

常见CNN网络及深度学习平台

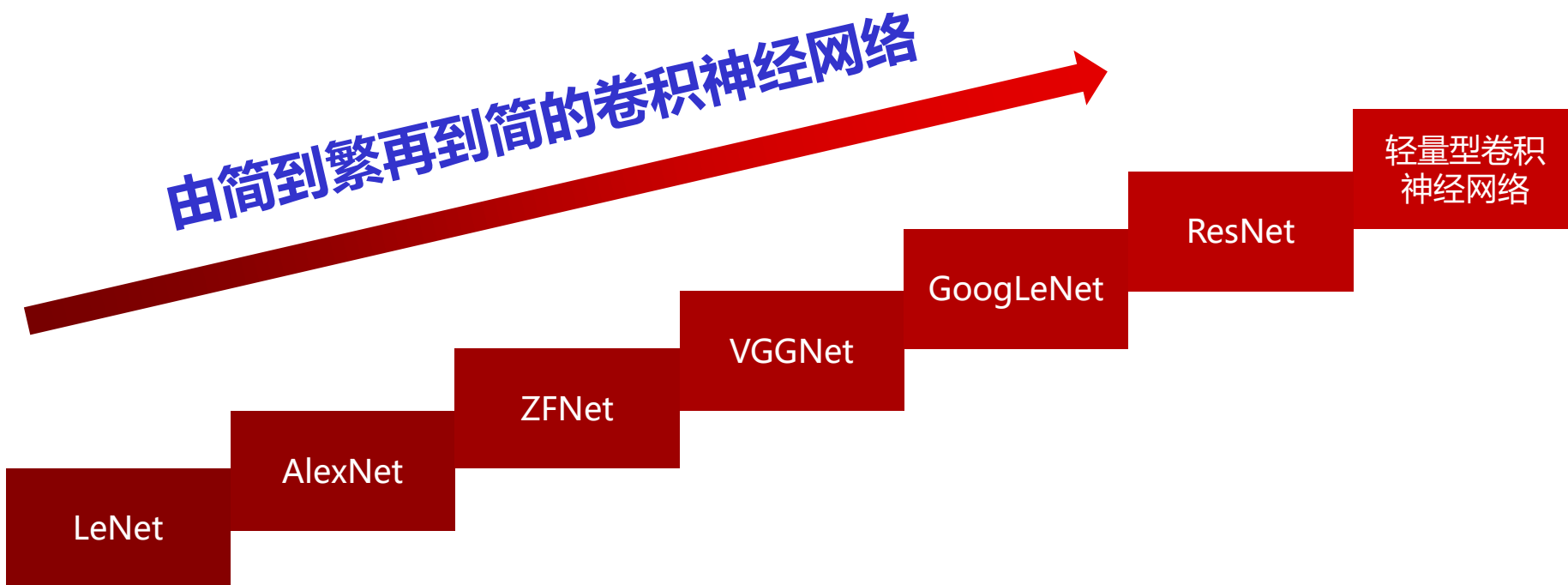
1

一、常见CNN网络

二、深度学习平台

常见CNN网络

2



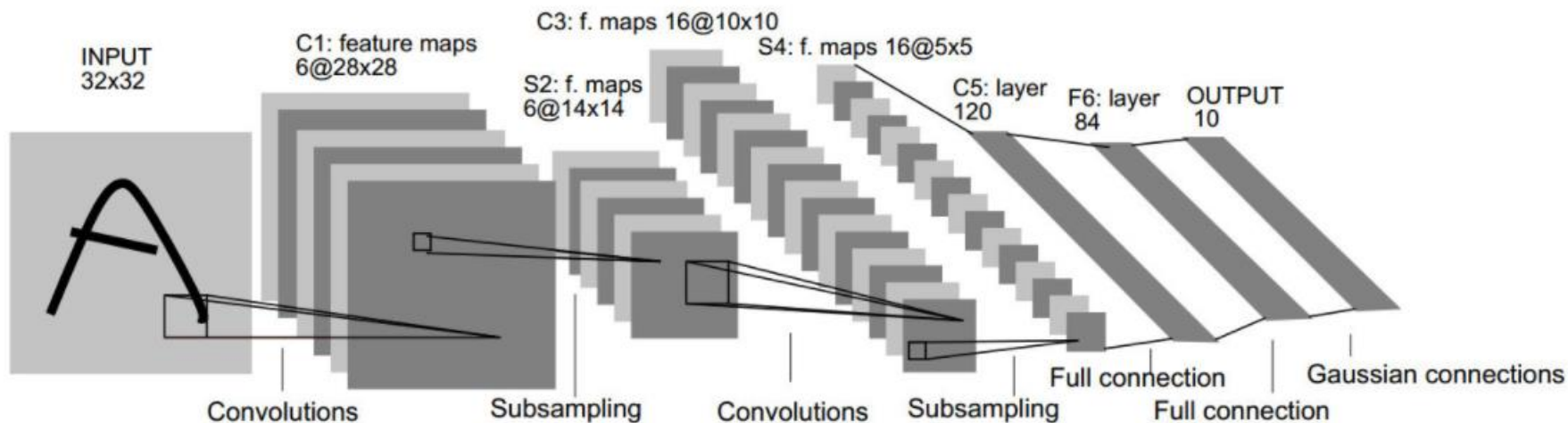
常见CNN网络

3

1. LeNet, 1998

LeNet是卷积神经网络的开山鼻祖，最初用于解决手写数字识别任务。

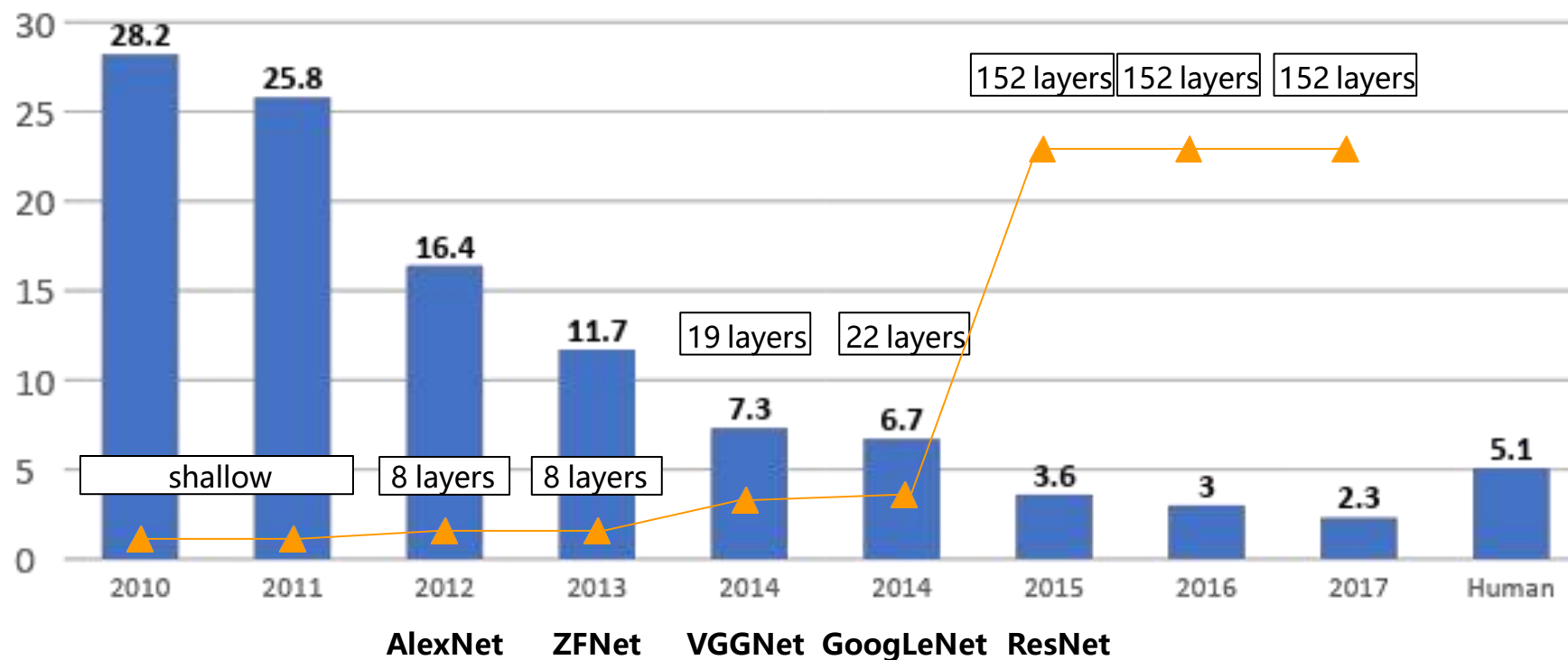
- 定义了CNN的基本组件：卷积层、池化(下采样)层、全连接层
- 如今使用的是简化改进过的LeNet-5(包含5层)，将激活函数由Tanh改为ReLU



常见CNN网络

4

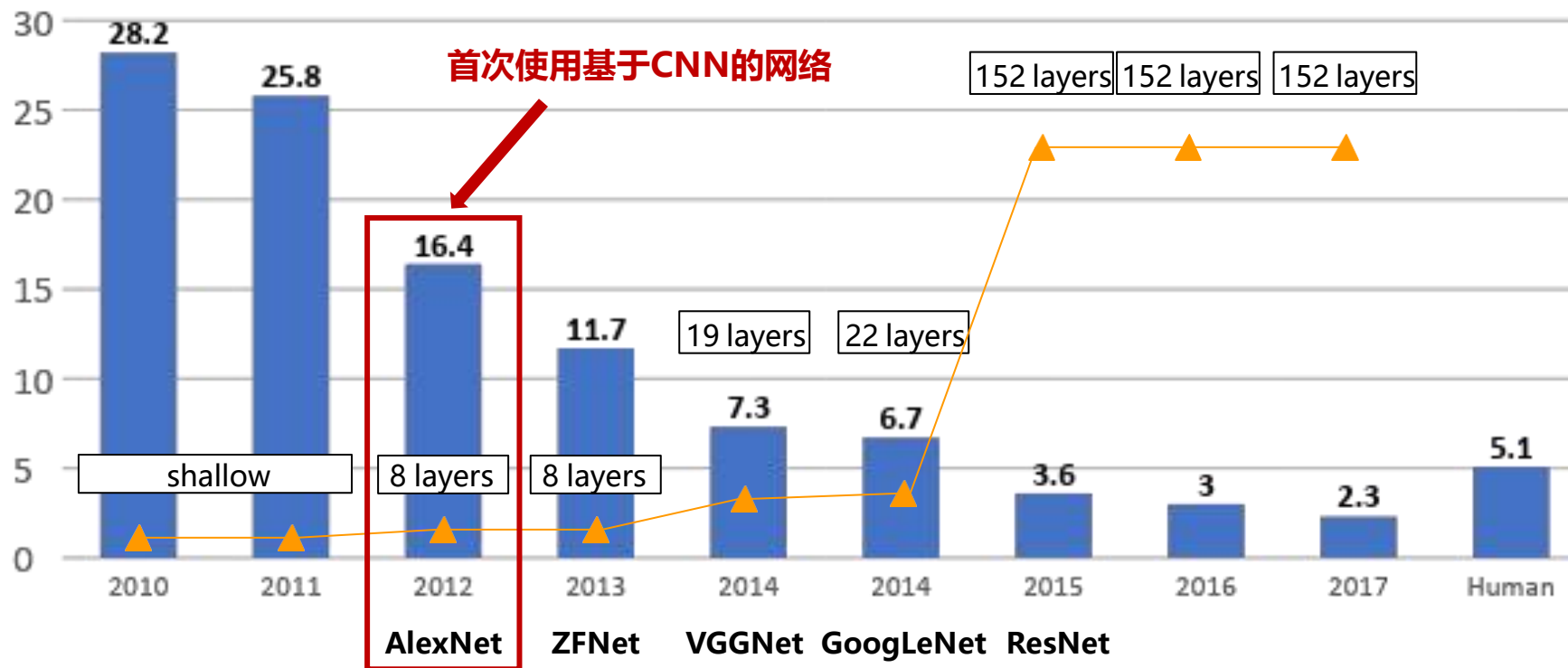
ImageNet Large Scale Visual Recognition Challenge(ILSVRC) winners



常见CNN网络

5

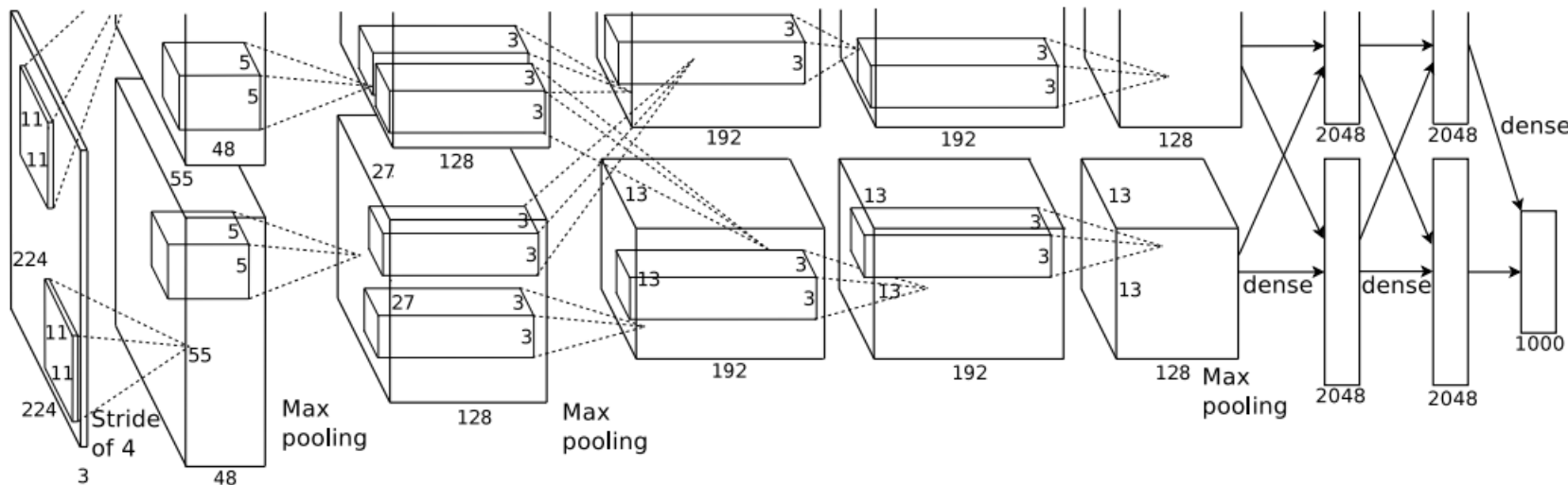
ImageNet Large Scale Visual Recognition Challenge(ILSVRC) winners



2. AlexNet, 2012

AlexNet在2012年ILSVRC中以超过第二名10.9个百分点的绝对优势一举夺冠，掀起了深度学习的研究热潮。

- 使用ReLu作为激活函数，增强模型的非线性表达能力，成为以后卷积层的标配
- 使用Dropout来缓解过拟合，成为以后全连接层的标配
- 引入标准化层LRN(Local Response Normalization)，通过局部响应归一来降低错误率
- 引入数据增广策略，增强模型泛化能力



常见CNN网络

7

2. AlexNet, 2012

CONV1

MAX POOL1

NORM1

CONV2

MAX POOL2

NORM2

CONV3

CONV4

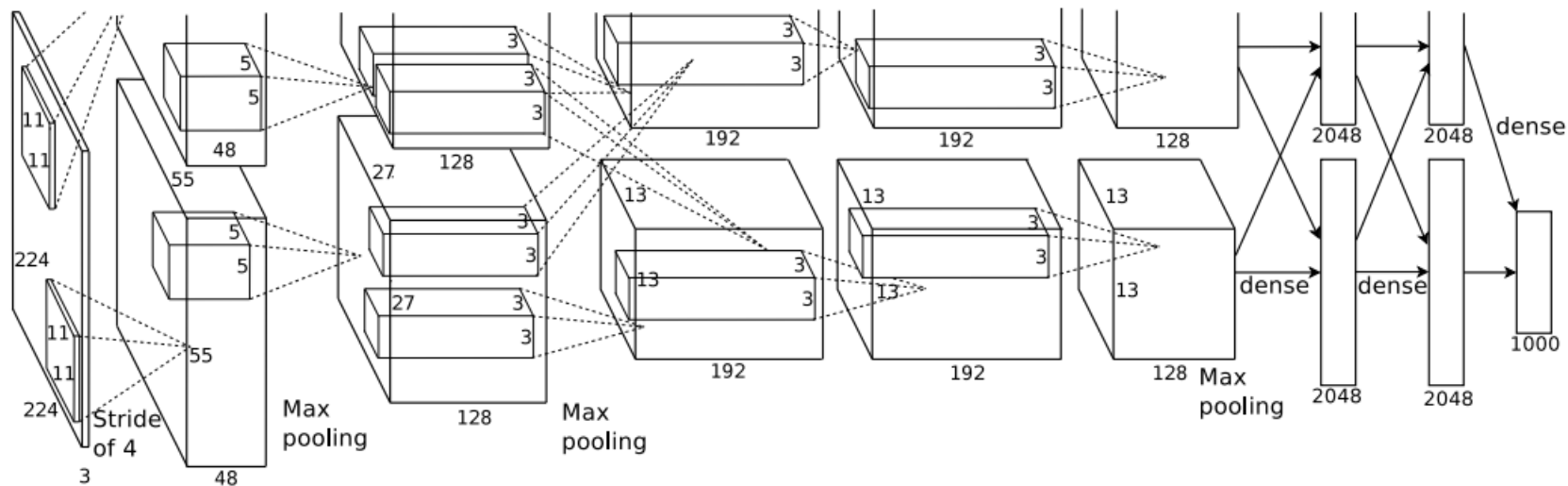
CONV5

Max POOL3

FC1

FC2

FC3



Input: 大小为227*227的3通道图像(3*227*227)

CONV1: 卷积核大小为11*11, 步长为4, 数量为96

→输出大小? $96*55*55$, $55=(227-11)/4+1$

→参数量? $(11*11*3)*96=35K + 96$

常见CNN网络

8

2. AlexNet, 2012

CONV1

MAX POOL1

NORM1

CONV2

MAX POOL2

NORM2

CONV3

CONV4

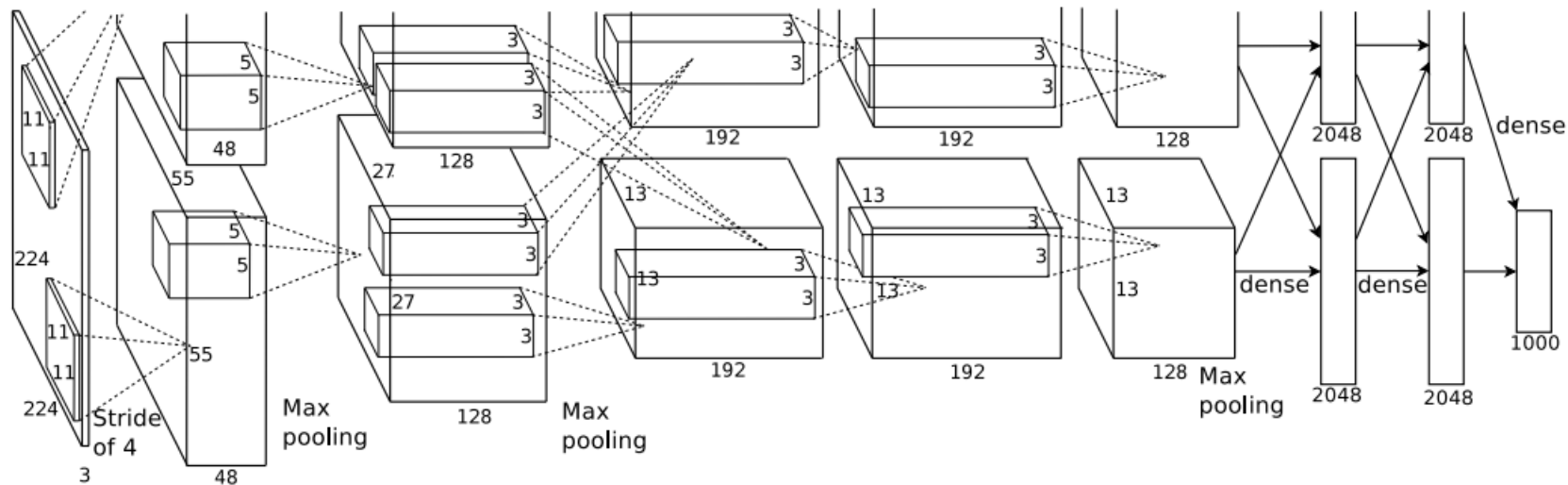
CONV5

Max POOL3

FC1

FC2

FC3



Input: 大小为227*227的3通道图像(3*227*227)

After CONV1: 96*55*55

MAX POOL1: pooling大小为3*3, 步长为2

→输出大小? $96*27*27$, $27=(55-3)/2+1$

→参数量? 0

常见CNN网络

9

2. AlexNet, 2012

CONV1

MAX POOL1

NORM1

CONV2

MAX POOL2

NORM2

CONV3

CONV4

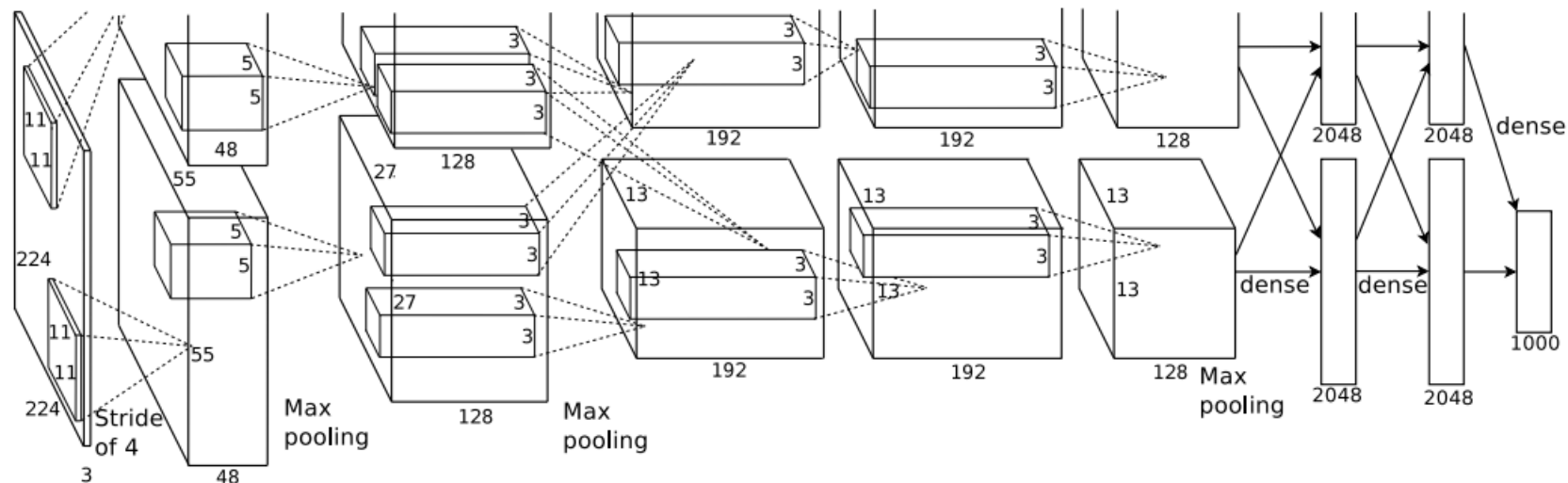
CONV5

Max POOL3

FC1

FC2

FC3



Input: 大小为227*227的3通道图像(3*227*227)

After CONV1: 96*55*55

After MAX POOL1: 96*27*27

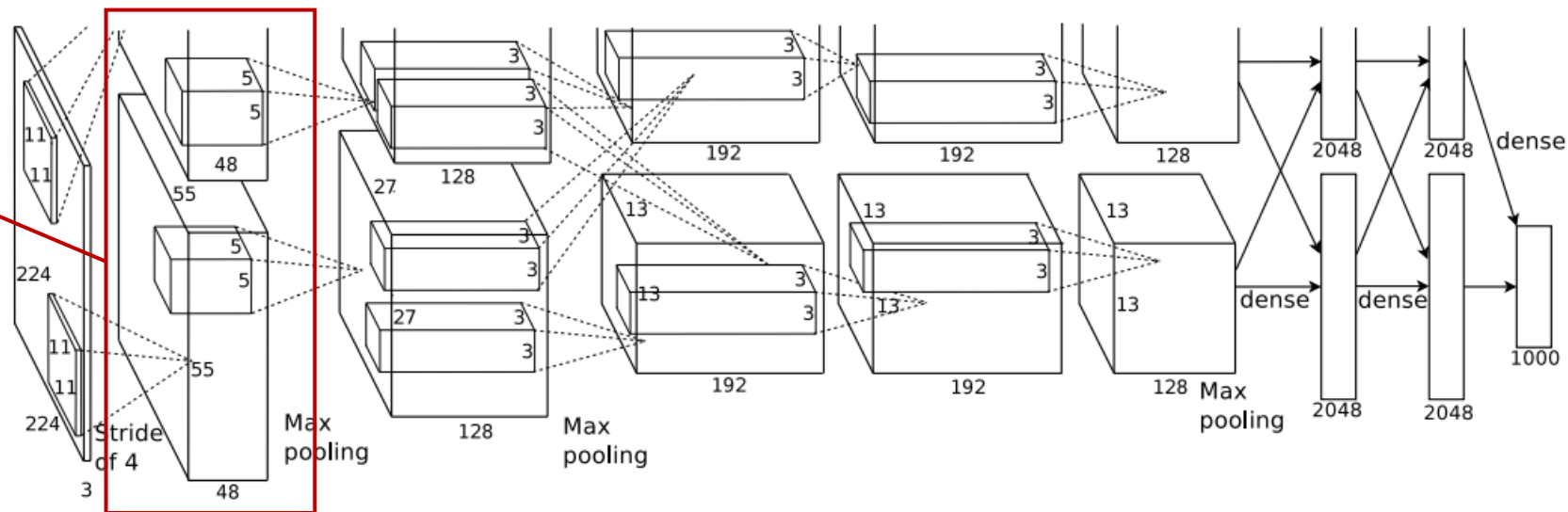
.....

常见CNN网络

10

2. AlexNet, 2012

CONV1
MAX POOL1
NORM1
CONV2
MAX POOL2
NORM2
CONV3
CONV4
CONV5
Max POOL3
FC1
FC2
FC3



常见CNN网络

11

2. AlexNet, 2012

CONV1

MAX POOL1

NORM1

CONV2

MAX POOL2

NORM2

CONV3

CONV4

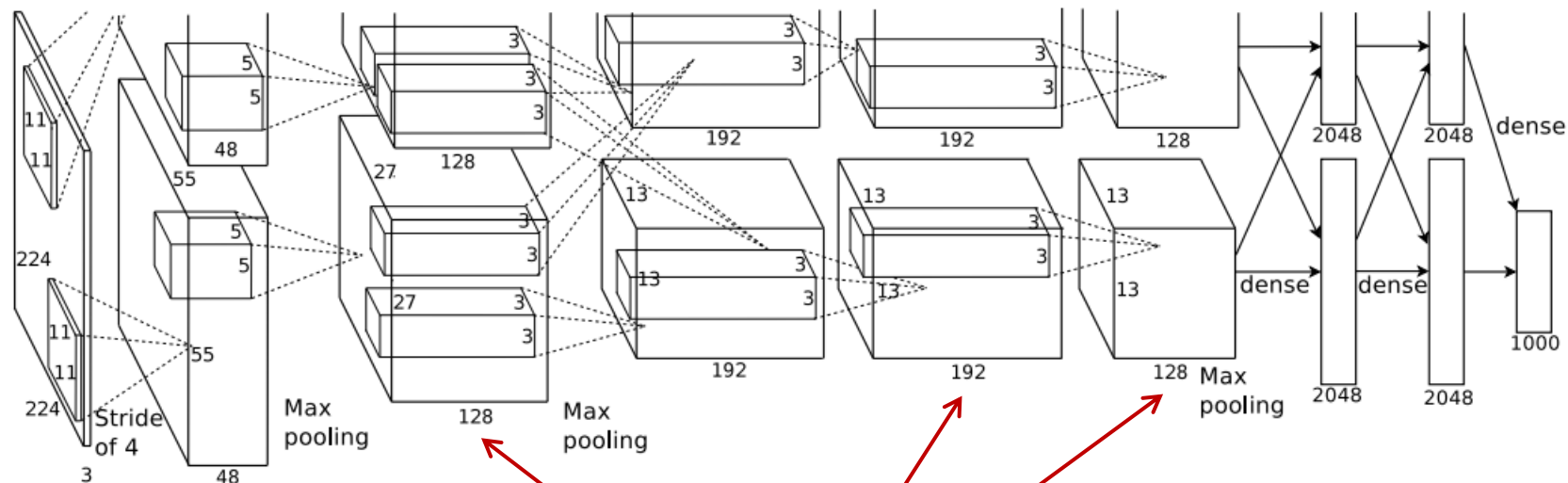
CONV5

Max POOL3

FC1

FC2

FC3



CONV1、CONV2、CONV4、CONV5:
只与同一GPU上的特征图进行连接

常见CNN网络

12

2. AlexNet, 2012

CONV1

MAX POOL1

NORM1

CONV2

MAX POOL2

NORM2

CONV3

CONV4

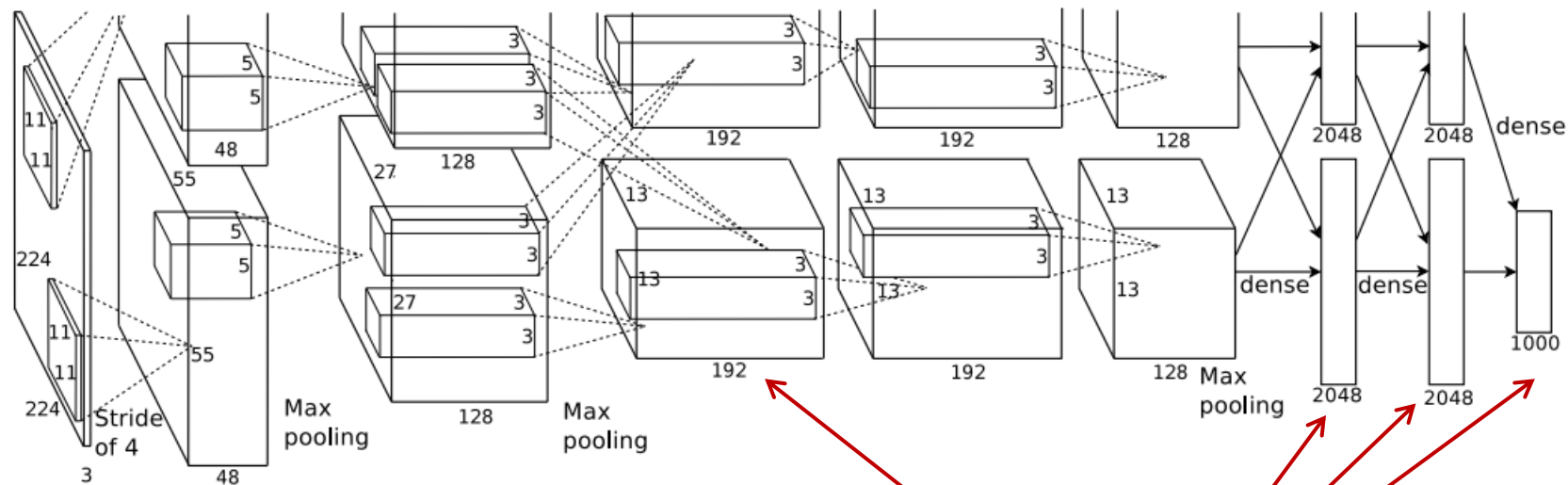
CONV5

Max POOL3

FC1

FC2

FC3

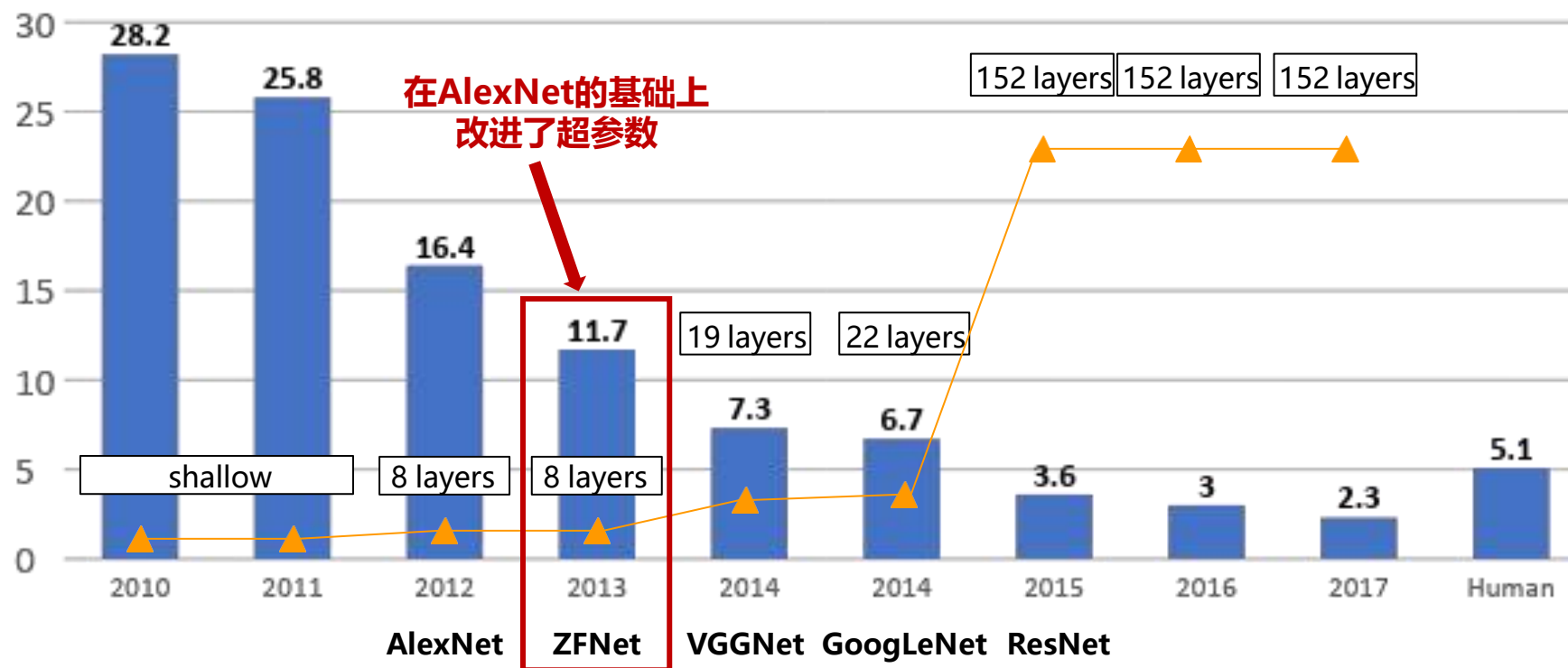


CONV3、FC1、FC2、FC3:
与前层的所有特征图进行连接, 跨GPU通信

常见CNN网络

13

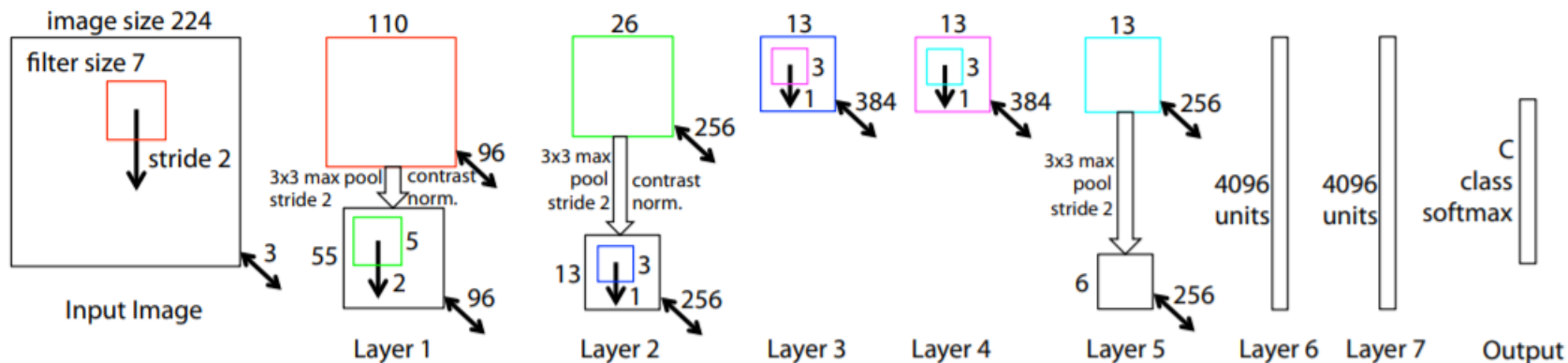
ImageNet Large Scale Visual Recognition Challenge(ILSVRC) winners



3. ZFNet, 2013

ZFNet是2013年ILSVRC分类任务的冠军，在AlexNet的基础上进行了调整。

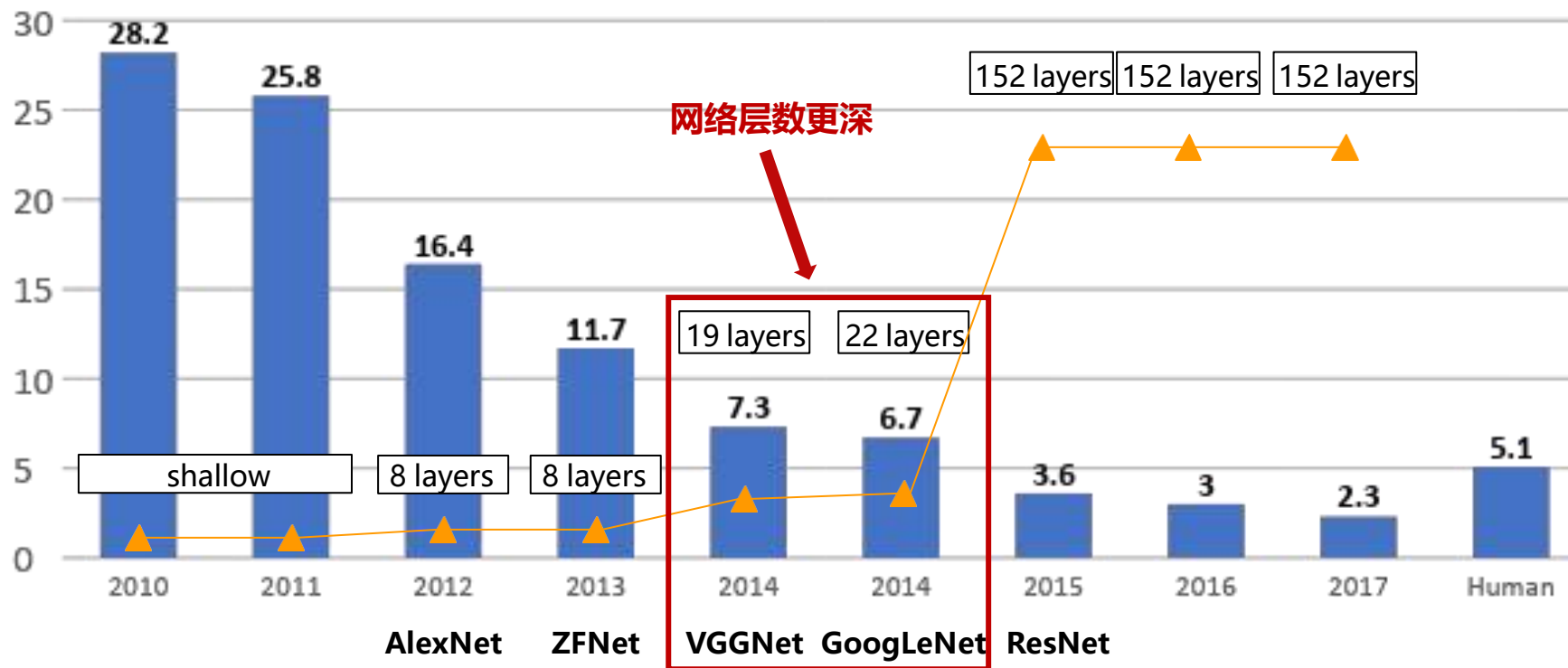
- 将Conv1的卷积核大小由11*11改为7*7，步长由4改为2
- 通过更小的卷积核和更小的步长，能够提取更多的信息，确保网络提取出来的信息更有效



常见CNN网络

15

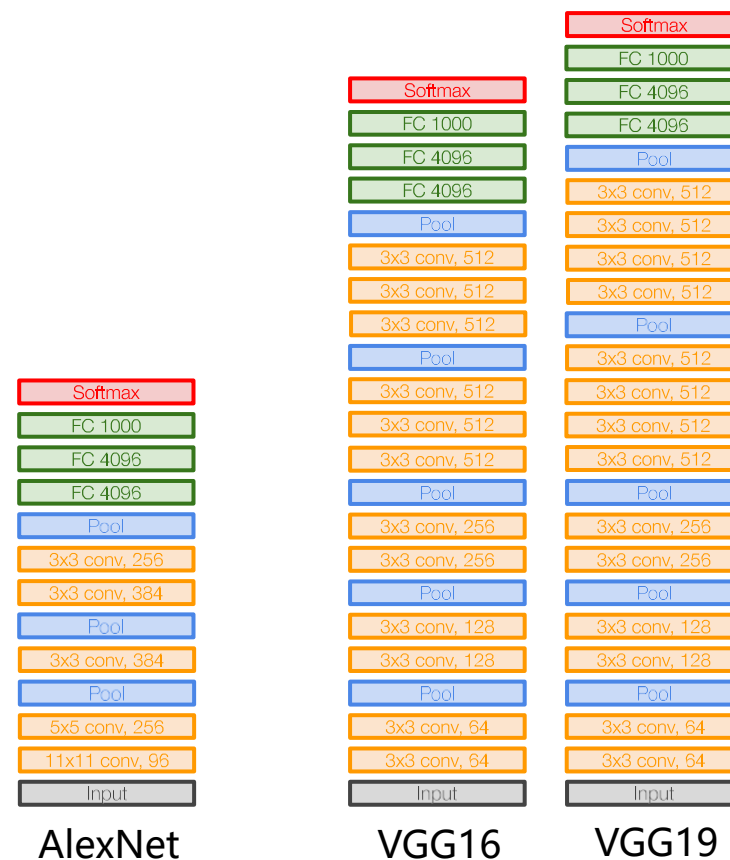
ImageNet Large Scale Visual Recognition Challenge(ILSVRC) winners



4. VGGNet, 2014

VGGNet在2014年ILSVRC中取得了定位任务的第一名和分类任务的第二名，可以看成是加深版本的AlexNet。

- 使用更小的卷积核、更深的网络
- 将AlexNet的8层网络加深到16/19层
- CONV的大小为3*3、步长为1、padding为1
- MAX POOL的大小为2*2，步长为2
- 去除了LRN



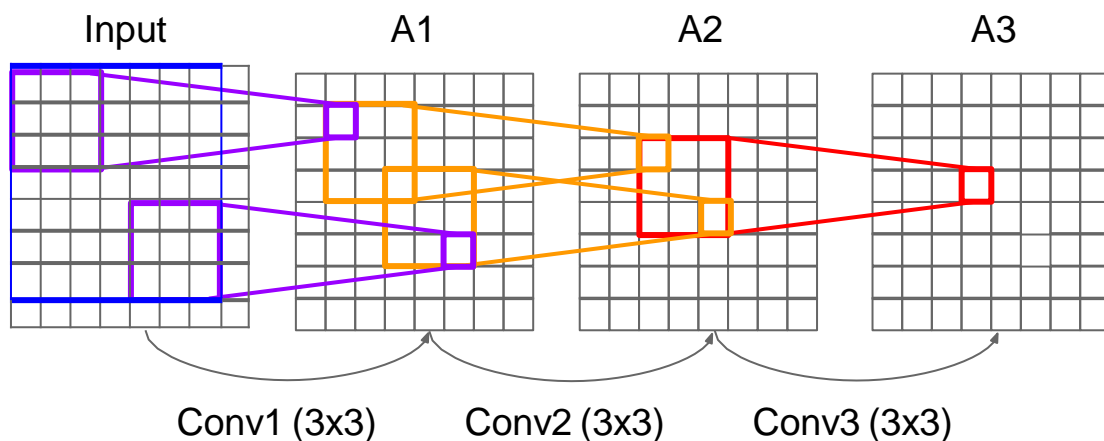
常见CNN网络

17

4. VGGNet, 2014

为什么要使用更小的卷积核? (3*3 CONV)

3个3*3卷积层与1个7*7卷积层的有效感受野相同(步长为1)

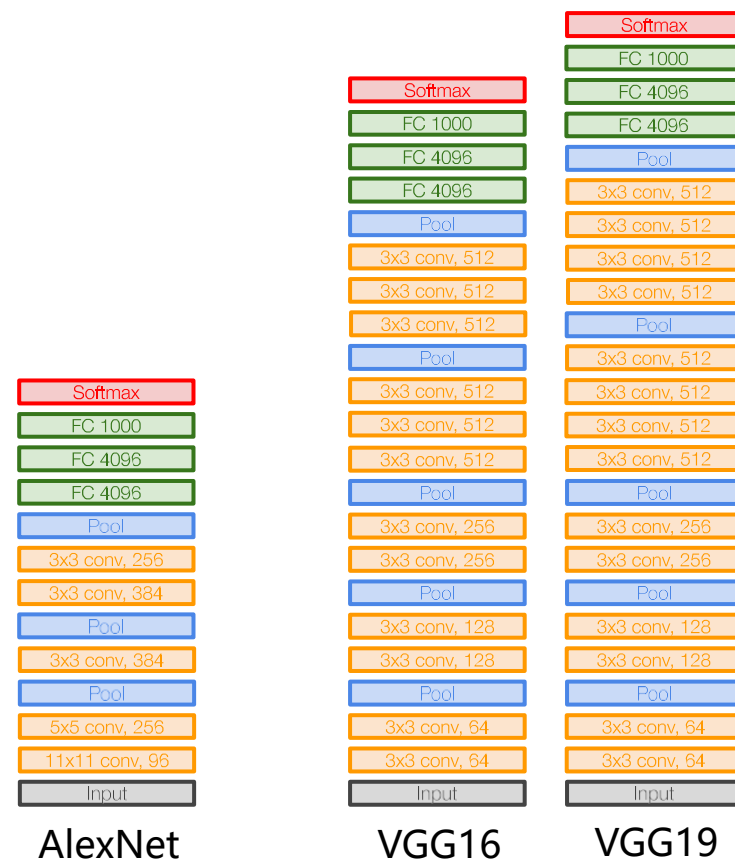


而增加层数能够提高非线性拟合能力, 并且需要的参数量更少

3个3*3卷积层: $3 \cdot (3^2 C^2)$

1个7*7卷积层: $7^2 C^2$

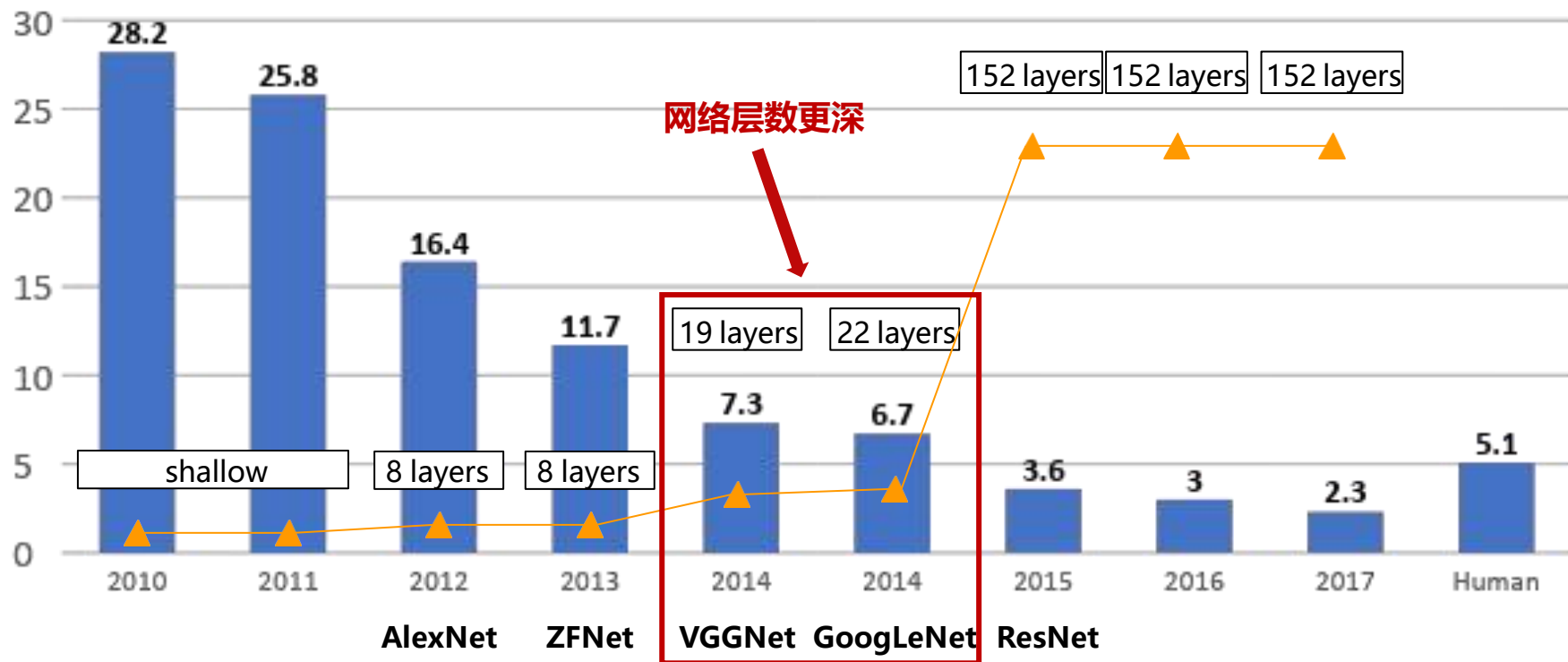
(C为通道数)



常见CNN网络

18

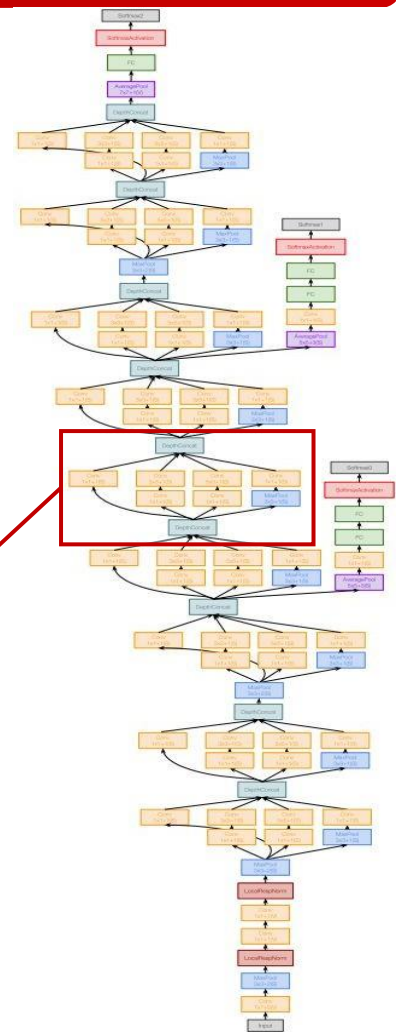
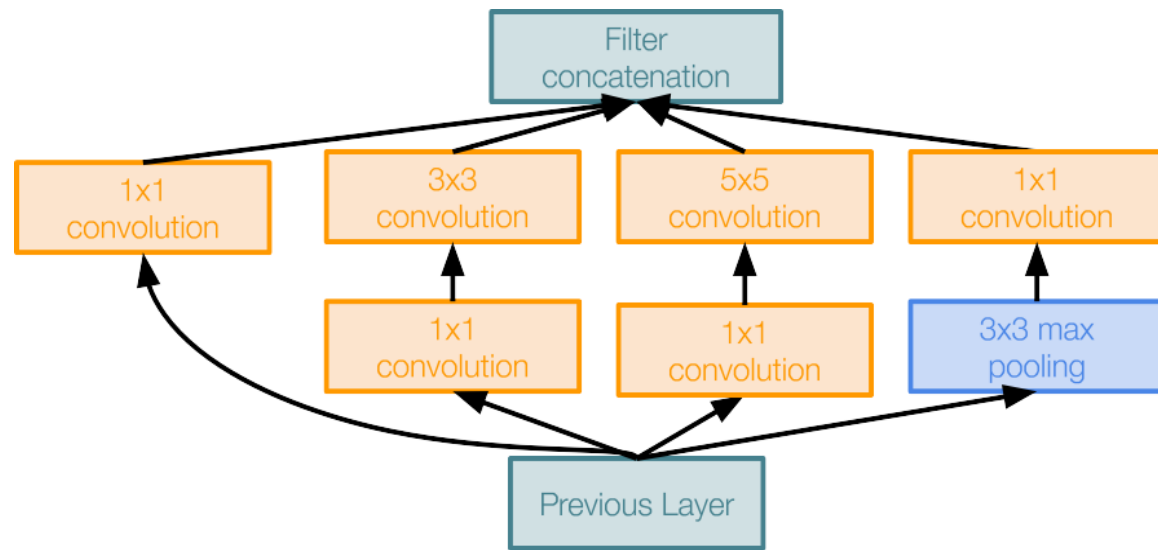
ImageNet Large Scale Visual Recognition Challenge(ILSVRC) winners



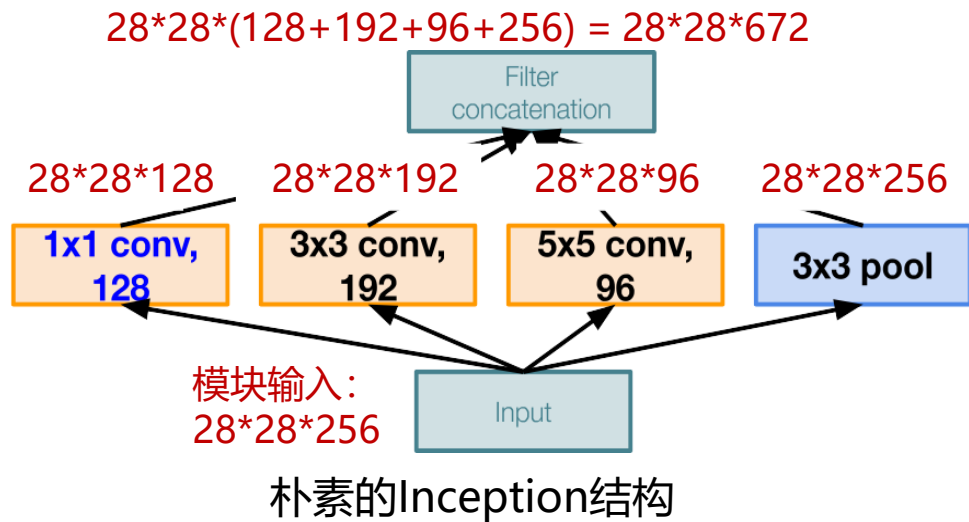
5. GoogLeNet, 2014

GoogLeNet是2014年ILSVRC的冠军，在加深网络的同时(22层)，也在结构上进行了创新。

- 引入Inception结构代替传统的卷积+激活操作
- 在中间层加入辅助计算的LOSS单元
- 将全连接层替换为简单的全局平均池化
- 参数量更少，比AlexNet少12倍，比VGG16少27倍

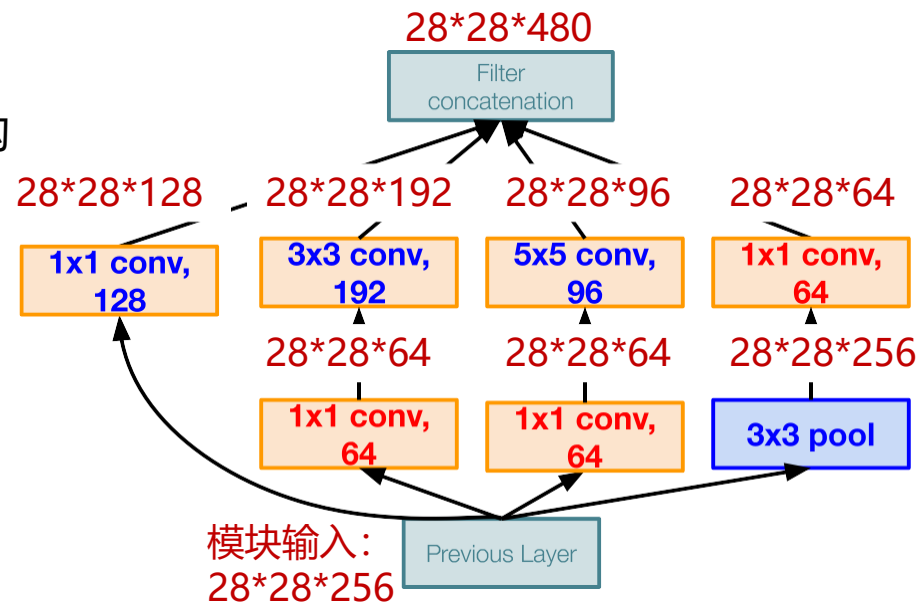


5. GoogLeNet, 2014



上述结构保留了特征的深度，连接操作后的总深度只能在一层内增长，需要非常复杂的计算。

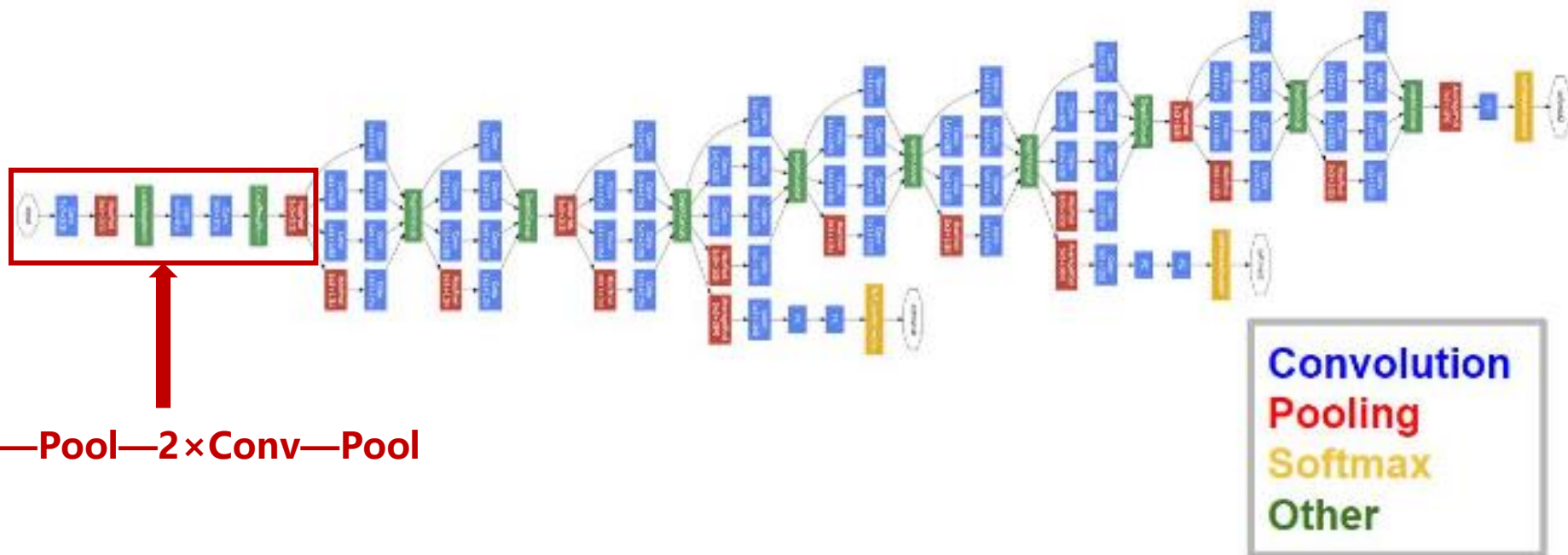
因此在上述结构中加入**维数缩减**：



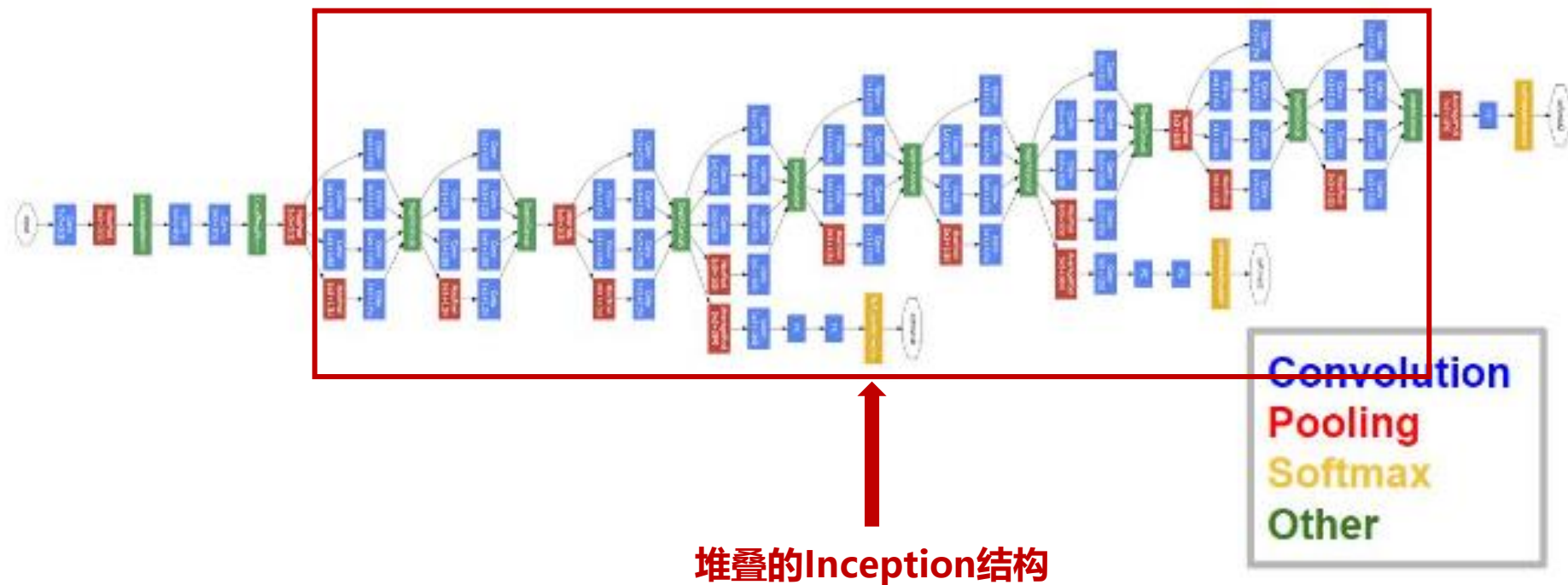
常见CNN网络

21

5. GoogLeNet, 2014



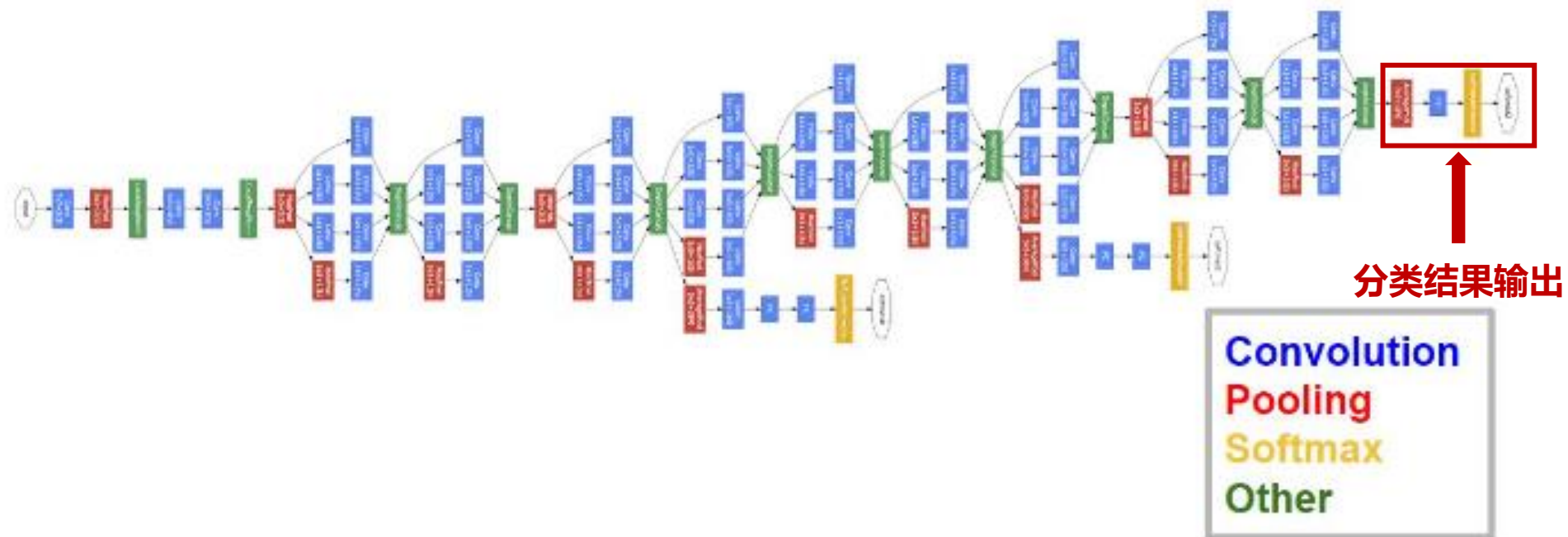
5. GoogLeNet, 2014



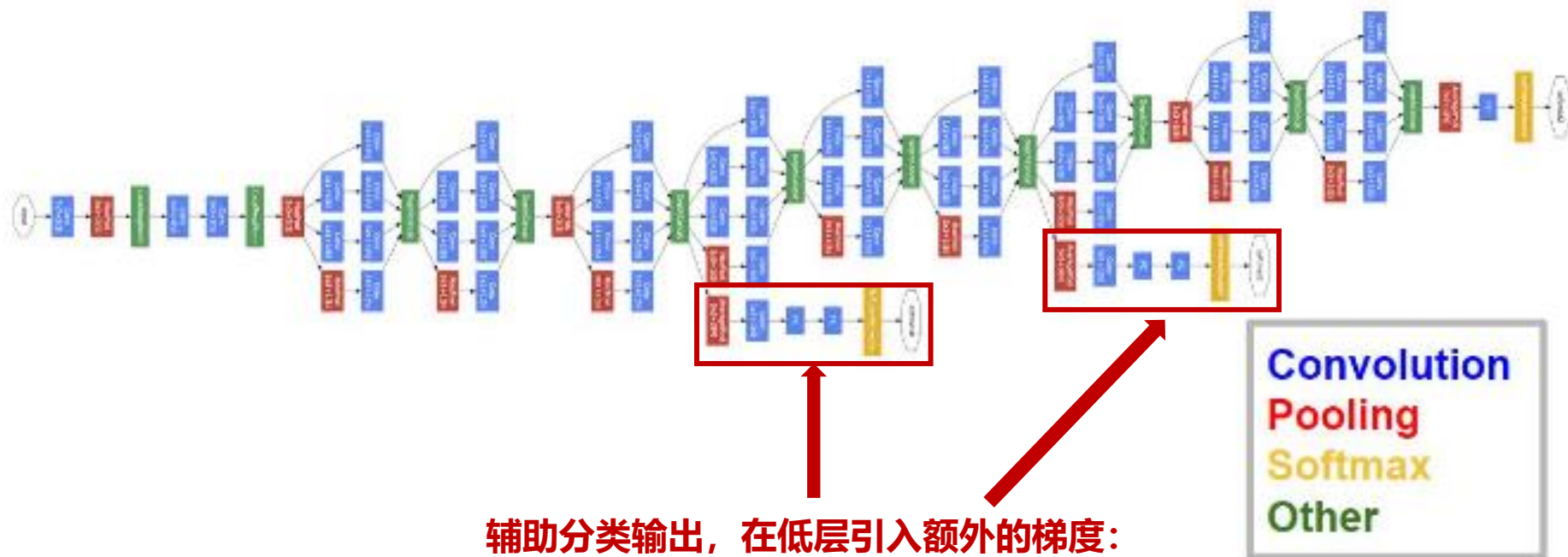
常见CNN网络

23

5. GoogLeNet, 2014



5. GoogLeNet, 2014

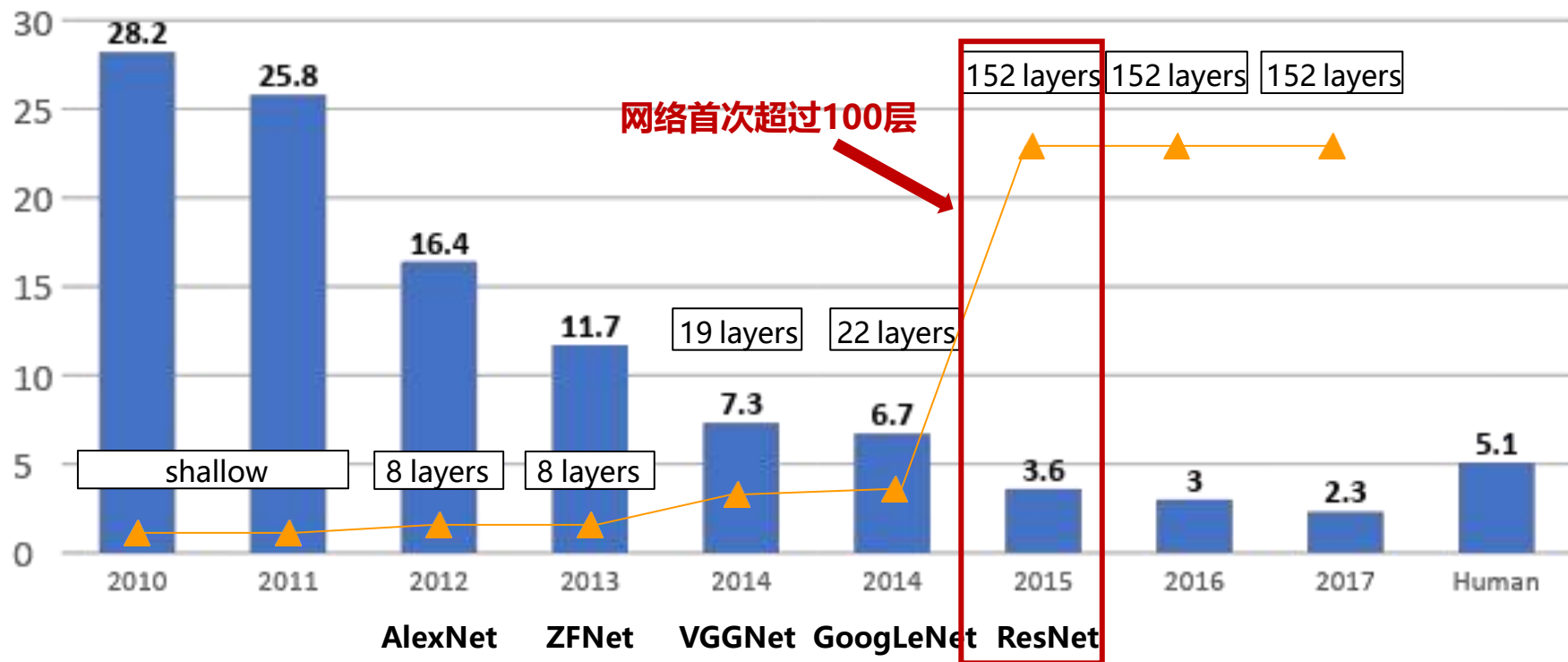


辅助分类输出，在低层引入额外的梯度：
AvgPool—Conv—FC—FC—Softmax

常见CNN网络

25

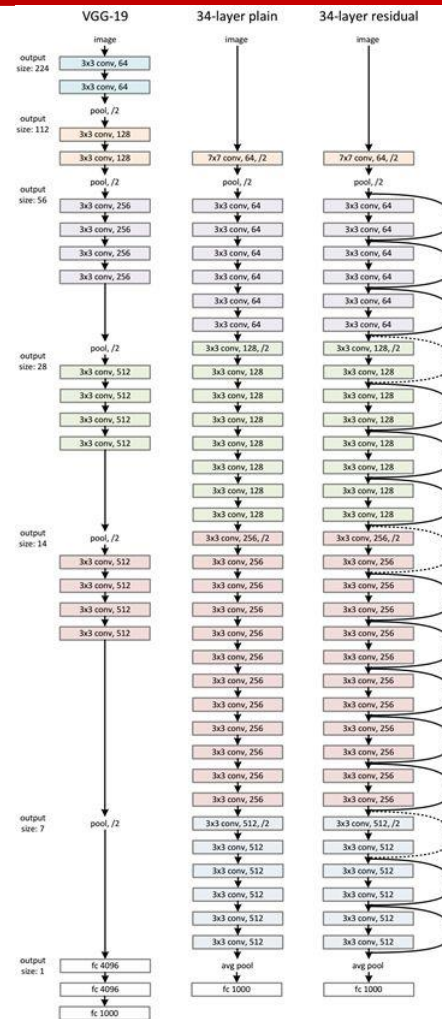
ImageNet Large Scale Visual Recognition Challenge(ILSVRC) winners



6. ResNet, 2015

ResNet在2015年ILSVRC中获得了冠军。ResNet不再是简单地堆积层数，而是在网络结构上做了大创新，是深度学习发展历程上里程碑式的事件。

- 层数非常深，已经超过百层
- 引入残差单元来解决退化问题



6. ResNet, 2015

- 梯度消失问题

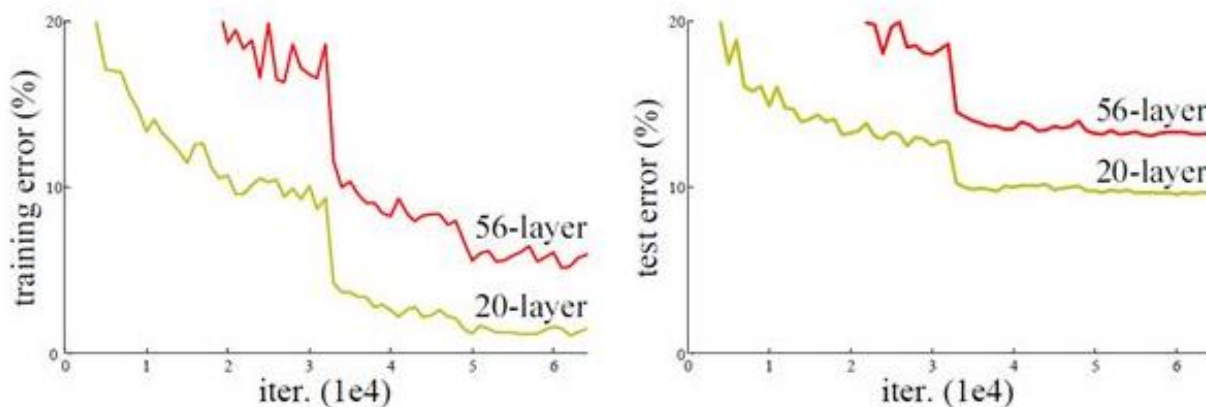
随着深度增加，网络的准确度应该同步增加，但这些增加的层是参数更新的信号(梯度从后向前传播)

增加网络深度后，靠前的层梯度会很小，这意味着这些层基本上学习停滞了

- 退化问题

随着深度增加，参数空间更大，这时即使网络已经收敛了，但其实正在开始退化，即增加网络层数却导致更大的误差

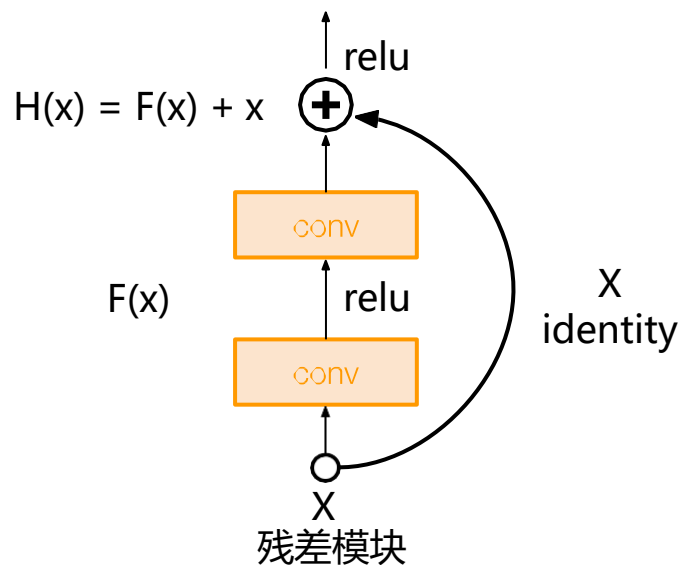
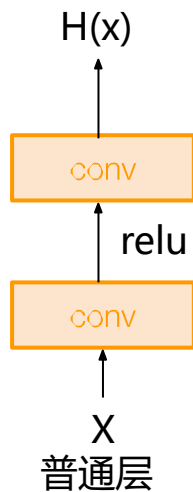
例：一个56层的网络性能却不如20层的性能好，这不是因为过拟合(训练集误差依然很大)，而是退化问题



6. ResNet, 2015

解决方法：引入**残差模块**

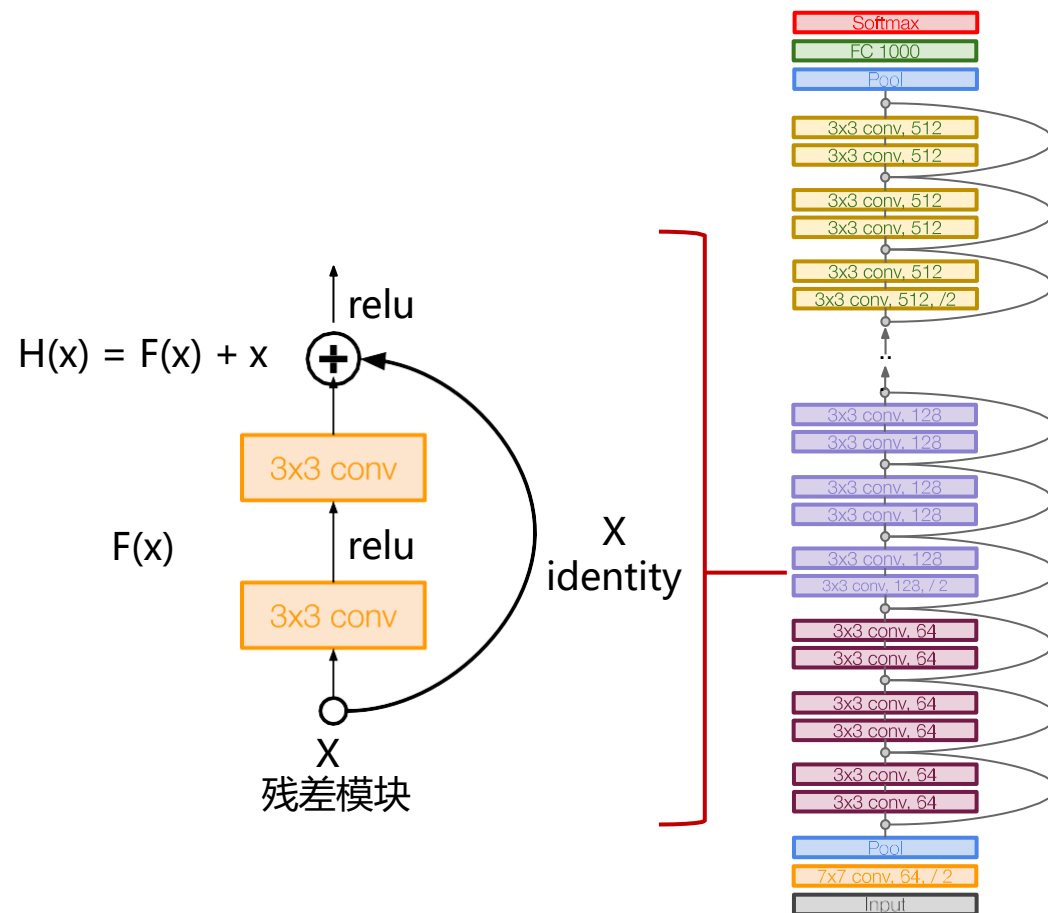
- 残差模块中，数据经过了两条路线，一条是常规路线，另一条是直接实现单位映射的捷径（shortcut）
- 将网络中模块的输入和输出关系看作 $y=H(x)$ ，那么直接通过梯度方法求 $H(x)$ 就会遇到退化问题；而如果使用这种带 shortcut 的结构，那么优化目标就不再是 $H(x)$ ，若用 $F(x)$ 来代表优化目标的话，则 $H(x)=F(x)+x$ ，即 $F(x)=H(x)-x$
- 由于 $y=x$ ， $F(x)$ 就是残差，因此只需学习输入和输出的差值，优化起来更为简单



6. ResNet, 2015

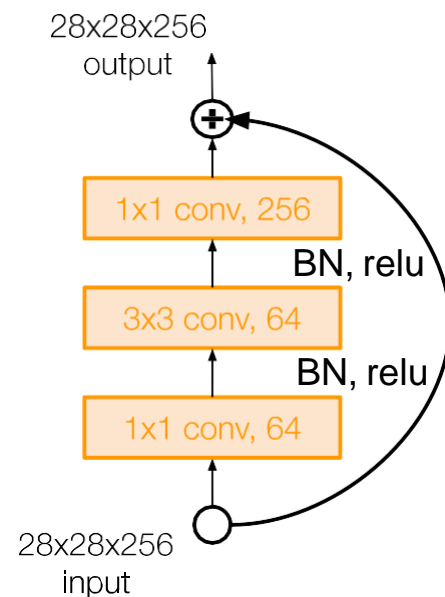
ResNet网络结构:

- 每个残差模块包含两个 3×3 卷积层
- 周期性地使用滤波器以及步长为2的空间下采样
- 在输入图像后使用额外的卷积层
- 在网络末端只使用一个FC 1000来输出类别, 没有其余的全连接层



6. ResNet, 2015

随着网络的进一步加深，通常采用右图所示的瓶颈残差模块。
依次由 1×1 、 3×3 、 1×1 这三个卷积层堆积而成， 1×1 卷积能够起到降维或升维的作用，从而使得 3×3 的卷积可以在相对较低维度的输入上进行，以达到提高计算效率的目的。



7. 轻量型卷积神经网络

- SqueezeNet
- MobileNet
- ShuffleNet
- Xception
- MixNet
- EfficientNet
-

Neural Architecture Search, Pruning, Quantization, etc.

常见CNN网络及深度学习平台

32

一、常见CNN网络

二、深度学习平台

1. Theano

Theano最初诞生于蒙特利尔大学LISA实验室，于2008年开始开发，是第一个有较大影响力的Python深度学习框架。

- 结合了计算机代数系统和优化编译器，可以最小化编译/解析的计算量
- 集成了numpy，更适合用于数值计算优化，特别是多维数组
- 更适合用于研究工作，困难调试、构建图慢等缺点为人所诟病
- 为早期深度学习研究人员提供了极大帮助，同时也为之后深度学习框架的开发奠定了基本设计方向：以计算图为核心，采用GPU加速计算
- 2017年9月28日，LISA实验室负责人、深度学习三巨头之一Yoshua Bengio宣布Theano即将停止开发

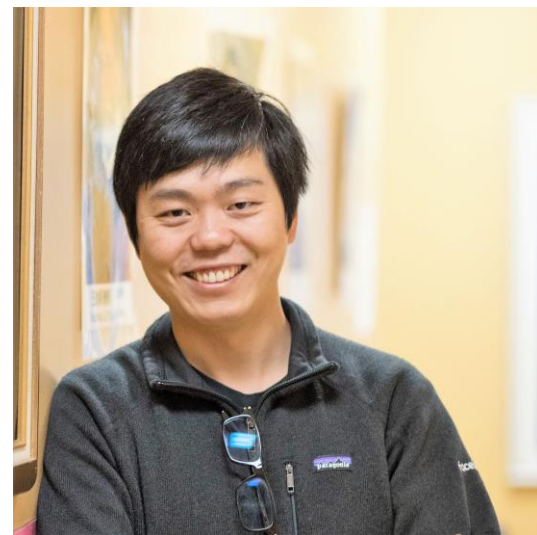


Yoshua Bengio

"Theano is Dead."

贾扬清

- 加州大学伯克利分校计算机科学博士学位
- 清华大学硕士学位和学士学位
- 曾就职于新加坡国立大学、微软亚洲研究院、NEC美国实验室、Google Brain、Facebook，现任阿里巴巴技术副总裁
- 在UC Berkeley读博期间，开发了深度学习框架**Caffe**
- 在Google Brain研究期间，参与了**TensorFlow**的开发以及**GoogLeNet**的创建
- 在Facebook AI Reseaech(FAIR)工作期间，担任工程主管，参与了深度学习框架**Caffe2**和**PyTorch**的开发



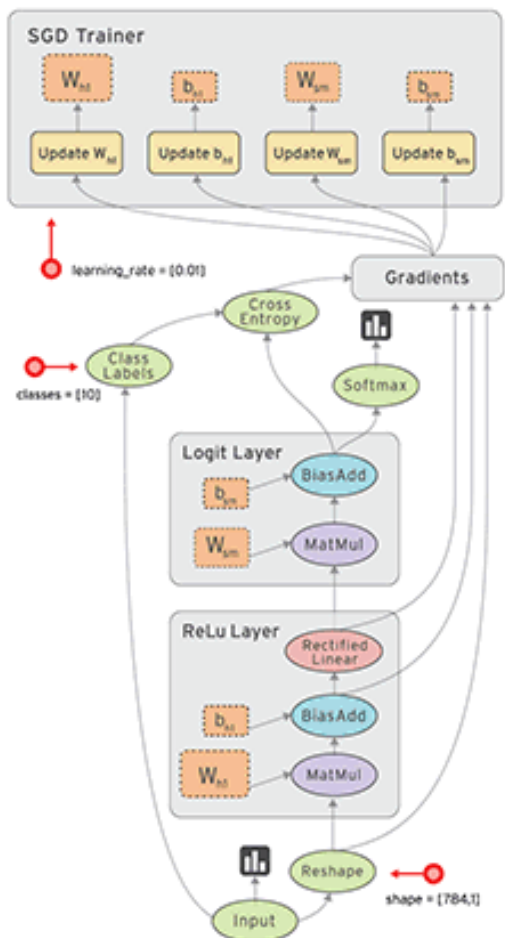
贾扬清

2. Caffe/Caffe2

Caffe(Convolutional Architecture for Fast Feature Embedding)的核心语言是C++，由贾扬清在UC Berkeley读博期间开发，毕业后贾扬清先后在Google、Facebook工作，并在FAIR任职期间开发了Caffe2。

- 支持命令行、Python和MATLAB接口，可以在CPU和GPU上运行
- 主要包括Blob、Layer和Net三个核心模块：Blob用于数据存储、交互和处理；Layer用于定义网络的层级结构，将Blob作为输入/输出；Net是一系列Layer的集合，并将这些层级结构连接形成一个网图
- Caffe简洁快速，但缺少灵活性，需要用C++/CUDA编写Layer的前向传播和反向传播代码
- Caffe2沿袭了大量的Caffe设计，在保证扩展性和高性能的同时，还强调了便携性，可部署在Linux、Windows、iOS、Android等多种环境下
- 极盛的时候，Caffe占据了计算机视觉研究领域的半壁江山，虽然如今Caffe已经很少用于学术界，但由于其稳定、出众的性能，仍有一些计算机视觉相关论文和公司还在使用Caffe部署模型

3. TensorFlow



TensorFlow最初是由Google机器学习研究部门的Google Brain团队开发，基于2011年Google开发的深度学习基础架构DistBelief构建起来的。

- 可以看作Theano的后继者，都是基于计算图实现自动微分系统
- 使用数据流图进行数值计算，图中的节点代表数学运算，边则代表在这些节点之间传递的多维数组(张量)
- 用户可以在各种服务器和移动设备上部署自己的训练模型，无须执行单独的模型解码器或加载Python解释器
- TensorFlow一经推出就获得了极大的关注，并迅速成为如今用户最多的深度学习框架之一
- 对它的批评也不绝于耳：过于复杂的系统设计、频繁变动的接口、接口设计过于晦涩难懂、文档混乱脱节

4. PyTorch

2017年1月，Facebook人工智能研究院(FAIR)团队在GitHub上开源了PyTorch，并迅速占领GitHub热度榜榜首。

- PyTorch的历史可追溯到2002年就诞生于纽约大学的Torch，Torch使用小众语言Lua作为接口，Lua虽然简洁高效，但用户太少
- PyTorch不是简单地封装Torch提供Python接口，而是对Tensor之上的所有模块进行了重构，并新增了最先进的自动求导系统
- 设计追求最少的封装，遵循tensor→variable(autograd)→nn.Module三个由低到高的抽象层次，分别代表高维数组(张量)、自动求导(变量)和神经网络(层/模块)，而且三者之间联系紧密，可以同时进行修改和操作
- 灵活性不以速度为代价，在许多评测中，PyTorch的速度表现胜过TensorFlow和Keras等框架
- 由于简洁、速度、易用、社区活跃等优点，PyTorch已经成为如今最流行的动态图框架

5. Keras

Keras是一个高层神经网络API，由纯Python编写而成并使用TensorFlow、Theano及CNTK作为后端，其创始人是Google AI研究员François Chollet。

- 是深度学习框架中最容易上手的一个，提供了一致而简洁的API，能够极大地减少一般应用下用户的工作量，避免重复造轮子
- 严格意义上讲，Keras并不能称为一个深度学习框架，它更像一个深度学习接口，构建于第三方框架之上
- 为了屏蔽后端的差异性，提供一致的用户接口，Keras做了层层封装
- 过度封装导致丧失灵活性，程序过于缓慢



François Chollet

6. MXNet



李沐



陈天奇

在2014年的NIPS上，同为上海交大校友的陈天奇与李沐碰头，发现大家普遍在做很多重复性的工作，于是他们决定组建DMLC (Distributed Machine Learning Community)，号召大家一起合作开发MXNet。

- 支持C++、Python、Scala、Julia、MATLAB、JavaScript、R等语言
- 支持命令和符号编程
- 可以运行在CPU、GPU、集群、服务器、台式机或移动设备上
- 在模型相同的情况下，MXNet往往占用更小的内存和显存
- 在分布式环境下，扩展性能明显优于其他框架
- MXNet最初由一群学生开发，缺乏商业应用，同时文档也不够完善，因此始终处于不温不火的状态

7. PaddlePaddle

PaddlePaddle(飞桨)以百度多年的深度学习技术研究和业务应用为基础，是中国首个开源开放、技术领先、功能完备的产业级深度学习平台，集深度学习核心训练和推理框架、基础模型库、端到端开发套件和丰富的工具组件于一体。

- 最初诞生于2018年7月，提供从数据预处理到模型部署在内的深度学习全流程的底层能力支持，首次开源CV/NLP/语音/强化学习等10个模型
- 2020年9月15日，以"万物智能"为主题的百度世界2020大会于线上召开，百度CTO王海峰宣布飞桨迎来全新升级，向观众揭示了核心技术自主可控、开源开放的重要价值
- 语法简洁，API的设计干净清晰
- 模型库丰富，可以非常容易地复现一些经典方法
- 能够充分利用GPU集群的性能，为分布式环境的并行计算进行加速

8. MindSpore



MindSpore是华为于2019年8月推出的新一代全场景AI计算框架，2020年3月28日，华为宣布MindSpore正式开源。

- MindSpore前端表示层包含Python API、MindSpore IR、计算图高级别优化(GHLO)三部分，分别用于提供统一接口，提供统一的中间表示，提供硬件无关优化、自动并行和自动微分功能
- MindSpore计算图引擎包含计算图低级别优化(GLLO)、图执行两部分，分别用于提供硬件相关优化和深度优化，提供离线图执行、分布式训练所需要的通信接口等功能
- MindSpore后端运行时包含云、边、端上不同环境中的高效运行环境