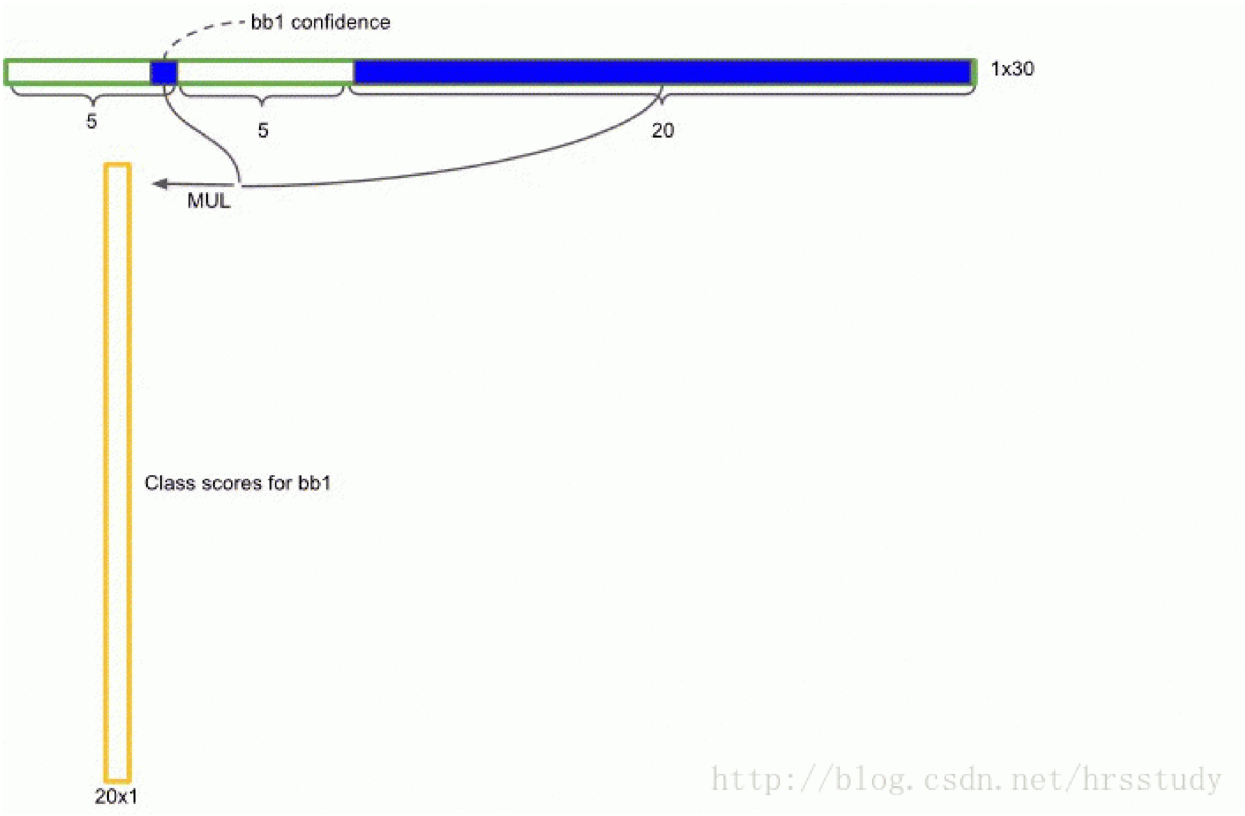
《You Only Look Once: Unified, Real-Time Object Detection》

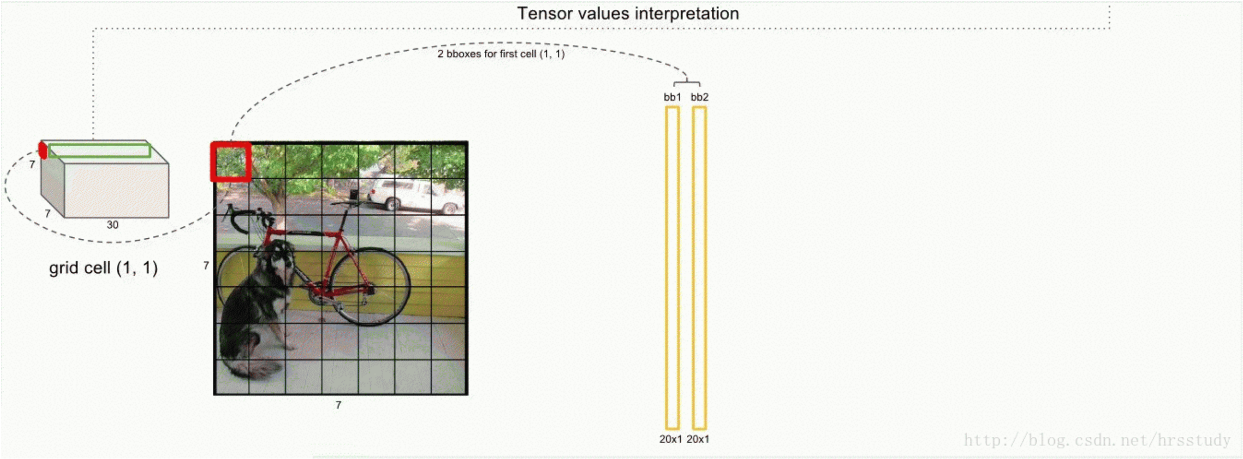
Key idea:

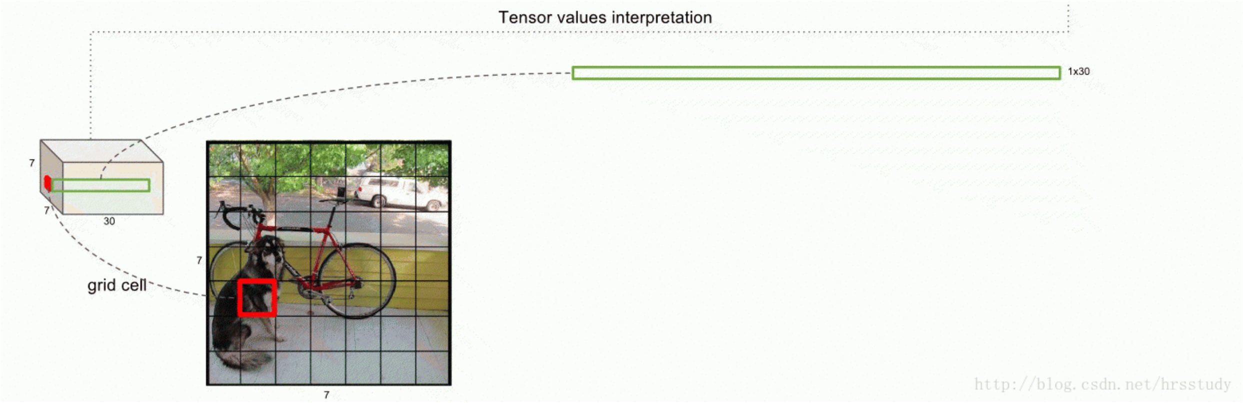
1. 将物体检测这个问题定义为bounding box和分类置信度的回归问题。

2. 将整张图像作为输入，划分成SxS grid，每个cell预测B个bounding box（x, y, w, h）及对应的分类置信度（class-specific confidence score）。分类置信度是 每个类别的概率 和 是物体的概率 以及 IOU 相乘的结果。

$$\Pr(\text{Class}_i|\text{Object}) * \Pr(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}} = \Pr(\text{Class}_i) * \text{IOU}_{\text{pred}}^{\text{truth}}$$




Tensor values interpretation
2 bboxes for first cell (1, 1)
bb1
bb2
7
7
30
grid cell (1, 1)
7
7
20x1
20x1
http://blog.csdn.net/hrsstudy



优点：

1. 速度快，基本YOLO模型达到45FPS，Fast YOLO模型达到45FPS。

2. YOLO使用图像的全局信息做预测，因而对背景的误识别率比Fast R-CNN低。

3. YOLO学习到的特征更加通用，在艺术品的检测上准确率高于DPM和R-CNN。

局限性：

1. 每个cell只能拥有一个label和两个bounding box，这个空间局限性，使得对小物体检测效果不好（尤其是密集的小物体）。

2. 对于物体长宽比的泛华能力较弱，当一类物体新的长宽比出现时，检测准确率减低。

3. 损失函数的设计，导致定位误差是影响检测效果的主要原因，在小的物体上，检测准确率较低。

损失函数设计的Tricks：

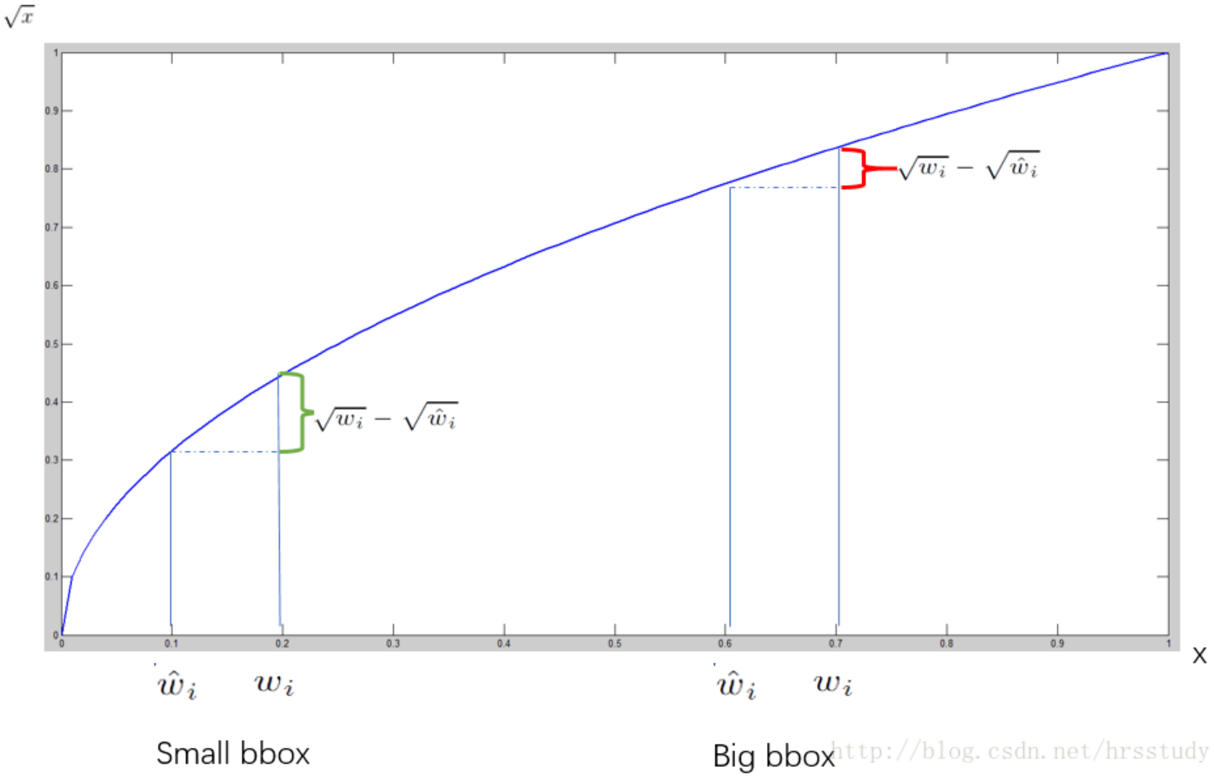
1. localization error（8维）和classification error（20维）的维数不同，赋予同样的权重显然不合理。

Solution：取5。

2. 大部分box是没有物体的，这些cell的confidence score就趋向于0，不包含物体的bbox对梯度更新的贡献会远大于包含物体的bbox对梯度更新的贡献，这会导致网络不稳定甚至发散。。

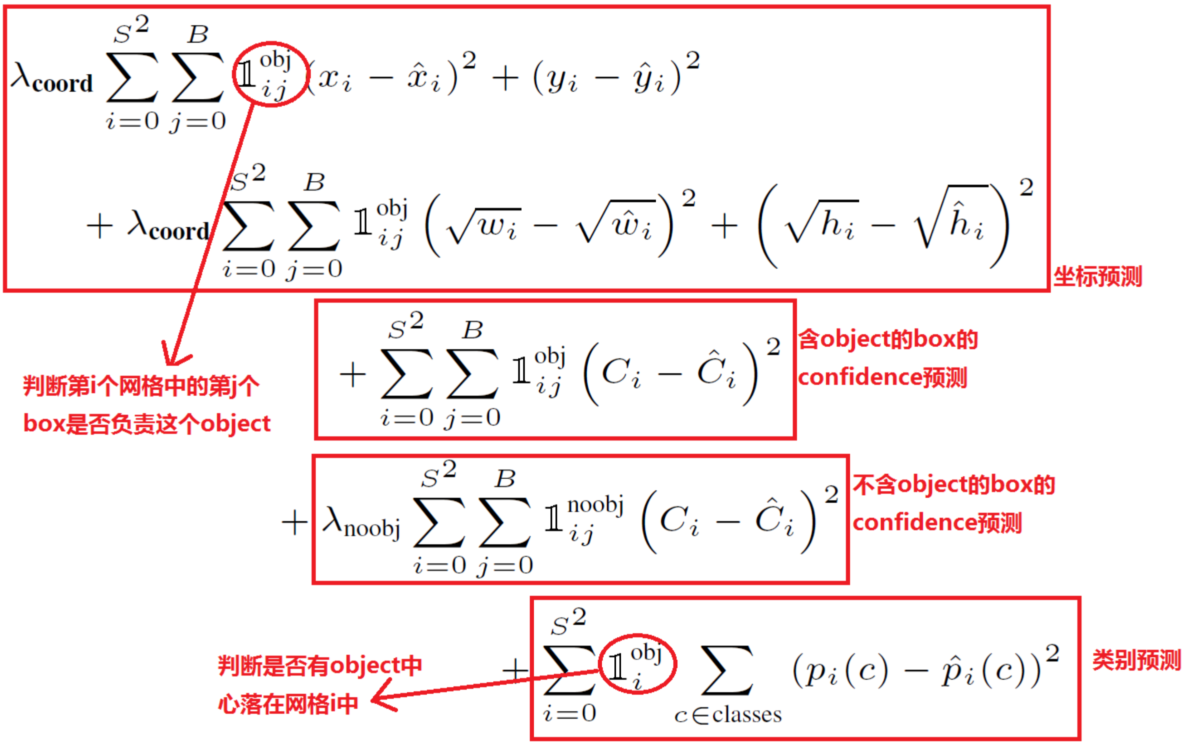
Solution：取0.5。

3. 对于大的box和小的box，sum-square error loss对于同样的偏移的loss是一样的，这也不合理。



Solution：对于width和height取平方根。

Loss Function：



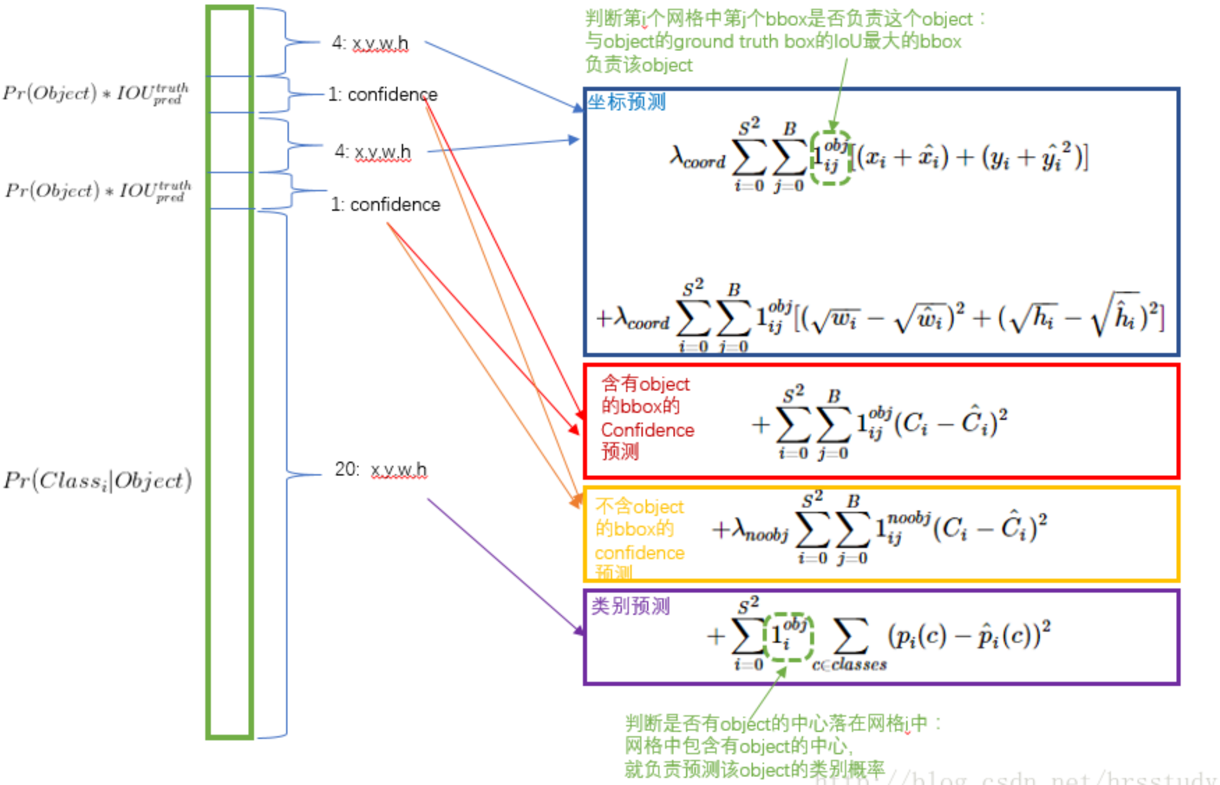
$$\lambda_{\textbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2$$

$$+ \lambda_{\textbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i}\right)^2$$

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i\right)^2$$

$$+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i\right)^2$$

$$+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2$$



$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} [(x_i + \hat{x}_i) + (y_i + \hat{y}_i{}^2)]$$

$$+\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} [(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2]$$

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} (C_i - \hat{C}_i)^2$$

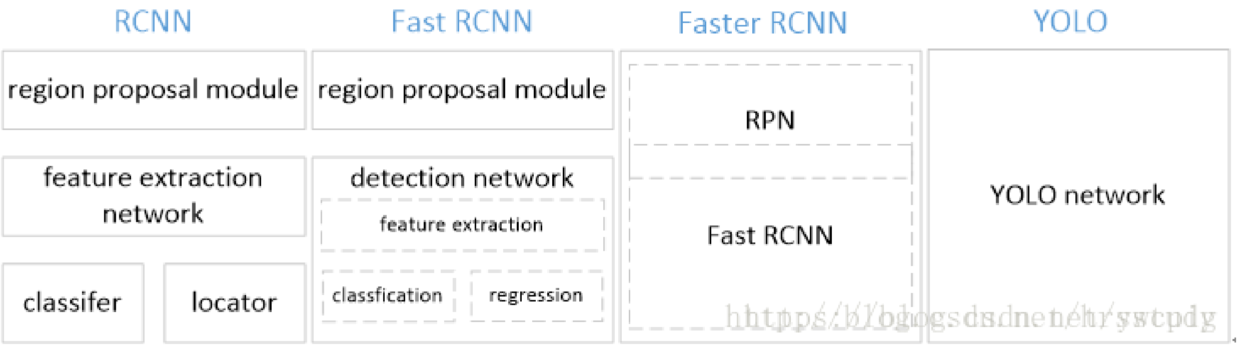
$$+\lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} (C_i - \hat{C}_i)^2$$

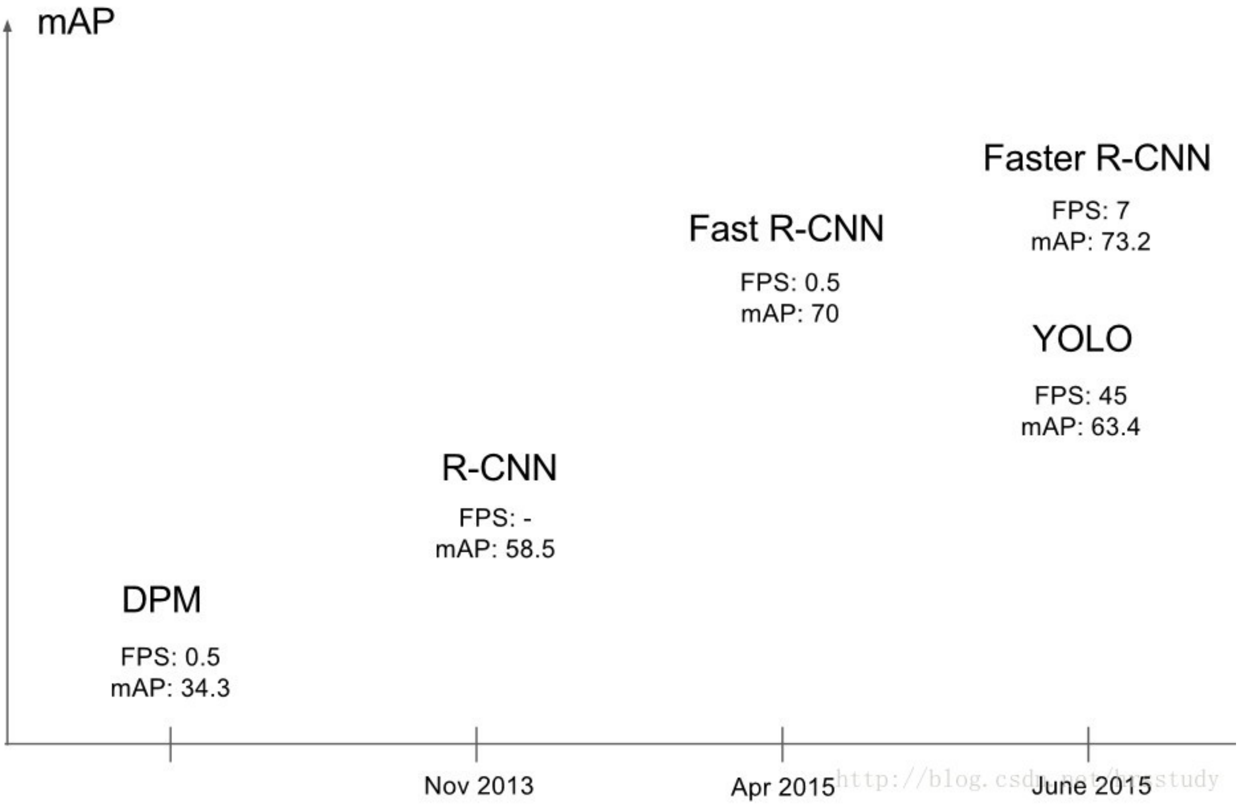
$$+ \sum_{i=0}^{S^2} 1_i^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2$$

$Pr(Object) * IOU_{pred}^{truth}$
$Pr(Object) * IOU_{pred}^{truth}$
$Pr(Class_i|Object)$
4: x,y,w,h
1: confidence
4: x,y,w,h
1: confidence
20: x,y,w,h
判断第i个网格中第j个bbox是否负责这个object：
与object的ground truth box的IoU最大的bbox
负责该object
坐标预测
含有object
的bbox的
Confidence
预测
不含object
的bbox的
confidence
预测
类别预测
判断是否有object的中心落在网格i中：
网格中包含有object的中心，
就负责预测该object的类别概率

非极大值抑制（NMS）：

YOLO与R-CNN系列的对比：





《SSD: Single Shot MultiBox Detector》

Key idea:

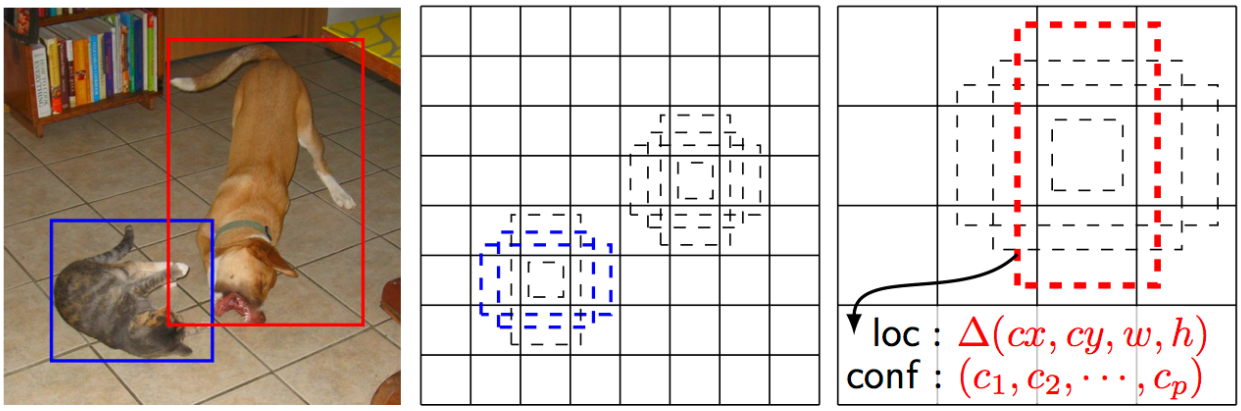
1. 将物体检测这个问题的解空间，抽象为一组预先设定好（尺度，长宽比）的bounding box。

2. 在每个bounding box，预测分类label，以及box offset来更好的框出物体。

3. 对一张图片，结合多个大小不同的feature map的预测结果，以期能够处理大小不同的物体。

优点：

1. 相比Fast RNN系列，删除了bounding box proposal这一步，及后续的重采样步骤，因而速度较快，达到59FPS。


loc : $\Delta(cx, cy, w, h)$
conf : $(c_1, c_2, \cdots, c_p)$

(a) Image with GT boxes (b) 8 × 8 feature map (c) 4 × 4 feature map

Prediction:

Input:

(1) m x n feature map

(2) k default bounding boxes

Filters: (c + 4) x k

Output: (c + 4) x k x m x n

Training:

Key difference between training SSD and training a typical detector that uses region proposals:

Ground truth information needs to be assigned to specific outputs in the fixed set of detector outputs(default bounding boxes). Once this assignment is determined, the loss function and back propagation are applied end-to-end.

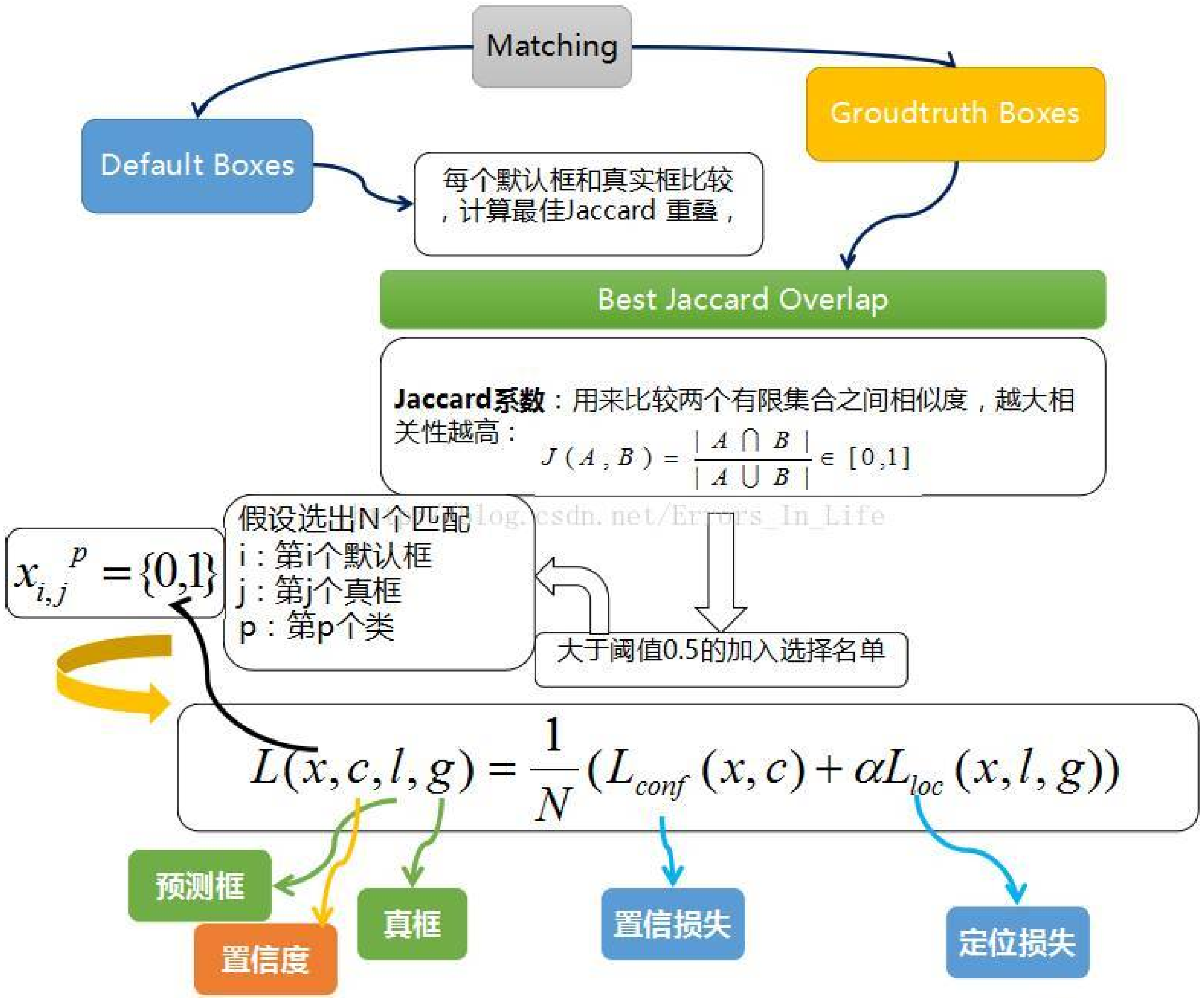
Matching strategy: Match default boxes to any ground truth with jaccard overlap higher than a threshold (0.5).

This simplifies the learning problem, allowing the network to predict high scores for multiple overlapping default boxes rather than requiring it to pick only the one with maximum overlap.

Training objective: The overall objective loss function is a weighted sum of the localization loss (loc) and the confidence loss (conf):

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g))$$

The weight term α is set to 1 by cross validation.


Matching
Default Boxes
Groudtruth Boxes
每个默认框和真实框比较，计算最佳Jaccard 重叠，
Best Jaccard Overlap
Jaccard系数：用来比较两个有限集合之间相似度，越大相关性越高：
$J(A,B)=\frac{|A\cap B|}{|A\cup B|}\in[0,1]$
$x_{i,j}^{p}=\{0,1\}$
假设选出N个匹配
i：第i个默认框
j：第j个真框
p：第p个类
大于阈值0.5的加入选择名单
$L(x,c,l,g)=\frac{1}{N}(L_{conf}(x,c)+\alpha L_{loc}(x,l,g))$
预测框
置信度
真框
置信损失
定位损失

(1) Localization loss: The localization loss is a Smooth L1 loss [6] between the predicted box (l) and the ground truth box (g) parameters.

$$L_{loc}(x,l,g)=\sum_{i\in Pos}^{N}\sum_{m\in\{cx,cy,w,h\}} x_{ij}^{k}\,\mathrm{smooth}_{\mathrm{L1}}(l_i^m-\hat{g}_j^m)$$

$$\hat{g}_j^{cx}=(g_j^{cx}-d_i^{cx})/d_i^w \qquad \hat{g}_j^{cy}=(g_j^{cy}-d_i^{cy})/d_i^h$$

$$\hat{g}_j^{w}=\log\left(\frac{g_j^w}{d_i^w}\right) \qquad \hat{g}_j^{h}=\log\left(\frac{g_j^h}{d_i^h}\right)$$

(2) Confidence loss: The confidence loss is the softmax loss over multiple classes confidences (c).

$$L_{conf}(x,c)=-\sum_{i\in Pos}^{N} x_{ij}^{p}log(\hat{c}_i^p)-\sum_{i\in Neg} log(\hat{c}_i^0) \quad \text{where} \quad \hat{c}_i^p=\frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)}$$

Choosing scales and aspect ratios for default boxes：

利用不同尺度（不同层）的 feature map 来学习不同尺度下的物体检测。

Hard negative mining:

Problem: Imbalance between the positive and negative training examples.

Solution: Instead of using all the negative examples, we sort them using the highest confidence loss for each default box and pick the top ones so that the ratio between the negatives and positives is at most 3:1.

优点：Faster optimization and a more stable training.

Data augmentation:

Target: To make the model more robust to various input object sizes and shapes.

Each training image is randomly sampled by one of the following options:

- Use the entire original input image.

- Sample a patch so that the minimum jaccard overlap with the objects is 0.1, 0.3, 0.5, 0.7, or 0.9.

- Randomly sample a patch.

After the aforementioned sampling step, each sampled patch is resized to fixed size and is horizontally flipped with probability of 0.5, in addition to applying some photo-metric distortions.

Model analysis on controlled experiments:

(1) **Data augmentation is crucial:** We can improve 8.8% mAP with this sampling strategy.

(2) **More default box shapes is better:** If we remove the boxes with 1/3 and 3 aspect ratios, the performance drops by 0.6%. By further removing the boxes with 1/2 and 2 aspect ratios, the performance drops another 2.1%. Using a variety of default box shapes seems to make the task of predicting boxes easier for the network.

(3) **Atrous is faster:** We used the atrous version of a subsampled VGG16, following DeepLab-LargeFOV. If we use the full VGG16, keeping pool5 with 22 s2 and not subsampling parameters from fc6 and fc7, and add conv5 3 for prediction, the result is about the same while the speed is about 20% slower.

(4) **Multiple output layers at different resolutions is better:** A major contribution of SSD is using default boxes of different scales on different output layers. To measure the advantage gained, we progressively remove layers and compare results. For a fair comparison, every time we remove a layer, we adjust the default box tiling to keep the total number of boxes similar to the original (8732). Table 3 shows a decrease in accuracy with fewer layers, dropping monotonically from 74.3 to 62.4.

(5) **Data Augmentation for Small Object Accuracy:** The random crops generated by the strategy can be thought of as a ” zoom in” operation and can generate many larger training examples. To implement a ” zoom out” operation that creates more small training

examples, we first randomly place an image on a canvas of 16 of the original image size filled with mean values before we do any random crop operation. Because we have more training images by introducing this new ”expansion” data augmentation trick, we have to double the training iterations. We have seen a consistent increase of 2%-3% mAP across multiple datasets, as shown in Table 6.

(6) Non-maximum suppression (nms): Considering the large number of boxes generated from our method, it is essential to perform non-maximum suppression (nms) efficiently during inference. By using a confidence threshold of 0.01, we can filter out most boxes.We then apply nms with jaccard overlap of 0.45 per class and keep the top 200 detections per image.

Comparison

A comparison between two single shot detection models: SSD and YOLO