Deep Residual Learning for Image Recognition

Abstract

Introduction

Related work

Deep residual learning

Experiment

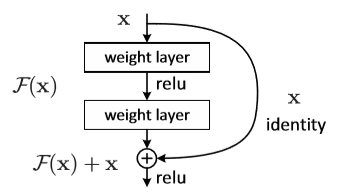
Abstract

当网络深度增加后，会变得很难训练。本文提出一种残差网络（Residual Network），将网络的层重新组合为新的学习残差网络，本文的经验和证据表明这些残差网络更容易优化，并且可以从显著增加的深度获得准确性

Introduction

深度卷积神经网络在图像分类上效果显著，网络可以集成从低级到高级的特征以及分类器，并且网络越深。特征级别越高。

网络深度增加后，会出现梯度消失和梯度爆炸的问题。这个问题通过归一初始化解决一部分。网络加深会出现退化问题（degradation）深度越深，越难以训练。

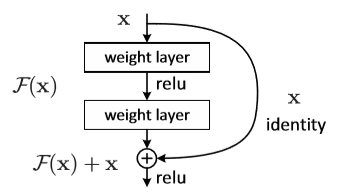


本文引入残差网络，将一个层的输入跳过一个或多个层的连接，到达后面层的输出中。此方法不增加额外参数也不增加计算复杂度。实验表明，深度残差网络很容易优化，并且可以轻松的从网络深度的增加中获得准确度增益

Identity mappings，恒等映射

3.Deep Residual Learning

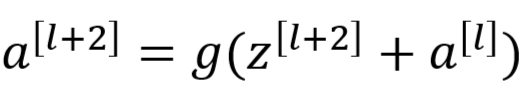
3.1. Residual Learning

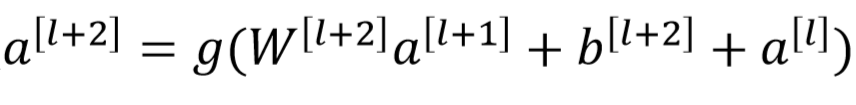


原来的函数是H(x)，作者将其改为F(x)+x，这两种表达的效果相同，但是优化的难度却并不相同，作者假设F(x)的优化 会比H(x)简单的多。

这一想法也是源于图像处理中的残差向量编码，通过一个reformulation，将一个问题分解成多个尺度直接的残差问题，能够很好的起到优化训练的效果。

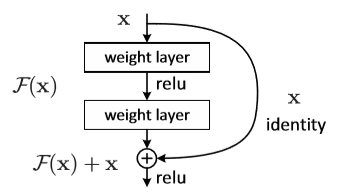
网络的退化问题使得更深的网络错误率比浅网络要高。求解器难以通过多个非线性层接近恒等映射，利用残差网络则很容易，如果恒等映射最优，只需将权重趋向0即可。如果恒等映射不是最优，只要最优函数本身更接近恒等映射而不是零映射，solver参考恒等映射来学习要比把该函数当成新的映射来学习要容易。





3.2 Identity Mapping by Shortcuts

通过shortcuts实现恒等映射



有两层网络，x是第一层的输入，为简化计算，忽略偏差b。F = W2\*g(W1x)是第二层网络的输出，现使用快捷连接（shortcuts connection），把第一层的输入连到第二层的输出，使得新的输出y = F + x，之后再使用ReLU函数。



这个公式并没有引入额外参数也没有增加计算复杂度。

X和F的维度必须相同才可以直接相加，如果维度不同，则可以增加一个线性投影（projection）Ws来使x维度达到要求。



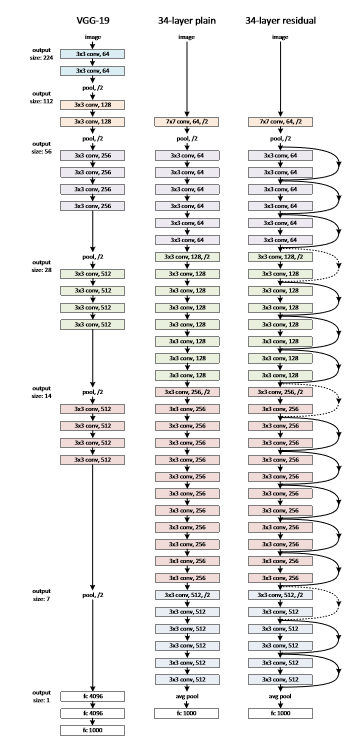
X跳跃的层数不固定，本文中是2-3层，但是如果只跳跃一层的话，没有观察到提升效果

Projection Shortcuts

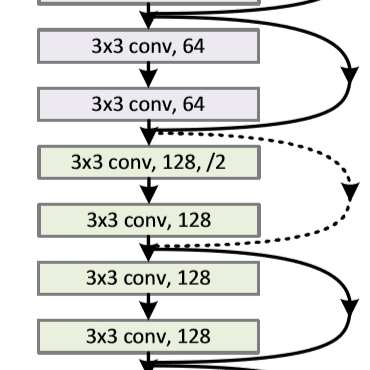
3.3 network architecture

本文设计了Plain Network和Residual Network两种网络结构作对比。

Plain Network参考了VGGNet，设计规则为：对于相同的输出特征图尺寸，层具有相同数量的filter；如果特征图尺寸减半，则filter数量加倍。网络通过步长为2的卷积层直接进行下采样，最后是平均池化和带有softmax的1000路全连接层。



Residual Network中引入快捷连接，如果输入输出尺寸相同，则直接连接（公式1），如果维度不同，有两种方式：一是shortcuts仍然执行恒等映射，额外的维度用0填充；二是使用公式2的方法，利用一个矩阵Ws。



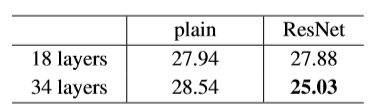
4.Experiments

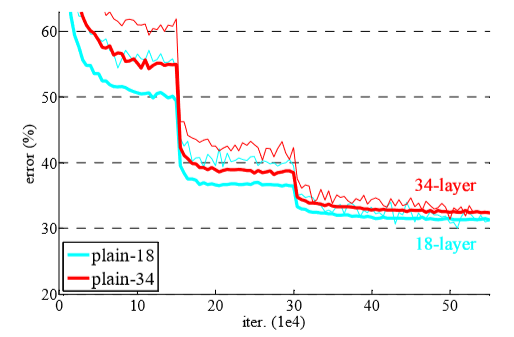
4.1 ImageNet Classfication

Plain Network

评估18层和34层的普通网络

34层网络的验证错误要高于比他浅的18层网络





文中认为这种优化困难不太可能是由于梯度消失造成的。推测深度普通平网可能具有指数级低的收敛速度，影响训了练误差的减少。

Residual Network

接下来，我们评估18层和34层残留网（ResNets）。 基线架构与上述普通网络相同，期望在每对3×3滤波器中添加快捷连接，如图3（右）所示。 在第一次比较中（表2和图4右），我们对所有快捷方式使用标识映射，为增加维度使用零填充（optionA）。因此，与普通对应项相比，它们没有额外的参数。

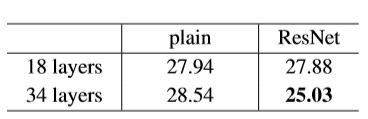
我们从表2和图4中得到了三个主要观察结果。首先，情况与剩余学习相反--34层ResNet优于18层ResNet（2.8％）。 更重要的是，34层ResNet表现出相当低的训练误差，并且可以推广到验证数据。 这表明在该设置中很好地解决了退化问题，并且我们设法从增加的深度获得准确性增益。

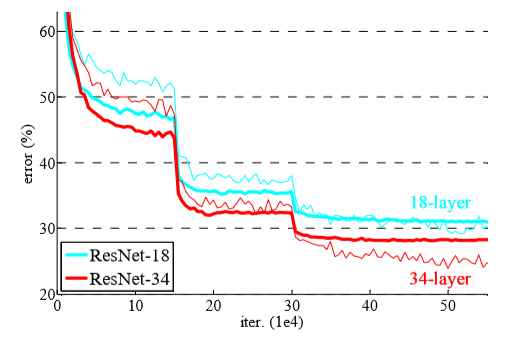
第一次比较中，使用恒等映射，增加维度使用第一种选项，零填充，不增加额外的参数。

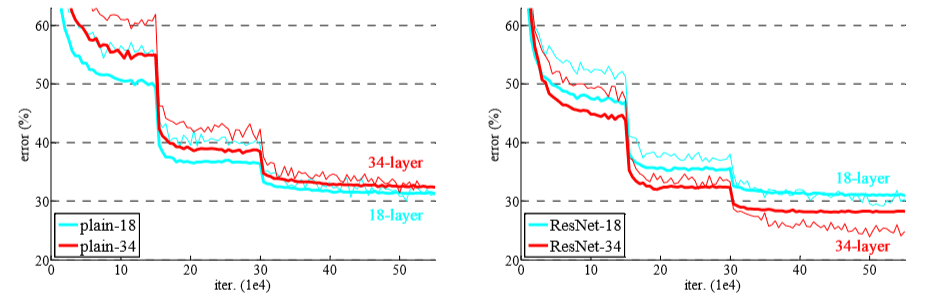
结果表明34层的结果优于18层1结果。3层表现出相当低的训练误差，表明该设置解决了退化问题。

同时横向对比，34层的ResNet比PlainNet效果要好，得益于训练误差的减少，这个对比证实了残差学习在极深系统中的有效性

3.图中表明，18层网络已经相对精确，但是18层ResNet收敛更快，对于不少那么深的网络来说，ResNet可以简化最优化







Identity VS. Projection Shortcuts

此部分评估Projection Shortcuts的效果，即公式2，在输入x上附加新的权重矩阵Ws。图3比较了三种方式

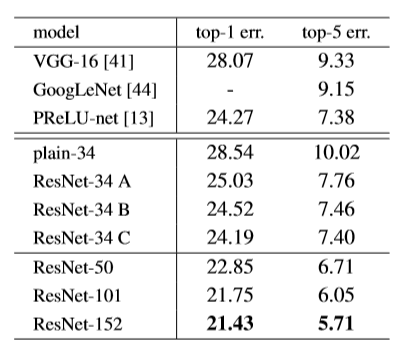
1. 使用零填充增加维度，所有shortcuts都不增加参数

B. 在增加维度时使用Projection Shortcuts，其他地方使用Identity Shortcuts

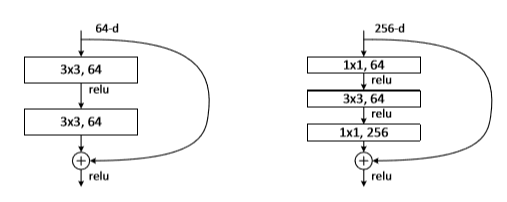
C. 所有的连接都使用Projection Shortcuts

表3表明三种方式的ResNet都比对应的普通网络效果要好

B比A好一点，C比B好一点，但三者之间区别很小，故之后不再使用C，来减少时间/存储复杂度和模型大小



Deeper Bottleneck Architectures



使用三层卷积（1×1,3×3,和1×1）替换之前的二层卷积（3×3和3×3），1×1卷积用来减少维度，从而减少训练时间。两种设计时间复杂度相似

Identity shortcuts对于这种bottleneck结构很重要，使用projection shortcuts的话相当于连接在了高维度的两端，会增加复杂度和模型大小

50-layer ResNet

使用bottleneck结构，把每两个卷积层换成三层，得到50层的ResNet，使用方式B增加维度。

101层和152层ResNet

使用更多的三层结构。尽管深度大大增加，但仍然没有VGG16/19复杂

50/101/152层的效果都比34层要好，没有发生退化问题