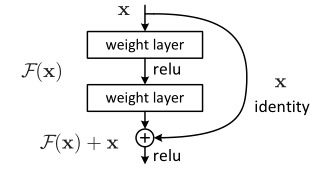
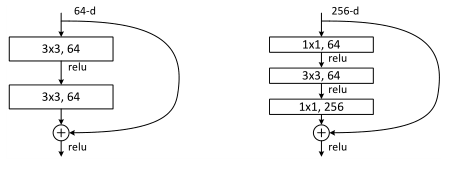
**[Deep Residual Learning for Image Recognition] CVPR2016 何凯明**

**主要思想：** F(x)=H(x)-x。ResNet学习的是残差方程F(x)，而不是基本映射H(x),因为F(x)更容易优化。通过**shortcut connection**，shortcut将block的输入和输出进行一个**element-wise的加叠**，这个简单的加法并不会给网络增加额外的参数和计算量，同时增加模型的训练速度、收敛更快，并且当模型的层数加深时，这个简单的结构能够很好的解决退化问题。

**贡献/缺点：**深度学习的模型有越来越深的趋势，但当模型层数增加时，错误率反而是降低的，这是模型的**“退化”问题**。退化的原因是优化难题，当模型变复杂时，SGD的优化变得更加困难。ResNet通过shortcut connection很好的解决退化问题，使**网络能达到更深，优化更好，并不增加计算量**。

**论文细节：**

**网络结构。**ResNet并没有提出一种新的网络结构，而是引入了一种深度残差学习的框架**F(x)=H(x)-x**。之前的网络学习是采用堆叠的网络来逼近基本映射方程H(x), 即输入x，输出是H(x)。残差学习中，将基本映射H(x)看做是两部分x+F(x)，x是**恒等映射**，F(x)是**残差方程**。深层的网络模拟的H(x)面临优化问题导致网络退化，残差网络的解决办法是不再求解H(x), 而是用网络模拟残差方程F(x)，F(x)加上输入的恒等映射x就可以得到H(x)。两种的方式的功能是一样的，但后者的训练难度明显是小的。实验表明，残差网络的优化更容易，且计算量不增反降，这意味着网络可以做的更深的同时不需要担心计算量暴增和网络退化的问题。

**残差模块**。定义为：y=F(x,wi)+x，其中，x代表输入，y代表输出，F(x,wi)代表需要学习的**残差mapping**。若shortcut两层网络，用F=W2·σ·(W1·x)表示，这里σ表示ReLU激活层。这里Wx是卷积操作，是线性的，ReLU是非线性的。其中x和F的维度一定要相同，如果不同的话，可以通过一个线性映射Ws来匹配维度：y=F(x,Wi)+Ws·x，这里F是比较灵活的，可以包含两层或者三层，甚至更多层。但是如果只有一层的话，就变成了y=Wi·x+x，这就是普通的线性函数了，就没有意义了。

**收敛和优化的解释。**假如我们需要的理想的mapping定义为H(x)，那么我们新加的非线性层就是F(x):=H(x)−x，原始的mapping就从x变成了F(x)+x。也就是说，如果我们之前的x是最优的，那么新加的残差F(x)就应该都是0，而不会是其他的值。这样整个残差网络是端对端（end-to-end）的，可以通过**随机梯度下降反向传播**，而且实现起来很简单（实际上就是两层求和）。至于它为什么收敛更快，error更低，有人这么理解：随机梯度下降是用链式求导法则，我们对H(x)求导，相当于对F(x)+x求导，那么这个梯度值就会在1附近(x的导数是1)，相比之前的plain网络，自然收敛更快。

**跨层连接。**Residual block中x的映射通过shortcut connection实现，通过shortcut将这个block的输入x和输出F(x)进行一个element-wise的加叠便可以得到H(x)，这个简单的加法并不会给网络增加额外的参数和计算量。Shortcut的方式作者提出三个选项：A)使用**恒等映射**，如果residual block的输入输出维度不一致，对增加的维度用0来填充;B)在block输入输出维度一致时使用恒等映射，不一致时使用**线性投影**以保证维度一致;C)对于所有的block均使用线性投影。实验发现虽然C好于B好于A的效果，但是差距很小，因此线性投影并不是必需，而使用0填充时，可以保证模型的复杂度最低，这对于更深的网络是更加有利的。

**Deeper Residual Block。**Shortcut跨层连接可以跨过两层，也可以跨过三层，好处是网络可以做的更深。ResNet的经典层数是ResNet50, ResNet101, ResNet152, 作者甚至探索了魔鬼般的ResNet1202。

ResNet的所有精髓：<https://blog.csdn.net/justpsss/article/details/77103077>

翻译：https://blog.csdn.net/Angle\_Cal/article/details/58025368