

法律声明

■ 本课件包括演示文稿、示例、代码、题库、视频和声音等内容，北风网和讲师拥有完全知识产权；只限于善意学习者在本课程使用，不得在课程范围外向任何第三方散播。任何其他人或者机构不得盗版、复制、仿造其中的创意和内容，我们保留一切通过法律手段追究违反者的权利。

■ 课程详情请咨询

◆ 微信公众号：北风教育

◆ 官方网址：<http://www.ibeifeng.com/>



人工智能之深度学习

卷积神经网络(CNN)

主讲人：Vincent

上海育创网络科技有限公司



课程要求

■ 课上课下 “九字” 真言

- ◆ 认真听，善摘录，勤思考
- ◆ **多温故，乐实践**，再发散

■ 四不原则

- ◆ **不懒散惰性，不迟到早退**
- ◆ **不请假旷课，不拖延作业**

■ 一点注意事项

- ◆ 违反 “四不原则” ， 不包就业和推荐就业

严格是大爱



寄语



做别人不愿做的事，
做别人不敢做的事，
做别人做不到的事。

课程内容

- 一、卷积神经网络
 - ◆ 层次结构
 - ◆ 数据处理
 - ◆ 训练算法
 - ◆ 优缺点
- 二、正则化与Dropout
- 三、CNN典型的结构与训练方式
- 四、数据增强
- **五、CNN经典网络结构**
- 六、CNN迁移学习

- 问题：由于梯度消失，深层网络很难训练。因为梯度反向传播到前面的层，重复相乘可能使梯度无穷小。结果就是，随着网络的层数更深，其性能趋于饱和，甚至迅速下降。
- 如下图：在cifar-10上的测试，深层CNN('plain' network)性能均比浅层的错误率要高。

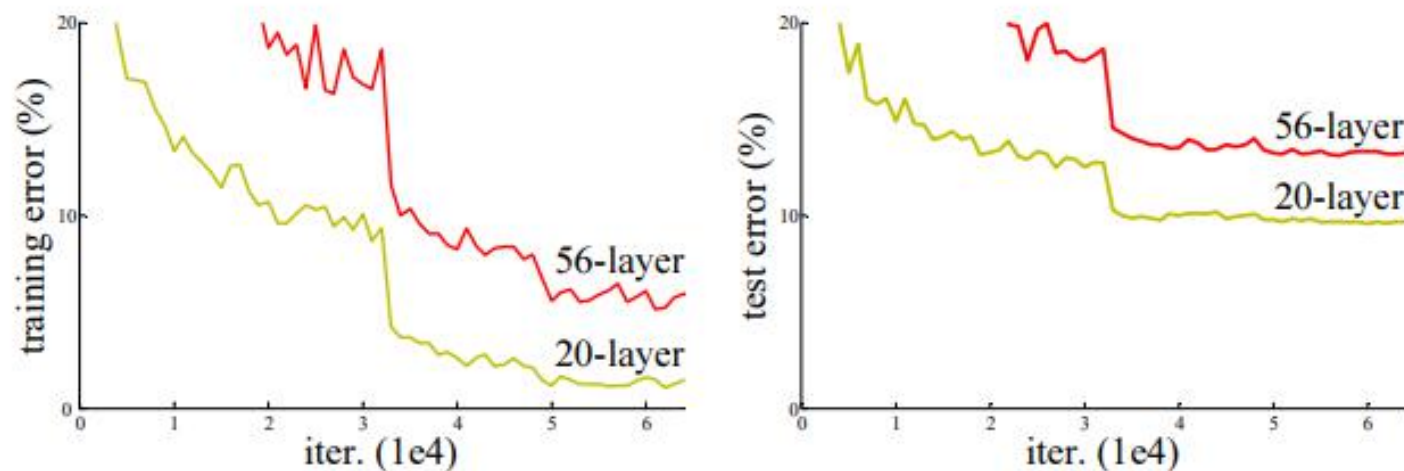
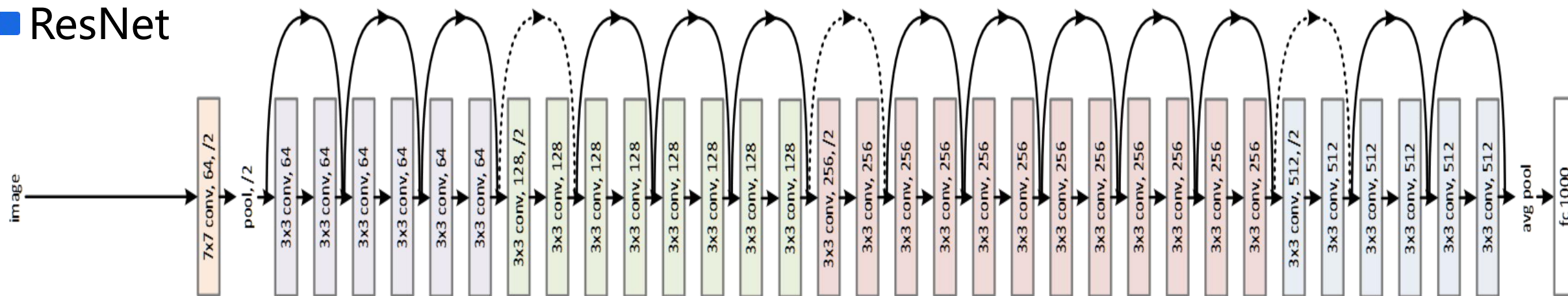


Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer “plain” networks. The deeper network has higher training error, and thus test error. Similar phenomena on ImageNet is presented in Fig. 4.

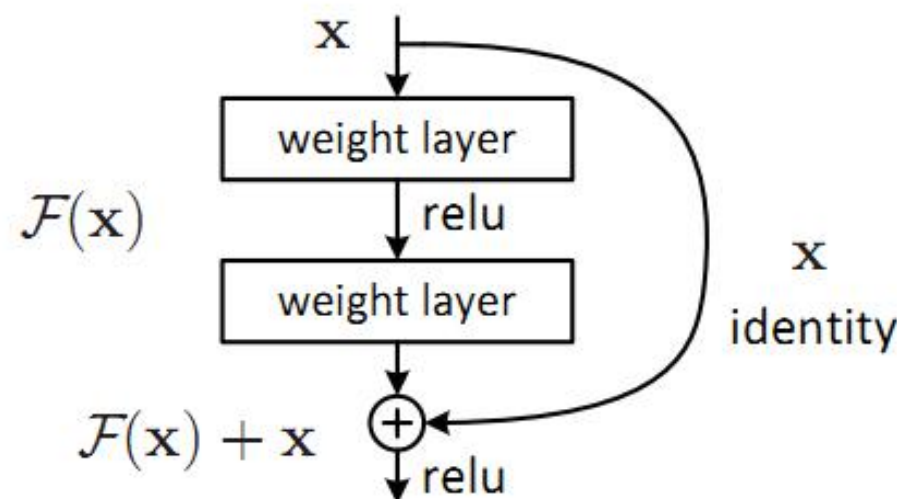
卷积神经网络典型CNN-ResNet

■ ResNet

34-layer residual



- ResNet 的核心思想是引入一个「恒等快捷连接」 (identity shortcut connection) , 直接跳过一个或多个层.
- 这种思想, 类似于LSTM中门开关, ResNet 可以认为是门开关的一种特殊情况.



两种 identity *Shortcut Connection*方式

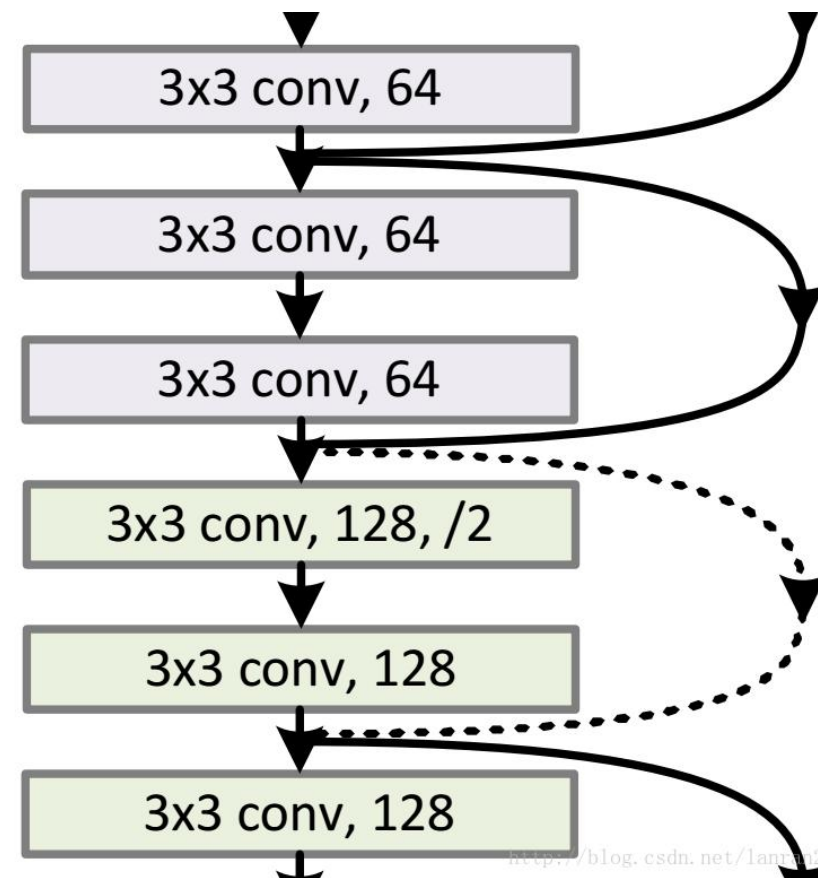
- 实线的Connection部分(“第一个粉色矩形和第三个粉色矩形”)都是3x3x64的特征图，他们的**channel个数一致**，所以采用计算方式：

$$y = F(x) + x$$

- 虚线的Connection部分(“第一个绿色矩形和第三个绿色矩形”)分别是3x3x64和3x3x128的特征图，他们的**channel个数不同(64和128)**，所以采用计算方式：

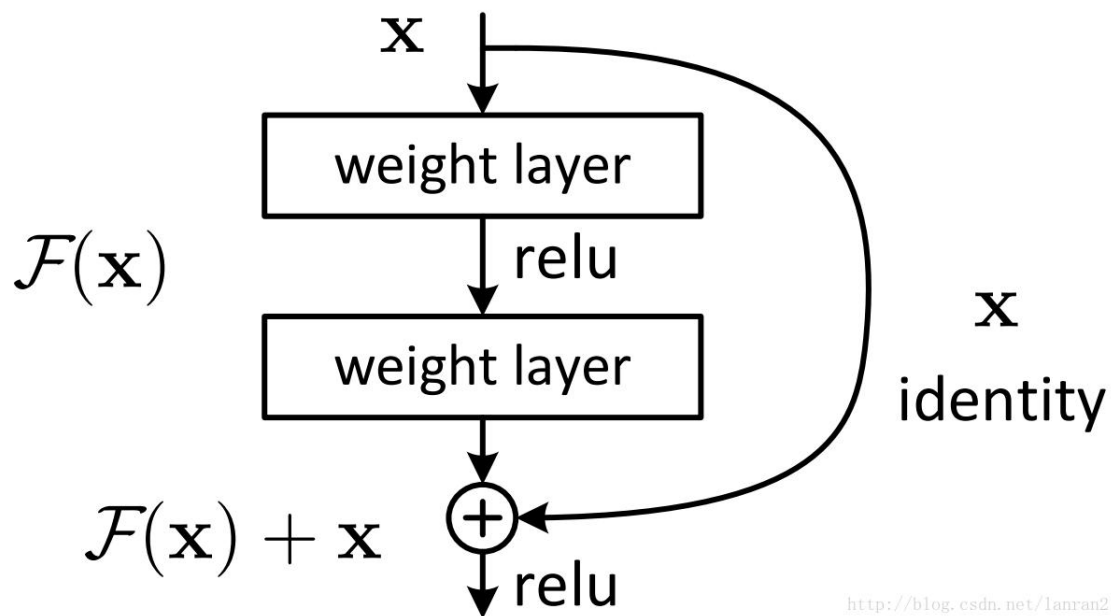
$$y = F(x) + xW$$

- 其中W是卷积操作，用来调整x的channel维度的。

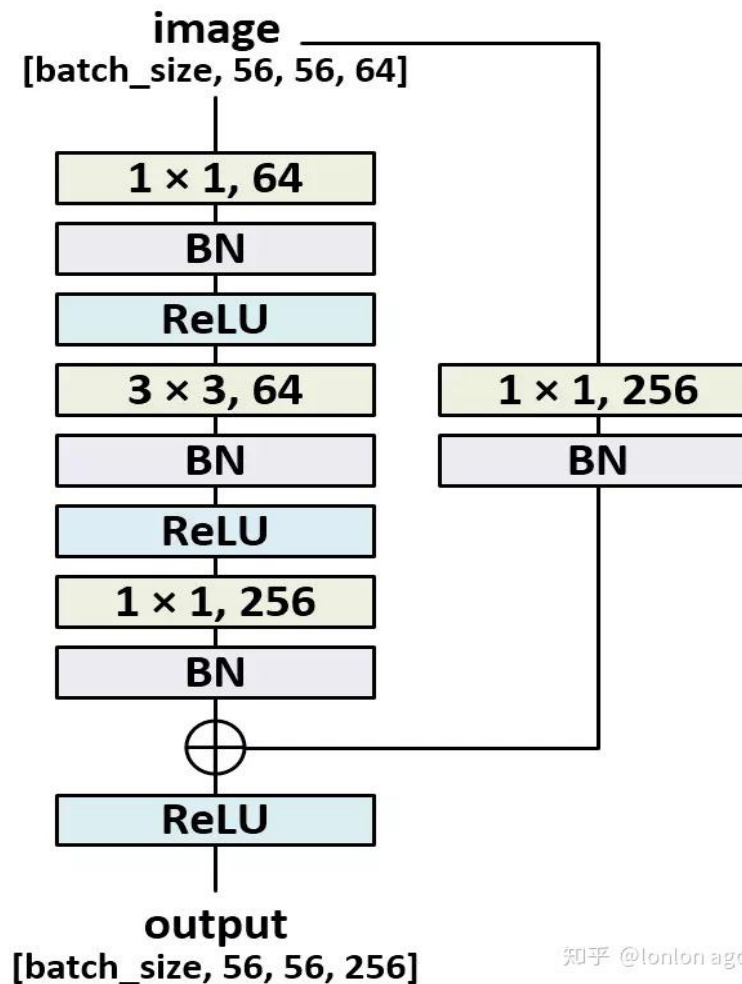


ResNet--identity shortcut connection

- 左图：若 X 和 $F(x)$ 输出通道数（且高宽）一致，则直接相加；
- 右图：若不一致（仅仅是通道数量不一致），方法一：使用 1×1 conv来调整 X 的shortcut输出通道；方法二：使用填充0补齐。



<http://blog.csdn.net/lanran2>



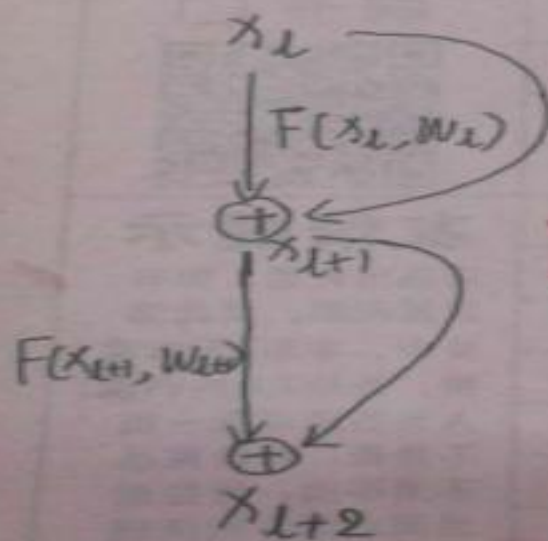
ResNet

$$x_{l+2} = x_{l+1} + F(x_{l+1}, w_{l+1})$$

$$= \overbrace{x_l + F(x_l, w_l) + F(x_{l+1}, w_{l+1})}^{\text{ResNet}}$$

$$= x_{l-1} + F(x_{l-1}, w_{l-1}) + F(x_l, w_l) + F(x_{l+1}, w_{l+1})$$

$$\Rightarrow x_l = x_0 + \sum_{i=0}^{l-1} F(x_i, w_i)$$



① 无 shortcut 时

$$\frac{\partial \text{Loss}}{\partial x_l} = \frac{\partial \text{Loss}}{\partial x_{l+2}} \cdot \frac{\partial x_{l+2}}{\partial x_{l+1}} \cdot \frac{\partial x_{l+1}}{\partial x_l}$$

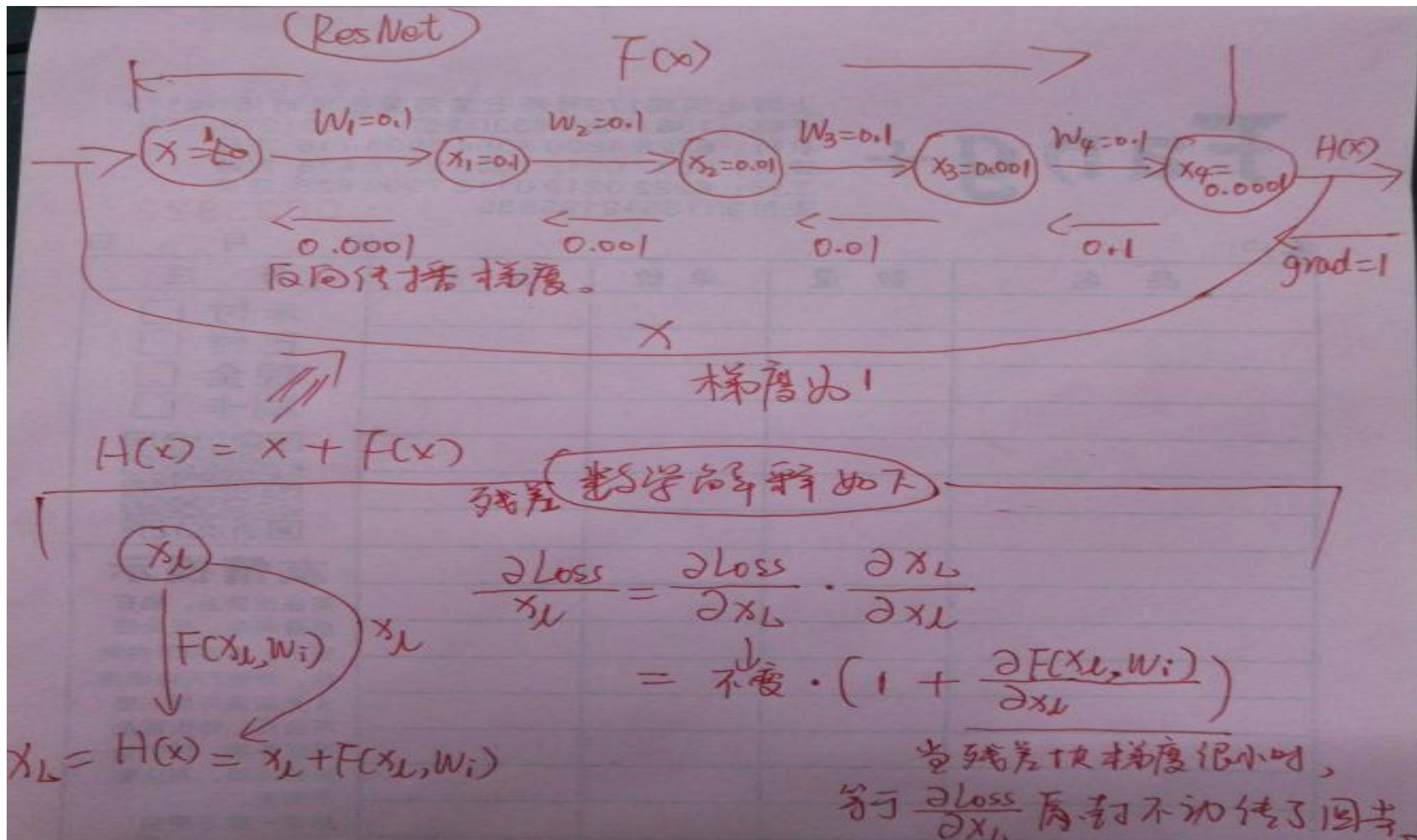
② 有 shortcut 时

$$\frac{\partial \text{Loss}}{\partial x_l} = \frac{\partial \text{Loss}}{\partial x_{l+2}} \cdot \frac{\partial (x_l + F(x_l, w_l) + F(x_{l+1}, w_{l+1}))}{\partial x_l}$$

$$= \frac{\partial \text{Loss}}{\partial x_{l+2}} \cdot \left[1 + \frac{\partial}{\partial x_l} \sum_{i=l}^{l+1} F(x_i, w_i) \right]$$

$$= \frac{\partial \text{Loss}}{\partial x_{l+2}} + \frac{\partial \text{Loss}}{\partial x_{l+2}} \cdot \frac{\partial}{\partial x_l} \sum_{i=l}^{l+1} F(x_i, w_i)$$

残差容易学习的数学解释



ResNet和vgg的对比

- 一、VGG19结构
- 二、34-layer plain(vgg加深)
- 三、上述34-layer plain+残差结构 (identity shortcut connection)

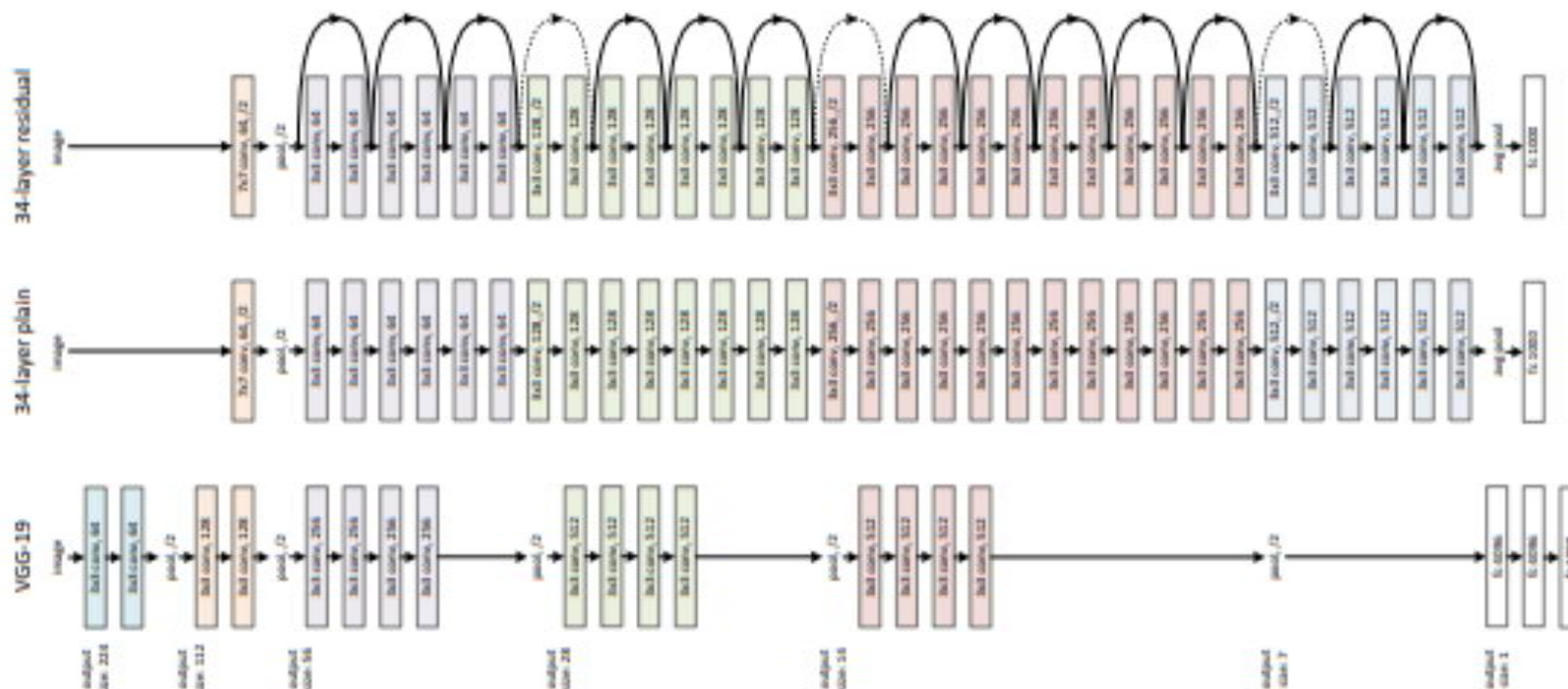


Figure 3. Example network architectures for ImageNet. **Left:** the VGG-19 model [41] (19.6 billion FLOPs) as a reference. **Middle:** a plain network with 34 parameter layers (3.6 billion FLOPs). **Right:** a residual network with 34 parameter layers (3.6 billion FLOPs). The dotted shortcuts increase dimensions. **Table 1** shows more details and other variants.

- 左图：plain结构错误率。 右图：残差结构错误率
- 结果：残差结构错误率显著优于plain结构。

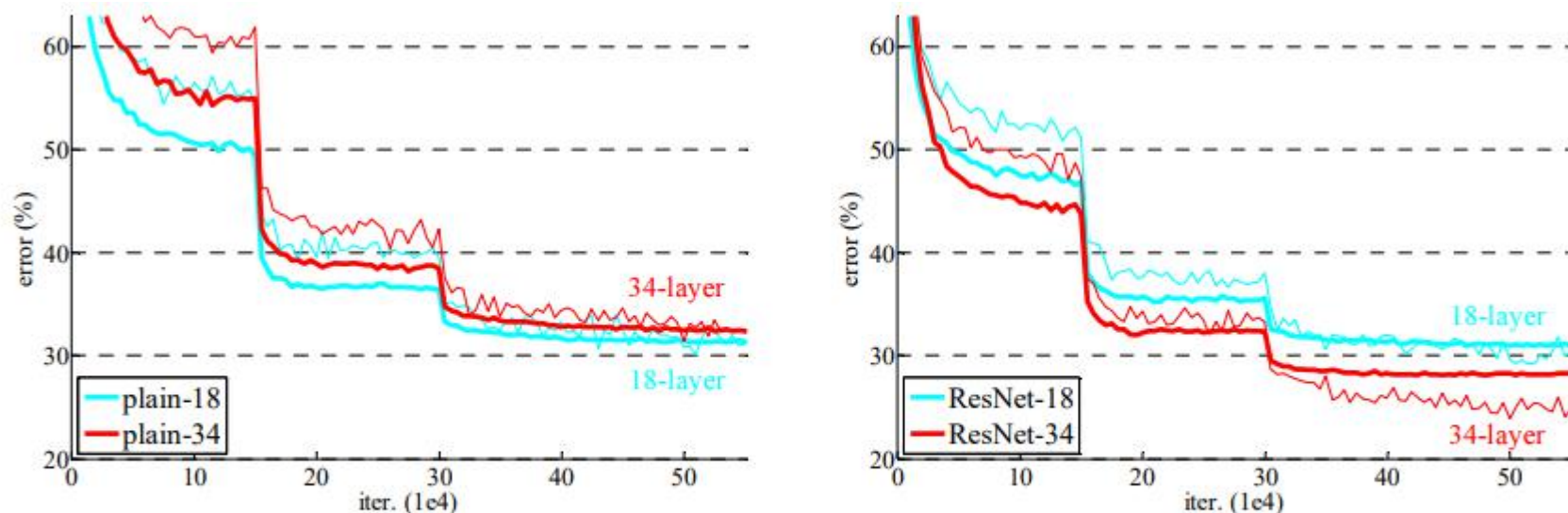


Figure 4. Training on **ImageNet**. Thin curves denote training error, and bold curves denote validation error of the center crops. Left: plain networks of 18 and 34 layers. Right: ResNets of 18 and 34 layers. In this plot, the residual networks have no extra parameter compared to their plain counterparts.

- ResNet中提出了一种bottleneck的结构块来代替常规的残差块，它借鉴了Inception网络中1x1 conv来缩减或扩张feature map维度
- 目的：不降低模型精度的前提下，降低参数数量和计算量。

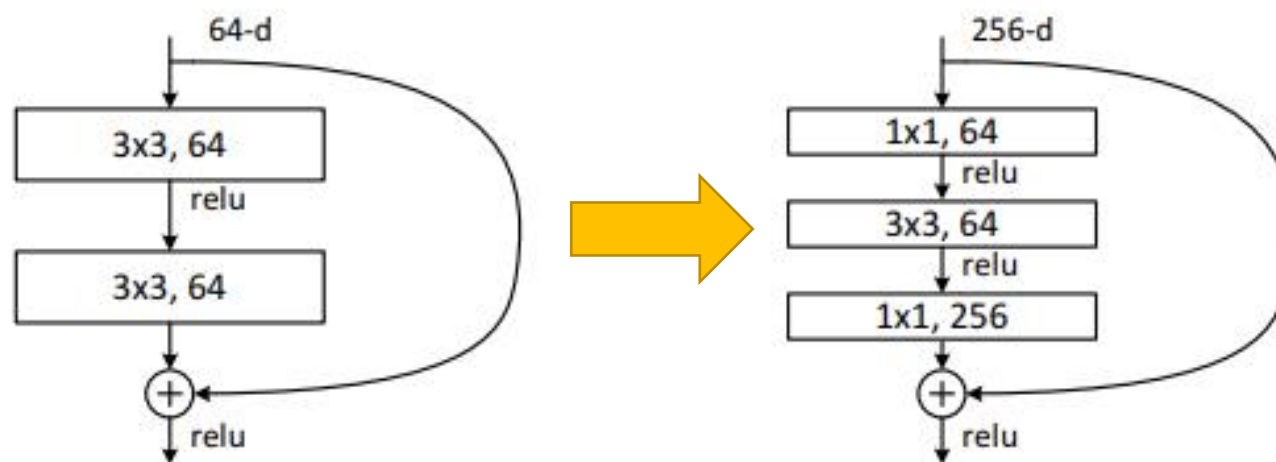


Figure 5. A deeper residual function \mathcal{F} for ImageNet. Left: a building block (on 56×56 feature maps) as in Fig. 3 for ResNet-34. Right: a “bottleneck” building block for ResNet-50/101/152.

ResNet模型结构图

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

resnet层数的计算--有权重的才算1层

101、152层的ResNet，而且不仅没有出现退化问题，错误率也大大降低，同时计算复杂度也保持在很低的程度。

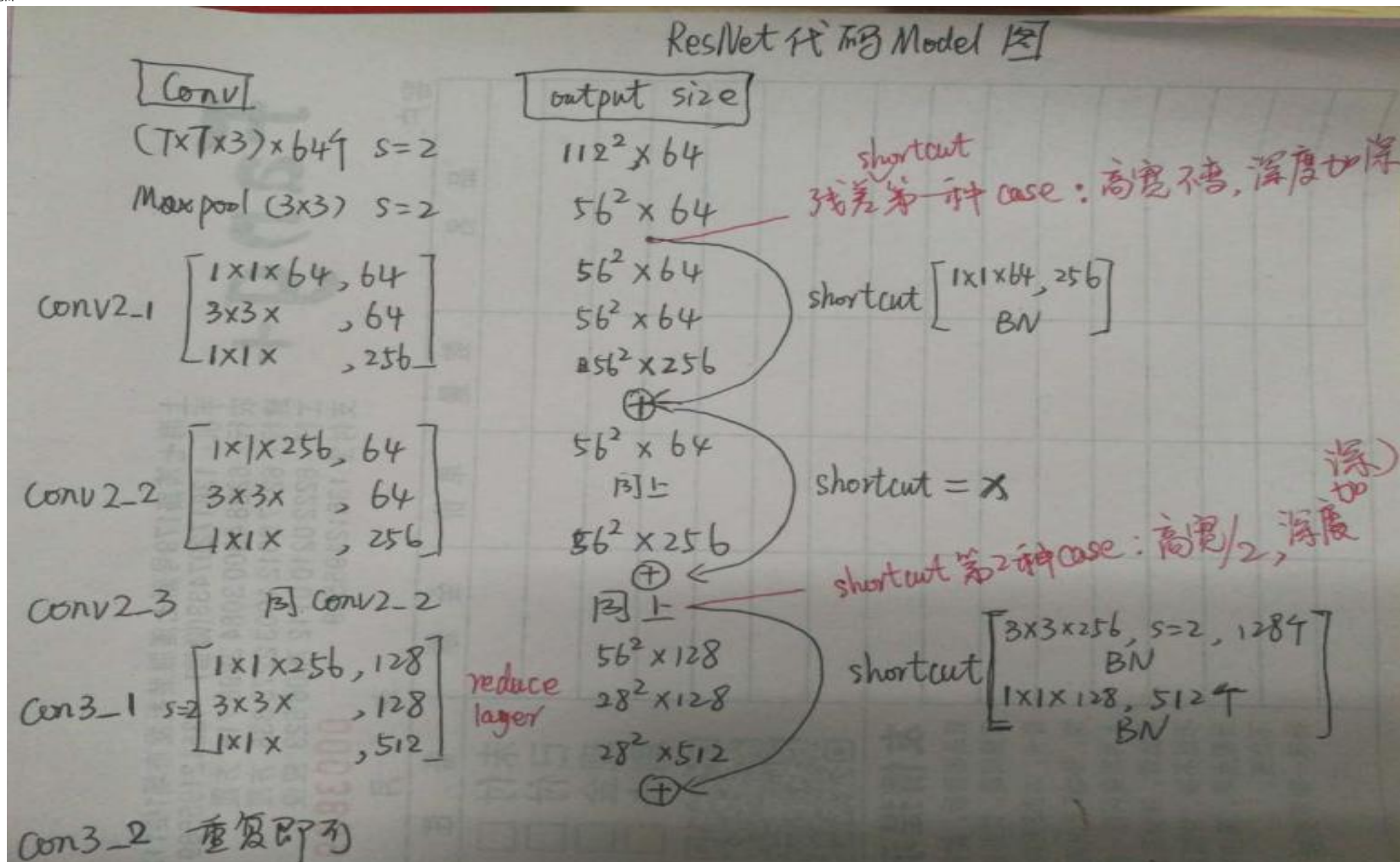
ResNet在不同层数时的网络配置，其中基础结构很类似，都是前面提到的两层和三层的残差学习单元的堆叠。

层计算

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

ResNet在ImageNet上的结果:

= 50层





THANK YOU

上海育创网络科技有限公司