DeepL
订阅DeepL Pro以翻译大型文件。
欲了解更多信息，请访问www.DeepL.com/pro。

arXiv:1409.4842v1 [cs.CV] 2014年9月17日

更深层次的转折

克里斯蒂安-塞格迪
谷歌公司

刘伟
北卡罗来纳大学教堂山分校

杨庆佳
谷歌公司

Pierre Sermanet
谷歌公司

斯科特-里德
密歇根大学

Dragomir Anguelov
谷歌公司

杜米特鲁-埃尔汗
谷歌公司

文森特-范豪克
谷歌公司

安德鲁-拉宾诺维奇
谷歌公司

摘要

我们提出了一种代号为 "Incep- tion "的深度卷积神经网络架构，该架构在 2014 年 ImageNet 大规模视觉识别挑战赛（ILSVRC14）中为分类和检测设定了新的技术标准。该架构的主要特点是提高了网络内部计算资源的利用率。这是通过精心设计实现的，在保持计算预算不变的情况下，可以增加网络的深度和宽度。为了优化质量，架构决策基于希比原理和多尺度处理的直觉。我们向 ILSVRC14 提交的论文中使用的一个特殊版本被称为 GoogLeNet，它是一个 22 层的深度网络，其质量是在分类和检测的背景下评估的。

1 导言

近三年来，主要由于深度学习（更具体地说是卷积网络[10]）的进步，图像识别和物体检测的质量一直在飞速提高。一个令人鼓舞的消息是，大部分进步不仅仅是更强大的硬件、更大的数据集和更大的模型的结果，而主要是新思路、新算法和改进的网络架构的结果。例如，在2014年ILSVRC竞赛中，除了用于检测目的的分类数据集外，排名靠前的参赛作品没有使用新的数据源。我们向 ILSVRC 2014 提交的 GoogLeNet 所使用的参数实际上比两年

前 Krizhevsky 等人[9]的获奖架构少了 *12* 倍，而准确度却高出很多。物体检测领域最大的进步并非来自于单独使用深度网络或更大的模型，而是来自于深度架构与经典计算机视觉的协同作用，例如 Girshick 等人[6]的 R-CNN 算法。

另一个值得注意的因素是，随着移动计算和嵌入式计算的不断发展，我们算法的效率（尤其是功耗和内存使用）变得越来越重要。值得注意的是，本文所介绍的深度架构的设计考虑了这一因素，而不是单纯地追求准确率。在大多数实验中，模型的设计都是为了在推理时保持 15 亿次乘法加法的计算预算，这样它们就不会最终成为纯粹的学术奇观，而是可以以合理的成本投入实际应用，即使是在大型数据集上。

在本文中，我们将重点讨论一种用于计算机视觉的高效深度神经网络架构，其代号为"Inception"，其名称源自 Lin 等人的网络论文[12]中的 "Network in network"，以及著名的"we need to go deeper "网络流行语[1]。在我们的案例中，"深入 "一词有两种不同的含义：首先是指我们以 "Inception 模块 "的形式引入了一个新的组织层次，还有更直接的含义，即增加网络深度。总的来说，我们可以把 Inception 模型视为 [12] 的逻辑结晶，同时从 Arora 等人 [2] 的理论研究中获得灵感和指导。该架构的优势在 ILSVRC 2014 分类和检测挑战赛上得到了实验验证，其性能明显优于目前的技术水平。

2 相关工作

从 LeNet-5 [10]开始，卷积神经网络（CNN）通常采用标准结构--堆叠卷积层（可选择对比度归一化和最大池化）之后是一个或多个全连接层。这种基本设计的变体在图像分类文献中非常普遍，并在 MNIST、CIFAR 以及最著名的 ImageNet 分类挑战中取得了迄今为止最好的结果[9, 21]。对于像 Imagenet 这样的大型数据集，最近的趋势是增加层数 [12] 和层大小 [21, 14]，同时使用 dropout [7] 来解决过拟合问题。

尽管有人担心最大池化层会导致准确空间信息的丢失，但与 [9] 相同的卷积网络架构也被成功用于定位 [9, 14]、物体检测 [6, 14, 18, 5] 和人体姿态估计 [19]。受到灵长类动物视觉皮层神经科学模型的启发，Serre 等人[15]使用了一系列不同大小的固定 Gabor 滤波器，以处理多个尺度，这与 Inception 模型类似。不过，与 [15] 中的固定 2 层深度模型不同，Inception 模型中的所有滤波器都是通过学习获得的。此外，Inception 层会重复多次，因此 GoogLeNet 模型会有 22 层深度模型。

网络中的网络（Network-in-Network）是 Lin 等人[12]提出的一种方法，旨在提高神经网络的表征能力。当应用于卷积层时，该方法可被视为额外的 1×1 卷积层，之后是典型的整流线性激活[9]。这样，它就能很容易地集成到当前的 CNN 管道中。我们在架构中大量使用了这种方法。不过，在我们的设置中，1×1 卷积具有双重目的：最关键的是，它们主要用作降维模块，以消除计算瓶颈，否则会限制我们网络的规模。这样，我们不仅可以增加网络的深度，还可以增加网络的宽度，而不会对性能造成显著影响。

Girshick 等人提出的区域卷积神经网络（Regions with Convolutional Neural Net- works，R-CNN）是目前最主要的物体检测方法[6]。R-CNN 将整体检测问题分解为两个子问题：首先利用颜色和超像素一致性等低级线索，以类别无关的方式提出潜在的物体建议，然后使用 CNN 分类器识别这些位置上的物体类别。这种两阶段方法充分利用了利用低级线索进行边界框分割的准确性，以及最先进的 CNN 强大的分类能力。我们在提交的检测报告中采用了类似的管道，但在这两个阶段都进行了改进，例如采用多边框[5]预测来提高物体边框的召回率，以及采用集合方法对边框提议进行更好的分类。

3 动机和高层考虑

提高深度神经网络性能的最直接方法就是扩大其规模。这包括增加网络的深度（层级数量）和宽度（每个层级的单元数量）。这是训练更高质量模型的一种简单而安全的方法，尤其是在有大量标注训练数据的情况下。然而，这种简单的解决方案有两大缺点。

更大的规模通常意味着更多的参数，这使得扩大后的网络更容易出现过度拟合，尤其是在训练集中标注示例数量有限的情况下。这可能会成为一个主要瓶颈，因为创建高质量的训练集非常困难

(a) 西伯利亚哈士奇　　(b) 爱斯基摩犬

图 1：ILSVRC 2014 分类挑战赛 1000 个类别中的两个不同类别。

而且成本高昂，特别是如果需要人类专家来区分细粒度的视觉类别，如图 1 所示，ImageNet 中的类别（即使是在 1000 个类别的 ILSVRC 子集中）。

均匀增加网络规模的另一个缺点是会大幅增加计算资源的使用。例如，在深度视觉网络中，如果两个卷积层是连锁的，其滤波器数量的均匀增加会导致计算量的二次方增加。如果增加的容量使用效率不高（例如，如果大多数权重最终接近于零），那么就会浪费大量计算。由于在实践中计算预算总是有限的，因此即使主要目标是提高结果质量，有效分配计算资源也比盲目增加计算量要好。

解决这两个问题的根本方法是最终将完全连接的架构转变为稀疏连接的架构，甚至是在卷积内部。除了模仿生物系统外，由于阿罗拉等人的突破性工作[2]，这种方法还具有更坚实的理论基础。他们的主要成果指出，如果数据集的概率分布可以用一个大型、非常稀疏的深度神经网络来表示，那么就可以通过分析最后一层激活的相关性统计，并对具有高度相关输出的神经元进行聚类，从而逐层构建出最佳网络拓扑结构。虽然严格的数学证明需要非常强的条件，但这一说法与众所周知的海比原理--神经元一起发火，就会一起连线--产生了共鸣，这表明即使在不太严格的条件下，这一基本思想在实践中也是适用的。

另一方面，当今的计算基础设施在对非均匀稀疏数据结构进行数值计算时效率非常低。即使算术运算次数减少 *100* 倍，查找和高速缓存缺失的开销也非常大，因此转而使用稀疏矩阵不会带来任何收益。由于使用了不断改进、高度调整的数值库，可以利用底层 CPU 或 GPU 硬件的微小细节实现极快的密集矩阵乘法，因此差距进一步拉大[16, 9]。此外，非均匀稀疏模型需要更复杂的工程和计算基础设施。目前大多数面向视觉的机器学习系统只是通过卷积来利用空间领域的稀疏性。然而，卷积是作为与前一层斑块的密集连接集合来实现的。自[11]以来，为了打破对称性并提高学习效率，ConvNets 传统上在特征维度上使用随机稀疏连接表，而[9]为了更好地优化并行计算，趋势又回到了全连接。结构的统一性、

大量的滤波器和更大的批处理规模，使得密集计算的效率得以提高。

这就提出了一个问题，即是否有希望实现下一个中间步骤：即使是在滤波器层面，也能像理论所建议的那样，利用额外的稀疏性，但利用我们的

通过对密集矩阵的计算，稀疏矩阵的计算能力将大大提高。有关稀疏矩阵计算的大量文献（如 [3]）表明，将稀疏矩阵聚类为相对密集的子矩阵，往往能为稀疏矩阵乘法带来最先进的实用性能。在不久的将来，利用类似方法自动构建非均匀深度学习架构似乎并不牵强。

Inception 架构最初是第一作者的一项案例研究，用于评估一种复杂的网络拓扑结构构建算法的假设输出，该算法试图近似[2]中暗示的视觉网络稀疏结构，并通过密集、易读的组件覆盖假设结果。尽管这是一项高度推测性的工作，但在对拓扑结构的精确选择进行了两次迭代后，我们已经看到与基于[12]的参考架构相比有了适度的提高。在进一步调整学习率、超参数和改进训练方法后，我们发现生成的 Inception 架构作为 [6] 和 [5] 的基础网络，在定位和物体检测方面特别有用。有趣的是，尽管大多数最初的架构选择都受到了质疑和全面测试，但事实证明它们至少是局部最优的。

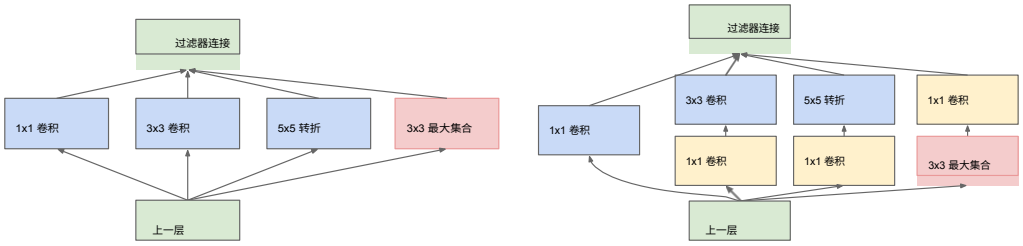
不过，我们必须谨慎行事：尽管所提出的架构在计算机视觉领域取得了成功，但其质量是否归功于构建该架构的指导原则仍值得怀疑。要确定这一点，需要进行更深入的分析和验证：例如，基于下述原则的自动化工具是否能为视觉网络找到类似但更好的拓扑结构。最有说服力的证明是，如果一个自动化系统能够使用相同的算法创建网络拓扑结构，从而在其他领域获得类似的收益，但全局架构却截然不同。至少，Inception 架构的初步成功为未来在这一方向开展令人兴奋的工作提供了坚实的动力。

4 建筑细节

Inception 架构的主要理念是，找出如何用现成的密集组件来近似和覆盖卷积视觉网络中的最佳局部稀疏结构。请注意，假设翻译不变性意味着我们的网络将由卷积积木构建而成。我们所需要做的就是找到最佳的局部构造，并在空间上进行重复。Arora 等人[2]建议采用逐层构建的方法，即分析上一层的相关性统计数据，并将其聚类为具有高相关性的单元组。这些群组构成下一层的单元，并与上一层的单元相连。我们假设前一层的每个单元都对应输入图像的某个区域，这些单元被分组为滤波器组。在下层（靠近输入的一层），相关单元将集中在局部区域。这就意味着，我们最终会发现很多集群都集中在一个区域，而下一层的 *1×1* 卷积层就可以覆盖这些集群，正如文献[12]所建议的那样。不过，我们也可以预期，在更大的斑块上，卷积可以覆盖的空间分布更广的集群数量会更少，而在越来越大的区域上，斑块的数量也会越来越少。为了避免斑块对齐问题，Inception 架构的当前版本仅限于滤波器大小为 1×*1*、3×3 和 5×*5*，但这一决定更多是基于方便性而非必要性。这也意味着所建议的架构是所有这些层的组合，其输出滤波器组串联成一个输出向量，构成下一阶段的输入。此外，由于池化操作是目前最先进的卷积网络取得成功的关键，因此在每个阶段增加一个并行池化路径也会产生额外的有益效果（见图 2(a)）。

随着这些 "入门模块 "的层层叠加，它们的输出相关性统计必然会发生变化：随着抽象程度较高的特征被更高的层捕获，它们的空间集中度预计会降低，这表明随着我们向更高的层移动，3×3 和 5×5 卷积的比例应该会增加。

上述模块的一个大问题是，至少在这种幼稚的形式下，即使是数量不多的 5×5 卷积，在具有大量滤波器的卷积层上也会过于昂贵。一旦加入池化单元，这个问题就会变得更加突出：池化单元的输出滤波器数量等于前一阶段的滤波器数量。将池化层的输出与卷积层的输出合并，将不可避免地产生以下结果



(a) 初始模块，原始版本　　(b) 维度缩减后的初始模块

图 2：启动模块

各阶段输出数量的增加。尽管这种结构可能涵盖了最佳稀疏结构，但其效率非常低，导致计算量在几个阶段内就会爆炸。

这就引出了拟议架构的第二个想法：在计算要求会增加太多的地方，明智地应用降维和投影。这是基于嵌入式的成功经验：即使是低维嵌入式也可能包含相对较大图像片段的大量信息。然而，嵌入式以密集、压缩的形式表示信息，而压缩信息更难建模。我们希望在大多数地方保持稀疏的表示（如 [2] 中的条件所要求的那样），只在需要对信号进行大规模聚合时才对信号进行压缩。也就是说，在昂贵的 3×3 和 5×5 卷积之前，先用 1×1 卷积来计算还原。除了用作还原之外，它们还包括使用整流线性激活，这使得它们具有双重用途。最终结果如图 2(b) 所示。

一般来说，Inception 网络是由上述类型的模块相互堆叠组成的网络，偶尔使用步长为 2 的最大池化层将网格分辨率减半。出于技术原因（训练时的内存效率），开始时只在较高层使用 Inception 模块，而将较低层保留为传统的卷积方式似乎是有益的。严格来说，这样做并无必要，只是反映了我们当前实现中一些基础设施的低效。

这种架构的一个主要优点是，它可以在不增加计算复杂度的情况下，大幅增加每个阶段的单元数量。无处不在的降维技术可以将上一阶段的大量输入滤波器屏蔽到下一层，首先降低它们的维度，然后再用大尺寸的补丁对它们进行卷积。这种设计的另一个实用之处在于，它符合视觉信息应在不同尺度上进行处理，然后汇总的直觉，这样下一阶段就能同时抽象出不同尺度的特征。

计算资源的使用得到改善后，可以增加每个阶段的宽度和 阶段数量，而不会陷入计算困难。利用初始架构的另一种方法是创建稍差但计算成本更低的版本。我们发现，所有包含的旋钮和杠杆都可以对计算资源进行可控的平衡，从而使网络的速度比采用非初始架构的类似网络快 2-3 *倍*，但这在目前还需要精心的手动设计。

5　GoogLeNet

在 ILSVRC14 比赛中，我们选择了 GoogLeNet 作为队名。这个名字是为了向 Yann LeCuns

首创的 LeNet 5 网络[10] 致敬。我们还使用 GoogLeNet 来指代我们在比赛中使用的 Inception 架构的特定化身。我们还使用了一个更深、更广的 Inception 网络，其质量略逊一筹，但将其添加到集合中似乎能略微改善结果。我们省略了该网络的细节，因为我们的实验表明，确切的架构参数对结果的影响相对较小。

类型	补丁大小/大步走	产量尺寸	深度	#1×1	#3×3减小	#3×3	#5×5减小	#5×5	汇集项目	参数	运筹
卷积	7×7/2	112×112×64	1							2.7K	34M
最大游泳池	3×3/2	56×56×64	0								
卷积	3×3/1	56×56×192	2		64	192				112K	360M
最大游泳池	3×3/2	28×28×192	0								
开始 (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
开始 (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
最大游泳池	3×3/2	14×14×480	0								
开始 (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
开始 (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
开始 (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
开始 (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
开始（4e）		14×14×832	2	256	160	320	32	128	128	840K	170M
最大游泳池	3×3/2	7×7×832	0								
开始 (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
开始 (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
平均泳池	7×7/1	1×1×1024	0								
辍学（40）		1×1×1024	0								
线形		1×1×1000	1							1000K	1M
软上限		1×1×1000	0								

表 1：GoogLeNet Inception 架构的化身

小。表 1 介绍了最成功的特定实例（命名为 GoogLeNet），以作示范。在我们的 7 个模型中，有 6 个使用了完全相同的拓扑结构（使用不同的采样方法训练）。

所有卷积，包括 Inception 模块内部的卷积，都使用了整流线性激活。我们网络的感受野大小为 224×224，采用平均次牵引的 RGB 颜色通道。"#3×3 reduce "和 "#*5*×*5* reduce "代表在 3×3 和 5×5 卷积之前使用的还原层中 *1*×*1* 过滤器的数量。在 "池 proj "一栏中，我们可以看到内置最大池化后的投影层中 1×1 滤镜的数量。所有这些还原/投影层也都使用了整流线性激活。

该网络的设计考虑到了计算效率和实用性，因此推理可以在个人设备上运行，甚至包括那些计算资源有限的设备，特别是内存占用较低的设备。如果只计算有参数的层，网络深度为 22 层（如果还计算池化，则为 27 层）。用于构建网络的总层数（独立构建模块）约为 100 层。不过，这个数字取决于所使用的机器学习基础系统。在分类器之前使用平均池化是基于文献[12]，不过我们的实现方式有所不同，我们使用了一个额外的线性层。这使得我们的网络可以很容易地针对其他标签集进行调整和微调，但这主要是为了方便，我们并不指望它会产生重大影响。研究发现，从全连接层到平均池化，top-1 的准确率提高了约 0.6%，但即使去掉了全连接层，使用 dropout 仍然是必不可少的。

由于网络的深度相对较大，能否以有效的方式将梯度传播回所有层是一个值得关注的问题。一个有趣的现象是，相对较浅的网络在这项任务中表现出色，这表明网络中间层产生的特征应该具有很强的辨别能力。通过添加连接到这些中间层的辅助分类器，我们希望能提

高分类器低级阶段的辨别能力，增加传播回来的梯度信号，并提供额外的正则化。这些分类器的形式是将较小的卷积网络放在起始（4a）和（4d）模块的输出之上。在训练过程中，它们的损失会以折扣权重（辅助分类器的损失权重为 0.3）加到网络的总损失中。在推理时，这些辅助网络将被丢弃。

包括辅助分类器在内的额外网络的具体结构如下：

- 平均池化层的滤波器大小为 5×5，步长为 3，因此 (4a) 阶段的输出为 4×4×512，(4d) 阶段的输出为 4×4×*528*。

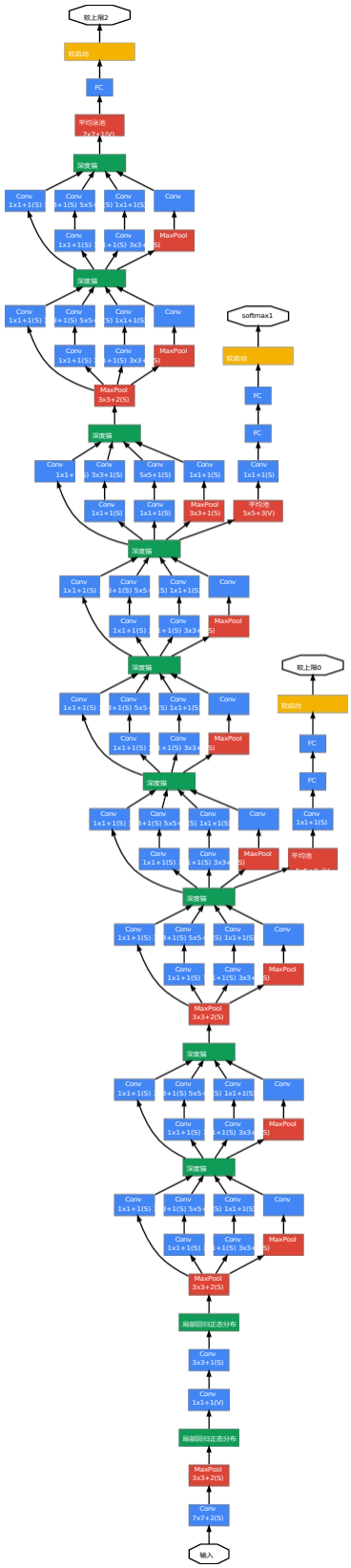


图 3：功能齐全的 GoogLeNet 网络

- 使用 128 个滤波器进行 1×1 卷积，以降低维度并进行整流线性激活。
- 全连接层有 1024 个单元，采用整流线性激活。
- 丢弃层的丢弃输出比例为 70%。
- 以软最大损失为分类器的线性层（预测与主分类器相同的 1000 个类别，但在推理时移除）。

图 3 是所生成网络的示意图。

6 培训方法

我们的网络是使用 DistBelief [4] 分布式机器学习系统训练的，使用了适量的模型和数据并行性。虽然我们只使用了基于 CPU 的实现方式，但粗略估计，GoogLeNet 网络可以在一周内使用少量高端 GPU 训练到收敛，主要限制是内存使用量。我们的训练采用异步随机梯度下降法，动量为 0.9 [17]，固定学习率计划（每 8 个历元学习率下降 4%）。在推理时使用 Polyak 平均法 [13] 创建最终模型。

在比赛前的几个月里，我们的图像采样方法发生了很大变化，而且已经收敛的模型还采用了其他方法进行训练，有时还改变了超参数，如辍学率和学习率，因此很难对训练这些网络最有效的单一方法给出明确的指导。更复杂的是，受文献[8]的启发，一些模型主要在相对较小的作物上训练，另一些则在较大的作物上训练。不过，有一种方法在比赛后被证实非常有效，其中包括对图像中不同大小的斑块进行采样，这些斑块的大小均匀分布在图像面积的 8% 到 100% 之间，长宽比在 3/4 和 *4/3* 之间随机选择。我们还发现，安德鲁-霍华德（Andrew Howard）[8] 提出的光度失真法在一定程度上有助于消除过度拟合。此外，我们开始使用随机插值法（双线性、面积、近邻和立方，概率相等）调整大小的时间相对较晚，而且是与其他超参数变化一起使用的，因此我们无法确定使用这些方法是否会对最终结果产生积极影响。

7 ILSVRC 2014 分类挑战赛的设置和结果

ILSVRC 2014 分类挑战赛的任务是将图像分类到 Imagenet 层次结构中的 1000 个叶节点类别之一。约有 120 万张图像用于训练，5 万张用于验证，10 万张用于测试。每张图像都与一个基本事实类别相关联，性能根据得分最高的分类器预测结果来衡量。通常会报告两个数字：top-1 准确率（将地面实况与第一个预测类别进行比较）和top-5 错误率（将地面实况与前 5 个预测类别进行比较）：如果地面实况位于前 5 位，则图像被视为分类正确，无论其在前 5 位中的排名如何。挑战赛使用前 5 名错误率进行排名。

我们参加挑战赛时没有使用外部数据进行训练。除了本文提到的训练技术外，我们还在测试中采用了一套技术，以获得更高的性能，下面我们将详细说明。

1. 我们独立训练了 7 个版本的相同 GoogLeNet 模型（包括一个更宽的版本），并用它们进行了集合预测。这些模型在训练时使用了相同的初始化（甚至是相同的初始权重，这主要是由于疏忽造成的）和学习率策略，它们之间的区别仅在于采样方法和看到输入图像的随机顺序。
2. 在测试过程中，我们采用了比 Krizhevsky 等人[9]更激进的裁剪方法。具体来说，我们将图像调整为 4 种比例，其中较短的尺寸（高或宽）分别为 256、288、320 和 352，然后取这些调整后图像的左、中、右三个正方形（如果是人像图像，则取上、中、下三个正方形）。然后，我们对每个正方形的 4 个角和中间的 224×224 裁剪以及

团队	年份	地点	错误（前 5 名）	使用外部数据
超级视觉	2012	第 1	16.4%	没有
超级视觉	2012	第 1	15.3%	Imagenet 22k
克拉里费	2013	第 1	11.7%	没有
克拉里费	2013	第 1	11.2%	Imagenet 22k
MSRA	2014	第 3 次	7.35%	没有
VGG	2014	第 2	7.32%	没有
GoogLeNet	2014	第 1	6.67%	没有

表 2：分类性能

型号数量	作物数量	费用	五大错误	与基数相比
1	1	1	10.07%	基础
1	10	10	9.15%	-0.92%
1	144	144	7.89%	-2.18%
7	1	7	8.09%	-1.98%
7	10	70	7.62%	-2.45%
7	144	1008	6.67%	-3.45%

表 3：GoogLeNet 分类性能细分

正方形调整为 224×224，以及它们的镜像版本。这样，每幅图像就有 $4\times3\times6\times2 = 144$ 种裁剪。安德鲁-霍华德（Andrew Howard）[8] 在前一年的参赛作品中也使用了类似的方法，我们通过经验验证，其性能略逊于所提出的方案。我们注意到，在实际应用中可能没有必要采用这种激进的裁剪方法，因为当裁剪数量达到一定程度后，更多裁剪的好处就变得微不足道了（正如我们稍后将展示的那样）。

3. 对多个作物和所有分类器的软最大概率取平均值，得出最终预测结果。在实验中，我们分析了验证数据的其他方法，如作物最大池化和分类器平均，但这些方法的性能都不如简单平均。

在本文的其余部分，我们将分析影响最终呈件整体性能的多种因素。

我们在挑战赛中提交的最终作品在验证和测试数据上的误差均为 6.67%，排名前五，在其他参赛者中名列第一。与 2012 年的 SuperVision 方法相比，相对误差减少了 56.5%，与前一年的最佳方法（Clarifai）相比，相对误差减少了约 40%，这两种方法都使用了外部数据来训练分类器。下表显示了一些表现最佳方法的统计数据。

我们还通过改变预测图像时使用的模型数量和作物数量，分析并报告了多种测试选择的性能，如下表所示。当我们使用一个模型时，我们选择在验证数据上误差率最低的模型。为了不过度拟合测试数据统计，所有数据都是在验证数据集上报告的。

8 ILSVRC 2014 检测挑战赛的设置和结果

ILSVRC 的检测任务是在图像中的 200 个可能类别中生成物体周围的边界框。如果检测到的物体与地面实况的类别相匹配，并且它们的边界框至少重叠 50%（使用 Jaccard 指数），则视为正确检测。不相干的检测结果将被视为假阳性并受到惩罚。与分类任务相反，每幅图像可能包含

团队	年份	地点	mAP	外部数据	建筑群	办法
UvA-Euvision	2013	第 1	22.6%	无	?	费舍尔向量
深入洞察	2014	第 3 次	40.5%	ImageNet 1k	3	美国有线电视新闻网
中大 DeepID-Net	2014	第 2	40.7%	ImageNet 1k	?	美国有线电视新闻网
GoogLeNet	2014	第 1	43.9%	ImageNet 1k	6	美国有线电视新闻网

表 4：探测性能

团队	mAP	情境模型	边界框回归
特里普斯-苏申	31.6%	没有	?
伯克利愿景	34.5%	没有	是
UvA-Euvision	35.4%	?	?
中大 DeepID-Net2	37.7%	没有	?
GoogLeNet	38.02%	没有	没有
深度洞察	40.2%	是	是

表 5：单一模型的检测性能

它们的规模可能很大，也可能很小。结果采用平均精度 (mAP) 进行报告。

GoogLeNet 采用的检测方法与 [6] 的 R-CNN 相似，但增加了 Inception 模型作为区域分类器。此外，通过将选择性搜索[20]方法与多方框[5]预测相结合，改进了区域建议步骤，以获得更高的对象边界方框召回率。为了减少误报的数量，超像素的大小增加了 2 *倍*。这使得来自选择性搜索算法的建议减半。我们又增加了 200 个来自多方框[5]的区域建议，结果是[6]所用建议的 60%，覆盖率从 92% 提高到 93%。在增加覆盖率的同时减少建议数量的总体效果是，单一模型的平均精度提高了 1%。最后，我们在对每个区域进行分类时使用了 6 个 ConvNets 的集合，结果准确率从 40% 提高到 43.9%。请注意，与 R-CNN 相反，由于时间不够，我们没有使用边界框回归。

我们首先报告了最高检测结果，并展示了自第一届检测任务以来取得的进展。与 2013 年的结果相比，准确率几乎翻了一番。成绩最好的团队都使用了卷积网络。我们在表 4 中报告了官方得分以及各团队的共同策略：使用外部数据、集合模型或上下文模型。外部数据通常是 ILSVRC12 分类数据，用于预训练模型，然后在检测数据上对模型进行改进。一些团队还提到了本地化数据的使用。由于大部分定位任务的边界框没有包含在检测数据集中，

因此可以使用这些数据预训练通用边界框回归器，就像使用分类数据进行预训练一样。GoogLeNet 条目没有使用定位数据进行预训练。

在表 5 中，我们比较了仅使用单一模型的结果。表现最出色的模型是 Deep Insight，但令人惊讶的是，它在 3 个模型的集合中仅提高了 0.3 分，而 GoogLeNet 在集合中的表现则明显更强。

9 结论

我们的研究结果似乎提供了一个确凿的证据，即用现成的密集构件来逼近预期的最佳稀疏结构，是改进计算机视觉神经网络的一种可行方法。这种方法的主要优势在于，与较浅和较宽的网络相比，只需适度降低计算要求，就能显著提高质量。还需注意的是，尽管我们既没有利用上下文，也没有执行边界框，但我们的检测工作仍具有竞争力。

这一事实进一步证明了 Inception 架构的优势。尽管预计类似深度和宽度的昂贵网络也能达到类似的结果质量，但我们的方法还是提供了确凿的证据，证明转向更稀疏的架构总体上是可行且有用的想法。这表明，在 [2] 的基础上，未来以自动化方式创建更稀疏、更精细的结构的工作大有可为。

10 致谢

我们要感谢 Sanjeev Arora 和 Aditya Bhaskara 就 [2] 进行的富有成效的讨论。我们还要感谢 DistBelief [4] 团队的支持，特别是 Rajat Monga、Jon Shlens、Alex Krizhevsky、Jeff Dean、Ilya Sutskever 和 Andrea Frome。我们还要感谢汤姆-杜里格（Tom Duerig）和叶宁（Ning Ye）在测光失真方面提供的帮助。此外，如果没有查克-罗森伯格和哈特维格-亚当的支持，我们的工作也不可能完成。

参考资料

[1] 了解你的备忘录：http://knowyourmeme.com/memes/ we-need-to-go-deeper。访问日期：2014-09-15。

[2] Sanjeev Arora, Aditya Bhaskara, Rong Ge, and Tengyu Ma.学习某些深度表征的可证明边界。*CoRR*，abs/1310.6343，2013。

[3] U¨ mit V. C ¸ atalyu¨rek, Cevdet Aykanat, and Bora Uc¸ar.关于二维稀疏矩阵分割：模型、方法和秘诀.*SIAM J. Sci. Comput.*, 32(2):656-683, February 2010.

[4] Jeffrey Dean、Greg Corrado、Rajat Monga、Kai Chen、Matthieu Devin、Mark Mao、Marc'aurelio Ranzato、Andrew Senior、Paul Tucker、Ke Yang、Quoc V. Le 和 Andrew Y. Ng。大规模分布式深度网络。见 P. Bartlett、F.c.n. Pereira、C.j.c. Burges、L. Bottou 和 K.q.Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1232-1240.2012.

[5] Dumitru Erhan、Christian Szegedy、Alexander Toshev 和 Dragomir Anguelov。使用深度神经网络的可扩展目标检测。*计算机视觉与模式识别，2014。CVPR 2014.IEEE Conference on*, 2014.

[6] Ross B. Girshick、Jeff Donahue、Trevor Darrell 和 Jitendra Malik。用于精确物体检测和语义分割的丰富特征层次。*计算机视觉与模式识别，2014。CVPR 2014.IEEE Conference on*, 2014.

[7] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhut- dinov.通过防止特征检测器的共同适应来改进神经网络*CoRR*，abs/1207.0580，2012。

[8] 安德鲁-G-霍华德基于深度卷积神经网络的图像分类的一些改进。*CoRR*，abs/1312.5402，2013.

[9] Alex Krizhevsky、Ilya Sutskever 和 Geoff Hinton。使用深度控制神经网络进行图像分类。*神经信息处理系统进展 25》*，第 1106-1114 页，2012 年。

[10] Y.LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel.反向传播应用于手写邮政编码识别。*神经计算》*，1（4）：541-551， 1989 年 12 月。

[11] Yann LeCun、Le´on Bottou、Yoshua Bengio 和 Patrick Haffner。基于梯度的学习应用于文档识别。*电气和电子工程师学会论文集》*，86（11）：2278-2324，1998 年。

[12] Min Lin, Qiang Chen, and Shuicheng Yan.网络中的网络*CoRR*，abs/1312.4400，2013。

[13] B.T. Polyak 和 A. B. Juditsky.用平均法加速随机逼近*SIAM J.控制优化》*，30（4）：838-855，1992 年 7 月。

[14] Pierre Sermanet、David Eigen、Xiang Zhang、Michae¨l Mathieu、Rob Fergus 和 Yann Le- Cun。Overfeat：使用卷积网络进行综合识别、定位和检测。*CoRR*，abs/1312.6229，2013。

[15] Thomas Serre、 Lior Wolf、 Stanley M. Bileschi、 Maximilian Riesenhuber 和 Tomaso Poggio。类皮质机制的鲁棒性物体识别。*IEEE Trans.Pattern Anal.机器。Intell.*，29（3）：411-426，2007.

[16] 宋丰光和杰克-东格拉在具有 1000 个 CPU 内核的共享内存多核系统上扩展矩阵计算。In *Proceedings of the 28th ACM International Con- ference on Supercomputing*, ICS '14, pages 333-342, New York, NY, USA, 2014.ACM.

[17] Ilya Sutskever、James Martens、George E. Dahl 和 Geoffrey E. Hinton。深度学习中初始化和动力的重要性。*第 30 届国际机器学习大会论文集，ICML 2013，美国佐治亚州亚特兰大，2013 年 6 月 16-21 日*，第 28 卷
of *JMLR Proceedings*, pages 1139-1147.JMLR.org, 2013.

[18] Christian Szegedy、Alexander Toshev 和 Dumitru Erhan。用于物体检测的深度神经网络克里斯托弗-伯格斯（Christopher J. C. Burges）、莱昂-博图（Le´on Bottou）、祖宾-加拉马尼（Zoubin Ghahramani）和基利安-温伯格（Kilian Q. Weinberger）编著的《*神经信息处理系统进展 26：2013 年第 27 届神经信息处理系统年会》（Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013*）中。*会议于 2013 年 12 月 5-8 日在美国内华达州太浩湖举行，会议论文集，*第 2553-2561 页，2013 年。

[19] Alexander Toshev 和 Christian Szegedy.Deeppose：通过深度神经网络进行人体姿态估计。*CoRR*，abs/1312.4659，2013。

[20] Koen E. A. van de Sande, Jasper R. R. Uijlings, Theo Gevers, and Arnold W. M. Smeulders. 作为对象识别选择性搜索的分割。In *Proceedings of the 2011 Interna- tional Conference on Computer Vision*, ICCV '11, pages 1879-1886, Washington, DC, USA, 2011.IEEE 计算机协会。

[21] Matthew D. Zeiler 和 Rob Fergus.可视化和理解卷积网络。戴维-J-弗利特（David J. Fleet）、托马斯-帕吉德拉（Toma´s Pajdla）、伯恩特-席勒（Bernt Schiele）和蒂内-图伊特拉斯（Tinne Tuytelaars）编著的《*计算机视觉
- ECCV 2014 - 第 13 届欧洲会议，瑞士苏黎世，2014 年 9 月 6-12 日，论文集，第一部分，《计算机科学讲座笔记》第*8689 卷，第 818-833 页。Springer, 2014.