射输入。然而，第4层的核只需要从第3层中位于同一GPU的那些核映射输入。选择连接模式是一个交叉验证的问题，但是这让我们可以精确地调整通信量，直到它的计算量在可接受的部分。

由此产生的体系结构有点类似于Ciresan等人提出的“柱状”CNN的体系结构[5]，不同之处在于我们的纵列不是独立的（见图2）。与在一个GPU上训练的每个卷积层有一半核的网络比较，该方案将我们的top-1与top-5误差率分别减少了1.7%与1.2%。训练双GPU网络比训练单GPU网络花费的时间略少一些（实际上单GPU网络与双GPU网络在最后的卷积层有着相同数量的核。这是因为大多数网络的参数在第一个全连接层，这需要上一个卷积层作为输入。所以，为了使两个网络有数目大致相同的参数，我们不把最后一个卷积层大小减半（也不把它后面跟随的全连接层减半）。因此，这种比较关系更偏向有利于单GPU网络，因为它比双GPU网络的“一半大小”要大）。

3.3 局部响应归一化

ReLU具有所希望的特性，它们不需要输入归一化来防止它们达到饱和。如果至少有一些训练样例对ReLU产生了正输入，学习就将发生在那个神经元。可是，我们仍然发现下列局部归一化方案有助于一般化。用$a^i_{x,y}$表示点(x,y)处通过应用核 计算出的神经元激活度，然后应用ReLU非线性，响应归一化活性$b^i_{x,y}$由下式给出

$$b^i_{x,y} = a^i_{x,y} \Big/ \left(k + \alpha \sum_{j=\max(0,i-n/2)}^{\min(N-1,i+n/2)} (a^j_{x,y})^2 \right)^\beta$$

其中求和覆盖了n个“相邻的”位于相同空间位置的核映射，N是该层中的核总数。核映射的顺序当然是任意的，且在训练开始前就确定。受到在真实神经元中发现的类型启发，这种响应归一化实现了一种侧向抑制，在使用不同核计算神经元输出的过程中创造对大激活度的竞争。常数k，n，α和β是超参数，它们的值要用验证集来确定；我们使用$k=2$，$n=5$，$\alpha=10^{-4}$，$\beta=0.75$。我们在某些层应用ReLU归一化后再应用这种归一化（见3.5节）。

该方案与Jarrett等人的局部对比度归一化方案具有一些相似之处[11]，但我们的方案更正确的命名为“亮度归一化”，因为我们不减去平均活跃度。响应归一化将我们的top-1与top-5误差率分别减少了1.4%与1.2%。我们也验证了该方案在CIFAR-10数据集上的有效性：四层CNN不带归一化时的测试误差率是13%，带归一化时是11%（由于版面有限我们不能详细描述该网络，但这里提供的代码和参数文件对其有精确详细的说明：http://code.google.com/p/cuda-convnet/）。

3.4 重叠Pooling

CNN中的Pooling层总结了同一核映射中邻近神经元组的输出。传统上，通过邻接pooling单元总结的邻近关系不重叠（例如，[17,11,4]）。更准确地说，一个pooling层可以被认为是由间隔s像素的pooling单元网格组成，每个网格总结出一个z×z大小的邻近关系，都位于pooling单元的中心位置。若设s=z，我们得到传统的局部pooling，正如常用于CNN中的那样。若设s

3.5 总体结构

现在，我们已经准备好描述CNN的总体结构。如图2所示，该网络包括八个带权层；前五层是卷积层，剩下三层是全连接层。最后一个全连接层的输出被送到一个1000-way的softmax层，其产生一个覆盖1000类标签的分布。我们的网络使得多分类的Logistic回归目标最大化，这相当于最大化了预测分布下训练样本中正确标签的对数概率平均值。

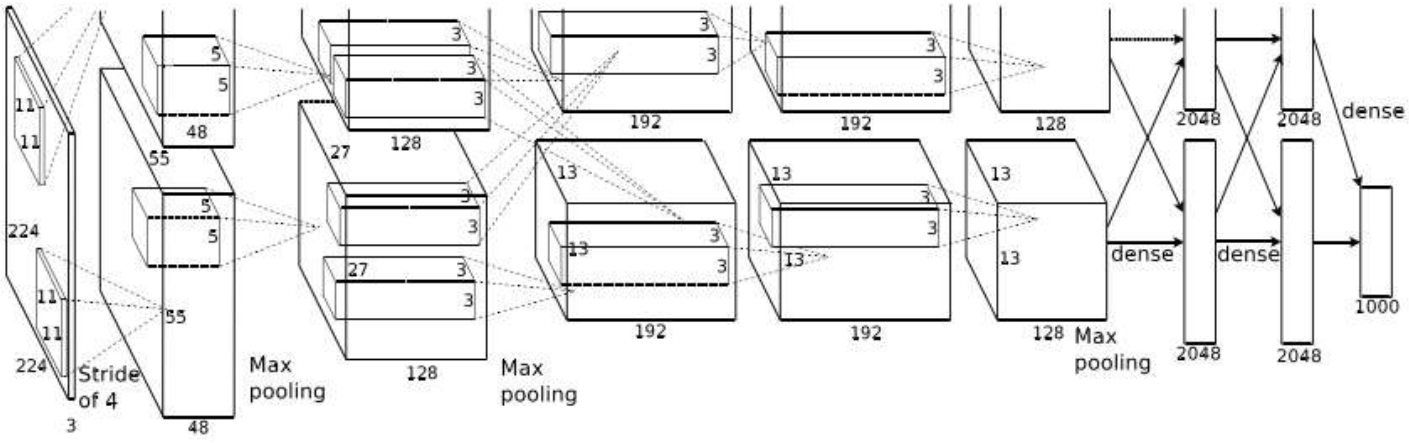

图2：CNN体系结构示意图，明确显示了两个GPU之间的职责划分。一个GPU运行图中顶部的层次部分，而另一个GPU运行图中底部的层次部分。GPU之间仅在某些层互相通信。该网络的输入是150,528维的，且该网络剩下各层的神经元数分别为253,440–186,624–64,896–64,896–43,264–4096–4096–1000。

第二、第四和第五个卷积层的核只连接到前一个卷积层也位于同一GPU中的那些核映射上（见图2）。第三个卷积层的核被连接到第二个卷积层中的所有核映射上。全连接层中的神经元被连接到前一层中所有的神经元上。响应归一化层跟

在第一、第二个卷积层后面。3.4节中描述的那种最大Pooling层，跟在响应归一化层以及第五个卷积层之后。ReLU非线性应用于每个卷积层及全连接层的输出。
第一个卷积层利用96个大小为11×11×3、步长为4个像素（这是同一核映射中邻近神经元的感受野中心之间的距离）的核，来对大小为224×224×3的输入图像进行滤波。第二个卷积层需要将第一个卷积层的（响应归一化及池化的）输出作为自己的输入，且利用256个大小为5×5×48的核对其进行滤波。第三、第四和第五个卷积层彼此相连，没有任何介于中间的pooling层与归一化层。第三个卷积层有384个大小为3×3×256的核被连接到第二个卷积层的（归一化的、池化的）输出。第四个卷积层拥有384个大小为3×3×192的核，第五个卷积层拥有256个大小为3×3×192的核。全连接层都各有4096个神经元。

4 减少过拟合

我们的神经网络结构有6000万个参数。虽然ILSVRC的1000个类别使得每个训练样本强加10比特约束到从图像到标签的映射上，这显示出要学习如此多的参数而不带相当大的过拟合，这些类别是不够的。下面，我们描述减少过拟合的两种主要方法。

4.1 数据增强

减少图像数据过拟合最简单最常用的方法，是使用标签-保留转换，人为地扩大数据集（例如，[25,4,5]）。我们使用数据增强的两种不同形式，这两种形式都允许转换图像用很少的计算量从原始图像中产生，所以转换图像不需要存储在磁盘上。在我们的实现中，转换图像是由CPU上的Python代码生成的，而GPU是在之前那一批图像上训练的。所以这些数据增强方案实际上是计算自由。
数据增强的第一种形式由生成图像转化和水平反射组成。为此，我们从256×256的图像中提取随机的224×224的碎片（还有它们的水平反射），并在这些提取的碎片上训练我们的网络（这就是图2中输入图像是224×224×3维的原因）。这使得我们的训练集规模扩大了2048倍，但是由此产生的训练样例一定高度地相互依赖。如果没有这个方案，我们的网络会有大量的过拟合，这将迫使我们使用小得多的网络。在测试时，该网络通过提取五个224×224的碎片（四个边角碎片和中心碎片）连同它们的水平反射（因此总共是十个碎片）做出了预测，并在这十个碎片上来平均该网络的softmax层做出的预测。
数据增强的第二种形式包含改变训练图像中RGB通道的强度。具体来说，我们在遍及整个ImageNet训练集的RGB像素值集合中执行PCA。对于每个训练图像，我们成倍增加已有主成分，比例大小为对应特征值乘以一个从均值为0，标准差为0.1的高斯分布中提取的随机变量。这样一来，对于每个RGB图像像素$I_{xy} = [I_{xy}^R, I_{xy}^G, I_{xy}^B]^T$，我们增加下面这项：

$$[\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3][\alpha_1\lambda_1, \alpha_2\lambda_2, \alpha_3\lambda_3]^T$$

其中$\mathbf{p}_i$与λ_i分别是RGB像素值的3×3协方差矩阵的第i个特征向量与特征值，α_i是前面提到的随机变量。每个α_i对于特定训练图像的全部像素只提取一次，直到那个图像再次被用于训练，在那时它被重新提取。这个方案大致抓住了自然图像的一个重要属性，即，光照强度与颜色是变化的，而对象识别是不变的。该方案将top-1误差率减少了1%以上。

4.2 Dropout

结合许多不同模型的预测是一种非常成功的减少测试误差的方式[1,3]，但它先前训练花了好几天时间，似乎对于大型神经网络来说太过昂贵。然而，有一个非常有效的模型组合版本，它在训练中只花费两倍于单模型的时间。最近推出的叫做“dropout”的技术[10]，它做的就是以0.5的概率将每个隐层神经元的输出设置为零。以这种方式“dropped out”的神经元既不利于前向传播，也不参与反向传播。所以每次提出一个输入，该神经网络就尝试一个不同的结构，但是所有这些结构之间共享权重。因为神经元不能依赖于其他特定神经元而存在，所以这种技术降低了神经元复杂的互适应关系。正因如此，要被迫学习更为鲁棒的特征，这些特征在结合其他神经元的一些不同随机子集时有用。在测试时，我们将所有神经元的输出都仅仅只乘以0.5，对于获取指数级dropout网络产生的预测分布的几何平均值，这是一个合理的近似方法。
我们在图2中前两个全连接层使用dropout。如果没有dropout，我们的网络会表现出大量的过拟合。dropout使收敛所需的迭代次数大致增加了一倍。

5 学习的详细过程

我们使用随机梯度下降法和一批大小为128、动力为0.9、权重衰减为0.0005的样例来训练我们的网络。我们发现，这少量的权重衰减对于模型学习是重要的。换句话说，这里的权重衰减不仅仅是一个正则化矩阵：它减少了模型的训练误差。对于权重w的更新规则为

$$v_{i+1} := 0.9 \cdot v_i - 0.0005 \cdot \varepsilon \cdot w_i - \varepsilon \cdot \left\langle \frac{\partial L}{\partial w}\Big|_{w_i} \right\rangle_{D_i}$$

$$w_{i+1} := w_i + v_{i+1}$$

其中i是迭代指数，v是动力变量，ε是学习率，$\left\langle \frac{\partial L}{\partial w}\Big|_{w_i} \right\rangle_{D_i}$是目标关于w、对$w_i$求值的导数在第i批样例$D_i$上的平均值。

我们用一个均值为0、标准差为0.01的高斯分布初始化了每一层的权重。我们用常数1初始化了第二、第四和第五个卷积层以及全连接隐层的神经元偏差。该初始化通过提供带正输入的ReLU来加速学习的初级阶段。我们在其余层用常数0初始化神经元偏差。

我们对于所有层都使用了相等的学习率，这是在整个训练过程中手动调整的。我们遵循的启发式是，当验证误差率在当前学习率下不再提高时，就将学习率除以10。学习率初始化为0.01，在终止前降低三次。我们训练该网络时大致将这120万张图像的训练集循环了90次，在两个NVIDIA GTX 580 3GB GPU上花了五到六天。

6 结果

我们在ILSVRC-2010测试集上的结果总结于表1中。我们的网络实现了top-1测试集误差率**37.5%**，top-5测试集误差率**17.0%**（若没有如4.1节所述的在十个碎片上平均预测，误差率是39.0%与18.3%）。ILSVRC-2010大赛中取得的最好表现是47.1%与28.2%，它的方法是用不同特征训练六个sparse-coding模型，对这些模型产生的预测求平均值[2]，自那以后公布的最好结果是45.7%与25.7%，它的方法是从两类密集采样的特征中计算出费舍尔向量（FV），用费舍尔向量训练两个分类器，再对这两个分类器的预测求平均值[24]。

Model	Top-1	Top-5
Sparse coding [2]	*47.1%*	*28.2%*
SIFT + FVs [24]	*45.7%*	*25.7%*
CNN	**37.5%**	**17.0%**

表1：ILSVRC-2010测试集上的结果比较。斜体字是他人取得的最好结果。

我们也在ILSVRC-2012大赛中输入了我们的模型，并在表2中报告结果。由于ILSVRC-2012测试集标签是不公开的，我们不能对试过的所有模型都报告测试误差率。在本段的其余部分，我们将验证误差率与测试误差率互换，因为根据我们的经验，它们之间相差不超过0.1%（见表2）。本文所描述的CNN实现了18.2%的top-5误差率。对五个相似CNN的预测求平均值得出了16.4%的误差率。训练一个在最末pooling层之后还有一个额外的第六个卷积层的CNN，用以对整个ImageNet 2011年秋季发布的图像（15M张图像，22K种类别）进行分类，然后在ILSVRC-2012上“微调”它，这种方法得出了16.6%的误差率。用在整个2011年秋季发布的图像上预训练的两个CNN，结合先前提到的五个CNN，再对这七个CNN作出的预测求平均值，这种方法得出了**15.3%**的误差率。比赛中的第二名实现了26.2%的误差率，用的方法是从不同类密集采样的特征中计算FV，用FV训练几个分类器，再对这几个分类器的预测求平均值[7]。

Model	Top-1 (val)	Top-5 (val)	Top-5 (test)
SIFT + FVs [7]	—	—	*26.2%*
1 CNN	40.7%	18.2%	—
5 CNNs	38.1%	16.4%	**16.4%**
1 CNN*	39.0%	16.6%	—
7 CNNs*	36.7%	15.4%	**15.3%**

表2：在ILSVRC-2012验证集与测试集上的误差率比较。斜体字是由他人取得的最好结果。带星号的模型是经过“预训练”以对整个ImageNet 2011年秋季发布的图像集进行分类的。详见第6节。

最后，我们还报告在ImageNet 2009年秋季版本上的误差率，该版本有10,184种类别与890万张图像。在这个数据集上，我们按照文献惯例，用一半图像来训练，用另一半图像来测试。由于没有确定的测试集，我们的划分必然不同于以前的作者使用的划分，但这并不会明显地影响到结果。我们在该数据集上的top-1误差率和top-5误差率分别为67.4%和40.9%，这是通过上述的网络得到的，但还有个附加条件，第六个卷积层接在最后一个pooling层之后。该数据集上公布的最佳结果是78.1%和60.9%[19]。

6.1 定性评价

图3显示了通过该网络的两个数据连接层学习到的卷积核。该网络已经学习到各种各样的频率与方向选择核，以及各种颜色的斑点。注意两个GPU显现出的特性，3.5节中描述了一个结果是限制连接。GPU1上的核大多数颜色不明确，而GPU2上的核大多数颜色明确。这种特性在每一次运行中都会出现，且独立于所有特定的随机权重初始化（以GPU的重新编数为模）。

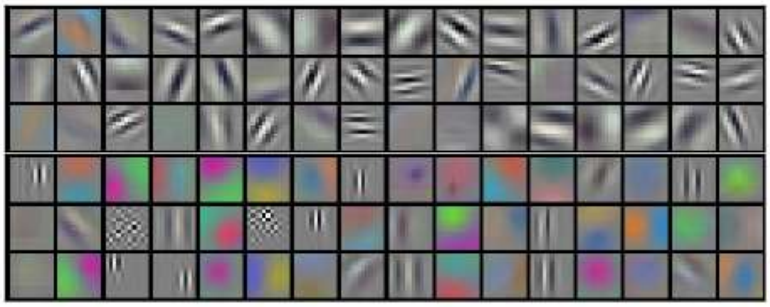

图3：通过 的输入图像上第一个卷积层学习到的96个大小为 的卷积核。顶部的48个核是从GPU1上学到的，底部的48个核是从GPU2上学到的。详见6.1节。

在图4左边面板上，通过计算该网络在八个测试图像上的top-5预测，我们定性地判断它学到了什么。注意到即使是偏离中心的物体，比如左上角的一小块，也可以被网络识别。大多数的top-5标签似乎合情合理。例如，只有其他类型的猫科动物被认为是对豹貌似合理的标签。在某些情况下（铁栅、樱桃），对于图片意图的焦点存在歧义。

图4：（左图）八个ILSVRC-2010测试图像，以及被我们的模型认为最有可能的五个标签。正确的标签写在每个图像下面，正确标签的概率也以红色条予以显示（若它在前5之内）。（右图）第一列是五个ILSVRC-2010测试图像。其余列显示了六个训练图像，它们在最后的隐层产生的特征向量与测试图像的特征向量有最小的欧氏距离。

探测网络的视觉知识有另一种方法，就是考虑由位于最后的4096维隐层上的图像引起的特征激活。如果两个图像用小欧氏分离产生了特征激活向量，我们可以说，在神经网络的更高级别上认为它们是相似的。图4显示了测试集中的五个图像，以及训练集中根据这一标准与其中每一个最相似的六个图像。注意，在像素级别，检索到的训练图像一般不会接近第一列中的查询图像。例如，检索到的狗和大象表现出各种各样的姿势。我们会在补充材料里给出更多测试图像的结果。
通过使用两个4096维实值向量之间的欧氏距离来计算相似性是低效的，但它可以通过训练一个自动编码器将这些向量压缩为短的二进制代码来变得高效。这应该会产生一个比应用自动编码器到原始像素要好得多的图像检索方法[14]，它不利用图像标签，此后还有一种用相似边缘图案来检索图像的倾向，而不论它们在语义上是否相似。

7 讨论

我们的研究结果表明，大型深度卷积神经网络在一个非常具有挑战性的数据集上使用纯粹的监督学习，能够达到破纪录的结果。值得注意的是，如果有一个卷积层被移除，我们的网络性能就会降低。例如，除去任何中间层都将导致该网络的top-1性能有2%的损失。所以该层次深度对于达到我们的结果确实是重要的。
为了简化实验，我们没有使用任何无监督的预训练，即使我们预计它将带来帮助，特别是我们可以获得足够的计算能力来显著地扩大网络规模，而不带来标记数据量的相应增加。到目前为止，我们的结果有所改善，因为我们已经让网络更

大，训练时间更久，但是为了匹配人类视觉系统的infero-temporal路径，我们仍然有更高的数量级要去达到。最终我们想要在视频序列上使用非常大型的深度卷积网络，其中的瞬时结构会提供非常有用的信息，这些信息在静态图像中丢失了或极不明显。

参考文献

[1] R.M. Bell and Y. Koren. Lessons from the netflix prize challenge. ACM SIGKDD Explorations Newsletter, 9(2):75–79, 2007.
[2] A. Berg, J. Deng, and L. Fei-Fei. Large scale visual recognition challenge 2010. www.image-net.org/challenges. 2010.
[3] L. Breiman. Random forests. Machine learning, 45(1):5–32, 2001.
[4] D. Cires¸an, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. Arxiv preprint arXiv:1202.2745, 2012.
[5] D.C. Cires¸an, U. Meier, J. Masci, L.M. Gambardella, and J. Schmidhuber. High-performance neural networks for visual object classification. Arxiv preprint arXiv:1102.0183, 2011.
[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In CVPR09, 2009.
[7] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. Fei-Fei. ILSVRC-2012, 2012. URL http://www.image-net.org/challenges/LSVRC/2012/.
[8] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Computer Vision and Image Understanding, 106(1):59–70, 2007.
[9] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007. URL http://authors.library.caltech.edu/7694.
[10] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R.R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580, 2012.
[11] K. Jarrett, K. Kavukcuoglu, M. A. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In International Conference on Computer Vision, pages 2146–2153. IEEE, 2009.
[12] A. Krizhevsky. Learning multiple layers of features from tiny images. Master’ s thesis, Department of Computer Science, University of Toronto, 2009.
[13] A. Krizhevsky. Convolutional deep belief networks on cifar-10. Unpublished manuscript, 2010.
[14] A. Krizhevsky and G.E. Hinton. Using very deep autoencoders for content-based image retrieval. In ESANN, 2011.
[15] Y. Le Cun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, et al. Handwritten digit recognition with a back-propagation network. In Advances in neural information processing systems, 1990.
[16] Y. LeCun, F.J. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, volume 2, pages II–97. IEEE, 2004.
[17] Y. LeCun, K. Kavukcuoglu, and C. Farabet. Convolutional networks and applications in vision. In Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on, pages 253–256. IEEE, 2010.
[18] H. Lee, R. Grosse, R. Ranganath, and A.Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In Proceedings of the 26th Annual International Conference on Machine Learning, pages 609–616. ACM, 2009.
[19] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Metric Learning for Large Scale Image Classification: Generalizing to New Classes at Near-Zero Cost. In ECCV - European Conference on Computer Vision, Florence, Italy, October 2012.
[20] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In Proc. 27th International Conference on Machine Learning, 2010.
[21] N. Pinto, D.D. Cox, and J.J. DiCarlo. Why is real-world visual object recognition hard? PLoS computational biology, 4(1):e27, 2008.
[22] N. Pinto, D. Doukhan, J.J. DiCarlo, and D.D. Cox. A high-throughput screening approach to discovering good forms of biologically inspired visual representation. PLoS computational biology, 5(11):e1000579, 2009.
[23] B.C. Russell, A. Torralba, K.P. Murphy, and W.T. Freeman. Labelme: a database and web-based tool for image annotation. International journal of computer vision, 77(1):157–173, 2008.
[24] J. Sánchez and F. Perronnin. High-dimensional signature compression for large-scale image classification. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 1665–1672. IEEE, 2011.
[25] P.Y. Simard, D. Steinkraus, and J.C. Platt. Best practices for convolutional neural networks applied to visual document analysis. In Proceedings of the Seventh International Conference on Document Analysis and

Recognition, volume 2, pages 958–962, 2003.
[26] S.C. Turaga, J.F. Murray, V. Jain, F. Roth, M. Helmstaedter, K. Briggman, W. Denk, and H.S. Seung. Convolutional networks can learn to generate affinity graphs for image segmentation. Neural Computation, 22(2):511–538, 2010.