《物理与人工智能》

9. 举例-卷积神经网络

授课教师： 马滟青

2025/10/13（第五周）

鸣谢：基于计算机学院《人工智能引论》课程组幻灯片

神经网络

北京大学 PEKING UNIVERSITY

$$y = f_\theta(x)$$

x：输入
θ：神经网络参数
y：输出

问题：
如何把输入内容转换成可计算的数字？
如何把输出数字转换成相应的操作？

需要确定每个问题（层次）的自由度，得到其表示

卷积神经网络（CNN）

北京大学 PEKING UNIVERSITY

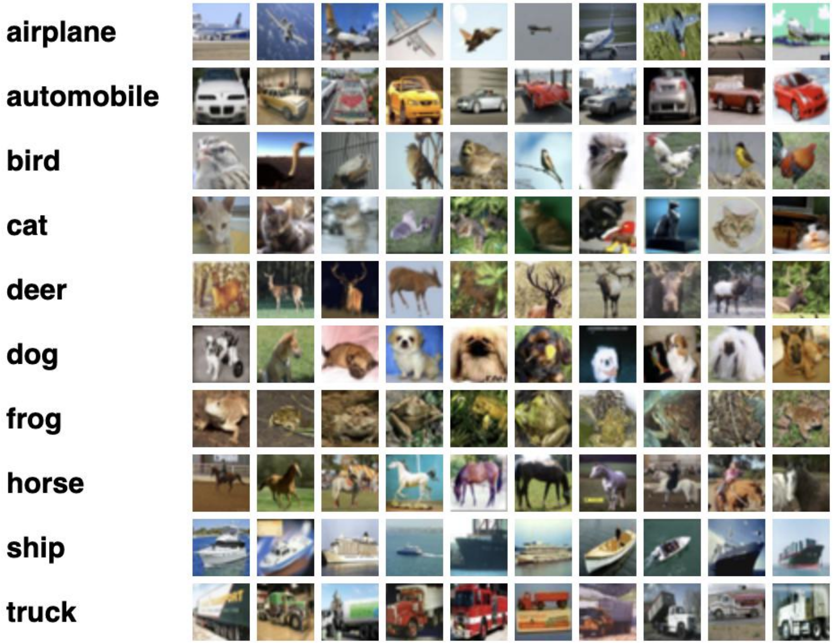
- 图像任务
- 卷积层
- CNN组件
- 常用CNN网络

基于Stanford cs231n, Lecture 5: Image Classification with CNNs
http://cs231n.stanford.edu/2021/slides/2021/lecture_5.pdf

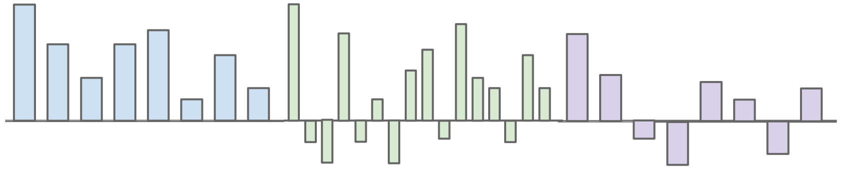
图像分类-计算机视觉的核心任务

f(x)

Class scores



图像特征

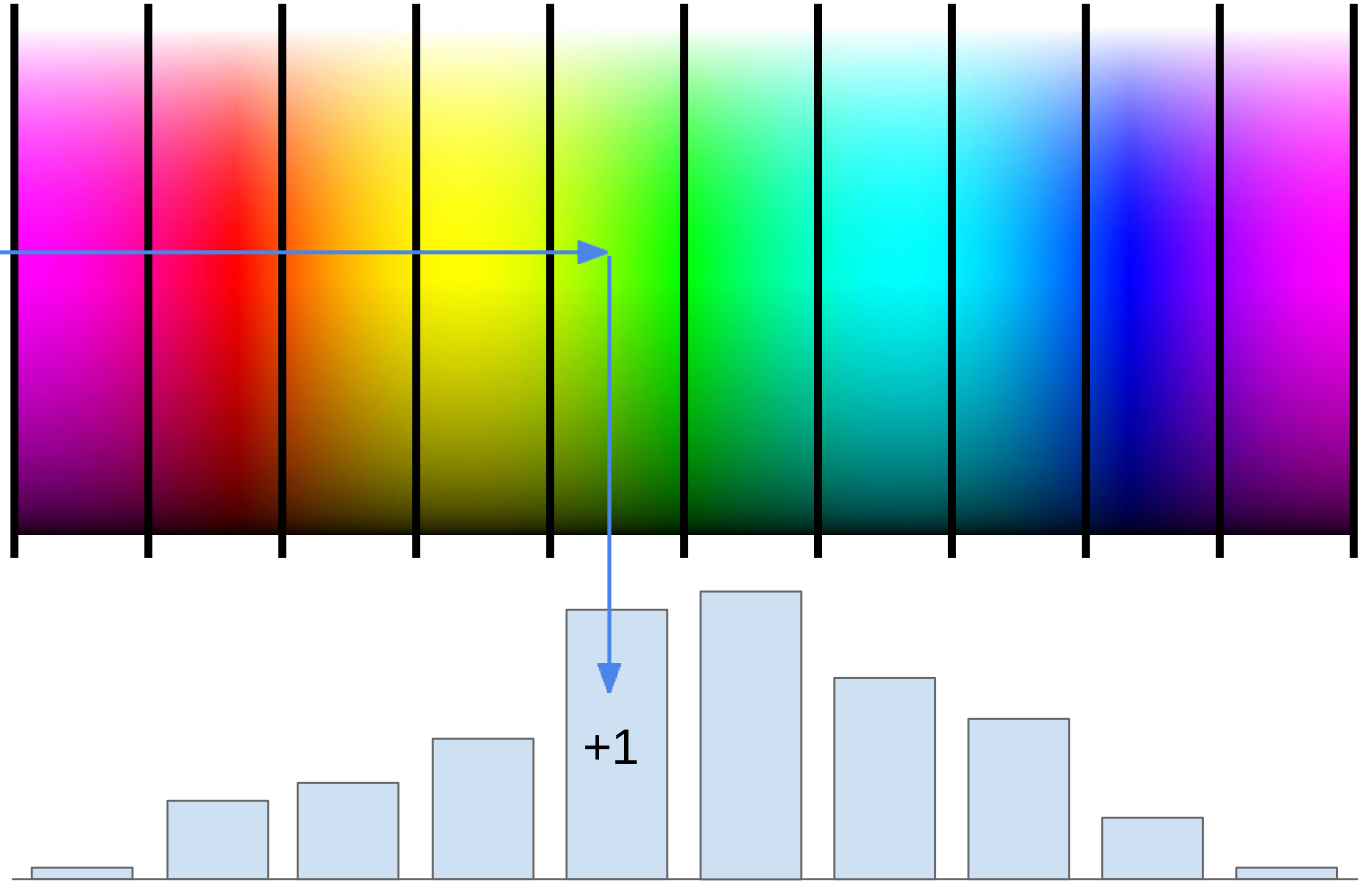


特征表示

f(x)

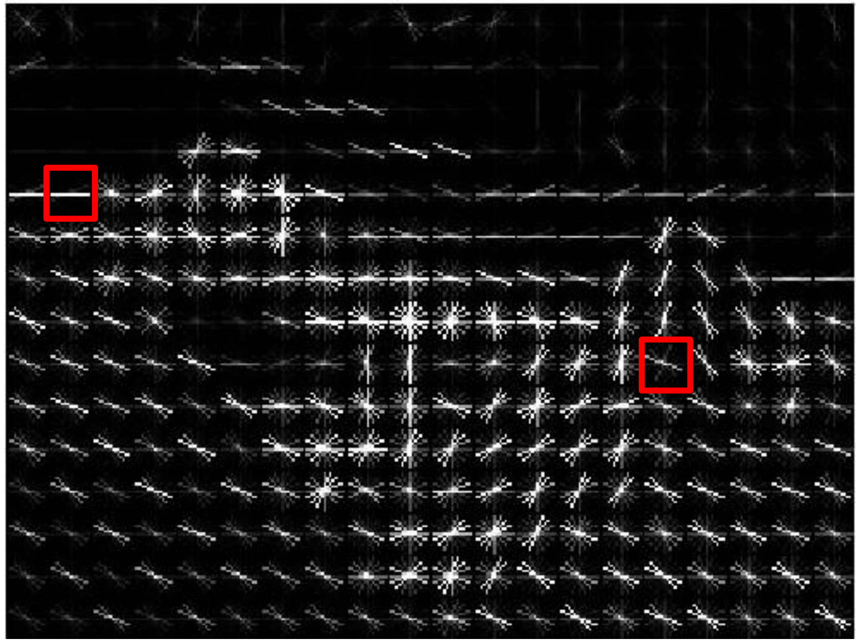
分类得分

手工特征-颜色



手工特征-纹理

Divide image into 8x8 pixel regions
Within each region quantize edge direction into 9 bins



Example: 320x240 image gets divided into 40x30 bins; in each bin there are 9 numbers so feature vector has 30*40*9 = 10,800 numbers

Dalal and Triggs, "Histograms of oriented gradients for human detection," CVPR 2005

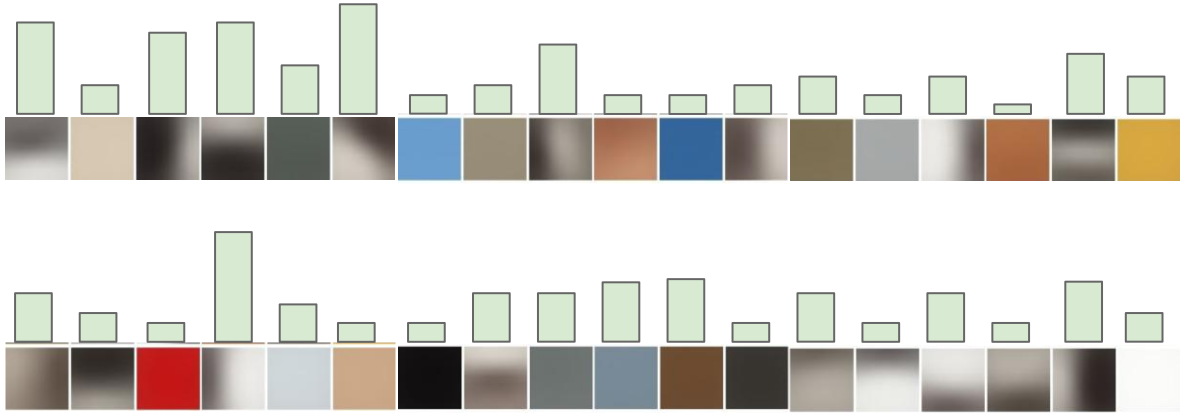
手工特征-词袋

Step 1: Build codebook

Extract random patches

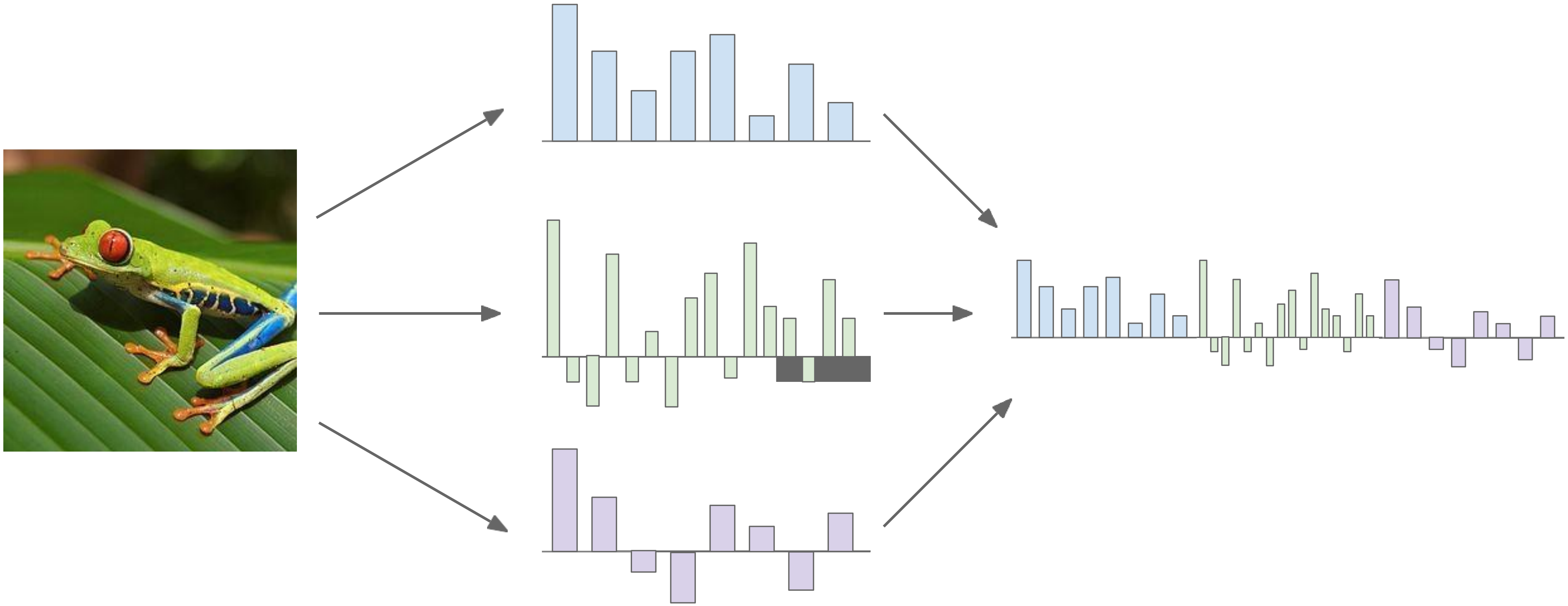
Cluster patches to form “codebook” of “visual words”

Step 2: Encode images



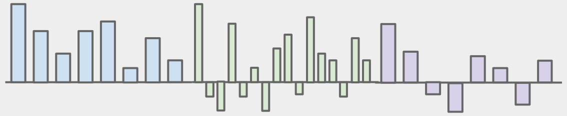
Fei-Fei and Perona, “A bayesian hierarchical model for learning natural scene categories”, CVPR 2005

集成多种特征



自动提取特征-卷积神经网络(CNN)

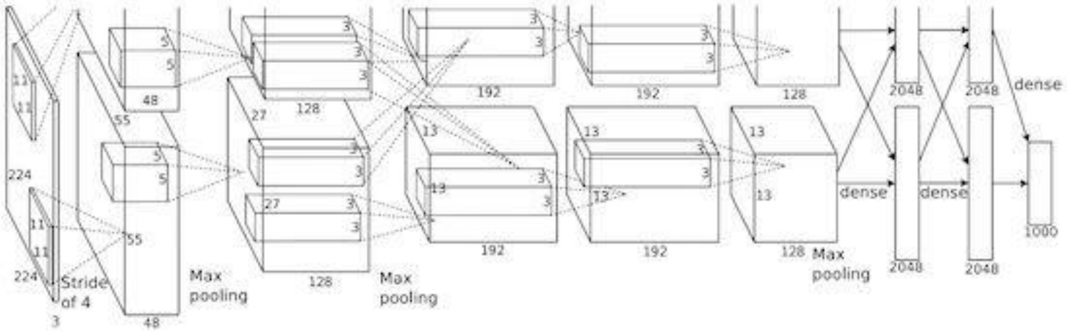
Feature Extraction



f

10 numbers giving scores for classes

training



Krizhevsky, Sutskever, and Hinton, “Imagenet classification with deep convolutional neural networks”, NIPS 2012.
Figure copyright Krizhevsky, Sutskever, and Hinton, 2012.
Reproduced with permission.

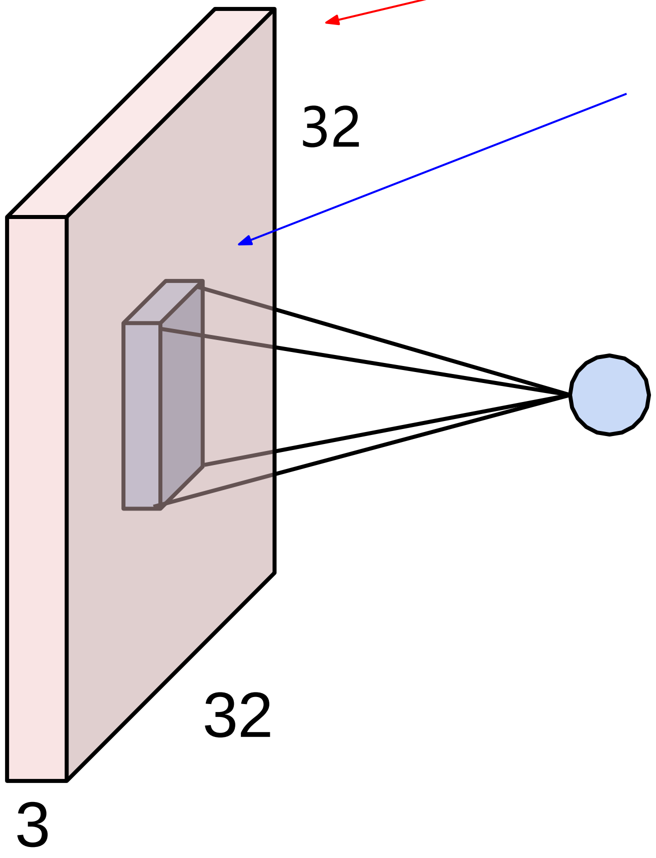
10 numbers giving scores for classes

training

卷积层（Convolution Layer）

32x32x3 图像(image)

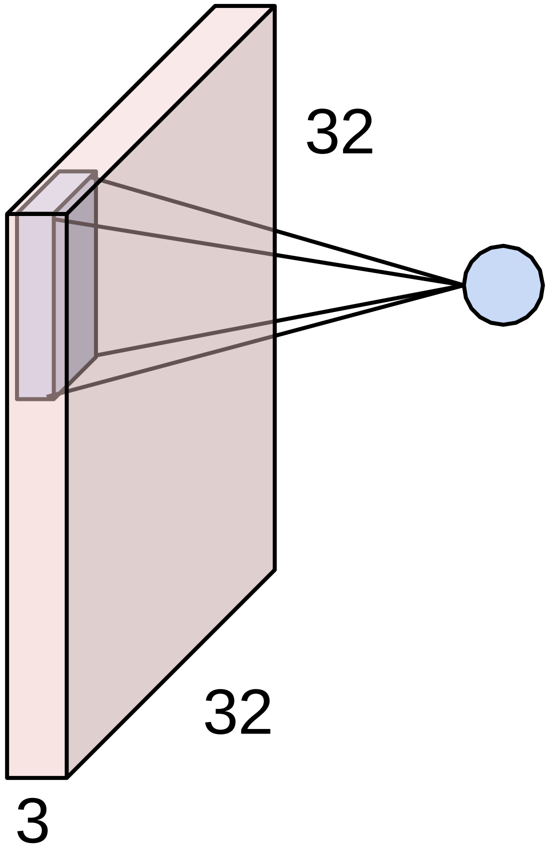
5x5x3 滤波器(filter) w

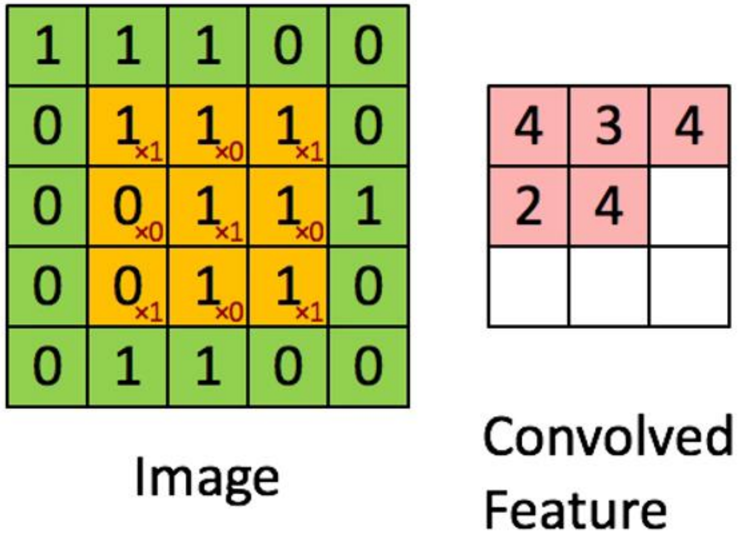


输出的每个值，是滤波器和图像上的一个5x5x3的块相乘，然后加上一个偏置量b。

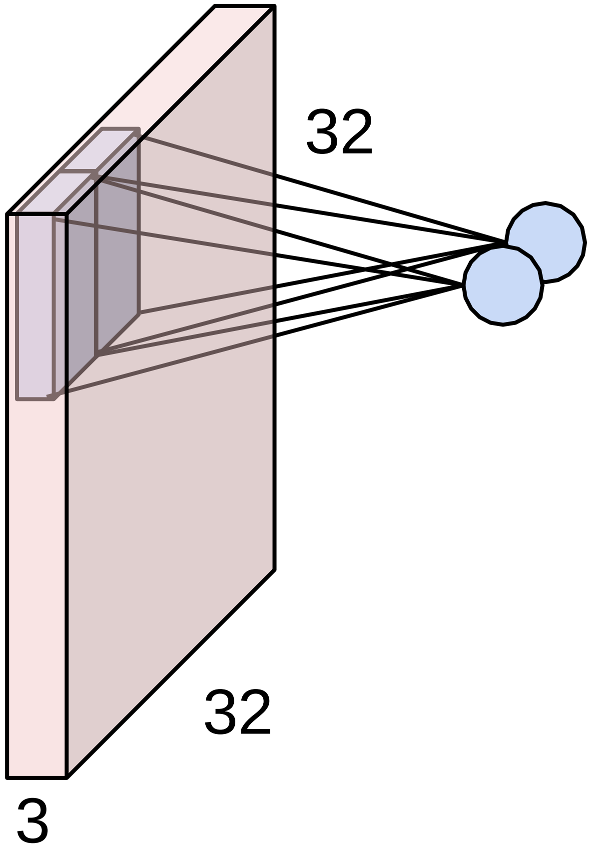
$$w^T x + b$$

卷积层（Convolution Layer）

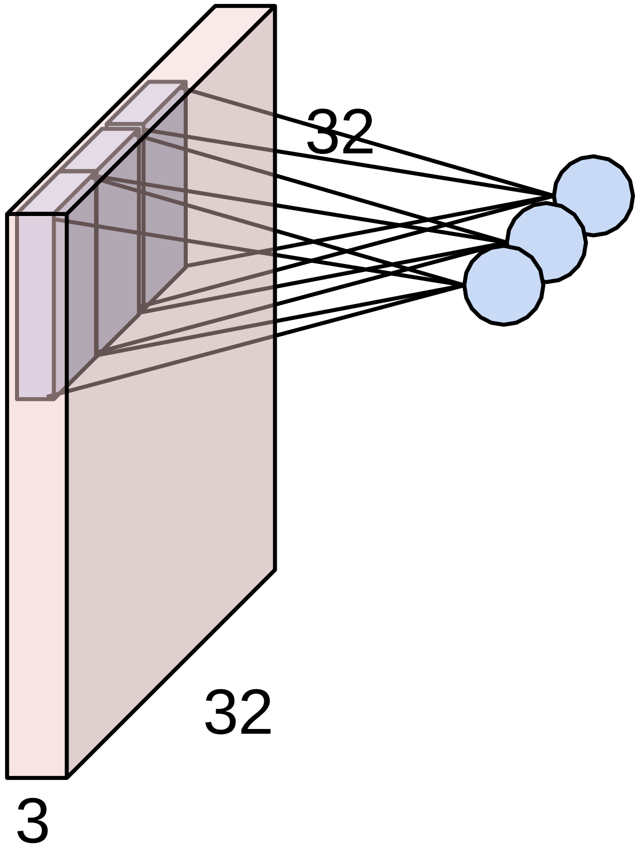




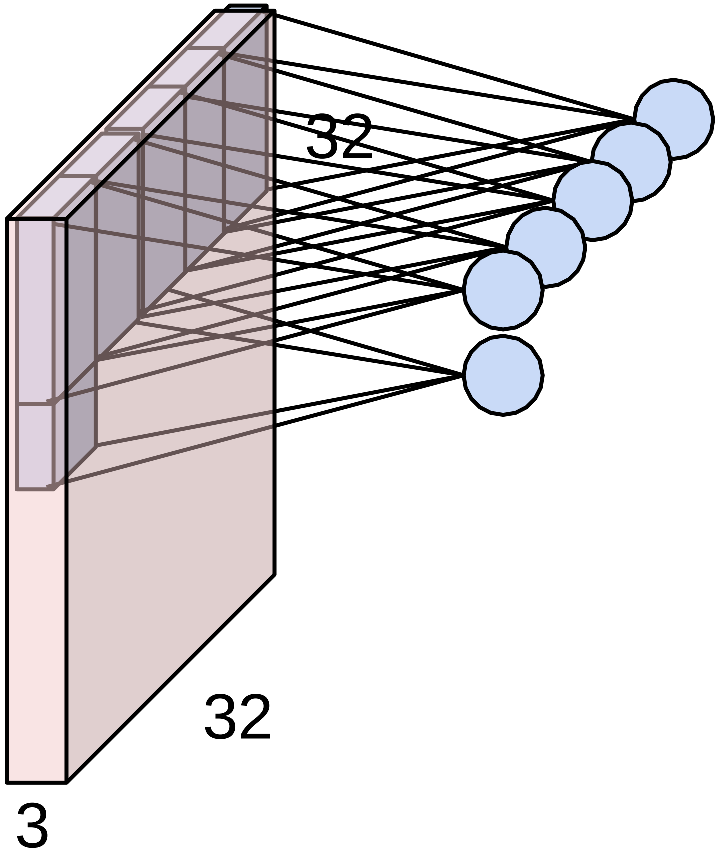
卷积层（Convolution Layer）



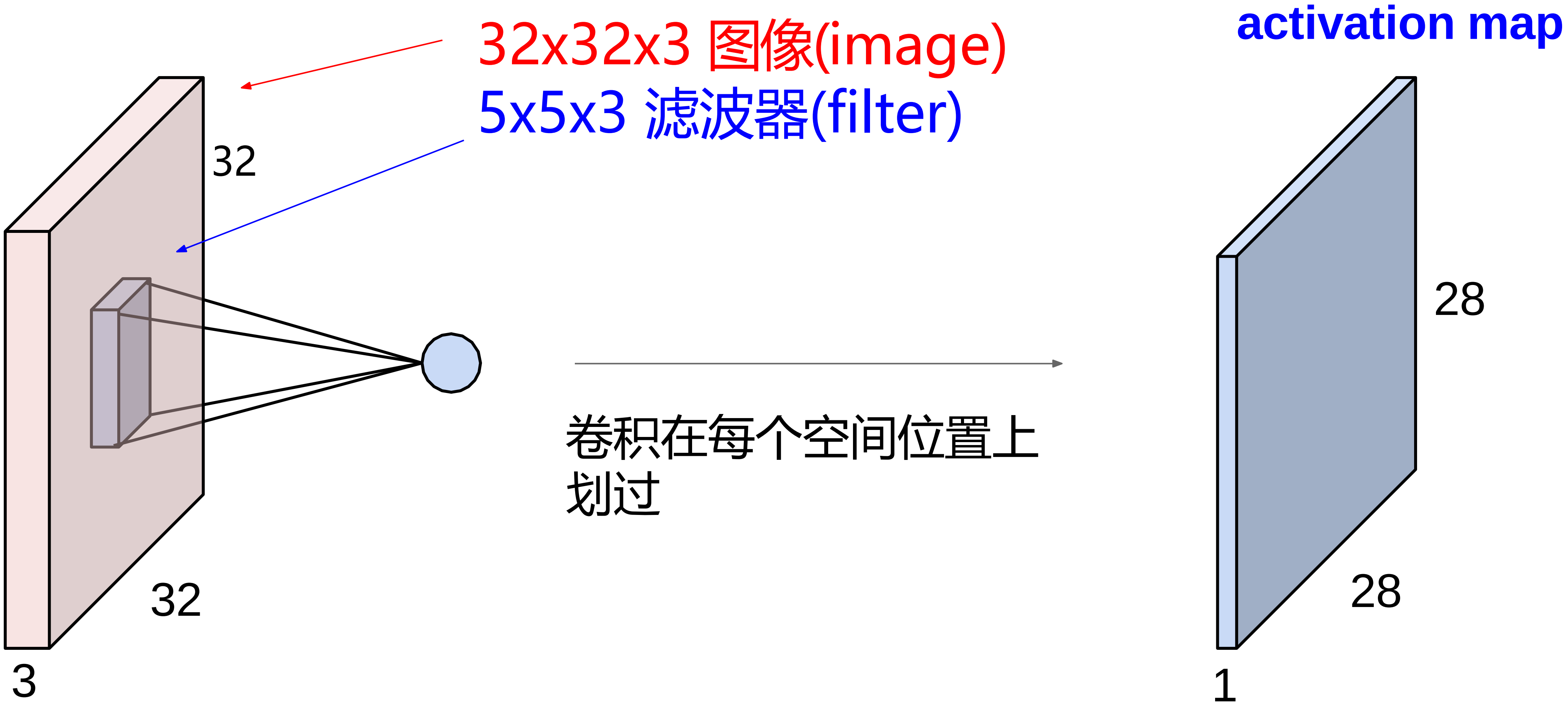
卷积层（Convolution Layer）



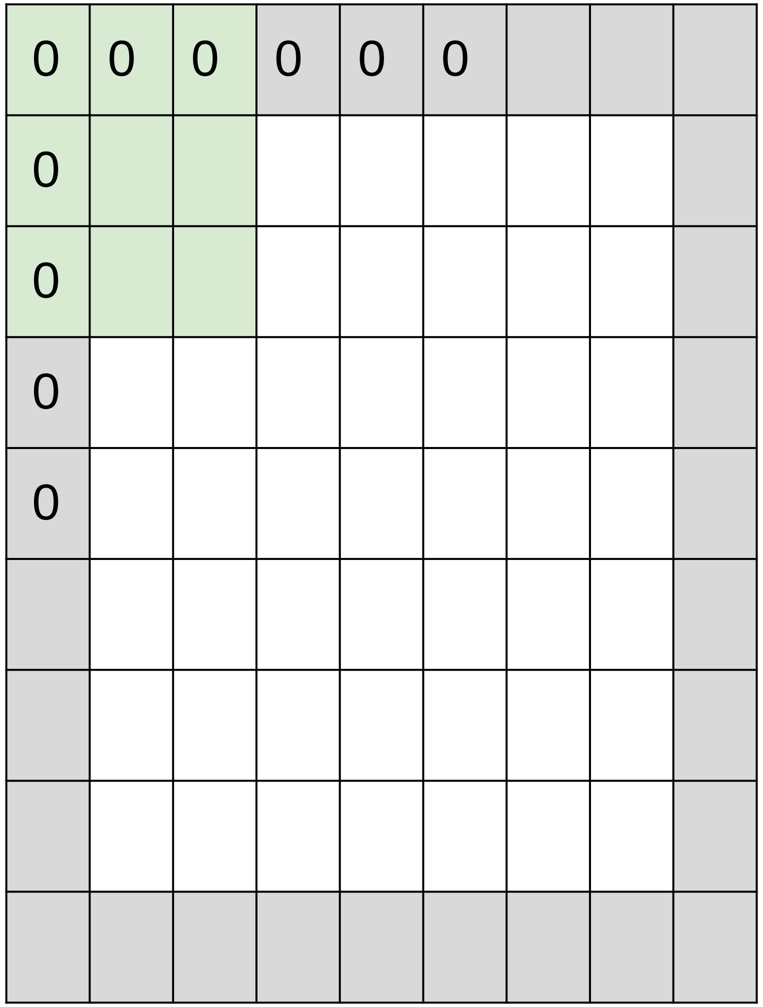
卷积层（Convolution Layer）



卷积层（Convolution Layer）



补零（Zero padding）



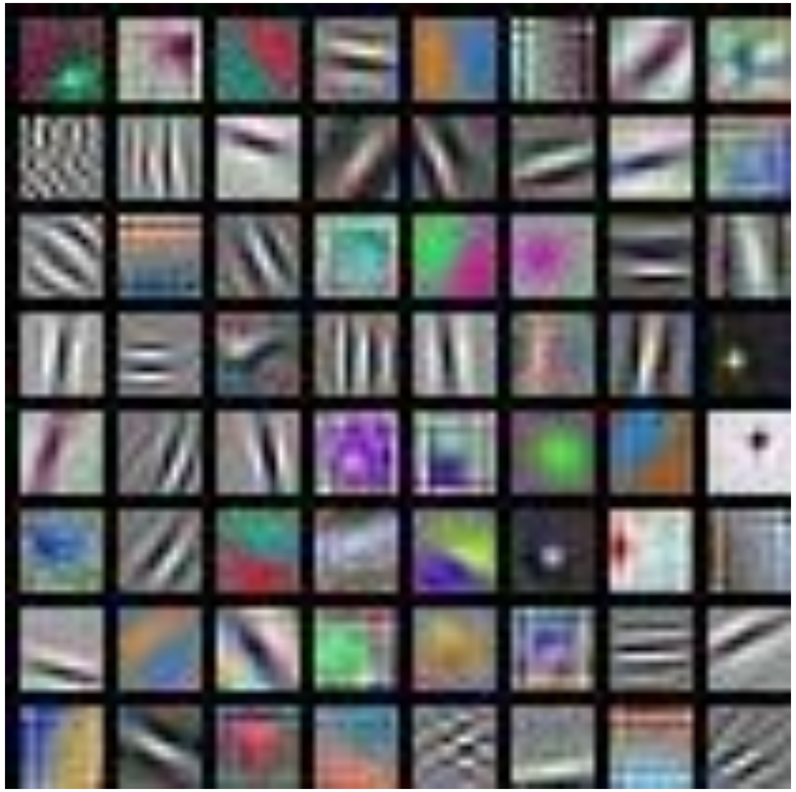
- 考虑一个输入为7x7的图片，
- 滤波器大小为3x3，padding with 1个像素=>
- **输出仍为7x7!**
- 通常如果滤波器大小为FxF， padding with (F-1)/2，就能保持原有特征图大小。

例如：

- F = 3 => zero pad with 1
- F = 5 => zero pad with 2
- F = 7 => zero pad with 3

滤波器学到了什么

滤波器：提取局部图像特征，如边缘特征，颜色。

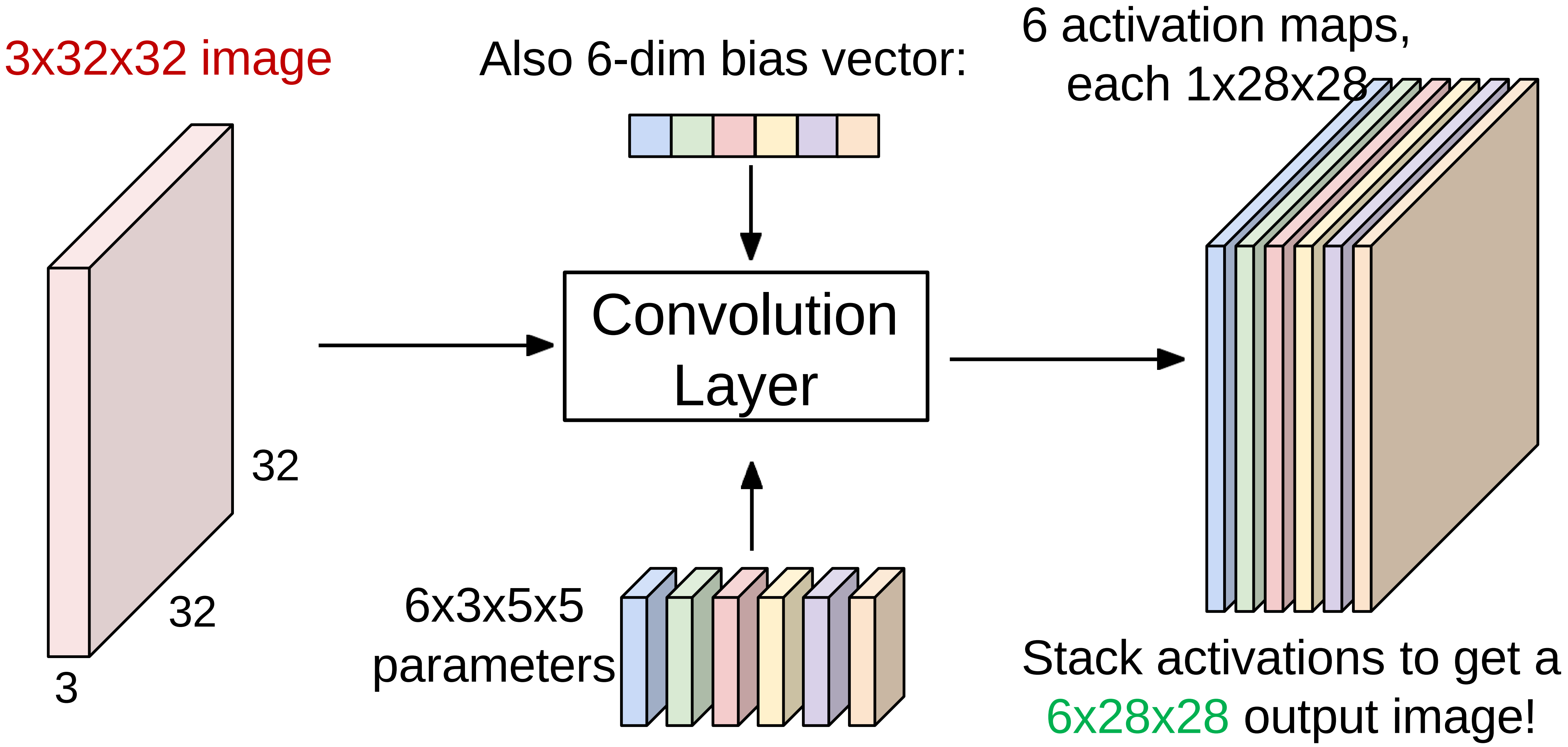


AlexNet: 64 filters, each 3x11x11

卷积层（Convolution Layer）

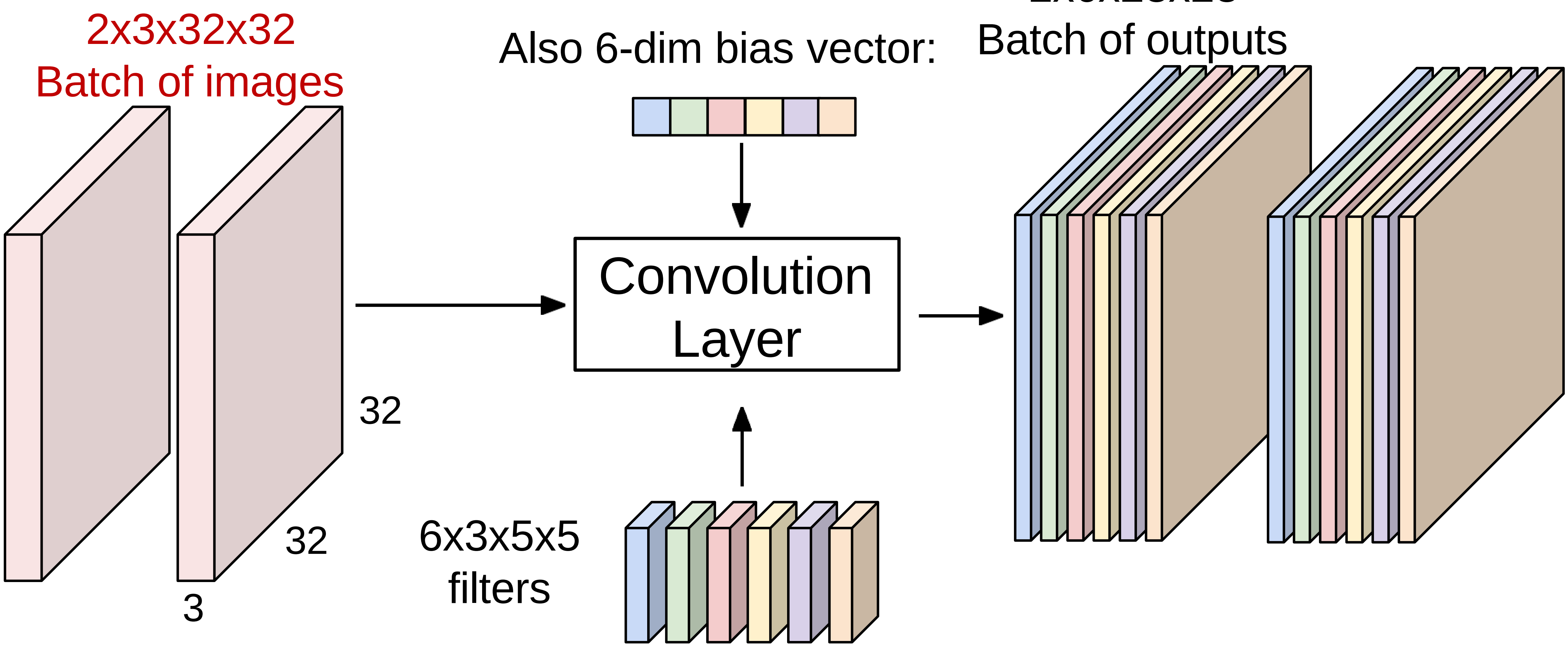
- 考虑第二个滤波器（绿色），同样地在图像上划过，产生绿色的特征图
- 每个滤波器对应一个输出通道（channel）

多个滤波器



考虑有6个 5x5x3 的滤波器

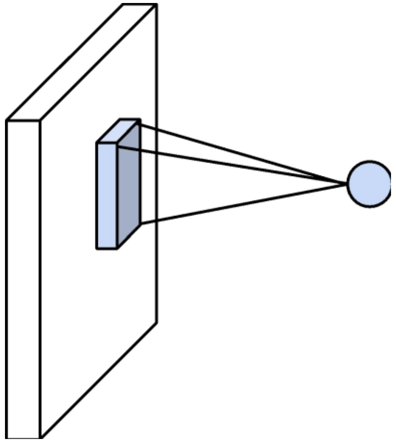
批量输入图片



2x3x32x32
Batch of images
32
32
3
Also 6-dim bias vector:
Convolution Layer
6x3x5x5
filters
2x6x28x28
Batch of outputs

CNN的常见组件

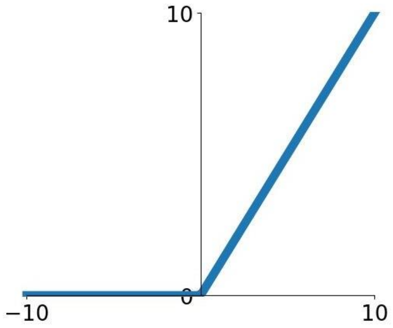
Convolution Layers



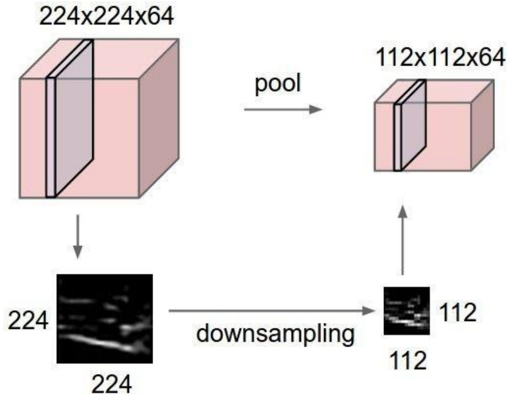
Normalization

$$\hat{x}_{i,j} = \frac{x_{i,j} - \mu_j}{\sqrt{\sigma_j^2 + \varepsilon}}$$

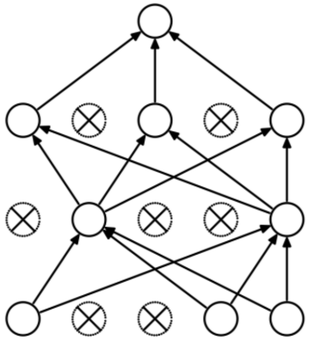
Activation Function



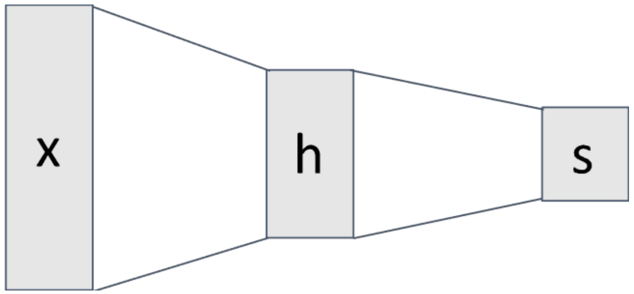
Pooling Layers



Dropout



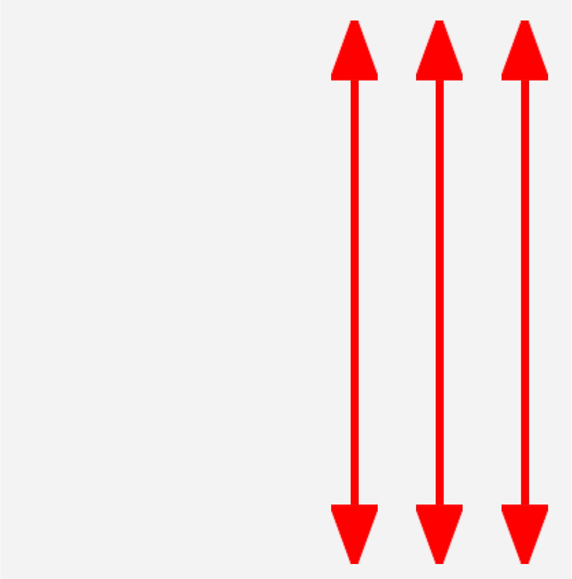
Fully-Connected Layers



批归一化（batch normalization）层

Input: $x : N \times D$

N



D

$$\mu_j = \frac{1}{N}\sum_{i=1}^{N} x_{i,j}$$

每个通道（维度）的均值，形状为D

$$\sigma_j^2 = \frac{1}{N}\sum_{i=1}^{N}(x_{i,j} - \mu_j)^2$$

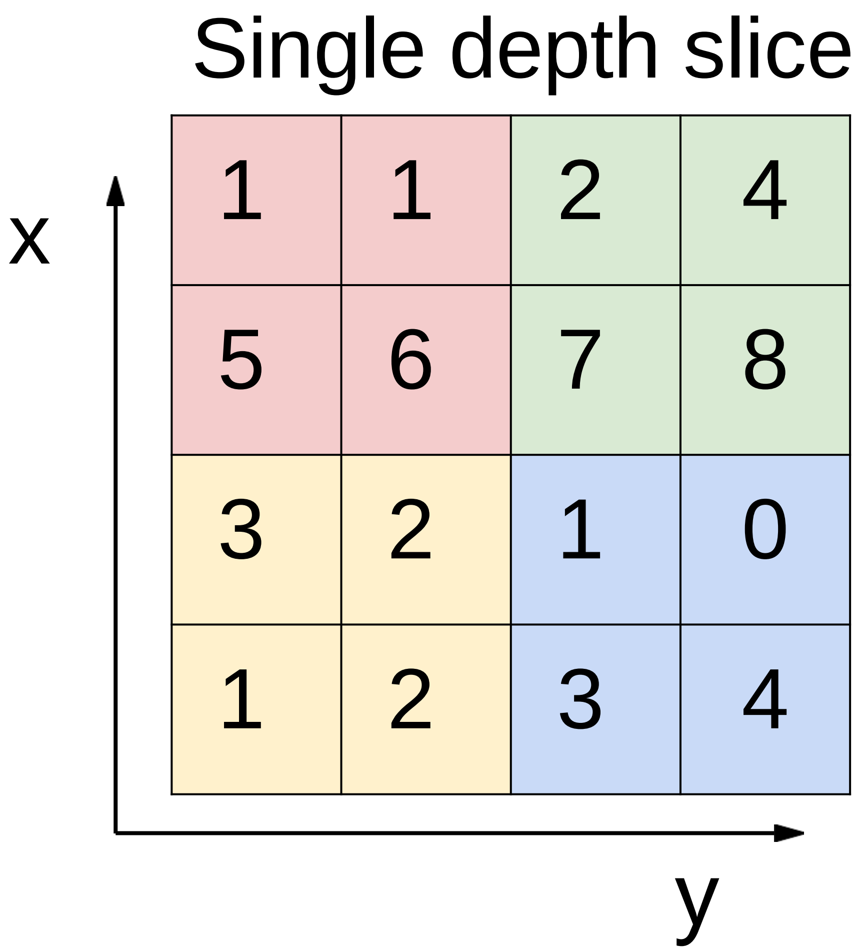
每个通道（维度）的方差，形状为D

$$\hat{x}_{i,j} = \frac{x_{i,j} - \mu_j}{\sqrt{\sigma_j^2 + \varepsilon}}$$

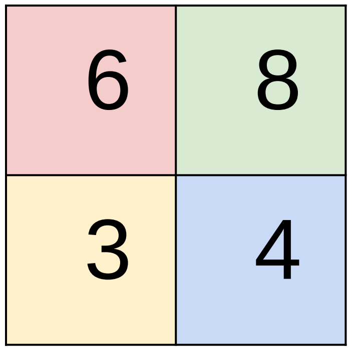
标准化 x，形状为N x D

- 将各维度特征标准化（0均值，1方差）
- 大致将各特征统一到相同尺度，利于优化
- 实际中非常有用

最大池化(MaxPooling)



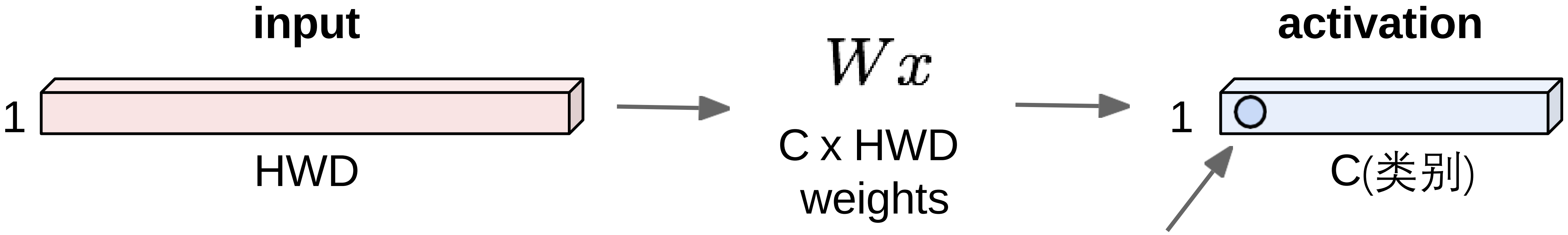
max pool with 2x2 filters and stride 2



- 没有可学习的参数
- 引入空间不变性

全连接层

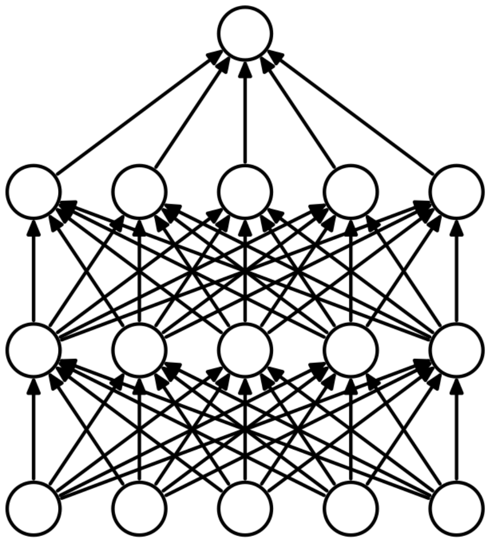
HxWxD(长宽高) -> 拉直为HWD x 1



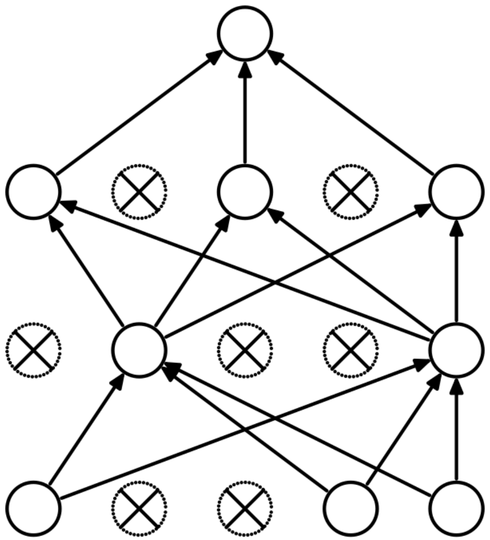
每个输出值都由全部的输入值和全连接层的一行做点乘计算得来。

Dropout

- 当特征很多而样本不足时，模型容易过拟合。
- 模型对微小扰动需要具有鲁棒性。
- Dropout：
  - 在训练过程的每一次迭代中，以p的概率随机丢弃一些神经元。
  - 保留的神经元需要除以1-p，以保持一层的期望不变。
  - $h' \begin{cases} 0 - \text{概率为} p \\ \frac{h}{1-p} - \text{其他情况} \end{cases}$
  - 在测试过程，使用全部的神经元。

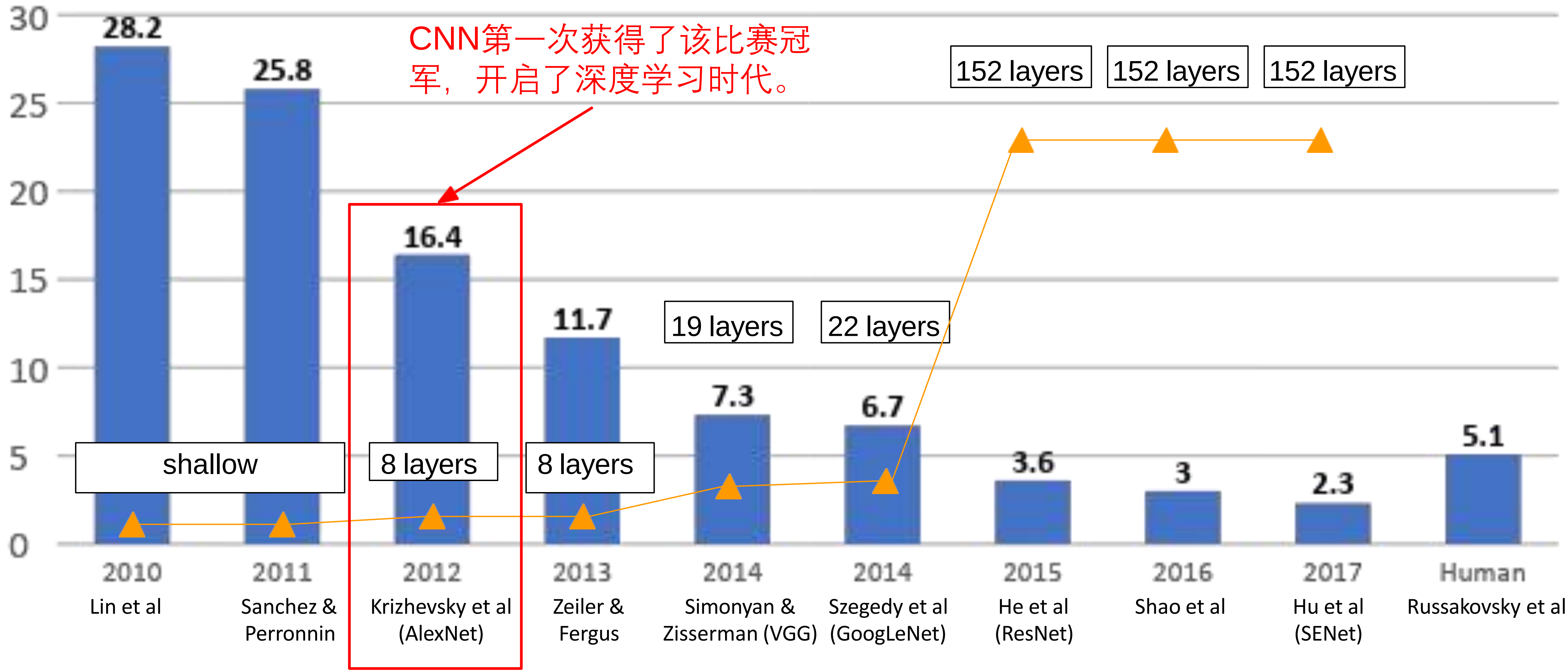


(a) Standard Neural Net



(b) After applying dropout.

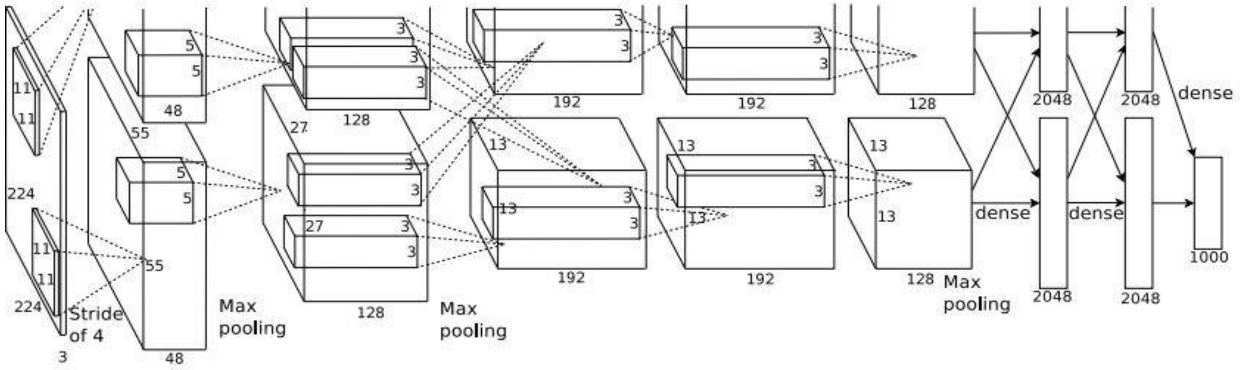
ImageNet视觉识别挑战赛



AlexNet-开启CNN时代

[Krizhevsky et al. 2012]

Full (simplified) AlexNet architecture:
[227x227x3] INPUT
[55x55x96] CONV1: 96 11x11 filters at stride 4, pad 0
[27x27x96] MAX POOL1: 3x3 filters at stride 2
[27x27x96] NORM1: Normalization layer
[27x27x256] CONV2: 256 5x5 filters at stride 1, pad 2
[13x13x256] MAX POOL2: 3x3 filters at stride 2
[13x13x256] NORM2: Normalization layer
[13x13x384] CONV3: 384 3x3 filters at stride 1, pad 1
[13x13x384] CONV4: 384 3x3 filters at stride 1, pad 1
[13x13x256] CONV5: 256 3x3 filters at stride 1, pad 1
[6x6x256] MAX POOL3: 3x3 filters at stride 2
[4096] FC6: 4096 neurons
[4096] FC7: 4096 neurons
[1000] FC8: 1000 neurons (class scores)

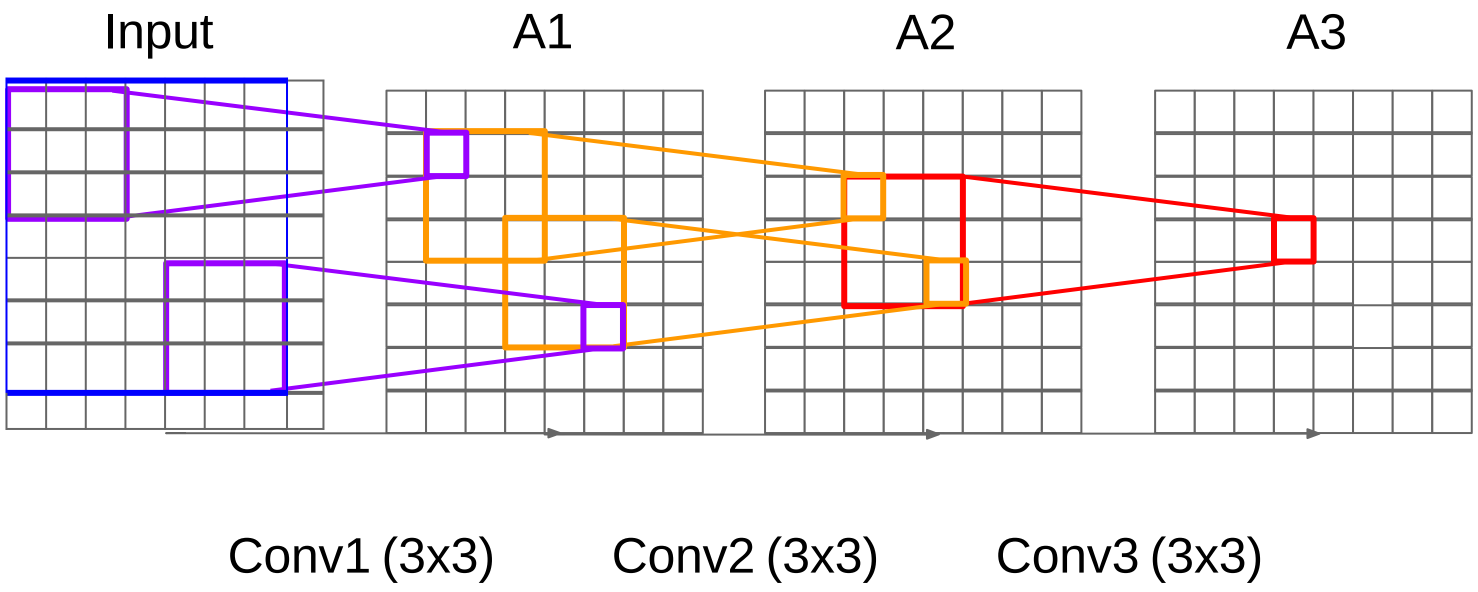


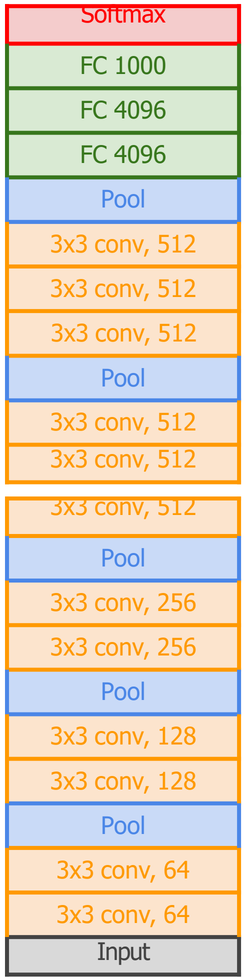
- 在只有3GB的GTX 580 GPU 上进行训练。
- 实现了2卡并行计算。

VGGNet-深度越来越深

[Simonyan and Zisserman, 2014]

堆叠三个3x3卷积可以获得和7x7卷积相同的感受。





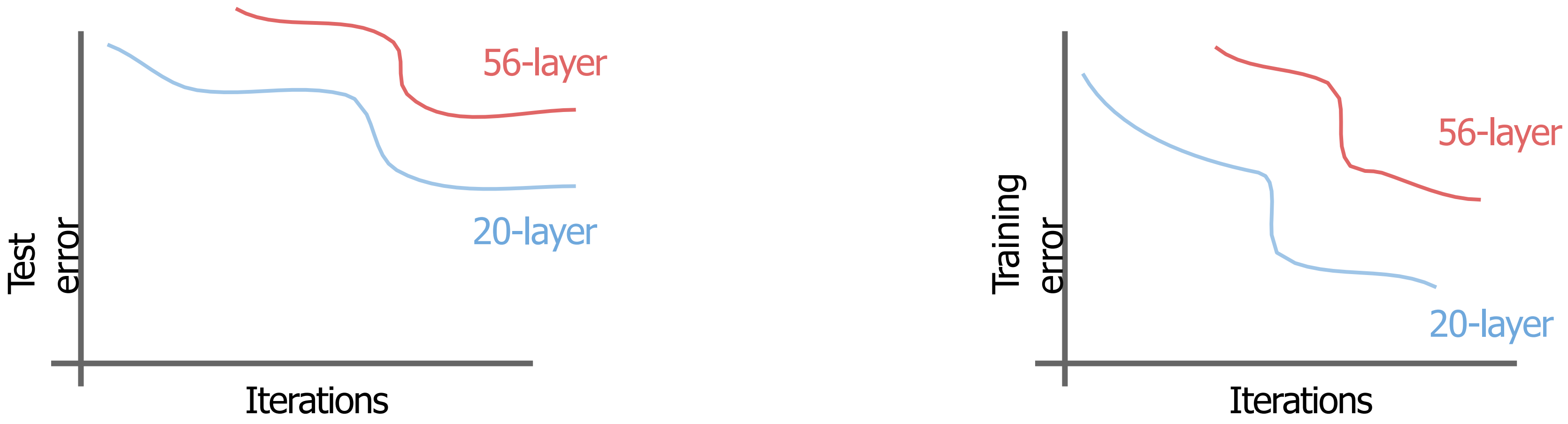
VGG16

VGG19

ResNet-残差连接

[He et al., 2015]

当深度足够深后，继续增加深度，模型表现反而会下降



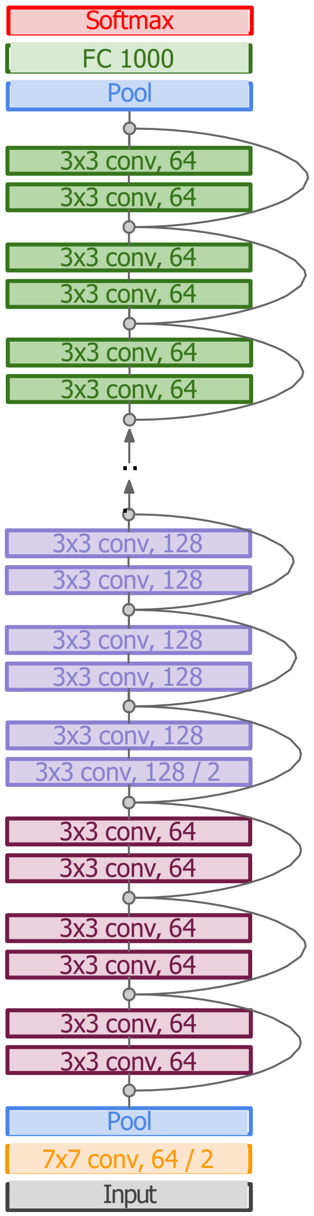
56层的模型的训练误差和测试误差都高于20层
->更深的网络，训练表现不好，说明不是过拟合导致的
->深层网络更难训练

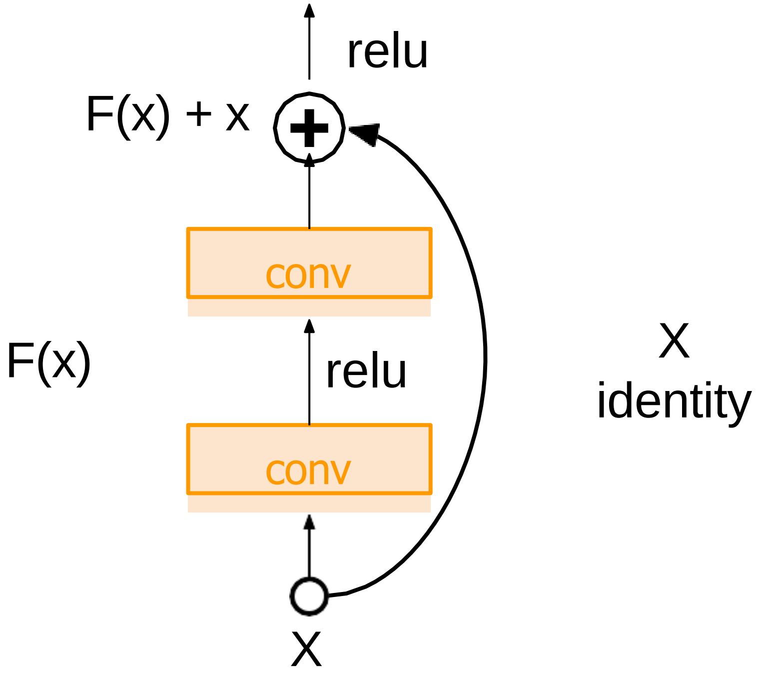
ResNet

[He et al., 2015]

使用残差连接训练深层网络

- ResNet-152 赢得了ImageNet ILSVRC’15 挑战赛冠军。仅仅只有3.57% top 5 error。
- 在ILSVRC’15 和 COCO’15, 横扫了其他分类任务和检测任务方法。





Residual block

谢谢

北京大学

PEKING UNIVERSITY