**深入研究卷积**

**Christian Szegedy谷歌Inc。**

**Pierre Sermanet谷歌公司**

**刘魏**

北卡罗来纳大学教堂山分校

**斯科特·里德密歇根大学**

**安圭洛夫谷歌公司**

**杨庆佳谷歌有限公司**

**Dumitru Erhan谷歌公司**

arXiv: 1409.4842 v1 [cs。2014年9月17日

**Vincent Vanhoucke谷歌公司**

**Andrew Rabinovich谷歌公司**

**摘要**

我们提出了一种代号为Inception的深度卷积神经网络架构，该架构负责在ImageNet大规模视觉识别挑战2014 (ILSVRC14)中设置分类和检测的新技术状态。这种体系结构的主要特点是提高了网络内计算资源的利用率。这是通过一个精心制作的设计实现的，允许增加网络的深度和宽度，同时保持计算预算不变。为了优化质量，架构决策是基于Hebbian原则和多尺度处理的直觉。我们为ILSVRC14提交的文件中使用的一个特定的化身叫做googleet，这是一个22层深度的网络，其质量在分类和检测的背景下进行评估。

**1介绍**

在过去的三年里，主要由于深度学习，更具体的是卷积网络[10]的进步，图像识别和目标检测的质量有了长足的进步。一个令人鼓舞的消息是，大部分的进展不仅是更强大的硬件、更大的数据集和更大的模型的结果，而且主要是新思想、算法和改进的网络架构的结果。例如，在ILSVRC 2014竞赛中，除了为了检测目的而使用同一竞赛的分类数据集之外，没有使用新的数据源。我们在谷歌网提交给ILSVRC 2014的报告实际上比两年前Krizhevsky等人的获奖架构[9]使用的参数少12倍，同时明显更准确。目标检测方面最大的进步并不是单独使用深度网络或更大的模型，而是深度架构和经典计算机视觉的协同，如Girshick等人[6]的R-CNN算法。

另一个值得注意的因素是，随着移动和嵌入式计算的持续发展，我们算法的效率——特别是它们的功率和内存使用——变得非常重要。值得注意的是，本文提出的深层架构设计的考虑因素包括了这个因素，而不是纯粹地固定在精度数字上。对于大多数的实验,模型计算15亿年的预算是为了保持multiply-adds在推理时,所以他们不最终是一个纯学术的好奇心,但可以把现实世界中使用,即使在大型数据集,以合理的成本。

1

在这篇论文中，我们将专注于一个高效的计算机视觉深度神经网络架构，代号为Inception，它的名字来源于Lin等人在网络论文[12]中的网络，以及著名的“我们需要深入”网络迷因[1]。在我们的案例中，“深度”一词有两种不同的含义:首先，我们以“初始模块”的形式引入了一个新的组织层次，另一种更直接的含义是增加了网络深度。一般来说，我们可以将Inception模型视为[12]的逻辑顶点，同时从Arora等人的理论工作中获得灵感和指导。该体系结构的优点在ILSVRC 2014分类和检测挑战上得到了实验验证，在这方面它的性能显著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)通常有一个标准的结构——堆叠的卷积层(可选地紧接着是对比度归一化和最大池化)后面是一个或多个完全连接的层。这种基本设计的变体在图像分类文献中普遍存在，并在MNIST、CIFAR和最显著的ImageNet分类挑战上取得了最好的结果[9,21]。对于较大的数据集，如Imagenet，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。

尽管人们担心最大池化层会导致准确的空间信息丢失，但与[9]相同的卷积网络结构也被成功地用于定位[9,14]、目标检测[6,14,18,5]和人体姿态估计[19]。受灵长类视觉皮层神经科学模型的启发，Serre等人[15]使用一系列不同大小的固定Gabor滤波器来处理多个尺度，类似于Inception模型。然而，与[15]固定的2层深层模型相反，Inception模型中的所有过滤器都是学习的。此外，先启层重复了许多次，导致了在googleet模型的情况下一个22层的深层模型。

网络中的网络(Network-in-Network)是Lin等人为了提高神经网络的表示能力而提出的一种方法。当应用到卷积层时，该方法可以看作是额外的1×1卷积层，通常是经过矫正的线性激活[9]。这使得它可以很容易地集成到当前的CNN管道中。我们在架构中大量使用这种方法。然而，在我们的设置中，1 × 1卷积有双重目的:最关键的是，它们主要用于降维模块，以消除计算瓶颈，否则将限制我们的网络规模。这不仅允许增加深度，还允许增加网络的宽度，而不会造成显著的性能损失。

目前主要的目标检测方法是Girshick等人[6]提出的卷积神经网络(R-CNN)区域。R-CNN将整个检测问题分解为两个子问题:首先以类别不可知的方式利用低级线索(如颜色和超像素一致性)来识别潜在的目标建议，然后使用CNN分类器在这些位置识别目标类别。这种两阶段的方法利用了低级别线索的包围盒分割的准确性，以及最先进的cnn的强大的分类能力。我们在我们的检测提交中采用了类似的管道，但在这两个阶段都探索了增强，比如多盒[5]预测用于更高的对象包围盒召回率，以及集成方法用于更好地分类包围盒建议。

**动机和高层次的考虑**

提高深度神经网络性能最直接的方法是增加它们的规模。这包括增加网络的深度(层数)和它的宽度(每层的单位数)。这是训练高质量模型的一种简单而安全的方法，特别是考虑到大量标记训练数据的可用性。然而，这个简单的解决方案有两个主要缺点。

尺寸越大，通常意味着参数的数量越多，这使得扩大后的网络更容易发生过拟合，特别是在训练集中标记的样本数量有限的情况下。这可能成为主要的瓶颈，因为创建高质量的训练集可能很棘手

2



(一)西伯利亚雪橇犬

(b)爱斯基摩犬

图1:ILSVRC 2014分类挑战的1000个分类中有两个不同的分类。

而且很昂贵，特别是如果需要专家人工评分器来区分细粒度的视觉类别，如图1所示的ImageNet中的那些(甚至是1000个ILSVRC子集)。

统一增加网络规模的另一个缺点是大大增加了计算资源的使用。例如，在深度视觉网络中，如果两个卷积层被链接起来，任何滤波器数量的均匀增加都会导致计算量的二次增加。如果添加的容量使用效率低下(例如，如果大多数权重最终接近于零)，那么就会浪费大量的计算。由于在实践中计算预算总是有限的，计算资源的有效分配比不加选择地增加规模更可取，即使主要目标是提高结果的质量。

解决这两个问题的根本方法是最终从完全连接的架构转移到稀疏连接的架构，甚至在convolutions内部。除了模拟生物系统，由于Arora等人的开创性工作，这也将具有更坚实的理论基础的优势。其主要结果状态的概率分布,如果先于可表示的是一个大型的、非常稀疏的深层神经网络,优化网络拓扑结构可以建造一层一层地分析相关统计数据的最后一层和集群神经元的激活与高度相关的输出。尽管严格的数学证明需要非常强的条件，但这一论断与著名的赫比原理(神经元一起放电，连在一起)产生共鸣的事实表明，这一基本思想即使在不那么严格的条件下也适用于实践。

缺点是，当前的计算基础设施在非均匀稀疏数据结构上进行数值计算时效率非常低。即使算术运算的数量减少了100×,查找的开销和缓存错过主导,切换到稀疏矩阵不会偿还。使用的差距进一步扩大稳步改善,高度调整,数值库允许极快稠密矩阵乘法,利用底层的细节CPU或GPU硬件(16日9)。而且，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通过使用卷积来利用空间域的稀疏性。然而，卷积被实现为与前一层补丁的密集连接的集合。从[11]开始，ConvNets传统上在特征维度上使用随机稀疏连接表，以打破对称性和提高学习能力，为了更好地优化并行计算，趋势又回到了与[9]的全连接。结构的均匀性、大量的过滤器和更大的批处理尺寸允许利用高效的密集计算。

这就提出了一个问题，是否有希望实现下一个中间步骤:一个利用额外稀疏性的体系结构，即使是在过滤级，就像理论建议的那样，但利用了我们的

3.

目前的硬件利用密集矩阵的计算。大量关于稀疏矩阵计算的文献(如[3])表明，将稀疏矩阵聚类为相对密集的子矩阵往往会为稀疏矩阵乘法提供最先进的实际性能。在不久的将来，类似的方法将被用于非均匀深度学习架构的自动化构建，这种想法似乎并不牵强。

Inception体系结构最初是第一作者的一个案例研究，用来评估复杂网络拓扑构造算法的假设输出，该算法试图近似于视觉网络中[2]所暗示的稀疏结构，并用密集的、现成的组件覆盖假设的结果。尽管这是一项高度投机性的工作，但在精确选择拓扑的两次迭代之后，我们已经可以看到相对于基于[12]的参考体系结构的适度收益。在进一步调整学习率、超参数和改进的训练方法后，我们确定了由此产生的Inception体系结构作为[6]和[5]的基础网络在定位和目标检测方面特别有用。有趣的是，虽然大多数最初的体系结构选择都经过了彻底的质疑和测试，但它们至少在局部是最优的。

但我们必须谨慎:尽管所提议的架构在计算机视觉方面取得了成功，但它的质量是否可以归因于引导其构建的指导原则仍存在疑问。确保这一点需要更彻底的分析和验证:例如，如果基于下面描述的原则的自动化工具可以为远景网络找到类似但更好的拓扑。最令人信服的证据是，一个自动化系统是否会创建网络拓扑，从而在使用相同算法但具有非常不同的全局架构的其他领域中获得类似的收益。至少，初始架构的最初成功为这个方向的激动人心的未来工作提供了坚定的动力。

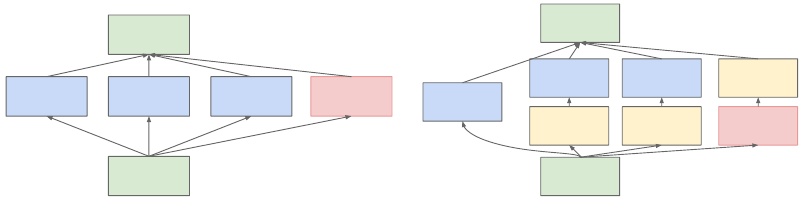
**4建筑上的细节**

Inception架构的主要思想是基于找出卷积视觉网络中的最优局部稀疏结构是如何被现成的密集组件逼近和覆盖的。注意，假设平移不变性意味着我们的网络将由卷积构建块构建。我们所需要的就是找到最优的局部构造，并在空间上重复它。Arora等人[2]提出了一种逐层构建的方法，即对最后一层的相关统计数据进行分析，并将它们聚成具有高相关性的单元组。这些簇形成了下一层的单元，并与上一层的单元相连。我们假设来自前面一层的每个单元对应于输入图像的某个区域，这些单元被分组成滤波器组。在较低的层(接近输入的层)，相关单元会集中在局部区域。这意味着，我们最终会有很多簇集中在一个区域，它们可以被下一层的1×1 convolutions覆盖，就像[12]中建议的那样。然而，我们也可以预期，在更大的斑块上，可以被卷积覆盖的集群数量会更少，在越来越大的区域上，斑块的数量会越来越少。为了避免补丁对齐问题，当前的Inception体系结构被限制在过滤器大小1×1, 3×3和5×5，然而这个决定更多的是基于方便性而不是必要性。这也意味着建议的架构是所有这些层的组合，它们的输出滤波器组连接成一个单独的输出向量形成下一阶段的输入。此外，由于池化操作对当前最先进的卷积网络的成功至关重要，因此建议在每个这样的阶段添加一个替代的并行池化路径也应该有额外的有益效果(见图2(a))。

这些《盗梦空间》模块堆叠在彼此之上,其产出相关统计数据绑定到不同:更高的抽象被更高层次的功能,其空间浓度的比例预计将减少表明3×3和5×5旋转应该增加我们移动到更高的层次。

以上模块的一个大问题是(至少在这种na¨ıve形式中是这样)，即使是少量的5×5卷积，在包含大量过滤器的卷积层之上也会变得非常昂贵。一旦将合用单元添加到混合中，这个问题就变得更加明显:它们的输出过滤器数量等于前一阶段的过滤器数量。将池化层的输出与卷积层的输出合并将导致不可避免的结果

4



过滤器连接

过滤器连接

3 x3的卷积

5 x5的卷积

1 x1的卷积

1 x1的卷积

3 x3的卷积

5 x5的卷积

3 x3马克斯池

1 x1的卷积

1 x1的卷积

1 x1的卷积

3 x3马克斯池

上一层

上一层

(a) Inception模块，没有ıve版本

(b)缩小尺寸的Inception模块

图2:Inception模块

从一个阶段到另一个阶段增加输出的数量。即使这个架构可能覆盖最优的稀疏结构，它也会非常低效，导致在几个阶段内的计算崩溃。

这就引出了所提议的体系结构的第二个想法:在计算需求可能过多增加的地方，明智地应用维缩减和投影。这是基于嵌入的成功:即使是低维的嵌入也可能包含一个相对较大的图像补丁的大量信息。然而，嵌入以密集、压缩的形式表示信息，压缩后的信息更难建模。我们希望在大多数地方保持我们的表示稀疏(如[2]条件所要求的那样)，并且只在需要全部聚合的时候才压缩信号。也就是说，在昂贵的3×3和5×5卷积之前，使用1×1卷积来计算缩减。除了用作还原外，它们还包括使用矫正线性激活，这使它们具有双重用途。最终的结果如图2(b)所示。

一般来说，初始网络是由上述类型的模块相互堆叠而成的网络，偶尔使用stride 2将网格分辨率减半的最大池层。出于技术原因(培训期间的内存效率)，只在较高的层开始使用Inception模块，而以传统的卷积方式保持较低的层似乎是有益的。这并不是必须的，只是反映了当前实现中一些基础设施的低效。

这种架构的一个主要好处是，它允许在每个阶段显著增加单元数量，而不会导致计算复杂度失控。降维的普遍使用允许将最后阶段的大量输入滤波器屏蔽到下一层，首先降低它们的维数，然后再用一个大的patch尺寸对它们进行卷积。这种设计的另一个实用方面是，它与直觉一致，即视觉信息应该在不同的尺度上进行处理，然后进行聚合，以便下一阶段能够同时从不同的尺度上提取特征。

改进计算资源的使用，可以增加每个压裂段的宽度以及压裂段的数量，而不会造成计算困难。另一种利用先启体系结构的方法是创建稍差但计算上便宜的版本。我们已经发现，所有包含的把手和杠杆都允许控制计算资源的平衡，从而可以产生比非初始架构的类似网络快2 - 3倍的网络，但这需要在这一点上仔细的手动设计。

**5 GoogLeNet**

在ILSVRC14比赛中，我们选择GoogLeNet作为我们的团队名称。这个名字是向Yann LeCuns先驱LeNet 5网络[10]致敬。我们还使用GoogLeNet来指代我们提交竞赛时使用的初始架构的特定化身。我们也使用了更深入更广泛的《盗梦空间》，其质量稍差，但将其添加到整体中似乎对效果略有改善。我们忽略了该网络的细节，因为我们的实验已经表明，精确的架构参数的影响是相对的

5

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **类型** | **块大小/大步** | **输出的大小** | **深度** | | # 1×1 | # 3×3减少 | # 3×3 | # 5×5减少 | # 5×5 | **池项目** | **参数个数** | **运维** |
| 卷积 | 7×7/2 | 112×112×64 |  | 1 |  |  |  |  |  |  | 2.7 k | 34米 |
| 马克斯池 | 3×3/2 | 56 56××64 |  | 0 |  |  |  |  |  |  |  |  |
| 卷积 | 3×3/1 | 56 56××192 |  | 2 |  | 64 | 192 |  |  |  | 112 k | 360米 |
| 马克斯池 | 3×3/2 | 28××28日192年 |  | 0 |  |  |  |  |  |  |  |  |
| 《盗梦空间》(3) |  | 28××28日256年 |  | 2 | 64 | 96 | 128 | 16 | 32 | 32 | 159 k | 128米 |
| 《盗梦空间》(3 b) |  | 28××28日480年 |  | 2 | 128 | 128 | 192 | 32 | 96 | 64 | 380 k | 304米 |
| 马克斯池 | 3×3/2 | 14×14×480 |  | 0 |  |  |  |  |  |  |  |  |
| 《盗梦空间》(4) |  | 14×14×512 |  | 2 | 192 | 96 | 208 | 16 | 48 | 64 | 364 k | 73米 |
| 《盗梦空间》(4 b) |  | 14×14×512 |  | 2 | 160 | 112 | 224 | 24 | 64 | 64 | 437 k | 88米 |
| 《盗梦空间》(4 c) |  | 14×14×512 |  | 2 | 128 | 128 | 256 | 24 | 64 | 64 | 463 k | 100米 |
| 《盗梦空间》(4 d) |  | 14×14×528 |  | 2 | 112 | 144 | 288 | 32 | 64 | 64 | 580 k | 119米 |
| 《盗梦空间》(4 e) |  | 14×14×832 |  | 2 | 256 | 160 | 320 | 32 | 128 | 128 | 840 k | 170米 |
| 马克斯池 | 3×3/2 | 7×7×832 |  | 0 |  |  |  |  |  |  |  |  |
| 《盗梦空间》(5) |  | 7×7×832 |  | 2 | 256 | 160 | 320 | 32 | 128 | 128 | 1072 k | 54米 |
| 《盗梦空间》(5 b) |  | 7×7×1024 |  | 2 | 384 | 192 | 384 | 48 | 128 | 128 | 1388 k | 71米 |
| avg池 | 7×7/1 | 1×1×1024 |  | 0 |  |  |  |  |  |  |  |  |
| 辍学(40%) |  | 1×1×1024 |  | 0 |  |  |  |  |  |  |  |  |
| 线性 |  | 1×1×1000 |  | 1 |  |  |  |  |  |  | 1000 k | 1米 |
| softmax |  | 1×1×1000 |  | 0 |  |  |  |  |  |  |  |  |

表1:初始架构的googleet化身

轻微的。为了便于演示，表1描述了最成功的特定实例(名为googleet)。7个模型中的6个使用了完全相同的拓扑结构(用不同的采样方法训练)。

所有的卷积，包括初始模块中的卷积，都使用矫正线性激活。采用均值相减的RGB颜色信道，网络中接收野的大小为224×224。“#3×3 reduce”和“#5×5 reduce”表示在3×3和5×5卷积之前使用的约简层中1×1过滤器的数量。在pool projj列中内置最大池之后，可以看到投影层中1×1过滤器的数量。所有这些减少/投影层使用矫正线性激活以及。

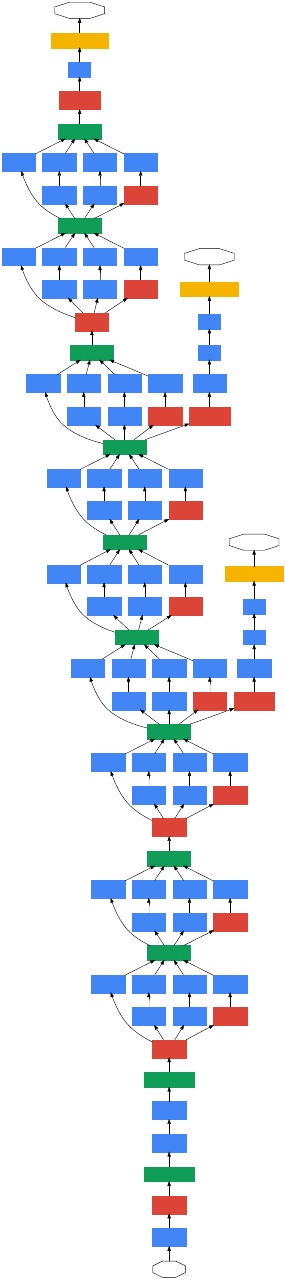
该网络的设计考虑了计算效率和实用性，因此推理可以在单个设备上运行，甚至包括那些计算资源有限的设备，特别是内存占用较少的设备。当只计算有参数的层时，网络是22层(或者27层，如果我们也计算池)。用于网络建设的层(独立的构件)的总数量约为100层。然而，这个数字取决于所使用的机器学习基础设施系统。在分类器之前使用平均池是基于[12]的，尽管我们的实现不同，我们使用了额外的线性层。这使得我们可以很容易地为其他标签集调整和微调我们的网络，但这主要是为了方便，我们不期望它有很大的影响。研究发现，从完全连接的层迁移到平均池化可以使top-1准确率提高约0.6%，然而，即使在删除完全连接的层之后，dropout的使用仍然是必不可少的。

由于网络的深度相对较大，以一种有效的方式将梯度传播回所有层的能力值得关注。一个有趣的发现是，相对较浅的网络在这个任务上的强大性能表明，网络中间的层产生的特征应该是非常有区别的。通过添加连接到这些中间层的辅助分类器，我们期望在分类器的较低阶段鼓励辨别，增加被传播回来的梯度信号，并提供额外的正则化。这些分类器采用在初始(4a)和(4d)模块输出之上的更小的卷积网络的形式。在训练过程中，它们的损失被加上一个折扣权重(辅助分类器的损失被加权为0.3)到网络的总损失中。在推理时，这些辅助网络被丢弃。

包括辅助分类器在内的侧面额外网络的确切结构如下:

•一个平均池化层5×5过滤器大小和步幅3，为(4a)产生4×4×512输出，为(4d)阶段4×4×528。

6



softmax2

SoftmaxActivation

足球俱乐部

AveragePool x7 7 + 1 (V)

DepthConcat

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

DepthConcat

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

softmax1

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

SoftmaxActivation

MaxPool 3 x3 + 2 (S)

足球俱乐部

DepthConcat

足球俱乐部

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

AveragePool 5 x5 + 3 (V)

DepthConcat

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

DepthConcat

softmax0

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

SoftmaxActivation

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

足球俱乐部

DepthConcat

足球俱乐部

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

AveragePool 5 x5 + 3 (V)

DepthConcat

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

MaxPool 3 x3 + 2 (S)

DepthConcat

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

DepthConcat

Conv 1 x1 + 1 (S)

Conv 3 x3 + 1 (S)

Conv 5 x5 + 1(年代)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

Conv 1 x1 + 1 (S)

MaxPool 3 x3 + 1 (S)

MaxPool 3 x3 + 2 (S)

LocalRespNorm

Conv 3 x3 + 1 (S)

Conv 1 x1 + 1 (V)

LocalRespNorm

MaxPool 3 x3 + 2 (S)

Conv 7 x7 + 2 (S)

输入

图3:带有各种花哨功能的google网络

7

•1×1卷积与128个滤波器降维和矫正线性激活。

•一个完全连接的层，1024个单位和矫正线性激活。

•下降层与下降输出70%的比率。

•一个以softmax损耗作为分类器的线性层(预测与主分类器相同的1000个类，但在推断时删除)。

结果网络的示意图如图3所示。

**6培训方法**

我们的网络训练使用DistBelief[4]分布式机器学习系统使用适度的模型和数据并行。尽管我们只使用基于CPU的实现，但粗略估计google网络可以在一周内使用少量高端gpu训练到收敛，主要的限制是内存使用。我们的训练采用0.9动量[17]的异步随机梯度下降，固定学习率调度(每8个epoch减少4%的学习率)。Polyak平均[13]用于创建推断时使用的最终模型。

图像采样方法大大改变了过去几个月来,导致竞争,并且已经聚合模型训练与其他选项,有时结合改变hyperparameters,辍学和学习速率,所以很难给出一个明确的指导,最有效的单一的方式来训练这些网络。更复杂的是，受[8]启发，一些模型主要针对较小的同类作物，而另一些模型针对较大的同类作物。然而，一种在比赛后被证实非常有效的方法是对图像的不同大小的小块进行采样，这些小块的大小均匀地分布在图像面积的8%到100%之间，宽高比在3/4到4/3之间随机选择。此外，我们发现，由Andrew Howard[8]的光度失真在一定程度上有助于对抗过拟合。此外，我们开始使用随机插值方法(双线性、面积、最近邻和等概率立方)较晚调整大小，并结合其他超参数的变化，所以我们不能确定最终结果是否受到它们的积极影响

使用。

**7 ILSVRC 2014分类挑战设置和结果**

ILSVRC 2014分类挑战涉及到将图像分类到Imagenet层次结构中1000个叶节点类别之一的任务。大约有120万张图像用于训练，5万张用于验证，10万张用于测试。每幅图像都与一个ground truth类别相关联，并且性能是基于得分最高的分类器预测来衡量的。两个数字通常报道:排名前准确率,比较了地面真理反对第一个预测类,前5名的错误率,比较了地面真理对前5预测类:一个图像被认为是正确的分类,如果地面真理是前5名中,无论其排名。该挑战使用前5名错误率进行排名。

我们在没有外部数据用于训练的情况下参加了挑战。除了本文中提到的训练技术外，我们在测试过程中采用了一组技术以获得更高的性能，我们将在下面进行阐述。

1.我们独立训练了7个版本的googleet模型(包括一个更宽的版本)，并用它们进行集成预测。这些模型使用相同的初始化(甚至相同的初始权值，主要是因为疏忽)和学习速率策略进行训练，它们只是在采样方法和看到输入图像的随机顺序上有所不同。

2.在试验中，我们采取了比Krizhevsky等人[9]更积极的种植方法。具体来说，我们将图像调整为4个比例，其中更短的尺寸(高度或宽度)分别为256、288,320和352，取这些调整大小的图像的左、中和右正方形(在肖像图像的情况下，我们取顶部、中心和底部正方形)。对于每个正方形，我们然后采取4个角落和中心224×224作物以及

8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **团队** | **一年** | **的地方** | **错误(前5名)** | **使用外部数据** |
| 监督 | 2012 | 1日 | 16.4% | 没有 |
| 监督 | 2012 | 1日 | 15.3% | Imagenet 22 k |
| Clarifai | 2013 | 1日 | 11.7% | 没有 |
| Clarifai | 2013 | 1日 | 11.2% | Imagenet 22 k |
| 同行 | 2014 | 3日 | 7.35% | 没有 |
| VGG | 2014 | 2日 | 7.32% | 没有 |
| GoogLeNet | 2014 | 1日 | 6.67% | 没有 |

表2:分类性能

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **数量的模型** | **许多农作物** | **成本** | **五大错误** | **相对于基础** |
| 1 | 1 | 1 | 10.07% | 基地 |
| 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分类性能分解

Square调整大小为224×224，以及它们的镜像版本。结果是4×3×6×2 =每张图像144个作物。Andrew Howard[8]在前一年的条目中使用了类似的方法，我们通过经验验证，它的表现比提议的方案略差。我们注意到，在实际应用中，这种激进的种植可能是不必要的，因为在合理数量的作物出现后，更多作物的效益就会变得微不足道(我们将在后面展示)。

3.在多个作物和所有单独的分类器上平均softmax概率，以获得最终的预测。在我们的实验中，我们分析了验证数据的替代方法，比如对作物进行最大池化和对分类器进行平均，但它们导致的性能低于简单平均。

在本文的其余部分，我们分析了影响最终提交作品整体表现的多种因素。

我们最终提交的挑战在验证和测试数据上获得了6.67%的前5名错误，在其他参与者中排名第一。与2012年的监督方法相比，这是56.5%的相对减少，与前一年的最佳方法(Clarifai)相比，大约40%的相对减少，这两种方法都使用外部数据来培训分类器。下表显示了一些性能最好的方法的统计数据。

我们还分析和报告多个测试选择的性能，方法是改变预测下表中图像时使用的模型数量和作物数量。当我们使用一个模型时，我们选择验证数据上前1个错误率最低的模型。为了不过拟合测试数据统计，所有的数字都报告在验证数据集上。

**8 ILSVRC 2014检测挑战设置与结果**

ILSVRC检测任务是在200个可能的类中，在图像中的对象周围产生边界框。如果检测到的对象与groundtruth的类匹配，并且它们的边界盒重叠至少50%(使用Jaccard索引)，则视为正确。不相关的检测算作假阳性，并予以处罚。与分类任务相反，每个图像可能包含

9

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **团队** | **一年** | **的地方** | **地图** | **外部数据** | **系综** | **方法** |
| UvA-Euvision | 2013 | 1日 | 22.6% | 没有一个 | 吗? | 费舍尔向量 |
| 深刻理解 | 2014 | 3日 | 40.5% | ImageNet 1 k | 3. | 美国有线电视新闻网 |
| 中大DeepID-Net | 2014 | 2日 | 40.7% | ImageNet 1 k | 吗? | 美国有线电视新闻网 |
| GoogLeNet | 2014 | 1日 | 43.9% | ImageNet 1 k | 6 | 美国有线电视新闻网 |

表4:检测性能

|  |  |  |  |
| --- | --- | --- | --- |
| **团队** | **地图** | **情景模式** | **边界框回归** |
| Trimps-Soushen | 31.6% | 没有 | 吗? |
| 伯克利的愿景 | 34.5% | 没有 | 是的 |
| UvA-Euvision | 35.4% | 吗? | 吗? |
| 中大DeepID-Net2 | 37.7% | 没有 | 吗? |
| GoogLeNet | 38.02% | 没有 | 没有 |
| 深刻理解 | 40.2% | 是的 | 是的 |

表5:检测单模型性能

有很多物体或没有，它们的尺度从大到小不等。结果报告使用平均精度(mAP)。

googleet采用的检测方法类似于[6]的R-CNN，但是用Inception模型作为区域分类器进行了扩展。此外，通过将选择性搜索[20]方法与多盒[5]预测相结合，改进了区域建议步骤，以获得更高的目标包围盒召回率。为了减少误报次数，超像素大小增加了2倍。这使得来自选择性搜索算法的提议减半。我们增加了200个来自多框[5]的区域提案，结果[6]总共使用了约60%的提案，同时将覆盖率从92%增加到93%。减少增加覆盖率的提案数量的总体效果是单个模型案例的平均精度提高1%。最后，我们使用6个卷积神经网络对每个区域进行分类，准确率从40%提高到43.9%。注意，与R-CNN相反，由于时间不够，我们没有使用包围盒回归。

我们首先报告top检测结果，并显示自第一版检测任务以来的进展情况。与2013年的结果相比，准确率几乎翻了一番。表现最好的团队都使用卷积网络。我们在表4中报告了官方分数和每个团队的常见策略:使用外部数据、集成模型或上下文模型。外部数据通常是用于预训练模型的ILSVRC12分类数据，该模型随后在检测数据上进行细化。一些团队还提到了本地化数据的使用。由于定位任务的很大一部分边界框不包括在检测数据集中，我们可以用这个数据预先训练一个一般的边界框回归器，就像使用分类方法进行预先训练一样。google网的条目并没有使用本地化数据进行预训练。

在表5中，我们只比较使用单一模型的结果。表现最好的模型是Deep Insight，令人惊讶的是，3个模型的集成只提高了0.3个点，而google et通过集成得到了显著更强的结果。

**9的结论**

我们的结果似乎提供了一个坚实的证据，即通过易于获得的密集构建块逼近预期的最优稀疏结构是一种可行的方法来改进计算机视觉神经网络。这种方法的主要优点是与较浅和较窄的网络相比，在适当增加计算需求的情况下显著提高质量。还要注意，我们的检测工作是有竞争力的，尽管既没有利用上下文，也没有执行边界框

10

回归和这一事实提供了先启体系结构强度的进一步证据。虽然我们期望通过类似深度和宽度的更昂贵的网络可以获得类似质量的结果，但我们的方法提供了坚实的证据，表明转移到稀疏体系结构通常是可行的和有用的想法。这预示着未来将在[2]的基础上以自动化的方式创建更稀疏和更精细的结构。

**10确认**

我们感谢Sanjeev Arora和Aditya Bhaskara就[2]进行的富有成果的讨论。我们也要感谢disbelief[4]团队对Rajat Monga, Jon Shlens, Alex Krizhevsky, Jeff Dean, Ilya Sutskever和Andrea Frome的支持。我们还要感谢Tom Duerig和Ning Ye对光度畸变的帮助。此外，如果没有查克·罗森博格和哈特维格·亚当的支持，我们的工作也不可能完成。

**参考文献**

了解你的表情包:我们需要更深入。http://knowyourmeme.com/memes/ we-need-to-go-deeper。访问:2014-09-15。

[2] Sanjeev Arora, Aditya Bhaskara, Rong Ge, and Tengyu Ma。学习某些深度表示的可证明界。*2013年,abs / 1310.6343。*

[3]乌米特¨V。所有的家庭都要在婚姻的举办权范围内发声。关于二维稀疏矩阵划分:模型，方法，和食谱。*暹罗j .科学。第一版。*， 32(2): 656-683, 2010年2月。

Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Marc 'aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, Quoc V. Le, and Andrew Y. Ng。大规模分布式深层网络。在P. Bartlett, F.c.n. Pereira, c.j.c. Burges, L. Bot- tou和K.q. Weinberger编辑的《神经信息处理系统进展》25，第1232-1240页。2012.

duitru Erhan, Christian Szegedy, Alexander Toshev和Dragomir Anguelov。利用深度神经网络进行可扩展的目标检测。计算机视觉与模式识别，2014。*CVPR 2014。IEEE会议，2014。*

Ross B. Girshick, Jeff Donahue, Trevor Darrell和Jitendra Malik。丰富的特征层次，以准确的目标检测和语义分割。计算机视觉与模式识别，2014。*CVPR 2014。IEEE会议，2014。*

Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever和Ruslan salakhudinov。通过防止特征检测器的共适应来改进神经网络。*2012年,abs / 1207.0580。*

Andrew G. Howard。基于深度卷积神经网络的图像分类的一些改进。*2013年,abs / 1312.5402。*

Alex Krizhevsky, Ilya Sutskever和Geoff Hinton。基于深度卷积神经网络的Imagenet分类。神经信息处理系统进展25，第1106-1114页，2012。

B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel。反向传播技术在手写邮政编码识别中的应用。*神经第一版。*， 1(4): 541-551, 1989年12月。

Yann LeCun, L´eon Bottou, Yoshua Bengio和Patrick Haffner。基于梯度的学习方法应用于文档识别。*电子与信息学报，34(11):388 - 391,1998。*

林敏，陈强，闫水成。网络在网络。*2013年,abs / 1312.4400。*

B. T. Polyak和A. B. Juditsky。平均的随机近似加速度。*SIAM J.控制Optim。*， 30(4): 838-855, 1992年7月。

bb0 Pierre Sermanet, David Eigen, Xiang Zhang, Micha¨el Mathieu, Rob Fergus, and Yann Le- Cun。Overfeat:使用卷积网络进行综合识别、定位和检测。*2013年,abs / 1312.6229。*

11

[15] Thomas Serre, Lior Wolf, Stanley M. Bileschi, Maximilian Riesenhuber和Tomaso Poggio。具有类皮质机制的鲁棒物体识别。*IEEE反式。模式肛门。马赫。智能。*29(3): 411 - 426年,2007年。

宋凤光和杰克·唐加拉。在具有1000个cpu核的共享内存多核系统上扩大矩阵计算。第28届ACM国际超级计算会议论文集，ICS ' 14，页333-342，纽约，纽约，美国，2014。ACM。

Ilya Sutskever, James Martens, George E. Dahl和Geoffrey E. Hinton。关于初始化和动量在深度学习中的重要性。第30届机器学习国际会议论文集，ICML 2013，亚特兰大，GA，美国，2013年6月16-21日，JMLR论文集第28卷，第1139-1147页。JMLR.org, 2013。

Christian Szegedy, Alexander Toshev和Dumitru Erhan。面向对象检测的深度神经网络。Christopher J. C. Burges, L´eon botou, Zoubin Ghahramani和Kilian Q. Weinberger，编辑，神经信息处理系统进展26:第27届神经信息处理系统年会2013。*2013年12月5-8日，美国内华达州太浩湖举行的会议记录。*，第2553-2561页，2013。

Alexander Toshev和Christian Szegedy。深度姿态:通过深度神经网络估计人体姿态。*2013年,abs / 1312.4659。*

Koen E. A. van de Sande, Jasper R. R. Uijlings, Theo Gevers, and Arnold W. M. Smeulders。分割作为目标识别的选择性搜索。2011年计算机视觉国际会议论文集，ICCV ' 11, 1879-1886页，华盛顿特区，美国，2011年。IEEE计算机协会。

Matthew D. Zeiler和Rob Fergus。可视化和理解卷积网络。David J. Fleet, Tom´as Pajdla, Bernt Schiele和Tinne Tuytelaars编辑，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，Proceedings, Part I, volume 8689 of Lecture Notes In Computer Science，第818-833页。施普林格,2014年。

12