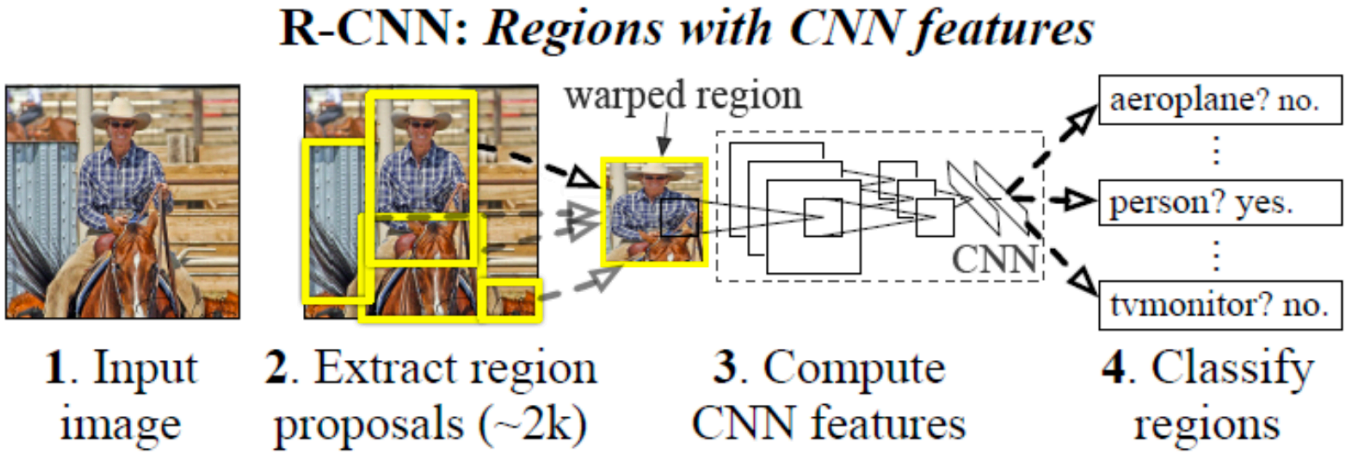
R-CNN系列

- Region-CNN的缩写，主要用于目标检测。
- 来自 2014 年 CVPR 论文“Rich feature hierarchies for accurate object detection and semantic segmentation”
- 在 Pascal VOC 2012 的数据集上，能够将目标检测的验证指标 mAP 提升到 53.3%，这相对于之前最好的结果提升了整整 30%
- 采用在ImageNet上已经训练好的模型，然后在PASCAL VOC数据集上进行 fine-tune

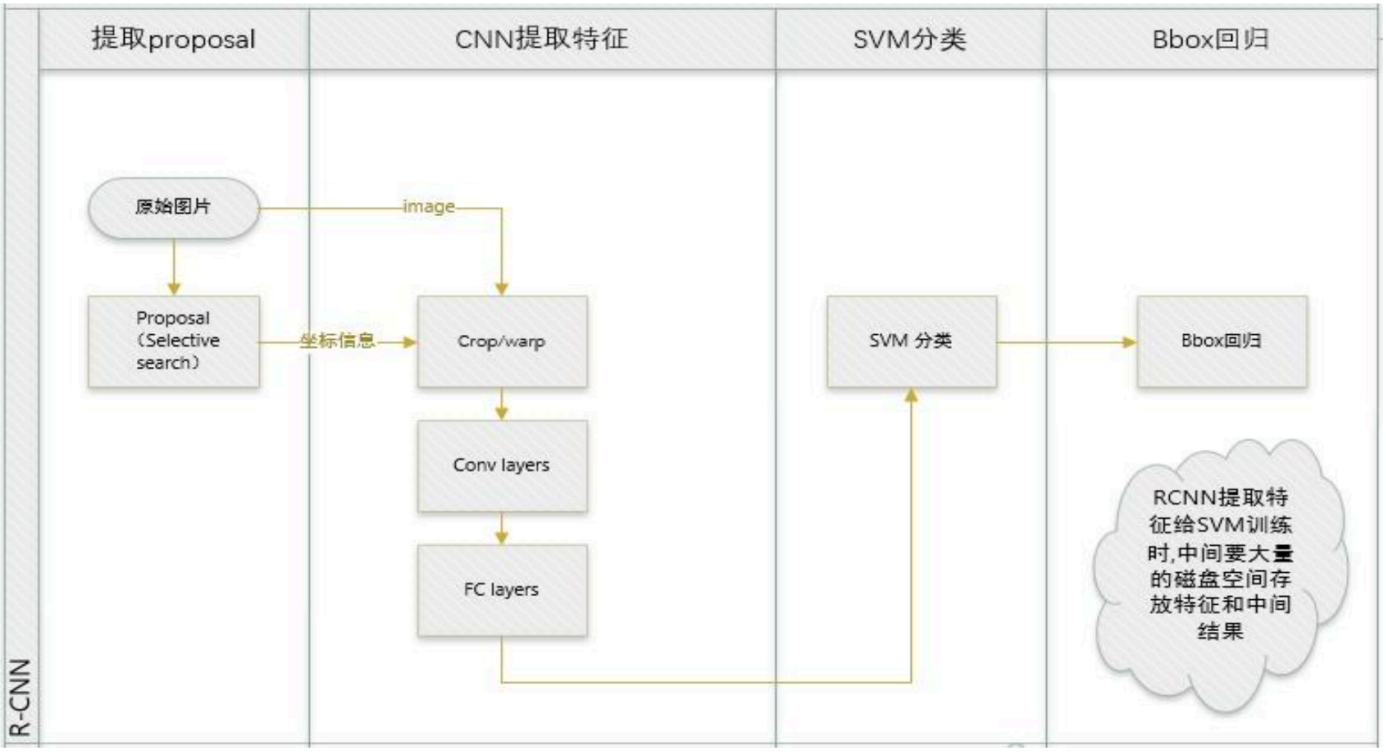


参考：Ross B. Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. CVPR 2014: 580-587

实现过程

- 区域划分:给定一张输入图片，从图片中提取2000个类别独立的候选区域，R-CNN 采用的是 Selective Search 算法
- 特征提取：对于每个区域利用CNN抽取一个固定长度的特征向量， R-CNN 使用的是 Alexnet
- 目标分类：再对每个区域利用SVM进行目标分类
- 边框回归：BoundingboxRegression(Bbox回归)进行边框坐标偏移

  优化和调整



- Crop就是从一个大图扣出网络输入大小的patch，比如227×227
- Warp把一个边界框bounding box的内容resize成227×227

Selective Search 算法

核心思想：图像中物体可能存在的区域应该有某些相似性或者连续性的，选择搜索基于上面这一想法采用子区域合并的方法提取 bounding boxes候选边界框。

- 首先，通过图像分割算法将输入图像分割成许多小的子区域
- 其次，根据这些子区域之间的相似性(主要考虑颜色、纹理、尺寸和 空间交叠4个相似) 进行区域迭代合并。每次迭代过程中对这些合并 的子区域做bounding boxes(外切矩形)，这些子区域的外切矩形就是 通常所说的候选框

算法步骤：

1. 生成区域集R，参见论文《Efficient Graph-Based Image Segmentation》
2. 计算区域集R里每个相邻区域的相似度$S = \{s1, s2, \dots\}$
3. 找出相似度最高的两个区域，将其合并为新集，添加进R
4. 从S中移除所有与step2中有关的子集
5. 计算新集与所有子集的相似度
6. 跳至step2，直至S为空

Algorithm 1: Hierarchical Grouping Algorithm

Input: (colour) image
Output: Set of object location hypotheses L

Obtain initial regions $R = \{r_1, \cdots, r_n\}$ using [13]
Initialise similarity set $S = \emptyset$
foreach *Neighbouring region pair* (r_i, r_j) **do**
    Calculate similarity $s(r_i, r_j)$
    $S = S \cup s(r_i, r_j)$

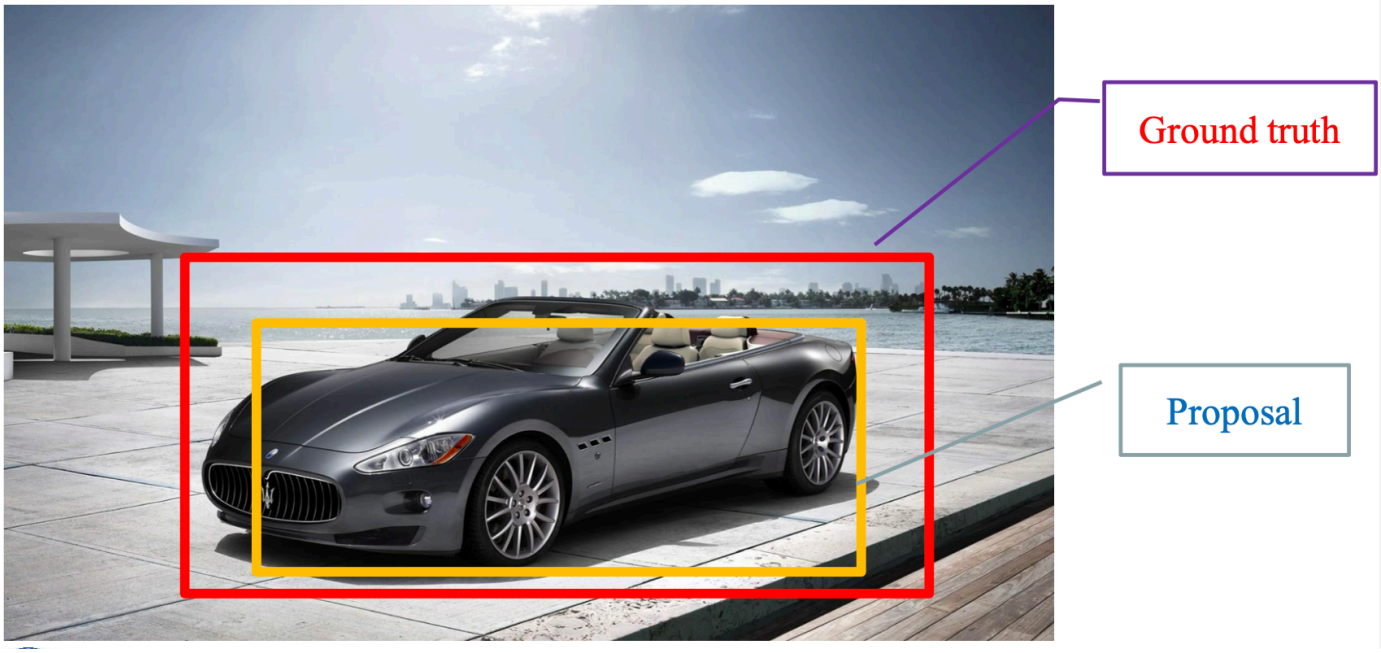
while $S \neq \emptyset$ **do**
    Get highest similarity $s(r_i, r_j) = \max(S)$
    Merge corresponding regions $r_t = r_i \cup r_j$
    Remove similarities regarding $r_i : S = S \setminus s(r_i, r_*)$
    Remove similarities regarding $r_j : S = S \setminus s(r_*, r_j)$
    Calculate similarity set S_t between r_t and its neighbours
    $S = S \cup S_t$
    $R = R \cup r_t$

Extract object location boxes L from all regions in R

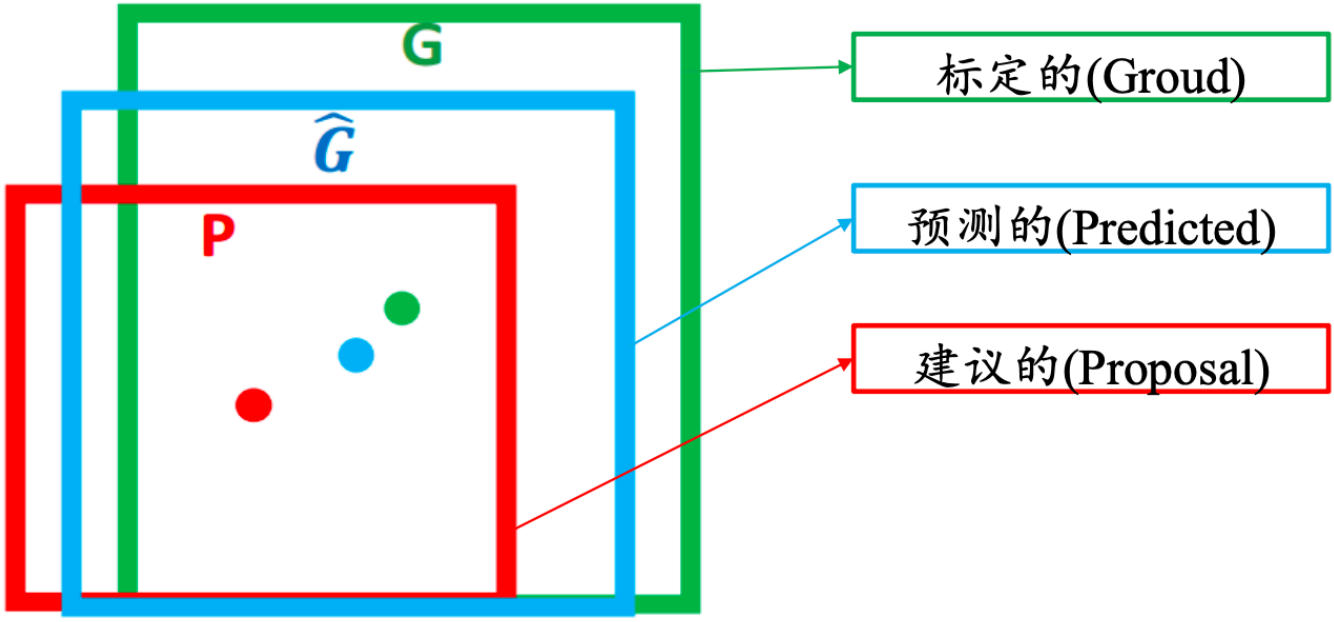
(a) (b)

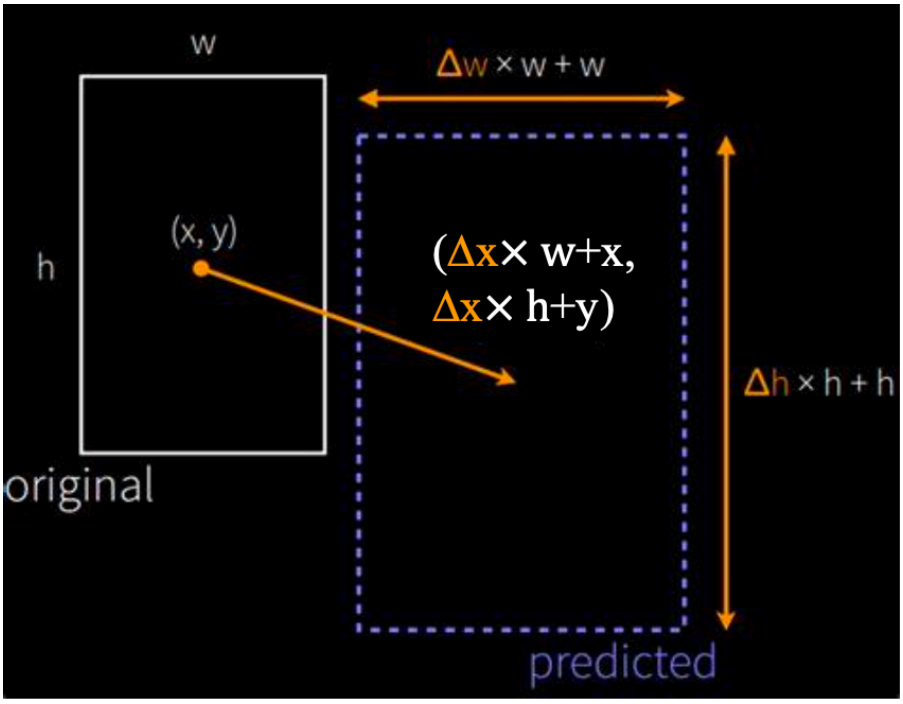
Bbox回归

核心思想：通过平移和缩放方法对物体边框进行调整和修正。



- bounding box的表示为(x, y, w, h)，即窗口的中心点坐标和宽高
- Bbox回归就是找到函数 f，将(P_x, P_y, P_w, P_h)映射为更接近 (G_x, G_y, G_w, G_h) 的 $(\hat{G}_x, \hat{G}_y, \hat{G}_w, \hat{G}_h)$





$$\hat{G}_x = P_w d_x(P) + P_x$$
$$\hat{G}_y = P_h d_y(P) + P_y$$
$$\hat{G}_w = P_w \exp\left(d_w(P)\right)$$
$$\hat{G}_h = P_h \exp\left(d_h(P)\right)$$

VOC 2010 test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
DPM v5 [17]†	49.2	53.8	13.1	15.3	35.5	53.4	49.7	27.0	17.2	28.8	14.7	17.8	46.4	51.2	47.7	10.8	34.2	20.7	43.8	38.3	33.4
UVA [32]	56.2	42.4	15.3	12.6	21.8	49.3	36.8	46.1	12.9	32.1	30.0	36.5	43.5	52.9	32.9	15.3	41.1	31.8	47.0	44.8	35.1
Regionlets [35]	65.0	48.9	25.9	24.6	24.5	56.1	54.5	51.2	17.0	28.9	30.2	35.8	40.2	55.7	43.5	14.3	43.9	32.6	54.0	45.9	39.7
SegDPM [15]†	61.4	53.4	25.6	25.2	35.5	51.7	50.6	50.8	19.3	33.8	26.8	40.4	48.3	54.4	47.1	14.8	38.7	35.0	52.8	43.1	40.4
R-CNN	67.1	64.1	46.7	32.0	30.5	56.4	57.2	65.9	27.0	47.3	40.9	66.6	57.8	65.9	53.6	26.7	56.5	38.1	52.8	50.2	50.2
R-CNN BB	**71.8**	**65.8**	**53.0**	**36.8**	**35.9**	**59.7**	**60.0**	**69.9**	**27.9**	**50.6**	**41.4**	**70.0**	**62.0**	**69.0**	**58.1**	**29.5**	**59.4**	**39.3**	**61.2**	**52.4**	**53.7**

Table 1: Detection average precision (%) on VOC 2010 test. R-CNN is most directly comparable to UVA and Regionlets since all methods use selective search region proposals. Bounding box regression (BB) is described in Section 3.4. At publication time, SegDPM was the top-performer on the PASCAL VOC leaderboard. †DPM and SegDPM use context rescoring not used by the other methods.

mAP：mean Average Precision，是多标签图像分类任务中的评价指标。AP衡量的是学出来的模型在给定类别上的好坏，而mAP衡量的是学出的模型在所有类别上的好坏。

SPPnet

SPPnet (Spatial Pyramid Pooling)：空间金字塔网络，R-CNN主要问题：每个Proposal独立提取CNN features，分步训练。

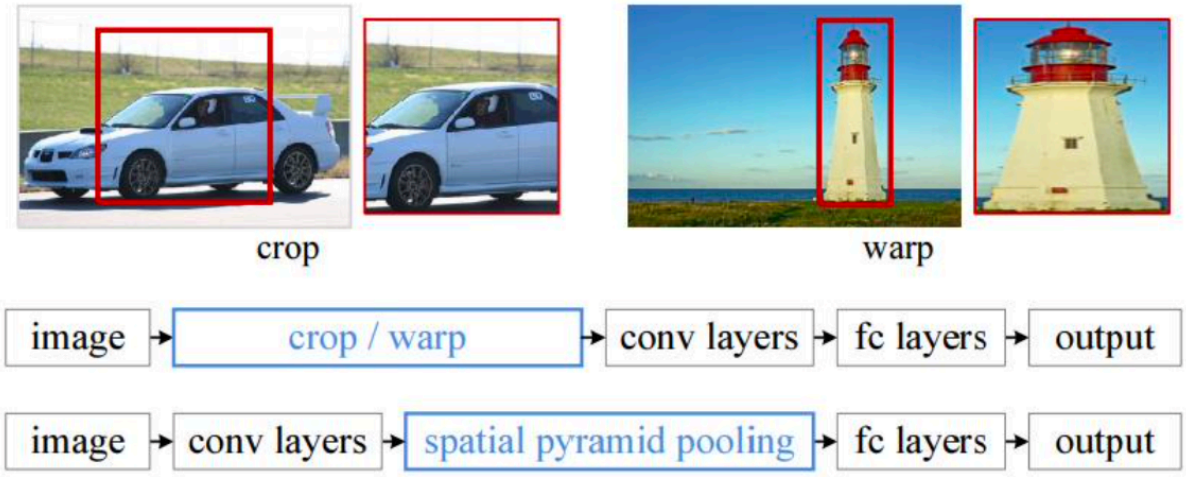


Figure 1: Top: cropping or warping to fit a fixed size. Middle: a conventional CNN. Bottom: our spatial pyramid pooling network structure.

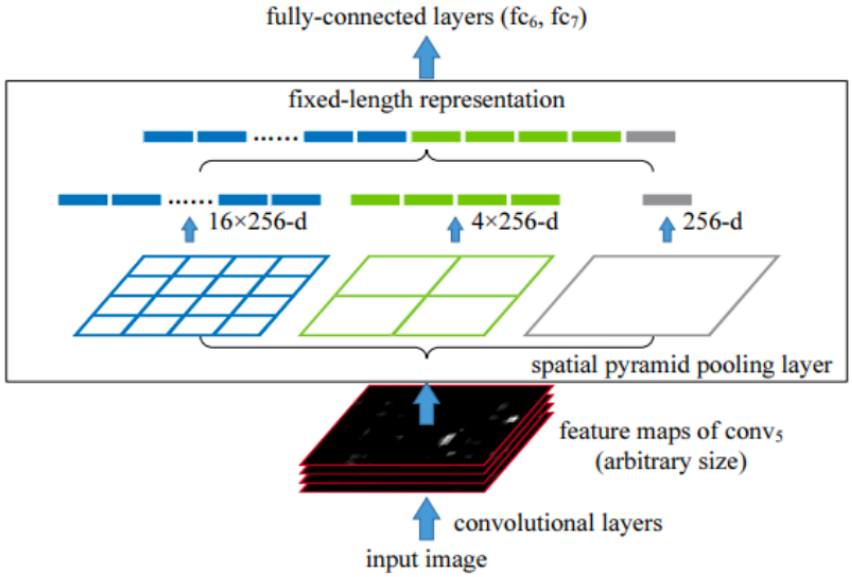
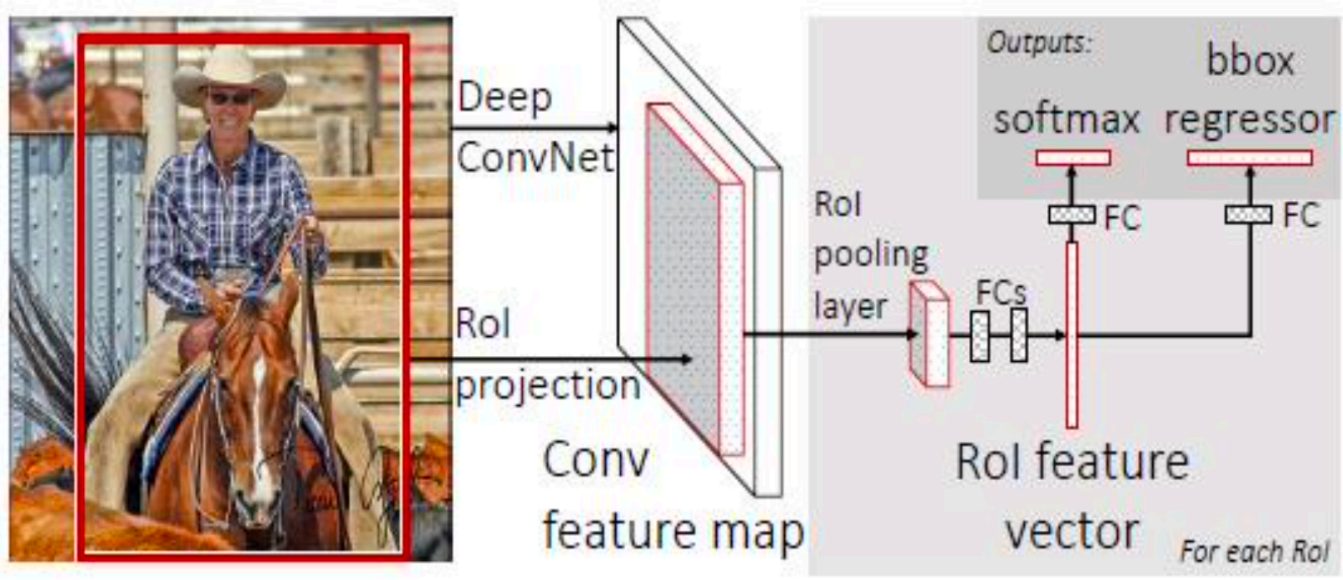


Figure 3: A network structure with a **spatial pyramid pooling layer**. Here 256 is the filter number of the $conv_5$ layer, and $conv_5$ is the last convolutional layer.

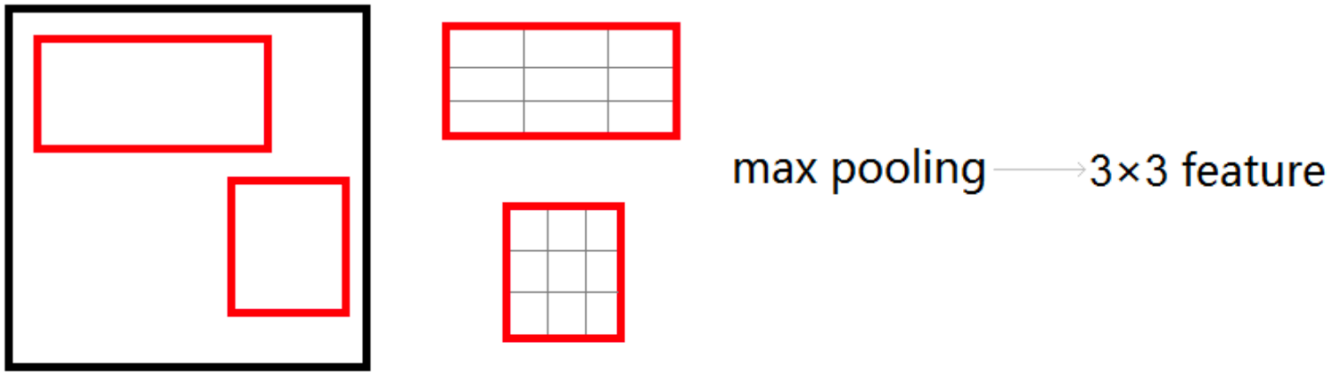
参考：Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. IEEE Trans. Pattern Anal. Mach. Intell. 37(9): 1904-1916 (2015)

Fast R-CNN

- R-CNN和SPPnet问题：训练步骤过多，需要训练SVM分类器，需要 额外的回归器, 特征也是保存在磁盘上。
- 联合学习(jointtraining)：把SVM、Bbox回归和CNN阶段一起训 练，最后一层的Softmax换成两个：一个是对区域的分类Softmax， 另一个是对Bounding box的微调。训练时所有的特征不再存到硬盘上，提升了速度。
- ROI Pooling层：实现了单尺度的区域特征图的Pooling。



ROI Pooling层：将每个候选区域均匀分成M×N块，对每块进行max pooling，将特征图上大小不一的候选区域转变为大小统一的数据，送入下一层。



性能对比

method	train set	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	mAP
SPPnet BB [11]†	07 \ diff	73.9	72.3	62.5	51.5	44.4	74.4	73.0	74.4	42.3	73.6	57.7	70.3	74.6	74.3	54.2	34.0	56.4	56.4	67.9	73.5	63.1
R-CNN BB [10]	07	73.4	77.0	63.4	45.4	**44.6**	75.1	78.1	79.8	40.5	73.7	62.2	79.4	78.1	73.1	64.2	**35.6**	66.8	67.2	70.4	**71.1**	66.0
FRCN [ours]	07	74.5	78.3	69.2	53.2	36.6	77.3	78.2	82.0	40.7	72.7	67.9	79.6	79.2	73.0	69.0	30.1	65.4	70.2	75.8	65.8	66.9
FRCN [ours]	07 \ diff	74.6	**79.0**	68.6	57.0	39.3	79.5	**78.6**	81.9	**48.0**	74.0	67.4	80.5	80.7	74.1	69.6	31.8	67.1	68.4	75.3	65.5	68.1
FRCN [ours]	07+12	**77.0**	78.1	**69.3**	**59.4**	38.3	**81.6**	**78.6**	**86.7**	42.8	**78.8**	**68.9**	**84.7**	**82.0**	**76.6**	**69.9**	31.8	**70.1**	**74.8**	**80.4**	70.4	**70.0**

Table 1. **VOC 2007 test** detection average precision (%). All methods use VGG16. Training set key: **07**: VOC07 trainval, **07 \ diff**: **07** without "difficult" examples, **07+12**: union of **07** and VOC12 trainval. †SPPnet results were prepared by the authors of [11].

method	train set	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	mAP
BabyLearning	Prop.	77.7	73.8	62.3	48.8	45.4	67.3	67.0	80.3	41.3	70.8	49.7	79.5	74.7	78.6	64.5	36.0	69.9	55.7	70.4	61.7	63.8
R-CNN BB [10]	12	79.3	72.4	63.1	44.0	44.4	64.6	66.3	84.9	38.8	67.3	48.4	82.3	75.0	76.7	65.7	35.8	66.2	54.8	69.1	58.8	62.9
SegDeepM	12+seg	**82.3**	75.2	67.1	50.7	**49.8**	71.1	69.6	88.2	42.5	71.2	50.0	85.7	76.6	81.8	69.3	**41.5**	**71.9**	62.2	73.2	**64.6**	67.2
FRCN [ours]	12	80.1	74.4	67.7	49.4	41.4	74.2	68.8	87.8	41.9	70.1	50.2	86.1	77.3	81.1	70.4	33.3	67.0	63.3	77.2	60.0	66.1
FRCN [ours]	07++12	82.0	**77.8**	**71.6**	**55.3**	42.4	**77.3**	**71.7**	**89.3**	**44.5**	**72.1**	**53.7**	**87.7**	**80.0**	**82.5**	**72.7**	36.6	68.7	**65.4**	**81.1**	62.7	**68.8**

Table 2. **VOC 2010 test** detection average precision (%). BabyLearning uses a network based on [17]. All other methods use VGG16. Training set key: **12**: VOC12 trainval, **Prop.**: proprietary dataset, **12+seg**: **12** with segmentation annotations, **07++12**: union of VOC07 trainval, VOC07 test, and VOC12 trainval.

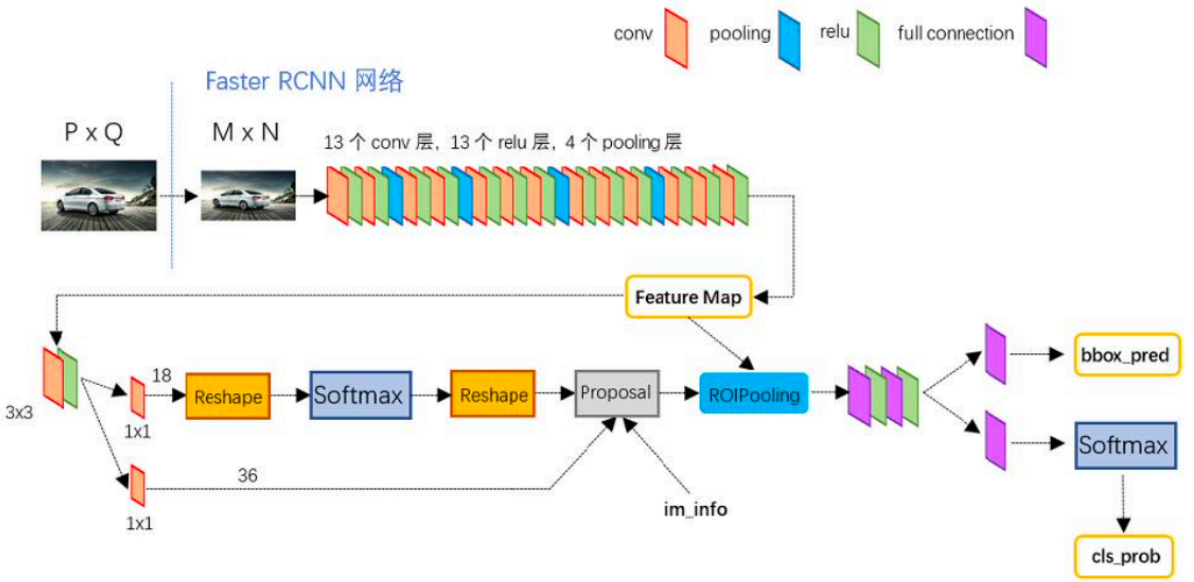
method	train set	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	persn	plant	sheep	sofa	train	tv	mAP
BabyLearning	Prop.	78.0	74.2	61.3	45.7	42.7	68.2	66.8	80.2	40.6	70.0	49.8	79.0	74.5	77.9	64.0	35.3	67.9	55.7	68.7	62.6	63.2
NUS_NIN_c2000	Unk.	80.2	73.8	61.9	43.7	**43.0**	70.3	67.6	80.7	41.9	69.7	51.7	78.2	75.2	76.9	65.1	**38.6**	**68.3**	58.0	68.7	63.3	63.8
R-CNN BB [10]	12	79.6	72.7	61.9	41.2	41.9	65.9	66.4	84.6	38.5	67.2	46.7	82.0	74.8	76.0	65.2	35.6	65.4	54.2	67.4	60.3	62.4
FRCN [ours]	12	80.3	74.7	66.9	46.9	37.7	73.9	68.6	87.7	41.7	71.1	51.1	86.0	77.8	79.8	69.8	32.1	65.5	63.8	76.4	61.7	65.7
FRCN [ours]	07++12	**82.3**	**78.4**	**70.8**	**52.3**	38.7	**77.8**	**71.6**	**89.3**	**44.2**	**73.0**	**55.0**	**87.5**	**80.5**	**80.8**	**72.0**	35.1	**68.3**	**65.7**	**80.4**	**64.2**	**68.4**

Table 3. **VOC 2012 test** detection average precision (%). BabyLearning and NUS_NIN_c2000 use networks based on [17]. All other methods use VGG16. Training set key: see Table 2, **Unk.**: unknown.

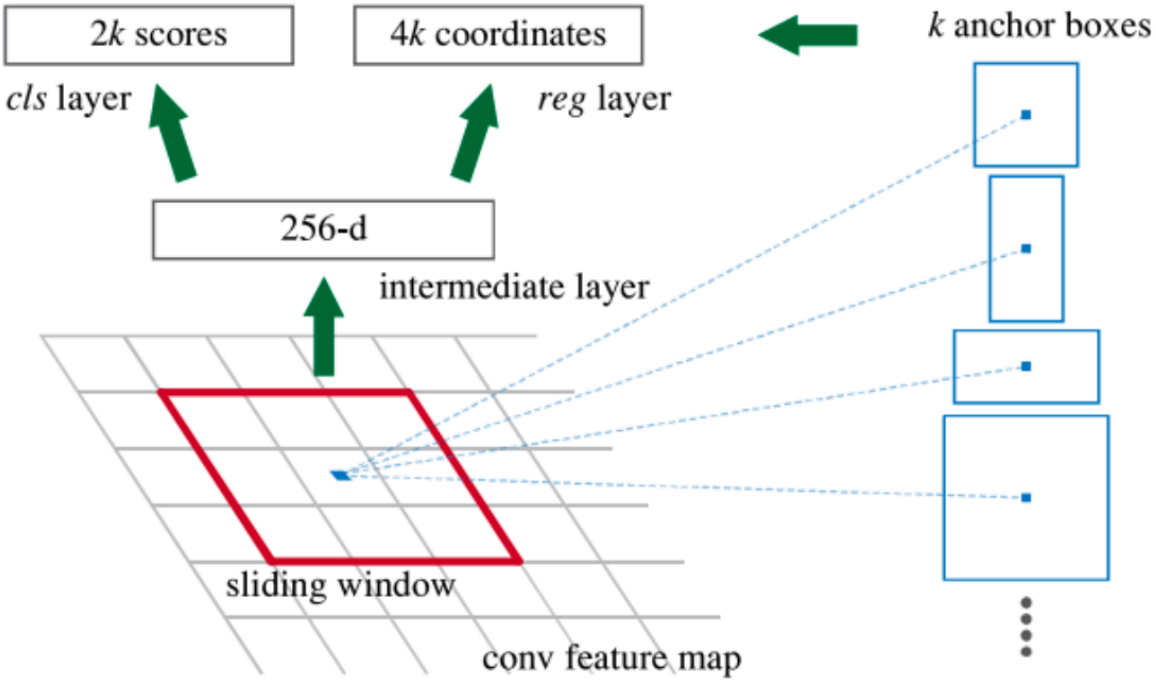
效率对比

- Fast R-CNN trains the very deep VGG16 network 9× faster than R- CNN, is 213× faster at test-time, and achieves a higher mAP on PASCAL VOC 2012.
- Compared to SPPnet, Fast R-CNN trains VGG16 3×faster, tests 10× faster, and is more accurate.

RPN(Region Proposal Network)：使用全卷积神经网络来生成区域建议(Region proposal)，替代之前的Selective search。

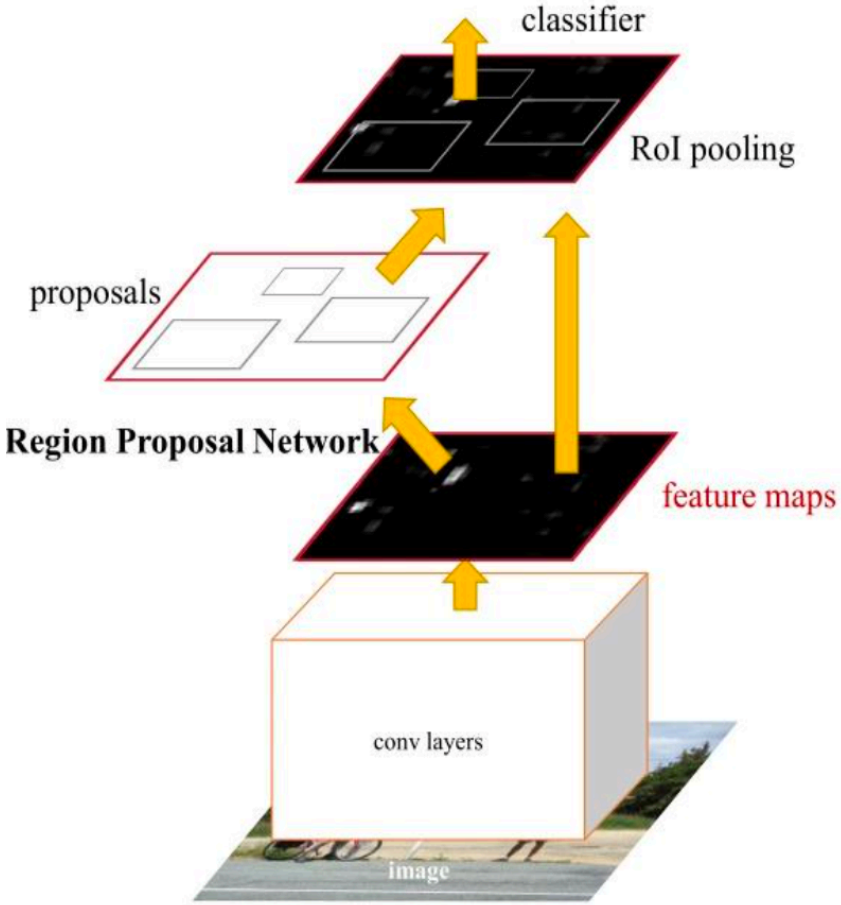


参考：Shaoqing Ren, Kaiming He, Ross B. Girshick, Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans. Pattern Anal. Mach. Intell. 39(6): 1137-1149 (2017)



Faster R-CNN训练方式

- Alternating training
- Approximate joint training

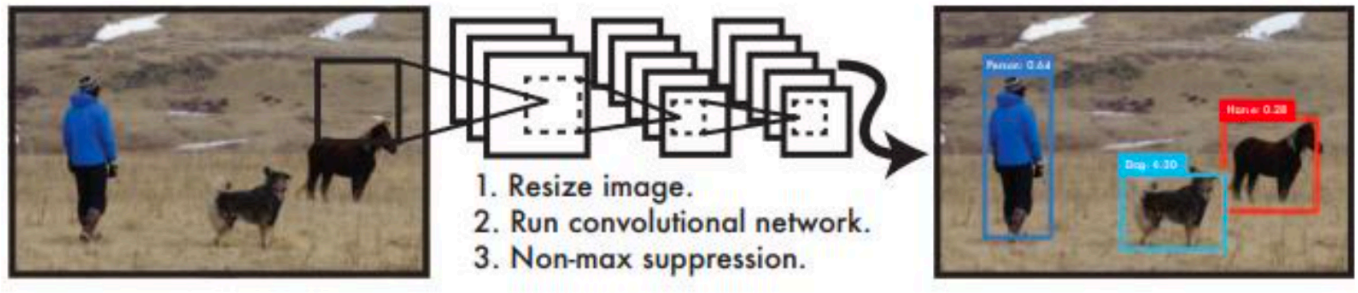


YOLO系列

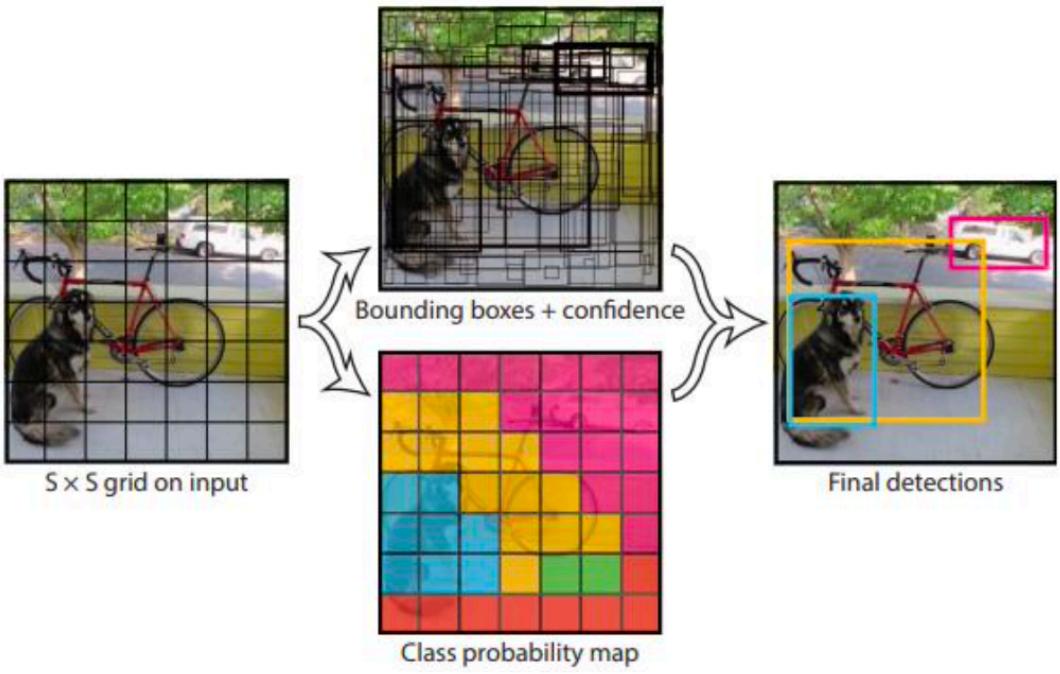
- 与R-CNN系列最大的区别是用一个卷积神经网络结构就可以从输入 图像直接预测bounding box和类别概率，实现了End2End训练
- 速度非常快，实时性好
- 可以学到物体的全局信息，背景误检率比R-CNN降低一半，泛化能力强
- 准确率还不如R-CNN高，小物体检测效果较差

参考：Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. CVPR 2016: 779-788

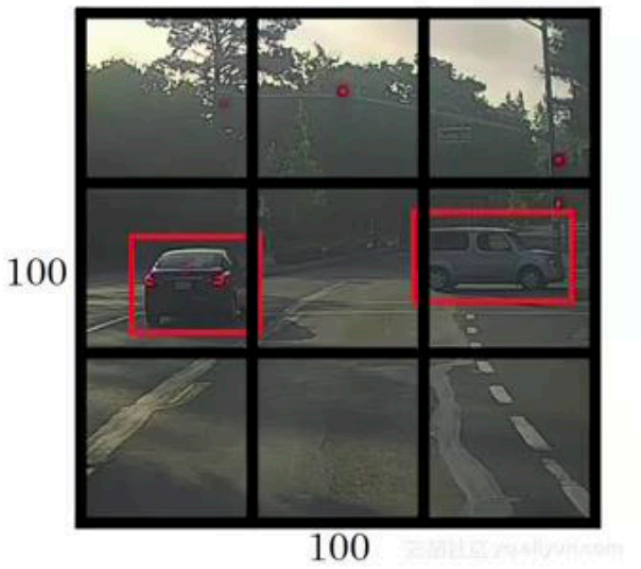
目标检测和识别



The YOLO Detection System. Processing images with YOLO is simple and straightforward. Our system (1) resizes the input image to 448 ×448, (2) runs a single convolutional network on the image, and (3) thresholds the resulting detections by the model's confidence.



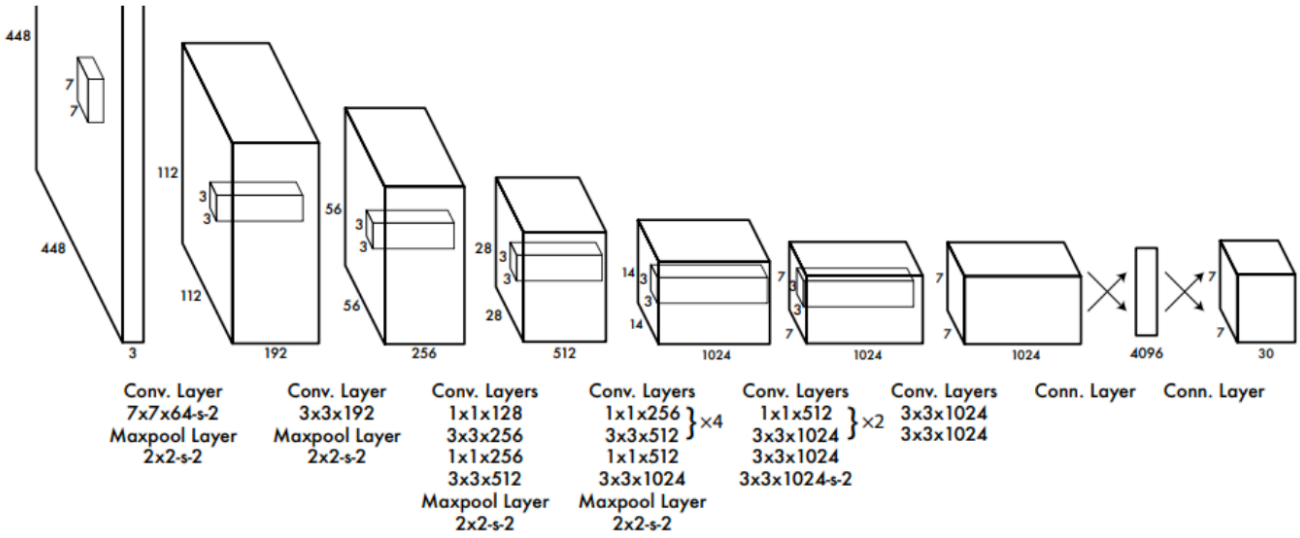
The Model. Our system models detection as a regression problem. It divides the image into an S×S grid and for each grid cell predicts B bounding boxes, confidence for those boxes, and C class probabilities. These predictions are encoded as an S×S×(B∗5+ C) tensor.



y =

pc
bx
by
bh
bw
c1
c2
c3

网络结构：24个卷积层和2个全连接层



The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224 ×224 input image) and then double the resolution for detection.

YOLO2和YOLO9000

参考：Joseph Redmon, Ali Farhadi. YOLO9000: Better, Faster, Stronger. CVPR 2017: 6517-6525

性能分析

	YOLO							YOLOv2
batch norm?		✓	✓	✓	✓	✓	✓	✓
hi-res classifier?			✓	✓	✓	✓	✓	✓
convolutional?				✓	✓	✓	✓	✓
anchor boxes?				✓	✓			
new network?					✓	✓	✓	✓
dimension priors?						✓	✓	✓
location prediction?						✓	✓	✓
passthrough?							✓	✓
multi-scale?								✓
hi-res detector?								✓
VOC2007 mAP	63.4	65.8	69.5	69.2	69.6	74.4	75.4	76.8	**78.6**