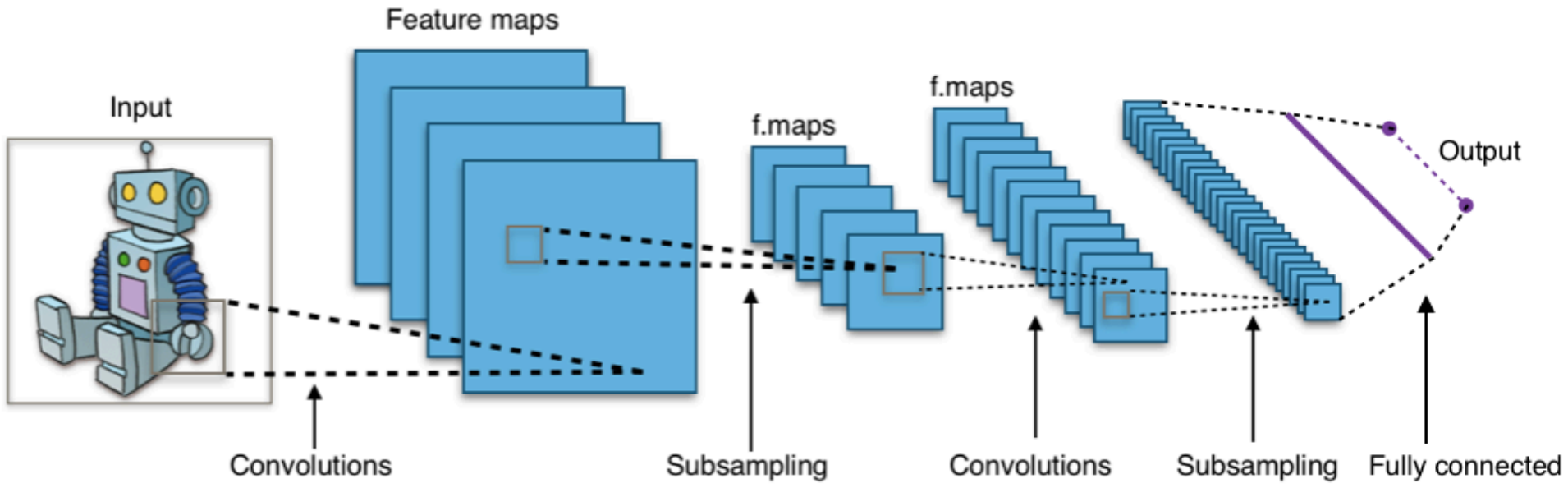
卷积神经网络

Convolutional neural network

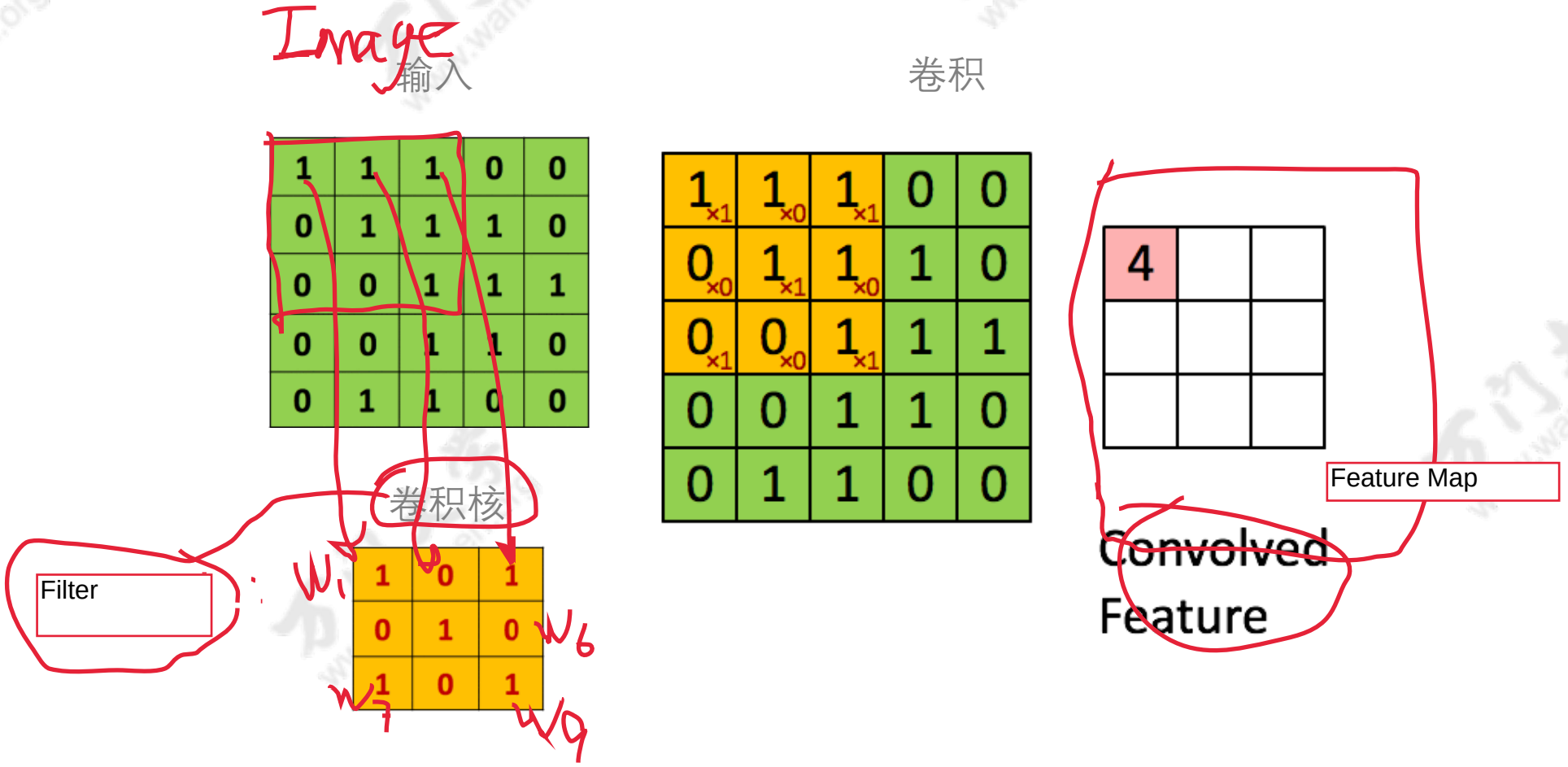
卷积神经网络



https://en.wikipedia.org/wiki/Convolutional_neural_network

Convolution

卷积



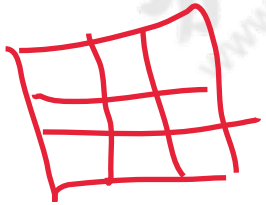
Convolution

卷积

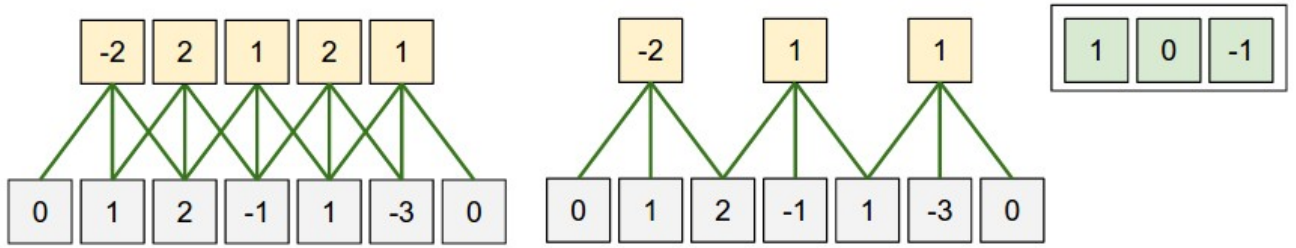
Filter

卷积核

- 尺寸(Size): 卷积核大小(F) e.g. 3x3



- 深度(Depth): 卷积核个数(K) will determine how many feature maps you get
- 步长(Stride): 卷积核每次移动的步数(像素点个数)(S)
- 零填充(Zero-padding): 控制输出图片大小(P) **在**input image matrix **周**围padding 0s, so that output size is the same as input matrix size.



http://cs231n.github.io/convolutional-networks/#architectures

Filter

卷积核

- 输入: $W_1 \times H_1 \times D_1$ input matrix size
- 输出:$W_2 \times H_2 \times D_2$ output matrix size

RELATIONSHIP BETWEEN INPUT AND OUTPUT MATRIX SIZE

$$W_2 = \frac{W_1 - F + 2P}{S} + 1$$

$$H_2 = \frac{W_2 - F + 2P}{S} + 1$$

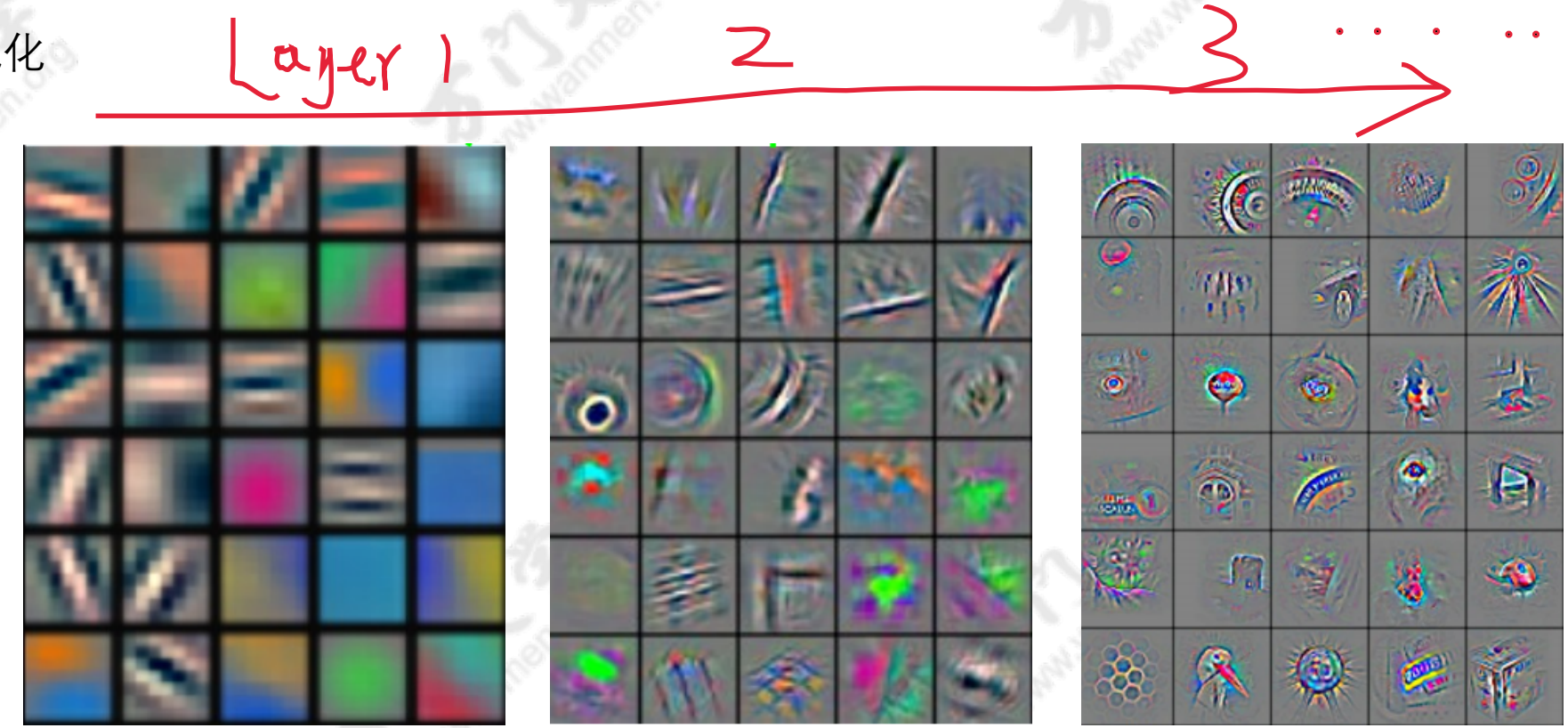
$$D_2 = K$$

- 参数: $(F \cdot F \cdot D_1) \cdot K$ 权重和K偏差

Note: CNN can be understood as NN (Dense Network) + Regularization

Filter Visualization

卷积核可视化



https://arxiv.org/pdf/1311.2901.pdf

Padding

填充

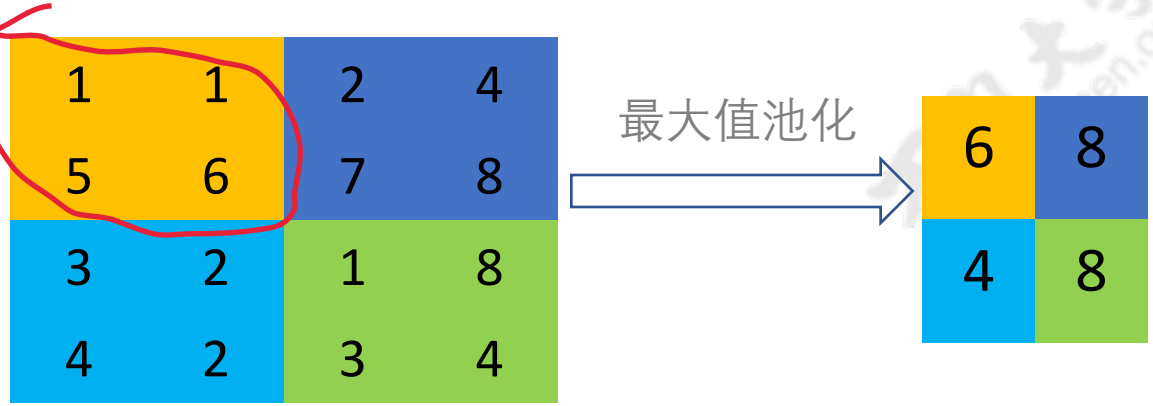
0	0	0	0	0	0
0	32	26	17	34	0
0	2	78	45	23	0
0	4	67	90	12	0
0	68	32	57	25	0
0	0	0	0	0	0

Pooling

Its function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network.

池化

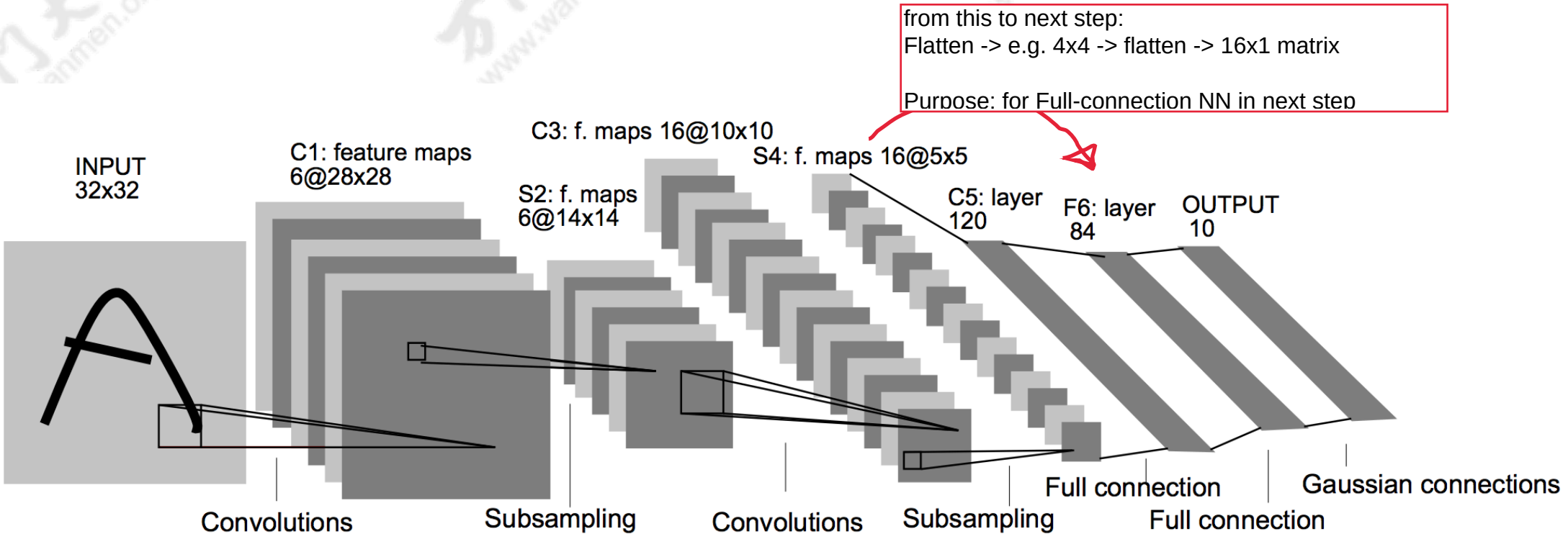
- 最大值池化(Max Pooling) $\max(x_1, x_2, \cdots, x_n)$
- 平均值池化(Average Pooling) $\text{mean}(x_1, \cdots, x_n)$
- L2-范数池化(L2-norm Pooling) $\sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$



For example:
max pooling for this pool = 6 (max value of this pool)

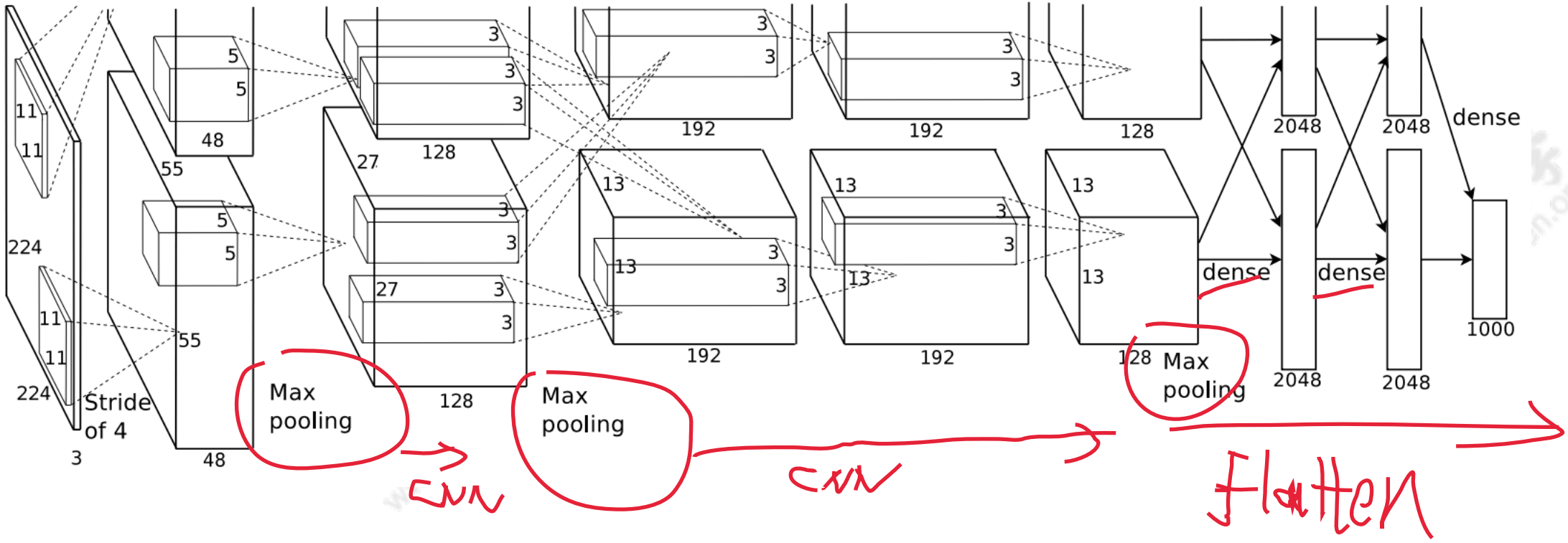
Here:
Each pool has size 2x2, therefore, original matrix (4x4) has 4 pools that are not overlapped.

深度卷积神经网络:LeNet-5



http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf

深度卷积神经网络:AlexNet

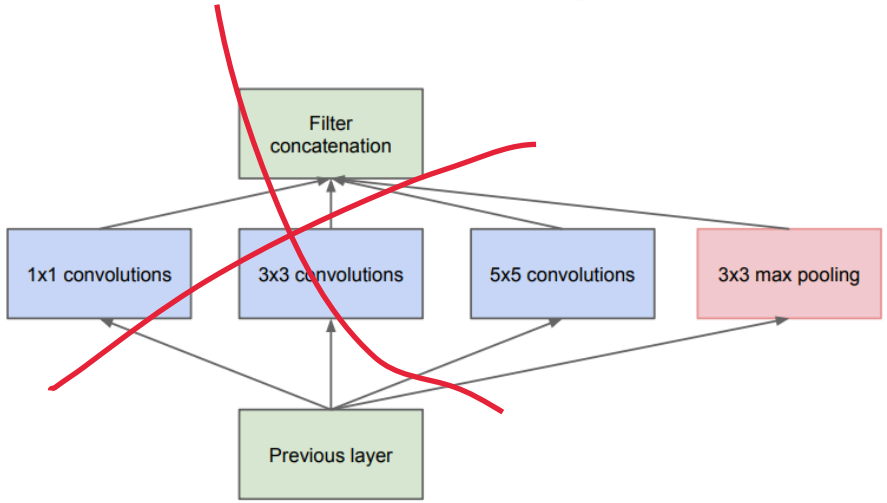


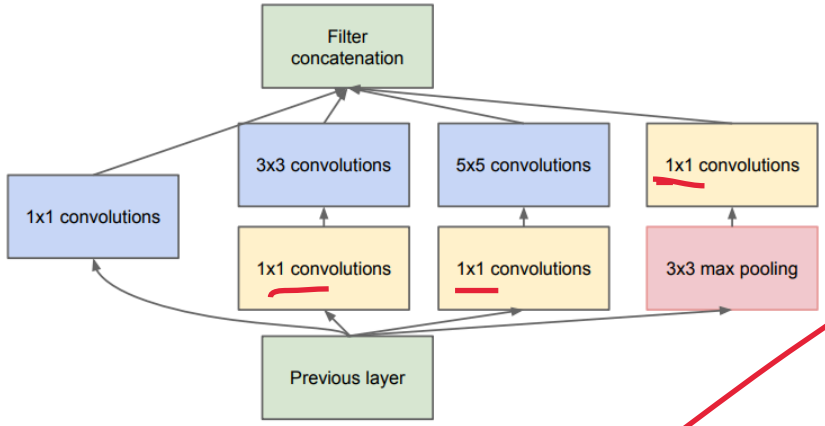
ImageNet Classification with Deep Convolutional Neural Networks

深度卷积神经网络：VGG-16

Very Deep Convolutional Networks for Large-scale Image Recognition

深度卷积神经网络 : Inception Network

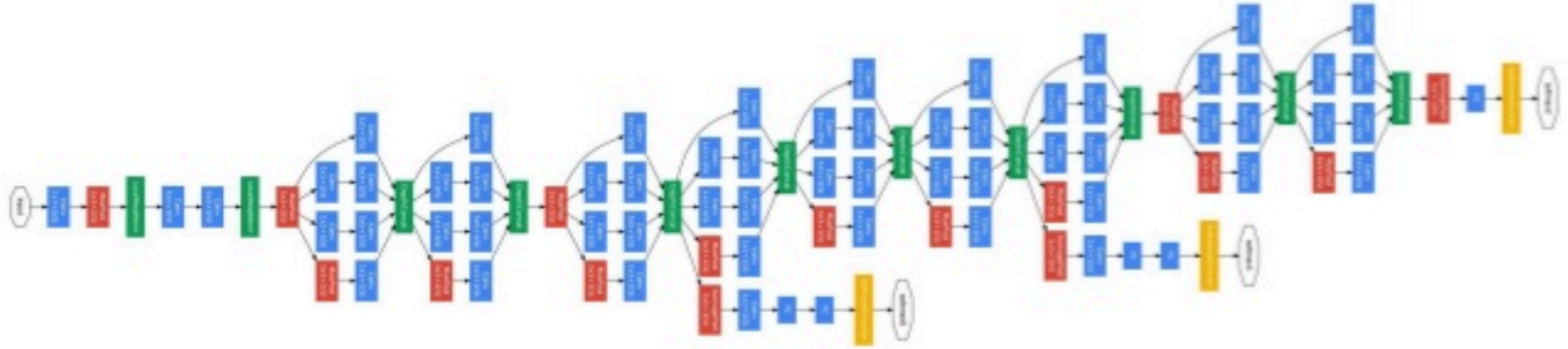




1x1 filter: can reduce the dimension of the matrix (reducing the # of parameters) by only choosing certain outputs over others.

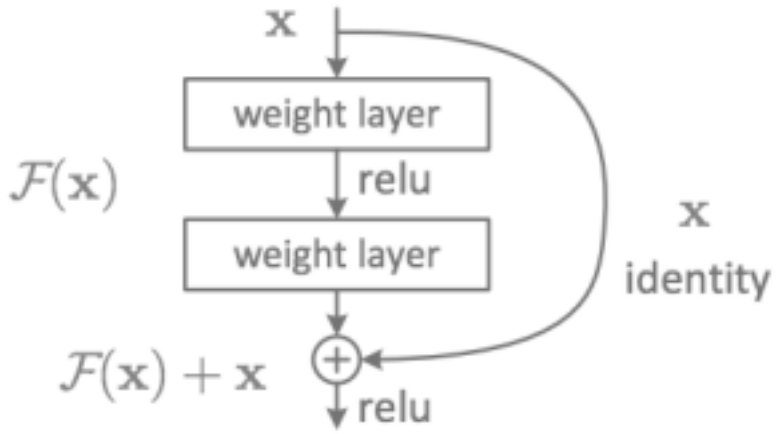
Going deeper with convolutions

深度卷积神经网络 : Inception Network



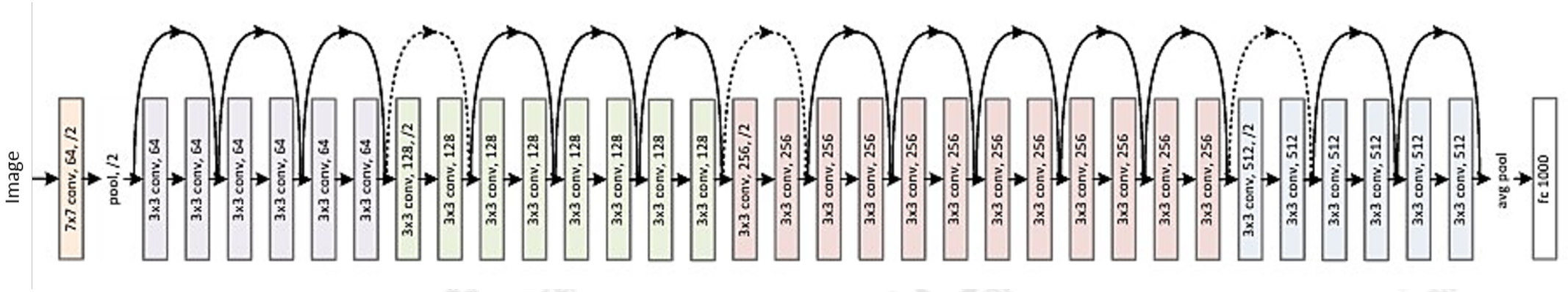
Going deeper with convolutions

深度卷积神经网络 : Residual Network



Deep Residual Learning for Image Recognition

深度卷积神经网络：Residual Network



Deep Residual Learning for Image Recognition

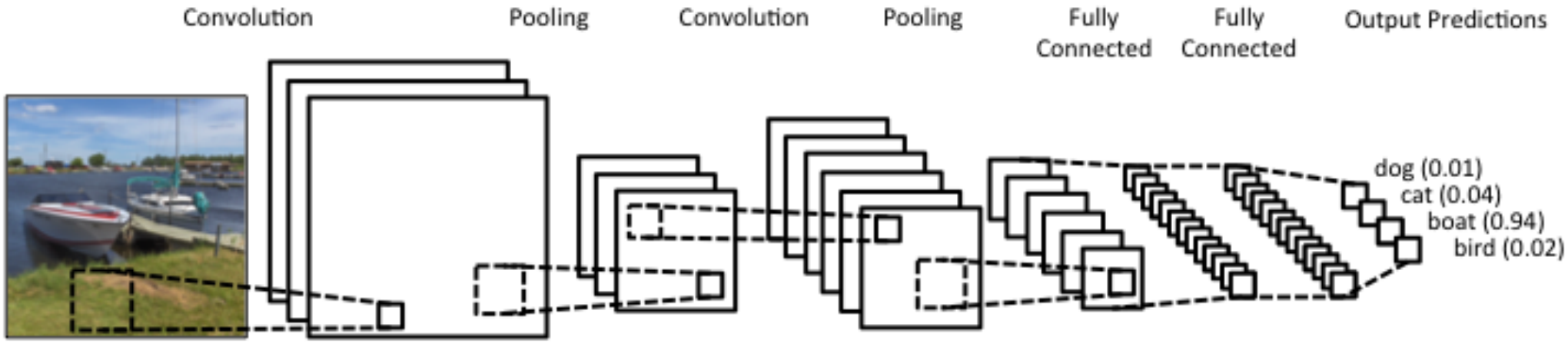
Addition:

DenseNet:
each layer is connected to all deeper layers

卷积神经网络的应用

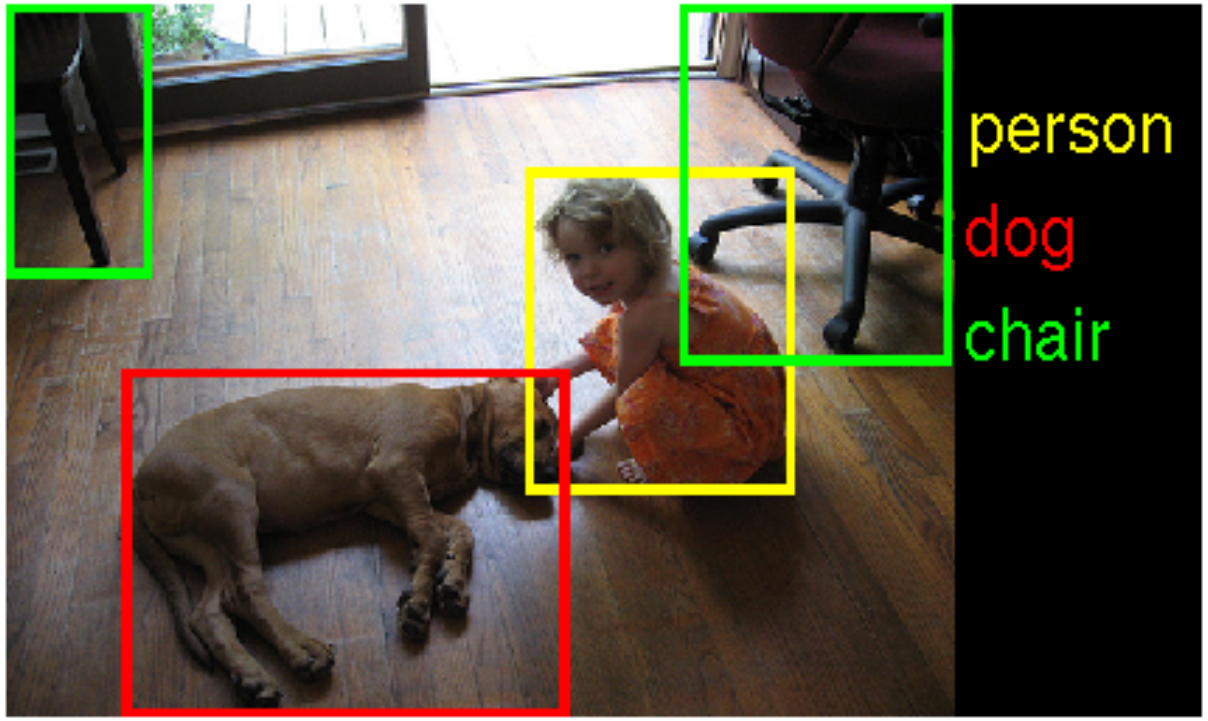
卷积网络的应用：物体分类

Input: one image: ->
Output: category of that image



卷积网络的应用：Object Detection

There may be many objects in one image, output needs to detect and label those objects.



物体检测：传统方法

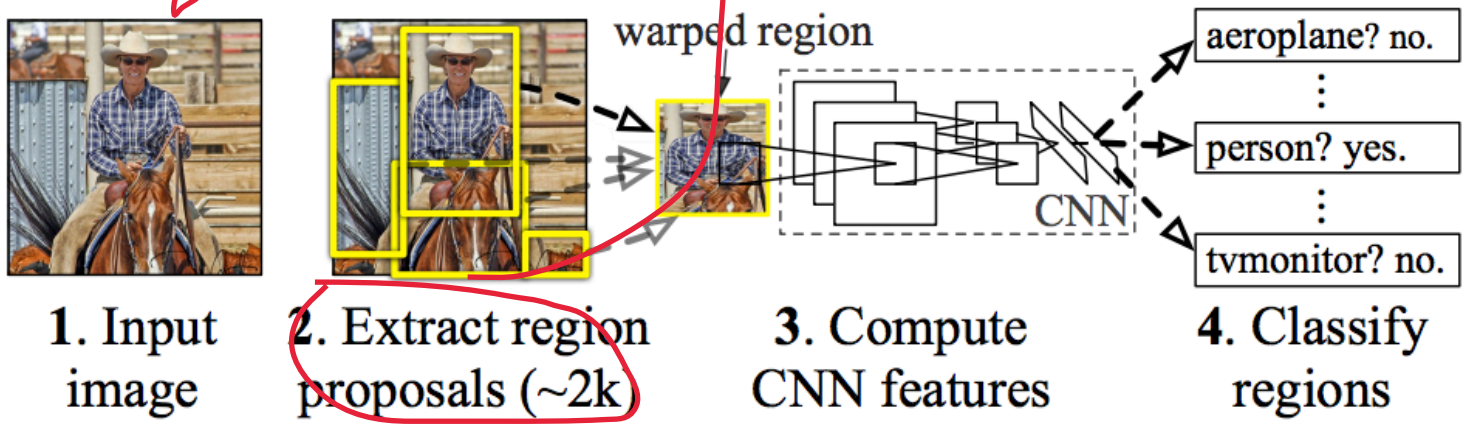

regional proposal
区域选择-
滑动窗口
特征提取-
SIFT
HOG
分类器分类-
SVM
Adaboost
In CNN method, we don't need to manually do feature extraction, instead, the NN itself can do it for us.

物体检测：RCNN

RCNN replace the step 2 (feature extraction) in traditional method (last page) to CNN

- Selective search算法在图像中提取2000个左右的region proposal
- 每个region proposal缩放成227x227的大小并输入到CNN
- AlexNet for CNN
- CNN的fc7的输出作为特征
- CNN特征输入到SVM进行分类

regional proposal (a.k.a. ROI: region of interest): i.e. each yellow block in the below picture.



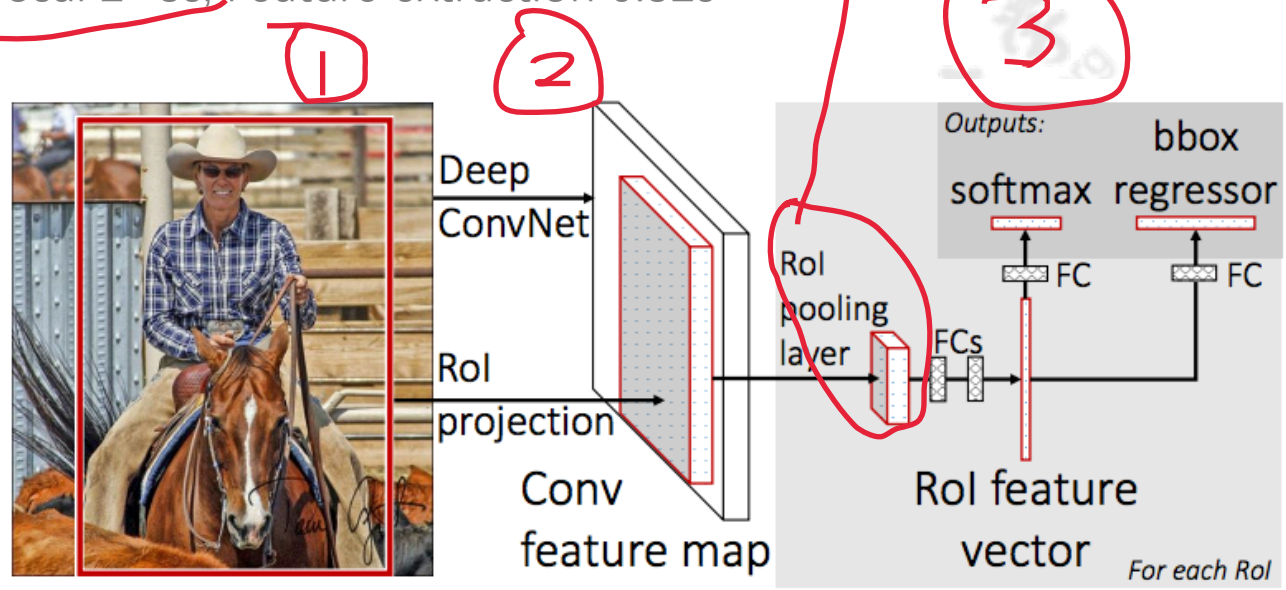
Rich feature hierarchies for accurate object detection and semantic segmentation

物体检测：Fast R-CNN

Fast R-CNN replace step 2 and 3 in traditional method to a NN network.

- 最后一个卷积层加入一个ROI pooling layer
- 损失函数使用多任务损失函数(multi-task loss)
- 接近端到端训练
- Region proposal 2~3s, Feature extraction 0.32s

It is close to end to end model, but regional proposal is still using traditional method (take longer to calculate), which is not a part of NN.

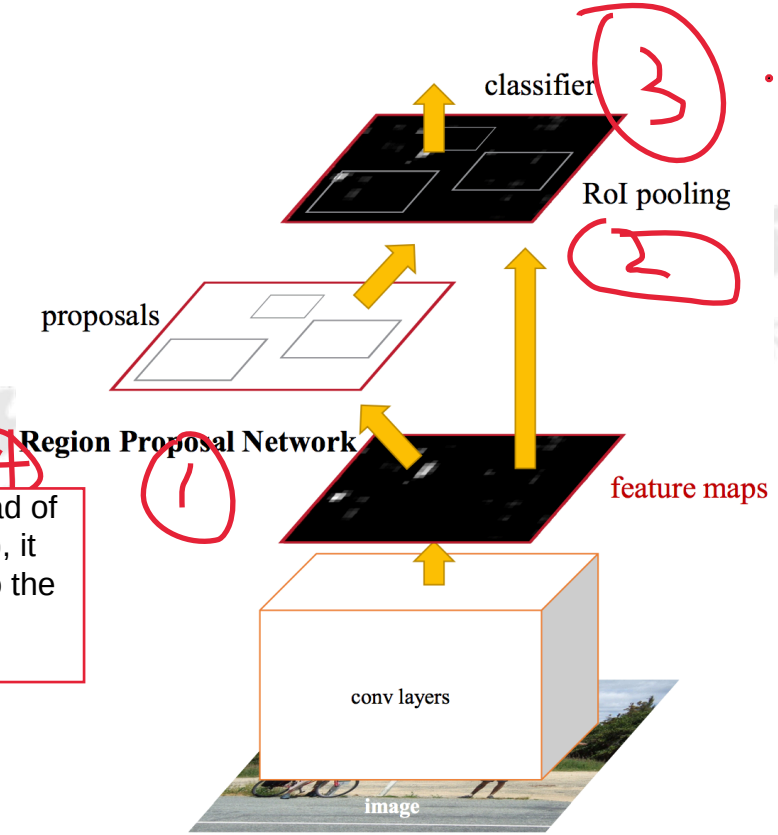


Fast R-CNN

物体检测：Faster R-CNN

Faster R-CNN replace all three steps in traditional method to NN network.

- Region Proposal Network
- 损失函数使用多任务损失函数(multi-task loss)
- 端到端训练
- 目标检测0.2s



Use Regional Proposal Network instead of Regional Proposal (traditional method), it can include regional proposal step into the NN, it is faster than traditional regional proposal

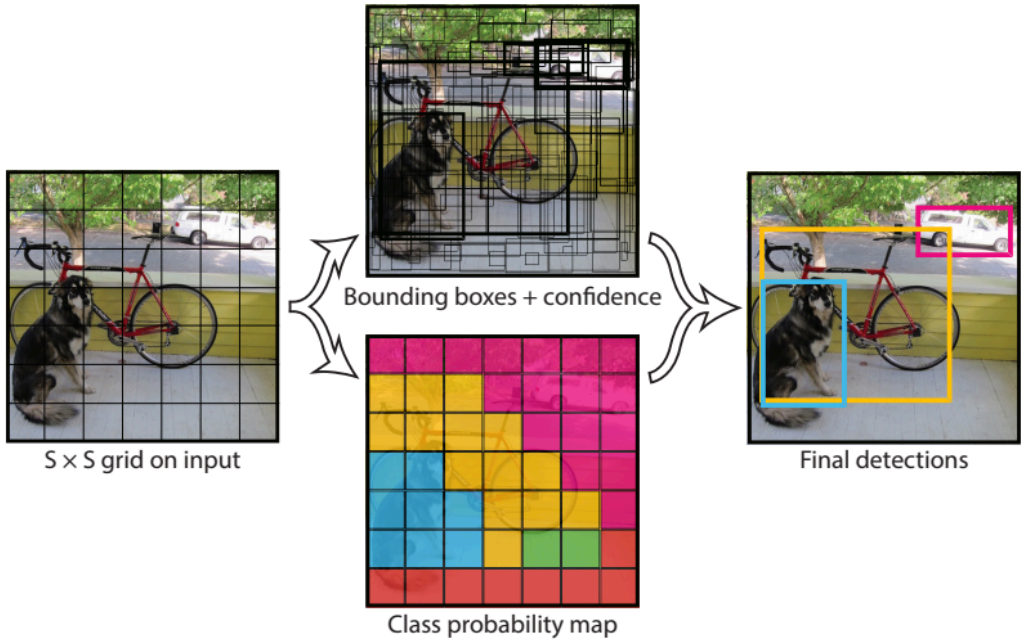
Faster R-CNN

物体检测：YOLO

Instead of doing regional proposal, YOLO just split a image to 7x7

Advantage: Fast
Disadvantage: not that accurate.

- 将图像划分为7x7的网格
- 对每个网格，预测两个边框(每个边框是目标的概率以及每个边框区域在多个类别上的概率
- 根据阈值去除可能性比较低的目标窗口
- 目标检测0.02s

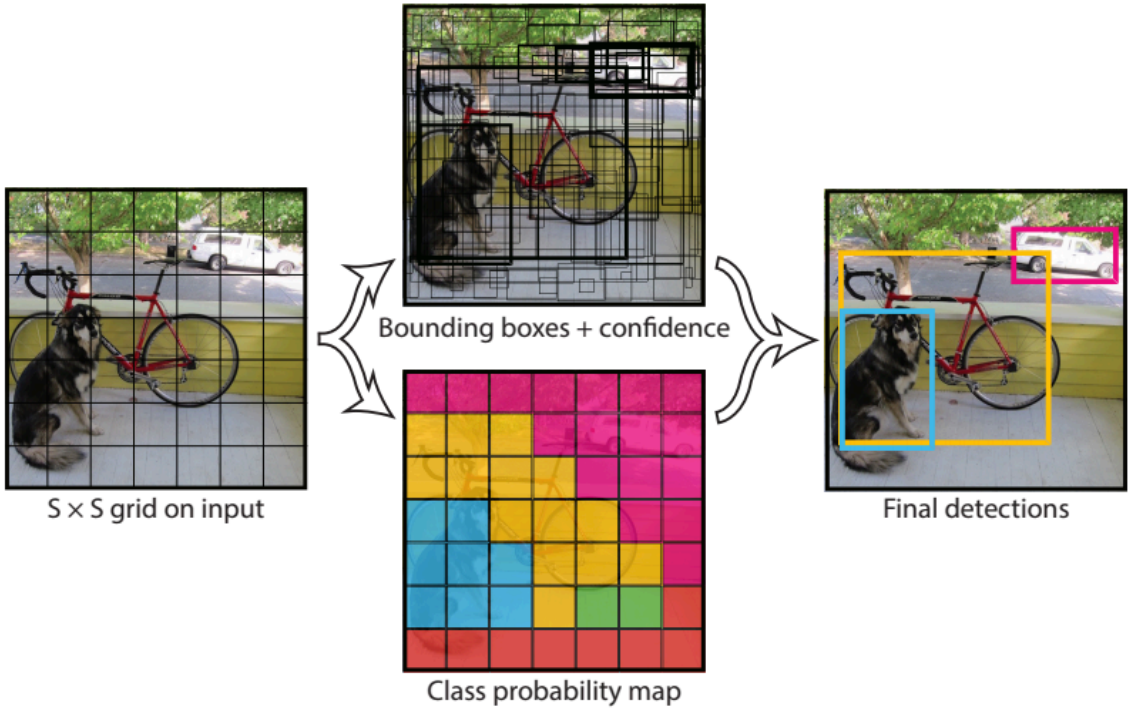


You Only Look Once: Unified, Real-Time Object Detection

物体检测：SSD

improved based on YOLO

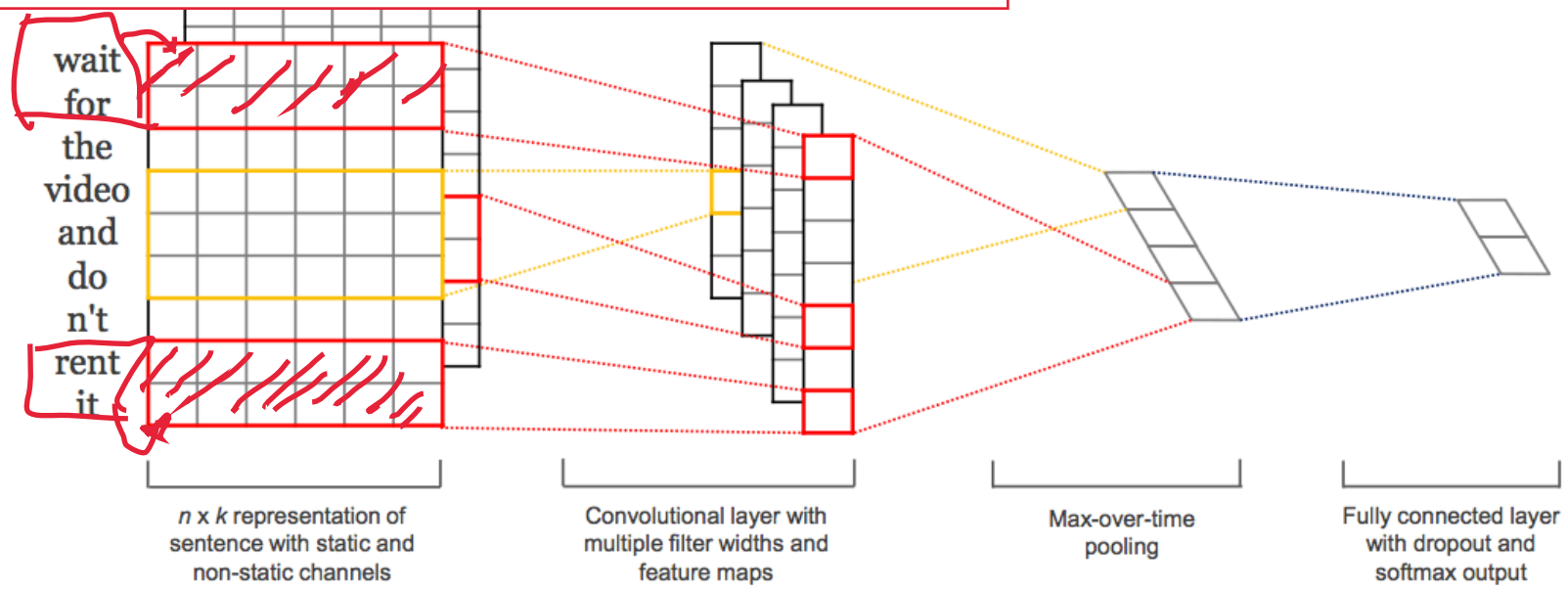
- 使用目标位置周围特征来预测目标位置
- 使用Faster R-CNN的anchor机制
- 目标检测0.02s



SSD: Single Shot MultiBox Detector

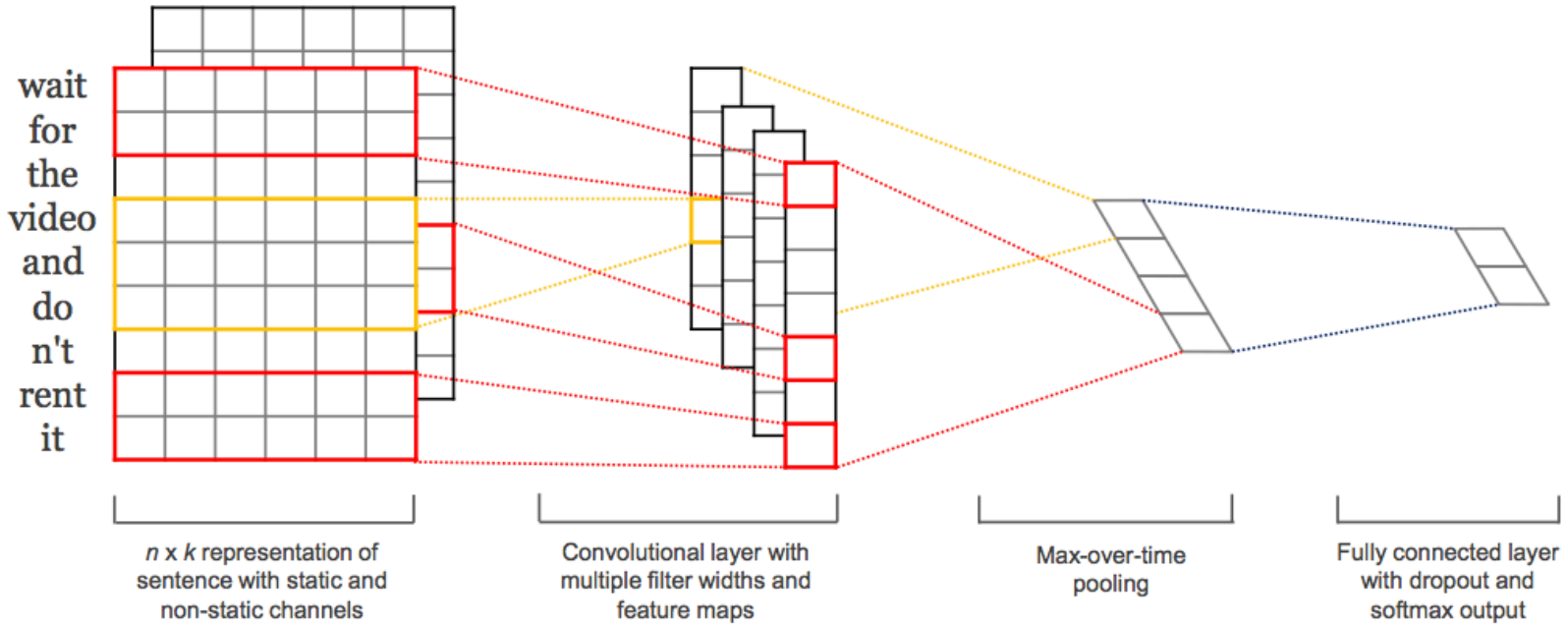
卷积网络的应用：文档的归类

Each word or words (n-gram) can be converted to a fixed sized vector (e.g. 10 x1);
And then we can do convolution vector(s) by vector(s).
Note: we cannot do convolution by splitting one vector to smaller vectors because one vector represents a word (e.g. we cannot split 10x1 vector to two 5x1 vectors)!!!!!!



Convolutional Neural Networks for Sentence Classification

卷积网络的应用：文档的归类



Convolutional Neural Networks for Sentence Classification

THANKS.