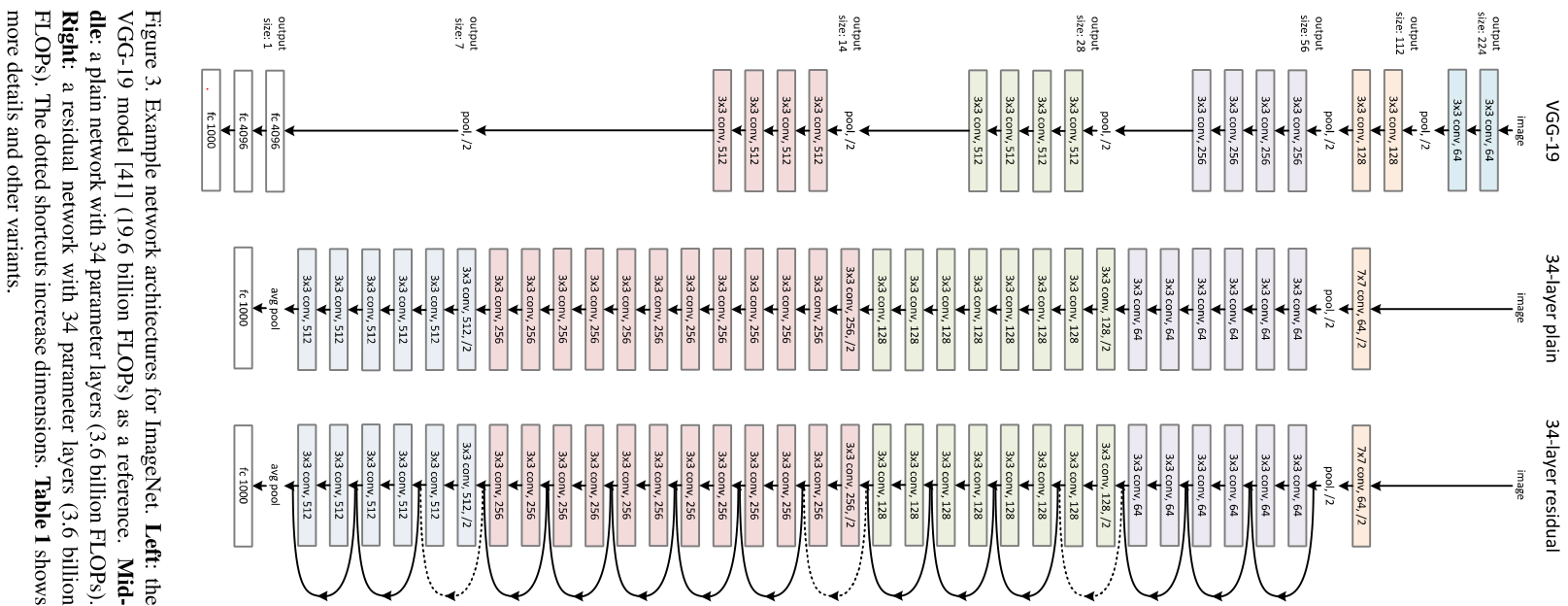
吉林大学本科毕业设计（论文）外文文献翻译

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 学 院 | 软件学院 | | | | 专业 | 软件工程 |
| 学生姓名 | 贾锋 | | | | 学号 | 55130218 |
| 指导教师 | 王欣 | | | | 职称 | 副教授 |
| 合作导师 |  | | 职称 |  | 单位 |  |
| 设计（论文）题目 | | 基于Qt的卷积神经网络辅助设计系统 | | | | |
| **使用深度残差学习进行图像识别**  Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun  Microsoft Research  {kahe,v-xiangz,v-shren,jiansun}@microsoft.com  **摘要:**  神经网络的层数越多，训练就越困难。我们提出了一种减轻网络训练负担的残差学习框架，这种网络比以前使用过的网络层次更深。我们把层定义为以本层输入为参考的残差函数，而不是训练没有参考的函数。同时，我们提供了全面的实验数据，这些数据证明残差网络更容易优化，并且可以通过增加深度来提高精度。我们在ImageNet数据集上用152层——8倍于VGG网络[41]的深度来测试残差网络，但它仍具有较低的复杂度。在ImageNet测试集中，组合使用的残差网络达到了3.57%的误差。该结果在2015年大规模视觉识别挑战赛分类任务中赢得了第一。此外，我们还在CIFAR-10数据集上分析了100层和1000层的深度残差网络。  对于大部分视觉识别任务，表示的深度是非常重要的。在COCO目标检测数据集上，仅由于极深的表示，我们就得到了28%的相对提升。深度残差网络是我们参加ILSVRC和COCO2015竞赛的基础[[1]](#footnote-1)，而且在ImageNet的图像检测和图像定位，COCO图像检测和COCO图像分割中，我们获得了第一名。 | | | | | | |
| **1、简介**  神经网络[22，21]在图像分类方面，引发了一系列突破[21，50，40]。通过改变叠层的数量（深度），深度网络以层与层首尾连接的方式整合低/中/高水平的特征[50]和分类器，并且特征的“等级”会随着网络的加深而更加丰富。最近的证据[41，44]表明网络深度是至关重要的，在ImageNet数据集挑战赛[36]中取得领先的团队[41，44，13，16]，都利用了“很深”模型[41]，该深度为十六[41]至三十[16]。而且很多特殊的视觉识别任务[8，12，7，32，27]也从深度模型中大大受益。  由于网络深度的重要意义，人们提出了一个问题：训练更好的网络是否和叠加层数一样简单？解决该问题的一大障碍是臭名昭著的梯度爆炸与消失问题[1，9]，它从一开始就阻碍了收敛。然而，这个问题很大程度上被正态初始化[23，9，37，13]和正则化层[16]解决了，它确保几十层的网络在使用随机梯度下降（SGD）进行反向传播的时候可以收敛[22]。  当更深的网络能够收敛时，暴露出了“退化”问题：随着网络深度的增加，精度开始饱和（这并不令人吃惊）然后迅速下降。预料之外的是，这种退化并不是由过度拟合造成的。而且在深度适当的模型中增加更多的层数，会造成更高的训练误差，这种现象在[11，42]中曾被报道，我们的实验对其进行了验证。图1展示了一个经典的实例。  精度的退化表明了，并不是所有的系统都同样容易优化。我们考虑一个较浅的网络结构和一个与它结构相似但是带有更多层的网络结构。有这样一种方法来建立那个较深的模型：添加恒等映射层，这种层直接拷贝上一层的输出作为自己的输出，其他层直接从训练好的模型中复制过来。这表明至少存在一种方法，向一个网络中添加更多的层，且并不会增加他的训练误差。但实验显示我们目前无法找到一个与用这种方法构造出来的网络一样好，或者更好的解决方案（或者无法在可行的时间内完成）。  在本论文中，介绍了一个深度残差学习框架来解决精准度下降问题。我们明确地让一些层拟合残差映射，而不是寄希望于每一个层直接拟合一个理想的映射。形式上，把作为理想的基本映射，让一些非线性层拟合另一个映射。那么原映射便转化成：。我们假设优化残差映射比优化原来的没有参照的映射更容易。在最极端的情况下，如果恒等映射是最佳的，那么为了拟合恒等映射，将残差推到零，比使用很多非线性层来拟合更容易。 | | | | | | |

|  |
| --- |
| 公式可以通过在前馈神经网络中添加“捷径”实现（图2）。捷径指的是那些跨过网络中的一层或多层的连接[2，34，49]。在我们的情景中，跨层连接扮演恒等映射的角色，而且，它们的输出会添加到这些层的输出中（图2）。恒等跨层连接既不会添加额外的参数，也不会增加计算的复杂度。整个网络仍然可以逐层的使用SGD进行反向传播训练，并且可以在不修改求解器的情况下很容易的使用公开库（例如Caffe[19]）进行实现。  我们在ImageNet上进行详细实验[36]展示精准度下降问题，并评估我们的方法。我们发现：（1）特别深的残差网网络很容易优化，但深度增加时，对应的“普通”网络（即简单的增加层）表现出更高的训练误差。（2）深度残差网络的精度能够从深度的增加中获取更大的收益，实验结果明显优于以前的网络。  相似的现象同样出现在了CIFAR-10数据集中[20]，这表明了我们的方法的优化的难易度和有效性的并不依赖于一个特定的数据集。我们在这个数据集中成功的训练出了超过100层的模型，并探索了超过1000层的模型。  在ImagNet分类数据集中[36]，通过极深的残差网络，我们得到了非常好的结果。在ImageNet提出的模型中，我们的152层的残差网络是最深的网络，而且相比于VGG网络仍然具有较低的复杂性[41]。我们在ImageNet测试集达到了整体3.57%的前五项误差，并且在ILSVRC-2015分类竞赛中取得第一名。这种极深的网络在其它识别任务方面也有出色的泛化性能，我们在ILSVRC-2015和COCO-2015的以下项目中赢得了第一：ImageNet图像检测、ImageNet图像定位、COCO图像检测和COCO图像分割。这有力的证据表明，残差学习的原理是行得通的，我们期望它适用于其它的视觉和非视觉问题。 |

|  |
| --- |
| **2、相关工作**  **残差表示。**在图像识别中，VLAD[18]是一个以关于字典的残差向量进行编码的特征，而且费舍尔向量[30]可以被表述成概率版本的VLAD。它们两个对于图像检索与分类都是有效的浅层特征[4，48]。对于矢量化，编码残差向量比编码[17]原始向量更有效。  在低层次视觉和计算机图形学中，为了解偏微分方程（PDEs），它们广泛的使用多重网格法[3]，将系统重构成多尺度的子问题，每个子问题负责更粗和更细规模之间的残差求解。多重网格的一种替代方法是分层的基础预处理[45，46]，它依赖于表示两个尺度之间的残差向量的变量。[3，45，46]表明这些解决方案的收敛速度比不使用残差性质的标准解法快得多。这些方法表明，一个好的方法或预处理可以使网络优化更简单。  **跨层连接。**跨层连接的方法和理论已经被研究了很长时间[2，34，49]。训练多层感知器（MLPs）的早期实践是添加一个线性层连接网络的输入和输出[34，39]。在[44，24]中，一些中间层直接连接到辅助分类器，来解决梯度的爆发与消失问题。论文[39，38，31，47]提出有关中心层响应、梯度和传播误差的方法，该方法通过跨层连接实现。在[44]中，一个“开始”层是由一个跨层连接分支和一些更深的分支组成。  与我们的工作同时进行的[42，43]“highway network”提出了带有开关功能[15]的跨层连接。这些连接是与数据相关的并带有参数，而我们的恒等跨层连接无需参数。当一个带开关的跨层连接“关闭”（接近零）时，highway network中的层代表非残差函数。相反的是，我们的想法是总是学习残差函数，恒等跨层连接不会有关闭，它总是允许所有的信息通过，并且需要学习额外的残差函数。此外，highway network并没有表现出通过层次加深而获得的精度提高（例如，超过100层）。 |

|  |
| --- |
| **3、深度残差学习**  **3.1、残差学习**  将H（x）假设为由几个层拟合的（不一定是整个网络）基础映射，用x表示这些层中的第一层的输入。假设足够多的非线性层能逼近复杂的函数[[2]](#footnote-2)，也就相当于假设它们可以逼近残差函数，例如H（x）-x（假设输入和输出是同一维度的）。因此我们希望这些层能够逼近于残差函数F(x):=H(x)-x，而并非H（x）。所以原函数变成了：F（x）+x。尽管两种形式都能逼近期望函数（就像假设的那样），但训练它的难易度可能不同。新的构思源于反常的精准度下降问题（图1，左）。如我们在简介中讨论的一样，如果添加的层可以被构造为恒等映射，那么一个更深模型的训练误差，不应大于与其相应的更浅的模型训练误差。精度下降问题表明，多个非线性层构成的求解器在拟合恒等映射方面有困难。在残差学习过程中，如果恒等映射是最佳的方法，那么求解器可以简单地使多个非线性层的权重趋向于零，以便逼近恒等映射。  在现实情况中，恒等映射未必是最优的，但我们的方法可能有助于事先处理该问题。如果最优函数相比于零映射更趋近于恒等映射，那么与学习一个新函数相比，求解器更容易找到关于恒等映射的响应。我们的实验证实（图7），所学到的残差函数一般有小的响应，这表明恒等映射提供了合理的预处理。  **3.2、用跨层连接实现的恒等映射**  我们对每一个层都采用残差学习，一个网络单元（Building block）如图2所示。形式上，我们把一个网络单元定义为：  其中x和y是层的输入和输出向量。函数F(x,{})代表需要学习的残差函数。如图2所示，一个网络单元含有两个层，中表示ReLU函数[29]，为了简化符号，偏置项已经忽略。的操作是由跨层连接和逐元素的加法进行的。在相加之后我们采用了第二个非线性函数（例如，见图2）。  公式（1）中介绍的跨层连接，没有额外的参数和复杂的计算。这不仅在实践中有吸引力，它在对比普通网络和残差网络方面也同样重要。当普通网络和残差网络有着相同数量的参数，深度，宽度和计算成本时（除了可以忽略不计的基于元素的加法），我们可以对它们进行公平的对比。  在式(1)中x和F的维度必须相同。如果不同（例如改变了输入或输出通道）我们可以通过对跨层连接的输出进行线性投影来匹配维度：  我们也可以在式(1)中使用一个方阵。但我们会通过实验表明，恒等映射足以用于解决精准度下降的问题且该方法是非常经济的，因此只有在匹配维度时，才使用。  残差函数F的形式是灵活的，本文的实验涉及的函数F拥有两层或三层（图5），尽管更多的层也是可行的。但如果F仅仅只有1层，式(1)就类似于线性层：，因此我们没有观察到这样做的优势。  我们还注意到，虽然上面的符号为了简单起见是关于全连接层的，但它们也  适用于卷积层。函数F(x,{})可以代表多个卷积层。这时，逐元素的加法就变  成了两个特征图之间逐通道的加法。 |



|  |
| --- |
| **3.3、网络构架**  我们测试了不同的普通网络和残差网络，并且观察到了一致的现象。下面描述了两个以ImageNet为数据集的模型实例。**普通网络。**我们的普通网路基准（图3，中）主要受启发于VGG网络[41]理论（图3，左）。卷积层大多数具有3X3的卷积核，并遵循两个简单的设计规则：（1）如果两个层输出了相同大小的特征图，则他们拥有相同数量的滤波器；（2）特征图的大小被减半，过滤器的数量变为原来的2倍，以保持每层的时间复杂度。我们在步长为2的卷积层后面直接进行下采样。在网络的最后，有一个全局平均池化层和一个1000节点的全连接层，最后使用Softmax分类器。带有权值的层总共有34个。见图3（中）。  值得注意的是，我们的模型比VGG网[41]（图3，左）有着更少的滤波器和更低的复杂度。作为基准的34层网络有36亿FLOPs（乘法和加法），这只有VGG-19的18%（196亿FLOPs）。  **残差网络。**基于上面的普通网络，我们插入跨层连接（图3，右），将网络转变成其对应的残差版本。当输入和输出的维度相同的时候（实线，图3），跨层连接（公式（1））可以直接使用。当维度不同时（虚线，图3），我们考虑了两个选项：（A）添加0以进行延拓。该选项不会引入额外的参数；（B）使用公式（2）中的投影操作，用于匹配维度（通过1×1卷积完成）。无论使用哪种策略，跨层连接都跨越两个层。  **3.4、实现**  我们遵循了[21，41]中的方法，在ImageNet进行了实现。对图片进行缩放以进行数据扩增[41]，缩放时图像的短边在[256，480]中随机取值。从原图和它的水平翻转图中随机取样出224x224大小的结果，并减掉像素均值[21]。使用了[21]中的标准颜色增强。在每一次卷积之后和激活之前，我们采用了BN算法（batch normalization）[16]。我们如[13]中一样初始化权重，并从零开始训练普通网络和残差网络。我们使用SGD（随机梯度下降）进行训练，mini-batch为256，学习率从0.1开始，并在误差稳定时除以10，训练迭代次，权值衰减为0.0001，动量为0.9。依据[16]中的经验，我们不使用dropout[14]。  测试中，为了对比我们采用标准的10-crop试验[21]。为达到最佳效果，我们采用完全卷积形式如[41，13]中所示，并且平均了多尺度下的成绩（图片进行了大小调整，短边的长度取值为{224，256，384，480，640}）。  **4、实验**  **4.1、Imagenet分类**  我们在ImageNet-2012分类数据集[36]中评估了我们的方法，该数据集含有1000类图片。在128万张训练图像中训练模型，并且在5万张验证图像进行了评估。测试服务器报道，我们也在10万张的测试图像中获得了最终结果。我们评价第1和前5项错误率。  **普通网络。**我们首先评估了18层和34层普通网络网。34层的普通网络在图3（中）中。18层普通网络形式是类似的。详细架构见表1。  表2显示的结果表明，较深的34层普通网络具有比较浅的18层普通网络更高的验证错误率。为了揭示原因，在图4（左）中，我们比较了训练过程中的训练和验证错误。我们观察到退化问题--在整个训练过程中34层的普通网络错误更多，尽管18层普通网络的解空间是34层网络的解空间的一个子空间。 |

|  |
| --- |
| 我们主张优化困难不太可能是由梯度消失问题导致的。所有的普通网络都是通过BN[16]训练的，而这也保证了向前传播的信号是非零的。此外我们也验证了反向传播的梯度在BN中表现出良好。所以无论是向前或是向后传播的信号都没有消失。事实上，34层的普通网络仍然可以达到可观的精确度（如表3），而这在一定的程度上也暗示了求解器的工作范围。我们猜想深度普通网络的收敛率可能会指数级的降低，而收敛率影响了训练错误的降低[[3]](#footnote-3)。此类优化问题会在未来进行研究。  **残差网络。**接下来评价18层和34层的残差网络（ResNets）。作为基准的网络架构与普通网络一样，每一对3×3卷积层之间增加一条跨层连接（图3，右）。在第一个比较中（表2和图4，右），我们使用前文所述的策略A，在维度不同时，通过补0进行延拓。所以普通网络相比，残差网络并没有额外的参数。  我们从表2和图4中得出3个主要结论。第一，在残差学习模式上出现了与普通网络相反的情况--34层网络比18层网络表现要好（2.8%左右）。更重要的是，34层残差网络比18层残差网络错误率更低，并且验证数据也证明了这一点。而这也表明我们很好的解决了退化问题能，而且能通过增加层次来提升准确率。  第二，与普通网络相比，34层残差网络将最低错误率降低了3.5%（如图2所示）。这一比较证实了残差学习在超深系统中的优越性。  最后，我们注意到18层的普通和残差网络的正确率比较相近（表2），但是18层的残差网络收敛的更快（图4右VS左）。当网络没有“过深”（如18层）时，当前的SGD求解程序依然可以在使用普通网络的情况下找到好的解决方案。在此情况下，残差网络在早期阶段通过提供更快的收敛能使网络的优化更容易一些。  **恒等VS投影式跨层连接。**我们发现无参数的恒等跨层连接能帮助训练。接下来我们会研究投影式跨层连接（等式2）。在表3中我们比较了三种选择：（A）需要增加维度的时候使用补零延拓，且所有的跨层连接都无参数（如表2和图4，右）；（B）需要增加维度时使用投影，而其他的跨层连接都是恒等式的；（C）所有的跨层连接都进行投影。  表3显示三个方案就普通网络比较而言都表现得更出色。B比A表现略好。我们会怀疑这是因为A中使用零填充增加的维度没有参与残差学习。C比B表现稍好，我们将此归功于那些（13个）被投影式跨层连接引入的额外参数。但是A/B/C之间差别很小，不足以证明采用投影操作的跨层连接对于解决退化问题必不可少。所以为了控制空间和时间复杂度和模型规模，在本文的剩余部分我们不使用方案C。恒等跨层连接对于不为“瓶颈结构”增加额外的复杂度至关重要。 |

|  |
| --- |
| **更深层次的瓶颈结构。**接下来描述我们在ImageNet上使用的更深的网络。鉴于我们能接受的训练时间是有限的，我们改变了网络单元的结构，将其变成瓶颈式设计[[4]](#footnote-4)。对于每一个残差函数F，我们使用3层而不是2层（图5）。这3层分别是1×1，3×3和1×1的卷积层。1×1层用于减少和增加（恢复）维度，让3×3层的输入和输出维度更小，这就是所谓的“瓶颈”。图5展示了一个例子，这两个设计的时间复杂度都一样。  无参数恒等跨层连接对于瓶颈式结构尤为重要。如果把图5（右）中的恒等跨层连接换成投影式跨层连接，由于它连接的是两个高维端，时间复杂度和模型规模都会翻倍。所以直通捷径可以使瓶颈式设计更有效。  **50层的残差网络**。我们用3层的瓶颈式网络单元取代34层网络结构中的2层单元，这时34层结构就会变成50层残差结构。我们使用B方案增加维度。这一模型有38亿FLOPs。  **101层和152层的残差网络。**我们通过使用3层的瓶颈式网络单元构建101层和152层的残差网络（表1）。让人惊讶的是，尽管深度显著增加，152层的残差网络（113亿FLOPs）复杂性较VGG-16/19网络（153/196亿FLOPs）更小。  50/101/152层的网络明显比34层的更准确（表3，表4）。网络没有出现退化问题。更深的层次带来了更高的准确率。表3和表4在各种程度上体现了深度增加带来的收益。  **与先进方法之间的比较。**在表4中，我们比较了之前单一模型所取得的最好结果。我们作为基准的34层残差网络取得了不错的准确度。152层ResNet的单模型前五项误差为4.49%。而这也超过之前所有网络的成绩（表5）。我们结合6种不同深度的模型形成一个整体（最终提交版本中只包含两个152层的模型），这一模型在测试中的前五项错误率只有3.57%（表5），这一结果在ILSVRC-2015中取得了冠军。 |

|  |
| --- |
| **4.2、CIFAR-10和分析**  我们在CIFAR-10数据集中进行更多的研究[20]，CIFAR-10包括五万张的训练图像和一万张的测试图像，这些图像被分成了10类。我们将会展示在训练集中训练和在测试集中评估的试验。我们关注的焦点在于超深网络的行为，而不在于取得最好结果，所以我们有意的使用了简单的结构。  所有普通/残差网络都遵循图3（中，右）的流程。网络输入32×32的图像，每个像素都减去了图像均值。第一层是3×3卷积层。接下来是6n层的3×3卷积层，特征图的大小依次为{32，16，8}，每个特征图大小持续使用2n层。卷积层后面的池化层步长为2。网络的末端有一个全局平均池化层，一个10路全连接层，最后是softmax分类器。总共有6n+2个带有权重的层。下表总结了其结构构成： |

|  |
| --- |
| 使用跨层连接时，它们连接一组组的3×3层，这样总共有3n条捷径。在此数据集中，我们在所有的情况下都使用恒等跨层连接（A方案），所以我们的残差模型与普通模型有完全一样的深度，宽度和参数数量。  我们设置权重衰减为0.0001，动量为0.9，使用了[13]中的权值初始化方法和[16]中的BN方法，没有使用dropout，mini-batch设为128，使用两个GPU进行训练。我们开始的学习率是0.1，经过32000次和48000次迭代后将学习率除以10，在64000迭代时结束训练。这是由45K/5K的训练/验证比值决定的。我们使用了[24]中的简单的数据扩增方法：每一边填充4像素，再从填充完的图像或者其水平翻转中随机选取大小为32×32的结果。测试时我们只使用32×32的原始图像。  我们比较n={3，5，7，9}，即20层、32层、44层和56层的网络结构。图6（左）展示的是普通网络的表现。普通网络随着深度的增加错误率也会变高。这与在ImageNet（图4，左）和MNIST[42]上得到的结论是类似的，这说明对于普通网络，这种优化问题是一个基础性问题。  图6（中）向我们展示了ResNet的表现。与在ImageNet上类似（图4，右），ResNet克服了优化问题:随着深度的增加，准确率提高了。  当n=18，即残差网络达到110层时，我们发现初始学习率为0.1对于收敛还是太高了[[5]](#footnote-5)。所以我们使用0.01的初始学习率进行训练直到错误率低于80%，再使用0.1的学习率进行训练。接下来的步骤如之前所述。110层的网络收敛的十分好（图6，中）。比之于其他模型，如FitNet[35]和Highway[42]，ResNet的参数量更少，但是结果却是十分的优秀（6.43%，表6）。  **层响应的分析。**图7展示的是层响应的标准差。所谓层响应，指的是每个3×3层的输出，在BN之后，非线性函数（例如ReLU之类）之前。对于ResNet，这一分析反应的是残差函数的影响。图7显示ResNet通常比普通网络响应更小。这一结果证实了我们最初的想法（见3.1）：残差函数比非残差函数更接近于零。图7中ResNet-20、56和100的比较，证明更深层次的ResNet反应幅度更小。随着层数的增加，ResNet的每一层会倾向于更少的改变信号。  **超过1000层的探索。**我们大胆的探索了超过1000层的模型。设置n=200，此时的网络有1202个层。训练方式和上文所述一致。我们的方法没有出现优化难的问题。这个上千层的网络结构能实现训练错误率小于0.1%（图6，右）。其测试错误率也十分的低（7.93%，表6）。 |

|  |
| --- |
| 但是在这样的超深模型中还是有未解决的问题。1202层的测试结果比110层的要差，尽管两者的训练错误率差不多。我们认为是过度拟合造成的。也许是这个1202层的网络（19.4M）对于如此小是数据集太大了。在此数据集中要取得较好的结果[10，25，24，35]有[必要](http://www.leiphone.com/news/201511/tgBJPsUzzzgvFHLp.html)做一些调整，如使用Maxout[10]或Dropout[14]。在本文中，我们并未使用Maxout和Dropout方法，只是设计了深度和结构。但是通过更大的调整也许能取得更好的结果，这一课题我们会在未来进行研究。  **4.3、在PASCAL和MSCOCO上的目标检测**  我们的方法在其他的识别任务中有较好的泛化表现。表7和表8显示了在PASCAL VOC 2007和2012[5]、COCO[26]上的目标检测基准结果。我们采用Faster R-CNN[32]作为检测方法。我们对于使用ResNet取代VGG-16[41]方法所获得的进步很感兴趣。两种模型所使用的目标检测的具体实现（见附录）是一样的，所以所获结果的差别取决于网络结构。值得注意的是，在极具挑战性的COCO数据集中，我们在COCO的衡量标准（mAP@[.5，.95]）下取得了6.0%的进步，即28%的相对进步。这种进步完全得力于模型的深度。  基于深度残差网络，我们赢得了ILSVRC&COCO2015比赛的数个任务的冠军：ImageNet图像检测，ImageNet定位，COCO图像检测和COCO图像分割。详情请见附录。 |

|  |
| --- |
| **参考文献:**  [1] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks, 5(2):157–166, 1994.  [2] C. M. Bishop. Neural networks for pattern recognition. Oxford university press, 1995.  [3] W. L. Briggs, S. F. McCormick, et al. A Multigrid Tutorial. Siam, 2000.  [4] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In BMVC, 2011.  [5] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. IJCV, pages 303–338, 2010.  [6] R. Girshick. Fast R-CNN. In ICCV, 2015.  [7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.  [8] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In AISTATS, 2010.  [9] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio. Maxout networks. arXiv:1302.4389, 2013.  [10] K. He and J. Sun. Convolutional neural networks at constrained time cost. In CVPR, 2015.  [11] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In ECCV, 2014.  [12] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In ICCV, 2015.  [13] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing coadaptation of feature detectors. arXiv:1207.0580, 2012.  [14] S. Hochreiter. Untersuchungen zu dynamischen neuronalen netzen. Diploma thesis, TU Munich, 1991.  [15] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.  [16] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In ICML, 2015.  [17] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. TPAMI, 33, 2011.  [18] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid. Aggregating local image descriptors into compact codes. TPAMI, 2012.  [19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. arXiv:1408.5093, 2014.  [20] A. Krizhevsky. Learning multiple layers of features from tiny images. Tech Report, 2009.  [21] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, 2012.  [22] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. Neural computation, 1989.  [23] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Muller. Efficient backprop. ¨ In Neural Networks: Tricks of the Trade, pages 9–50. Springer, 1998.  [24] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeplysupervised nets. arXiv:1409.5185, 2014.  [25] M. Lin, Q. Chen, and S. Yan. Network in network. arXiv:1312.4400, 2013.  [26] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick. Microsoft COCO: Common objects in ´ context. In ECCV. 2014.  [27] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015.  [28] G. Montufar, R. Pascanu, K. Cho, and Y. Bengio. On the number of ´ linear regions of deep neural networks. In NIPS, 2014.  [29] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In ICML, 2010.  [30] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In CVPR, 2007.  [31] T. Raiko, H. Valpola, and Y. LeCun. Deep learning made easier by linear transformations in perceptrons. In AISTATS, 2012.  [32] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015.  [33] B. D. Ripley. Pattern recognition and neural networks. Cambridge university press, 1996.  [34] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. In ICLR, 2015.  [35] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. arXiv:1409.0575, 2014.  [36] A. M. Saxe, J. L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. arXiv:1312.6120, 2013.  [37] N. N. Schraudolph. Accelerated gradient descent by factor-centering decomposition. Technical report, 1998.  [38] N. N. Schraudolph. Centering neural network gradient factors. In Neural Networks: Tricks of the Trade, pages 207–226. Springer, 1998.  [39] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In ICLR, 2014.  [40] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.  [41] R. K. Srivastava, K. Greff, and J. Schmidhuber. Highway networks. arXiv:1505.00387, 2015.  [42] R. K. Srivastava, K. Greff, and J. Schmidhuber. Training very deep networks. 1507.06228, 2015.  [43] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In CVPR, 2015.  [44] R. Szeliski. Fast surface interpolation using hierarchical basis functions. TPAMI, 1990.  [45] R. Szeliski. Locally adapted hierarchical basis preconditioning. In SIGGRAPH, 2006.  [46] T. Vatanen, T. Raiko, H. Valpola, and Y. LeCun. Pushing stochastic gradient towards second-order methods–backpropagation learning with transformations in nonlinearities. In Neural Information Processing, 2013.  [47] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms, 2008.  [48] W. Venables and B. Ripley. Modern applied statistics with s-plus. 1999.  [49] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional neural networks. In ECCV, 2014. |

|  |
| --- |
| **附录:**  **A．目标检测基准**  在这一节中来介绍我们基于Faster R-CNN系统的检测方法。这个模型用ImageNet分类模型进行初始化，然后用目标检测数据进行微调。在ILSVRC&COCO 2015检测竞赛中我们验证了ResNet-50/101。  与参考文献[32]中的VGG-16不一样，ResNet没有全连接隐层。我们将“卷积特征图的网络”（ Noc）[33]理念运用到这个问题中。使用那些在步长不超过16像素的层（例如conv1，conv2\_x，conv3\_x，conv4\_x等ResNet-101中的共91个卷积层；表1）计算出全图共享的卷积特征图。考虑到这些层和VGG-16中的13个卷积层类似，这样ResNet和VGG-16就都有同样的总步长了（16像素）。一个区域建议网络（RPN，产生300个建议）[32]和一个Fast R-CNN检测网络[7]共享这些层。在conv5\_X之前感兴趣区域池化（ROI pooling）。在这个ROI池化后的特征上，每个区域都会被conv5\_x及以上的层使用，这与VGG-16中的全连接层起相同的作用，最终的分类层被并列的两个同级层取代（分类和矩形回归（box regression）[7]）。  对于BN层的使用，在预训练之后，我们在ImageNet训练集上对每个层都进行了BN统计（均值和方差）。BN层在进行针对目标检测的微调时被固定。这样一来，BN层变成了带有固定偏置和缩放的线性层。固定BN层主要是为了减少在Faster-CNN训练过程中的内存消耗。  **PASCAL VOC**  针对PASCAL VOC 2007数据组，我们和参考文献[7，32]中使用的方法一样，我们使用VOC2007中的5000张图片和VOC2012中的16000张训练图片进行训练（“07+12”）。对于PASCAL VOC2012测试集，我们使用VOC2007中的10000张训练和测试图片与VOC2012中的16000张图进行训练（“07++12”）。用来训练Faster R-CNN的超参数和参考文献[32]中提到的一样，表7展示的是结果。ResNet-101在VGG-16的基础上提高了超过3%的mAP成绩。这些提高来源于ResNet学习到的新特征。  **MS COCO**  MS COCO数据集包括80种对象类别。我们的评价标准是PASCAL VOC度量（mAP @IoU=0.5）和标准COCO度量（mAP @IoU=.5:.05:.95）。我们使用80000张图像训练，使用验证集中的40000张图像进行评价。我们在COCO上和PASCAL上使用的检测系统时类似的。COCO模型的训练使用了8个GPU，所以RPN使用的nimi-batch是8（每个GPU为1），Fast R-CNN使用的mini-batch是16。在训练过程中，RPN和Fast R-CNN都使用0.001的学习率迭代24万次，然后用0.0001的学习率迭代8万次。 |

|  |
| --- |
| 表8展示的是在MS COCO验证集上的结果。[ResNet-101相比于VGG-16在mAP@[.5,.95](mailto:ResNet-101相比于VGG-16在mAP@[.5,.95)]取得了6%的进步，即28%的相对提高，这完全得益于更好的网络学习到的特征。值得一提的是，[在mAP@[.5,.95](mailto:在mAP@[.5,.95)][上6%的绝对提高接近于mAP@.5](mailto:上6%25的绝对提高接近于mAP@.5)上的6.9%。这些结果证明更深的网络的确能提高识别和定位的能力。  **B．对象检测提升**  为了追求完美，我们报道了针对竞赛做出的提升。这些提升都是基于深度特征的，而且残差学习对其有益。  **MS COCO**  **边界提取。**我们的边界改良沿用的是参考文献[6]中的迭代定位。在F-R-CNN中，最终的输出是一个回归矩形，而不是建议矩形。我们集中回归矩形中的特征，获取一个新的分类分数和一些新的回归矩形。我们将300个新的预判与原始的300个预判结合起来。根据矩形投票方法[6]，我们在预测矩形的并集上应用了非最大抑制[NMS]，IoU阈值设置为0.3[8]。边界改良大概提升了2%的mAP成绩（如表9所示）。  **全局上下文。**在Fast R-CNN上，我们综合了全局上下文。为了使用全图卷积特征图，我们对特征使用了空间金字塔池化[12]（使用单层金字塔），这可以通过将ROI设为整个图片的边缘矩形，并使用ROI池化来实现。池化后的特征被注入到后面的ROI层，以产生全局上下文。全局特征是与原始的每个区域的特征连锁的，后面层是并列的分类器和矩形区域回归层。这个新的结构是首尾相接训练的，全局上下文在mAP@.5上提高了1%（表9）。  **多尺度测试。**和[32]一样，以上所有的结果都是在单一尺度内进行训练、测试的，如参考文献[32]中提到的那样图片的短边是600像素的。[12，7]中使用从特征金字塔中选取尺度的方法实现多尺度训练和测试，在[33]中使用的是maxout层。在我们目前的实现方案中，我们沿用了参考文献[33]中的方法进行了多尺度测试；但由于时间限制我们还没有进行多范围训练。另外，我们只针对Fast R-CNN步骤进行了多尺度测试（RPN步骤没有进行）。在训练好的模型中，我们在图片金字塔中计算卷积特征图，图片的短边长为s∈{200，400，600，800，1000}。根据[33]，我们在金字塔中选取相邻的两个尺度。ROI池化和后面的层在这两种尺度[33]的特征图上工作，然后使用maxout进行合并[33]。多尺度测试能提升2%的mAP成绩（表9）。 |

|  |
| --- |
| **使用验证数据。**接下来我们使用8万+4万的验证数据集来进行训练，2万的测试数据集来进行评估。测试数据集的数据不是开放的，其结果是由验证服务器给出的。我们的但模型是这样：[mAP@.5](mailto:mAP@.5)上的结果是55.7%，mAP@[.5,.95]上的结果是34.9%，如表9所示。  **组合。**Faster R-CNN系统被设计用来学习给出区域建议和目标分类，组合在一起对两者都有提高。表9展示了我们基于3个网络组合的结果。在测试数据集上，取得了59.0%和37.4%的mAP成绩。这是COCO 2015目标检测任务中的最佳成绩。  **PASCAL VOC**  基于上面的模型我们在PASCAL VOC数据集上进行了测试。以COCO数据集上的模型为基础（[55.7%的mAP@.5](mailto:55.7%25的mAP@.5)成绩，表9），我们对在PASCAL VOC数据集上对其进行了微调。同时也采用了全局上下文，多尺度测试等改进。结果是我们在PASCAL VOC 2007上取得了85.6%的mAP成绩（表10），在PASCAL VOC 2012上取得了83.8%的mAP成绩[[6]](#footnote-6)（表11）。在PASCAL VOC 2012上获得的成绩比目前最佳成绩还要好10%。  **ImageNet检测**  ImageNet目标检测任务（DET）包括200个对象类别。[准确率评估标准使用的是mAP@.5](mailto:准确率评估标准使用的是mAP@.5)。我们使用的检测算法和在MS COCO中的相同（表9）。网络在1000类的ImageNet分类数据集上进行了预训练，在DET数据集上进行了微调。根据[8]，把验证数据集分成了两部分（val1/val2）。我们在DET训练集和val1数据集上进行微调，使用val2进行验证。我们没有使用过其他的ILSVRC2015数据。单模型的ResNet-101拥有58.5%的mAP成绩，而在DET测试数据集上，我们的三模型组合网络拥有62.1%的mAP成绩（表12）。这个结果赢得了ILSVRC2015的第一名，超过第二名8.5%（绝对值）。 |

|  |
| --- |
| **C．ImageNet定位**  ImageNet定位（LOC）任务[36]要求对物体分类和定位。根据参考文献[40，41]，假设首先用图像分类器来预测图像类别，然后根据预测类别使用定位算法开始进行定位。我们采用了“逐类回归策略”（PCR）[40，41]，为每个类别训练一个边界回归器。在ImageNet分类数据集上进行预训练，然后在定位数据集上将进行微调。我们在ImageNet的1000类训练数据集上训练这个网络。  我们的定位算法是基于RPN框架[32]进行小部分修改得到的。和参考文献[32]的类别无关方法不同，我们的RPN定位是针对每一类设计的。在参考文献[32]中，RPN最后的阶段是并列的用于二分类的1×1卷积层（cls）和边界框定组件（reg）。在我们的网络中，cls和reg层都是逐类设计的形式。特别的，cls层有一个1000维的输出，每一个维度都是二进制逻辑回归值，代表的是现在处理的图像是不是某一类。Reg层的输出是1000×4维的，存储的是1000的类别的矩形回归。就像在[32]中一样，我们的边界矩形在每个位置都有多重平移不变性的锚点。  在3.4中的ImageNet分类训练中，我们随机使用224x224的图像切割来进行数据扩增。设置mini-batch为256进行微调。为了避免阴性样本占据主导，每张图片上都随机取8个锚点，取样位置和阴性锚点比例是1:1[32]。为了测试，网络被设置成全卷积。  表13对定位结果进行了对比。根据[41]，我们首先进行“oracle”测试。VGG的文章[41]报道，使用真实类别时有33.1%的中心切割错误（表13）。在相同的测试中，我们的RPN方法使用ResNet-101将中心分割错误减少到了13.3%。  比较的结果可以证明我们的网络的优越性。使用密集（全卷积）和多尺度测试，我们的ResNet-101的错误率为11.7%。当使用ResNet-101来预测类别，前5的定位错误结果是14.1%，前五项分类错误率为4.6%（表4）。  上述结果仅基于Faster R-CNN[32]中的RPN。人们可以在Faster R-CNN中使用检测网络（Fast R-CNN[7]）来改善结果。但我们注意到，在此数据集，一个图像通常包含一个主要对象，并且建议区域相互高度重叠，故ROI池化特征非常接近。由于上述原因，以图像为核心的Fast R-CNN[7]所产生的的样本变化很小，这是随机训练所不愿看到的情况。在我们目前的实验中，我们使用原来的以ROI为核心的R-CNN[8]来取代Fast R-CNN。 |

|  |
| --- |
| 我们的R-CNN实现如下：把如上所述训练的逐类RPN用来预测正确类的边界矩形。这些预测出来的边界被作为基于类的建议。对于每一张训练图像，评分最高的200个建议被导出，作为R-CNN分类器的训练样本。和R-CNN中一样，从建议区域中裁剪下来的224×224像素大小的图片被送入分类器[8]。网络的输出层包括并列的cls和reg，他们也是被设计成逐类形式的。R-CNN网络采用以ROI为核心的mini-batch为256的形式在训练集上进行微调。测试时，RPN针对每一个预测类生成评分最高的200个建议，R-CNN负责更新建议的评分和框定矩形的位置。  这种方法将前5项的定位错误降低到10.6%（表13）。这是我们的单模型在验证数据集上的成绩。将分类和定位组合起来使用，我们将前5项的定位错误率降低到了9.0%。这个数字明显比ILSVRC14的结果要好（表14），减少了64%的相对错误率。  这个方法在定位任务中将前五项错误率降低到10.6%（表13）。这是在验证数据集上的单模型结果。将分类网络和定位网络结合起来，在测试数据集上的前五项错误率达到9.0%。这个数字明显好于ILSVRC-14的结果（表14），相对减少了64%的错误率。这个结果在ILSVRC2015中的ImageNet定位任务上取得了第一名的成绩。 |

1. <Http://image-net.org/challenges/LSVRC/2015> and <Http://mscoco.org/dataset/#detections-challenge2015> [↑](#footnote-ref-1)
2. 然而，这个假设仍然是个未被解决的问题，见[28]。 [↑](#footnote-ref-2)
3. 我们尝试增加训练迭代次数（变为原来的三倍），退化问题仍然出现，这表明简单的增加迭代次数并不能解决这个问题。 [↑](#footnote-ref-3)
4. 更深的非瓶颈式ResNet（图5，左）也可以通过增加深度来提高精确度（在CIFAR-10上证实），但是其开销比瓶颈式ResNet高，所以从实际出发，我们使用了瓶颈式网络。需要进一步指出的是，普通网路中出现的退化问题在瓶颈式网络中也会出现。 [↑](#footnote-ref-4)
5. 初始学习率设为0.1时，数个最大训练次数后开始收敛（错误率小于90%），但是最终的精度是类似的。 [↑](#footnote-ref-5)
6. <http://host.robots.ox.ac.uk:8080/anonymous/30j40j.html>,提交日期：2015-11-26 [↑](#footnote-ref-6)