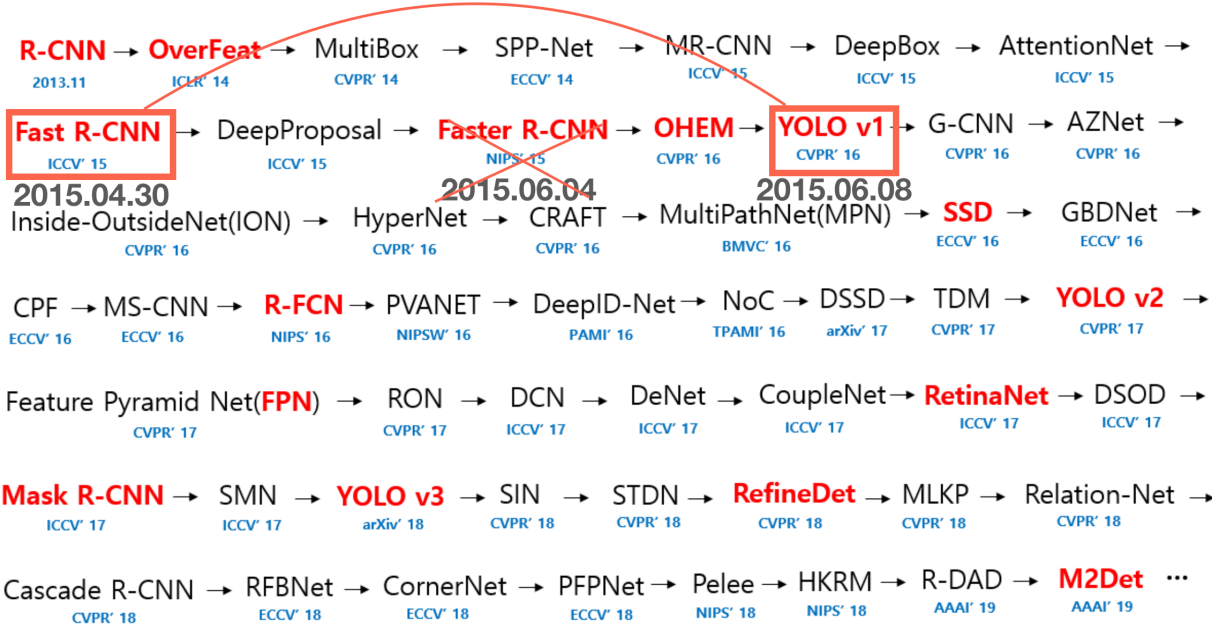


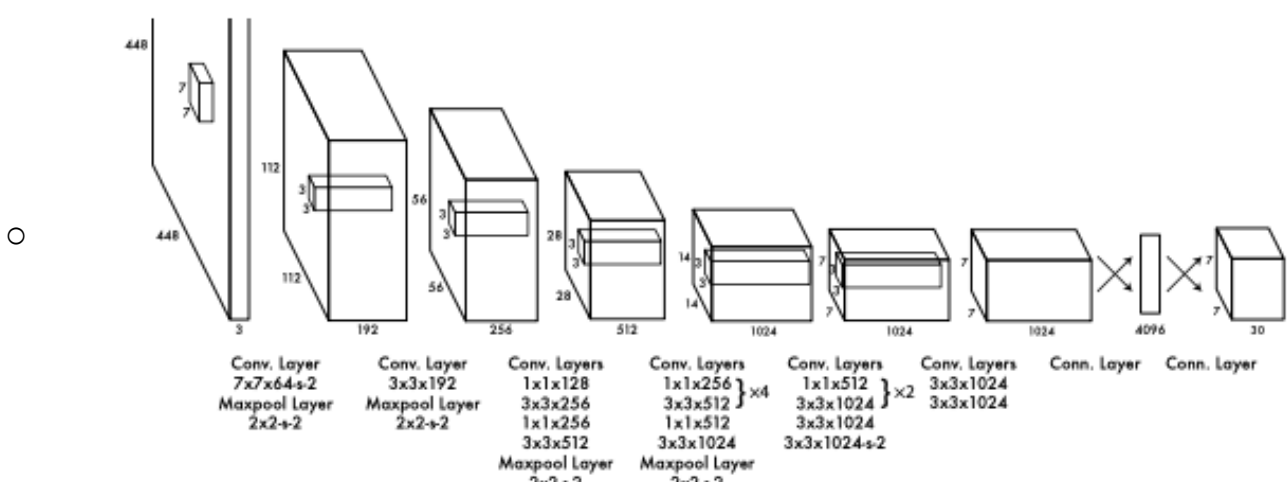
Yolo v1解读

2020/3/14 12:57

- 发布时间比Faster Rcn晚



- 通篇对比对象
 - Fast rcnn
 - DPM
 - RCNN
 - Faster rcnn（当时的sota，轻微对比对象）
- Backbone



- 原是darknet，使用VGG也可以
- New features
 - First end2end neural network detector, unified detector
 - 引入了 slide windows，尤其强调cell和cell 的center point，包括损失函数的回归
 - yolo系列有一个非常有意思地方，就是应该是首次将detection从classification和regression 直接简化为regression 问题，具体（标杆）展现在不用softmax，而用sigmoid多分类

Softmax 分类：（排它性）

0, 1, 2, 3

P1, p2, p3, p4

0.3 0.1 0.3, 0.3

P1+p2+p3+p4 =1

类别 0 在 四个类别中的概率是 0.3

二分类(logistic 回归)

	0	1	2	3
是	0.3	0.6	0.3	0.6
否	0.7	0.4	0.7	0.4

0.3+0.6+0.3+0.6 != 1

属于类别 0 的概率是 0.3，不属于类别 0 的概率是 0.7

- speed和acc在15年达到sota
- Final layer使用linear activation（这个很个很有意思的操作）
 - We use a linear activation function for the final layer and all other layers use the following leaky rectified linear activation:

$$\phi(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.1x, & \text{otherwise} \end{cases} \quad (2)$$

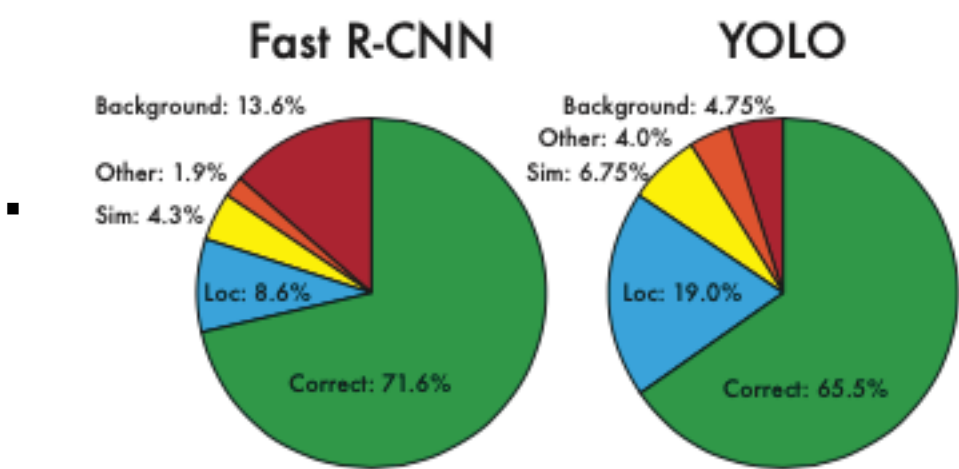
- unify loss function
 - Train

$$\begin{aligned} & \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \\ & + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \\ & + \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\ & + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\ & + \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2 \end{aligned}$$

- Coordinates loss +
- confidence loss +
- classification loss

- Test
 - 其他不变，confidence prediction value output
 - $\Pr(\text{Class}_i | \text{Object}) * \Pr(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}} = \Pr(\text{Class}_i) * \text{IOU}_{\text{pred}}^{\text{truth}} \quad (1)$

- Error analysis



- Regression 问题 导致 “重框”很多（localization :correct class .1 < iou < .5）
- TN：漏检率比较低 • Background: IOU < .1 for any object