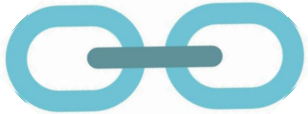
TensorFlow



bilibli: 霹雳吧啦Wz

PYTORCH

深度学习-目标检测篇

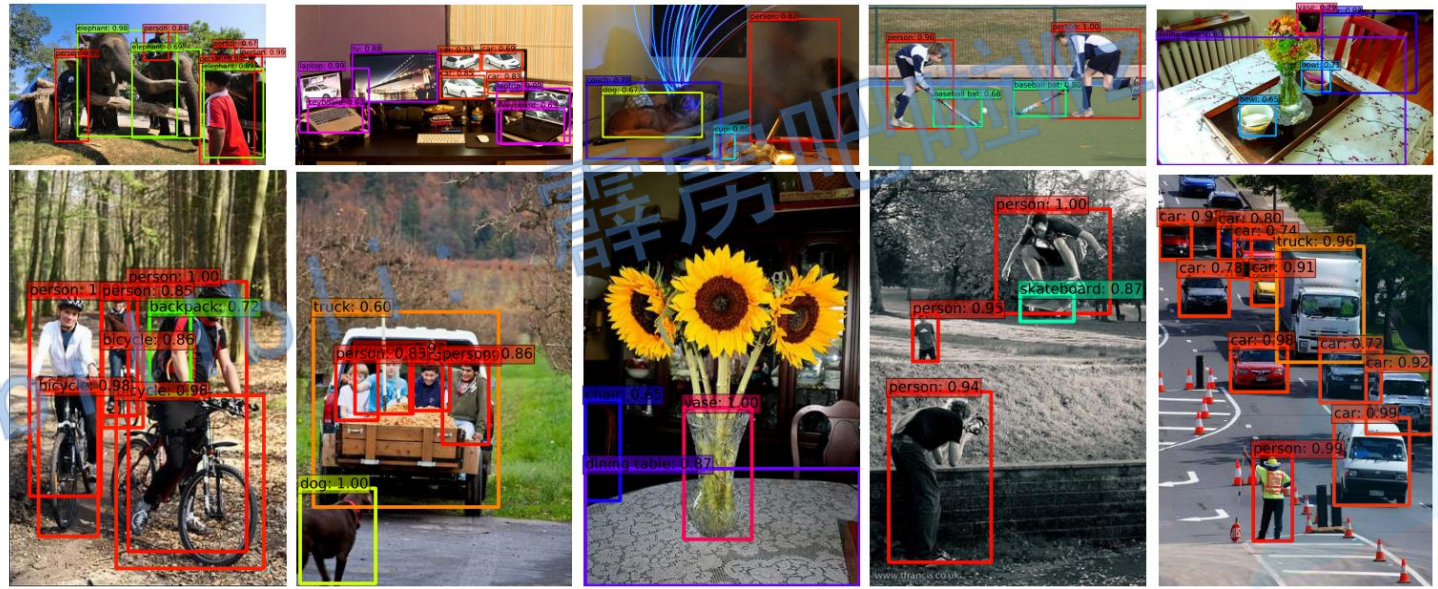
作者：神秘的wz

Single Shot MultiBox Detector

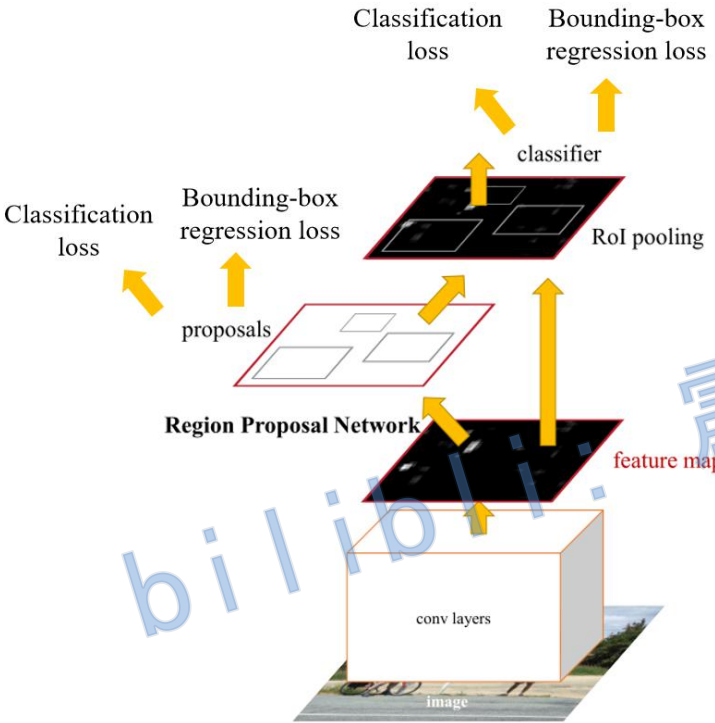
SSD: Single Shot MultiBox Detector

真正的实时

SSD网络是作者Wei Liu在ECCV 2016上发表的论文。对于输入尺寸300x300的网络使用Nvidia Titan X在VOC 2007测试集上达到74.3%mAP以及59FPS，对于512x512的网络，达到了76.9%mAP超越当时最强的Faster RCNN(73.2%mAP)。



Single Shot MultiBox Detector



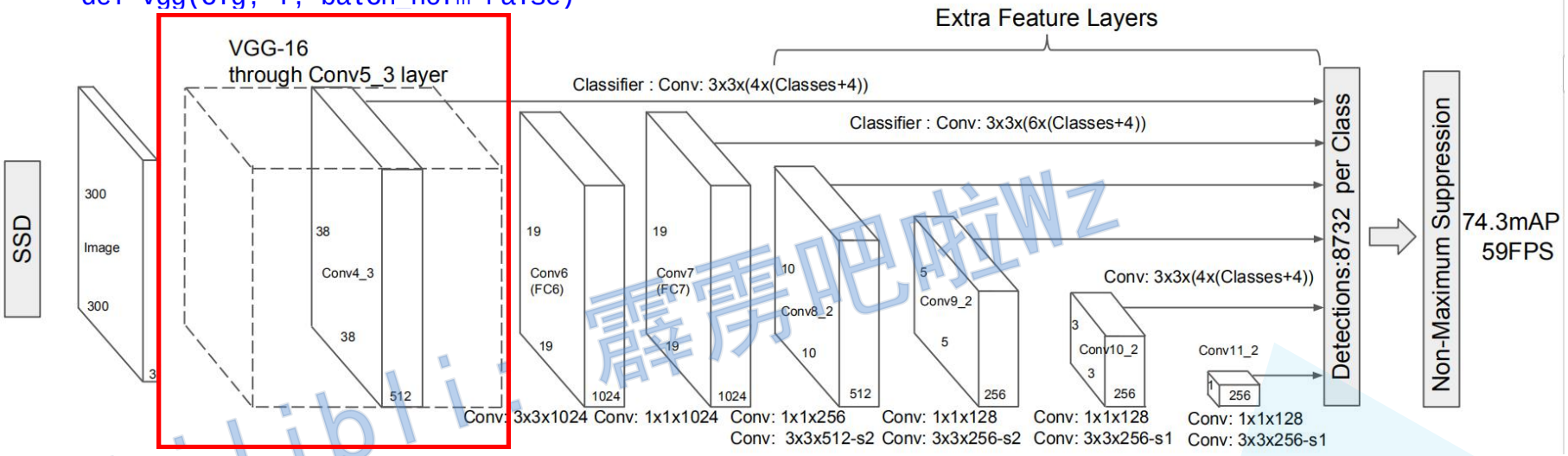
Faster RCNN存在的问题

- 对小目标检测效果很差
- 模型大，检测速度较慢

Single Shot MultiBox Detector

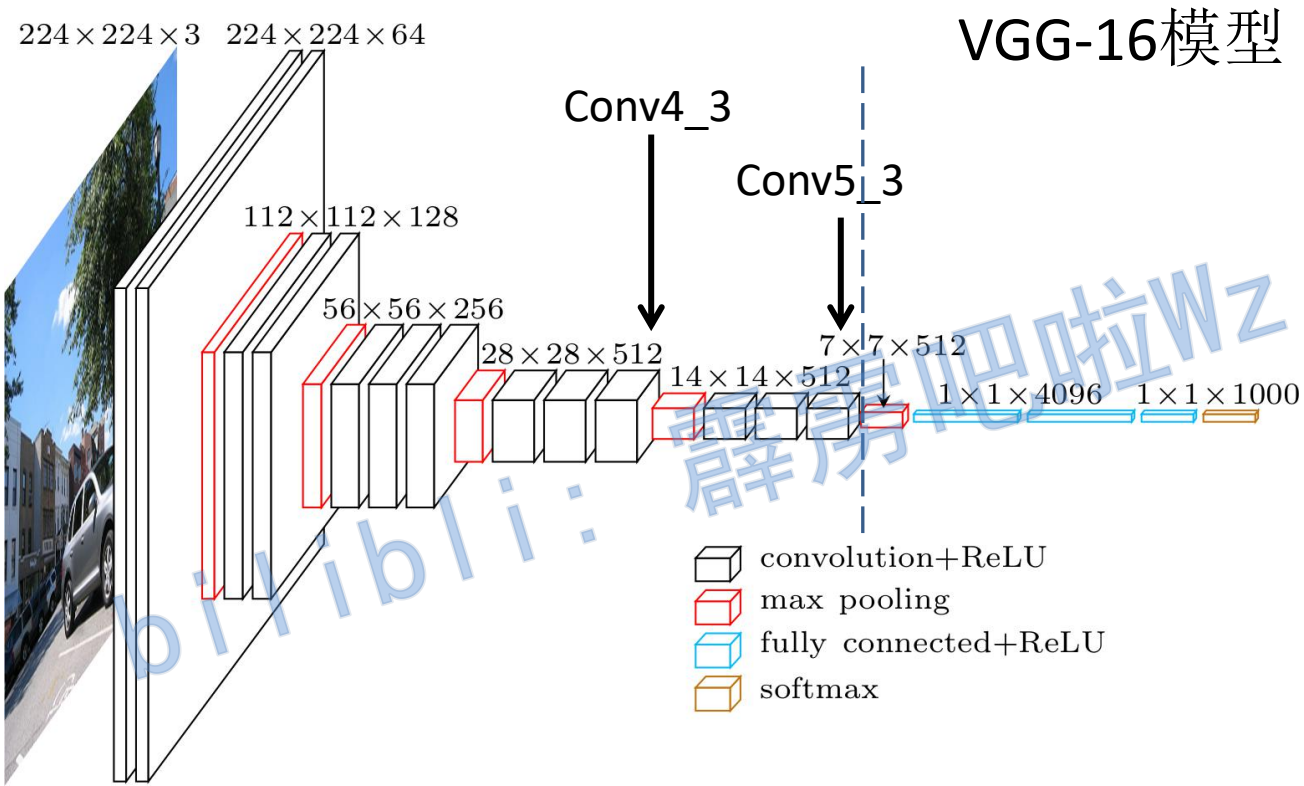
在不同特征尺度上预测不同尺度的目标

```
def vgg(cfg, i, batch_norm=False)
```

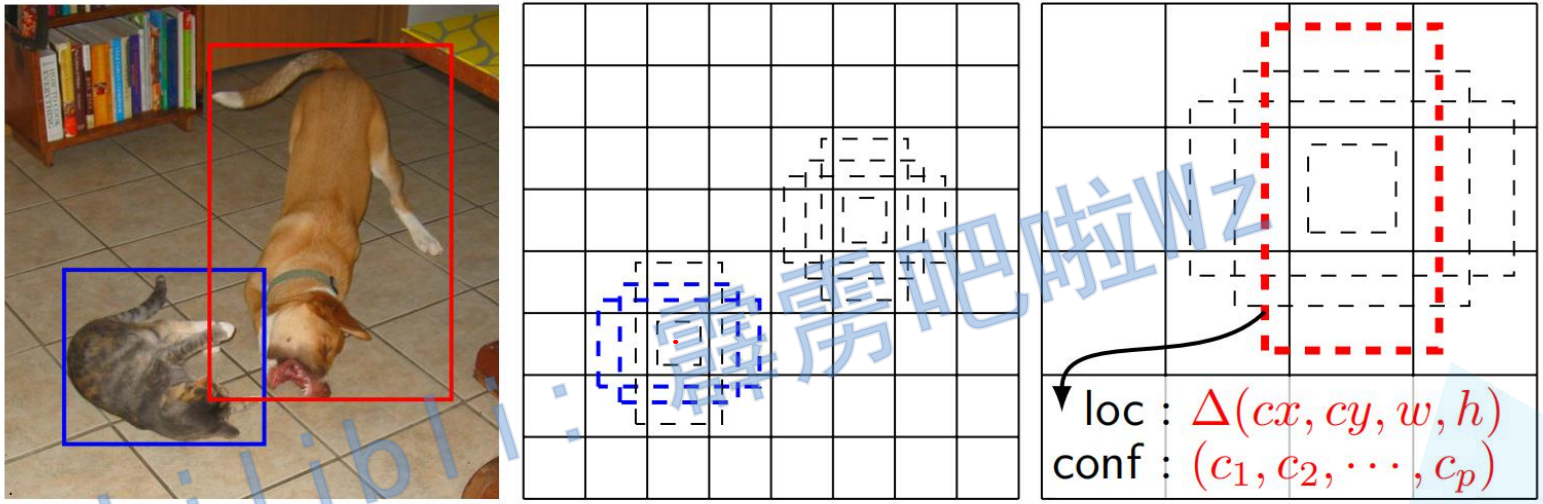


Base network Our experiments are all based on VGG16 [15], which is pre-trained on the ILSVRC CLS-LOC dataset [16]. Similar to DeepLab-LargeFOV [17], we convert fc6 and fc7 to convolutional layers, subsample parameters from fc6 and fc7, change pool5 from $2 \times 2 - s2$ to $3 \times 3 - s1$, and use the *à trous* algorithm [18] to fill the "holes". We remove all the dropout layers and the fc8 layer. We fine-tune the resulting

Single Shot MultiBox Detector



Single Shot MultiBox Detector



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

Single Shot MultiBox Detector

Default Box的scale以及aspect设定

Feature maps from different levels within a network are known to have different (empirical) receptive field sizes [13]. Fortunately, within the SSD framework, the default boxes do not necessary need to correspond to the actual receptive fields of each layer. We design the tiling of default boxes so that specific feature maps learn to be responsive to particular scales of the objects. Suppose we want to use m feature maps for prediction. The scale of the default boxes for each feature map is computed as:

$$s_k = s_{\min} + \frac{s_{\max} - s_{\min}}{m - 1}(k - 1), \quad k \in [1, m] \tag{4}$$

where $s_{\min}$ is 0.2 and $s_{\max}$ is 0.9, meaning the lowest layer has a scale of 0.2 and the highest layer has a scale of 0.9, and all layers in between are regularly spaced. We impose different aspect ratios for the default boxes, and denote them as $a_r \in \{1, 2, 3, \frac{1}{2}, \frac{1}{3}\}$. We can compute the width ($w_k^a = s_k\sqrt{a_r}$) and height ($h_k^a = s_k/\sqrt{a_r}$) for each default box. For the aspect ratio of 1, we also add a default box whose scale is $s'_k = \sqrt{s_k s_{k+1}}$, resulting in 6 default boxes per feature map location. We set the center of each default box to $(\frac{i+0.5}{|f_k|}, \frac{j+0.5}{|f_k|})$, where $|f_k|$ is the size of the k-th square feature map, $i, j \in [0, |f_k|)$. In practice, one can also design a distribution of default boxes to best fit a specific dataset. How to design the optimal tiling is an open question as well.

Single Shot MultiBox Detector

Default Box的scale以及aspect设定

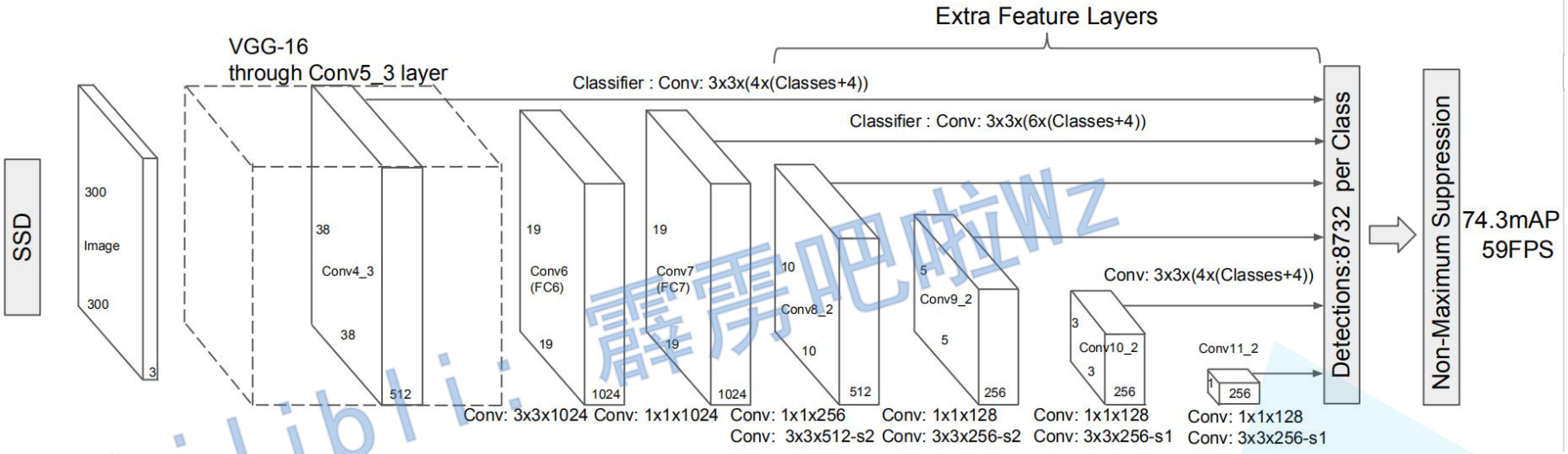
scale = [(21, 45),
(45, 99),
(99, 153),
(153, 207),
(207, 261),
(261, 315)]

aspect = [(1, 2, .5),
(1, 2, .5, 3, 1./3),
(1, 2, .5, 3, 1./3),
(1, 2, .5, 3, 1./3),
(1, 2, .5),
(1, 2, .5)]

Figure 2 shows the architecture details of the SSD300 model. We use conv4_3, conv7 (fc7), conv8_2, conv9_2, conv10_2, and conv11_2 to predict both location and confidences. We set default box with scale 0.1 on conv4_3[3]. We initialize the parameters for all the newly added convolutional layers with the "xavier" method [20]. For conv4_3, conv10_2 and conv11_2, we only associate 4 default boxes at each feature map location – omitting aspect ratios of $\frac{1}{3}$ and 3. For all other layers, we put 6 default boxes as described in Sec. 2.2. Since, as pointed out in [12], conv4_3 has a different feature

Single Shot MultiBox Detector

在不同特征尺度上预测不同尺度的目标



Base network Our experiments are all based on VGG16 [15], which is pre-trained on the ILSVRC CLS-LOC dataset [16]. Similar to DeepLab-LargeFOV [17], we convert fc6 and fc7 to convolutional layers, subsample parameters from fc6 and fc7, change pool5 from $2 \times 2 - s2$ to $3 \times 3 - s1$, and use the *à trous* algorithm [18] to fill the "holes". We remove all the dropout layers and the fc8 layer. We fine-tune the resulting

Single Shot MultiBox Detector

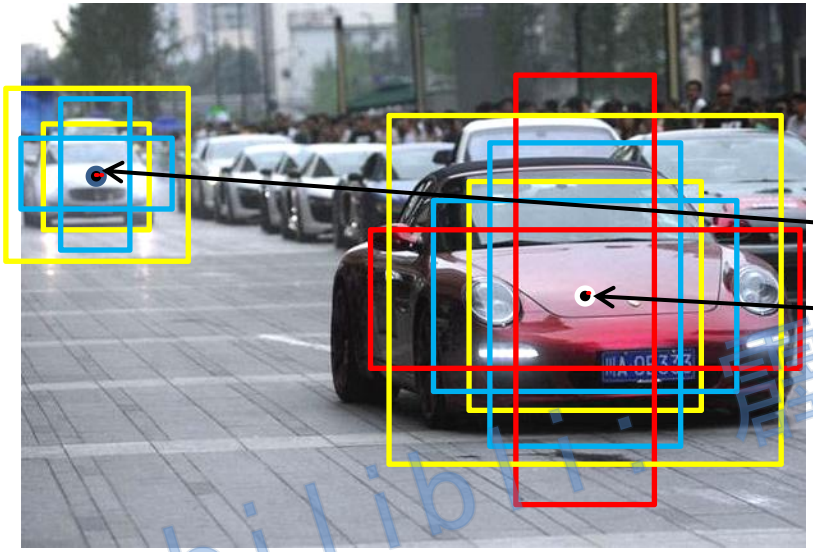
Default Box的scale以及aspect设定

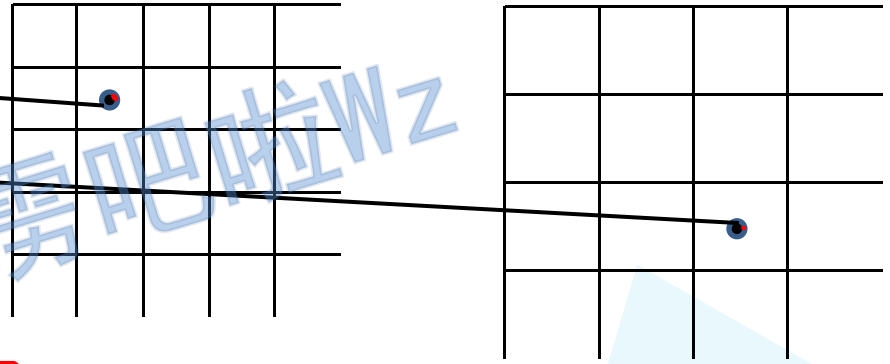
特征图层	特征图层的宽和高	默认框尺寸	默认框数量
特征图层①	38×38	21{1/2, 1, 2}; $\sqrt{21\times45}$ {1}	38×38×4
特征图层②	19×19	45{1/3, 1/2, 1, 2, 3}; $\sqrt{45\times99}$ {1}	19×19×6
特征图层③	10×10	99{1/3, 1/2, 1, 2, 3}; $\sqrt{99\times153}$ {1}	10×10×6
特征图层④	5×5	153{1/3, 1/2, 1, 2, 3}; $\sqrt{153\times207}$ {1}	5×5×6
特征图层⑤	3×3	207{1/2, 1, 2}; $\sqrt{207\times261}$ {1}	3×3×4
特征图层⑥	1×1	261{1/2, 1, 2}; $\sqrt{261\times315}$ {1}	1×1×4

38x38x4 + 19x19x6 + 10x10x6 + 5x5x6 + 3x3x4 + 1x1x4 = 8732

Single Shot MultiBox Detector

Default Box的scale以及aspect设定





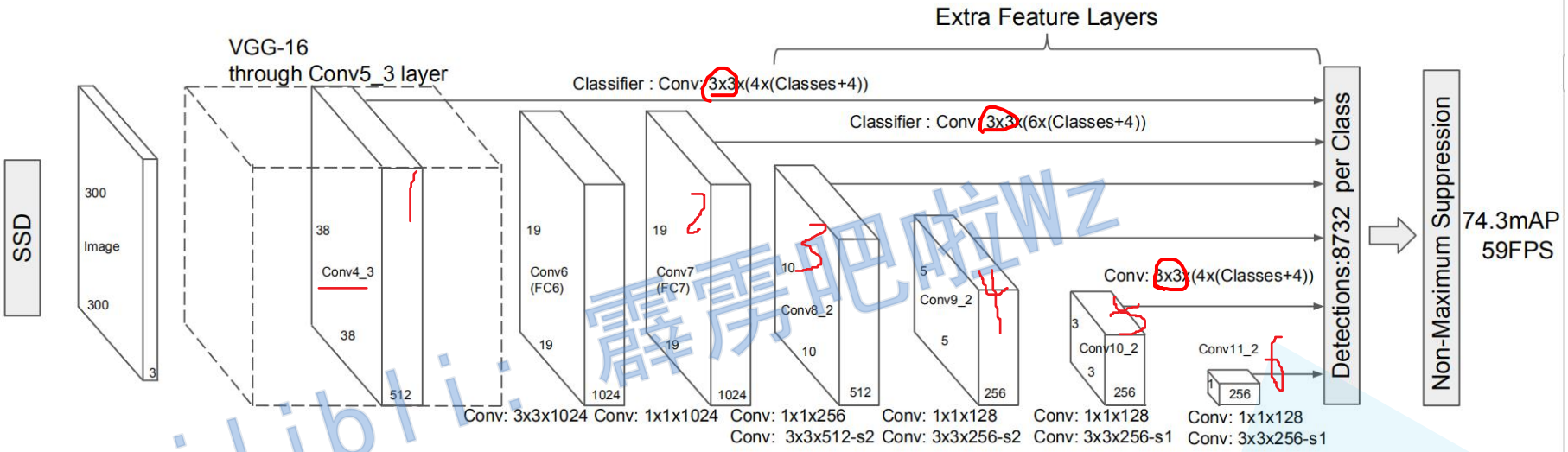
21{1/2, 1, 2}; $\sqrt{21\times45}$ {1}

153{1/3, 1/2, 1, 2, 3}; $\sqrt{153\times207}$ {1}

特征图层	特征图层的宽和高	默认框尺寸	默认框数量
特征图层①	38×38	21{1/2, 1, 2}; $\sqrt{21\times45}$ {1}	38×38×4
特征图层②	19×19	45{1/3, 1/2, 1, 2, 3}; $\sqrt{45\times99}$ {1}	19×19×6
特征图层③	10×10	99{1/3, 1/2, 1, 2, 3}; $\sqrt{99\times153}$ {1}	10×10×6
特征图层④	5×5	153{1/3, 1/2, 1, 2, 3}; $\sqrt{153\times207}$ {1}	5×5×6
特征图层⑤	3×3	207{1/2, 1, 2}; $\sqrt{207\times261}$ {1}	3×3×4
特征图层⑥	1×1	261{1/2, 1, 2}; $\sqrt{261\times315}$ {1}	1×1×4

Single Shot MultiBox Detector

Predictor的实现



Base network Our experiments are all based on VGG16 [15], which is pre-trained on the ILSVRC CLS-LOC dataset [16]. Similar to DeepLab-LargeFOV [17], we convert fc6 and fc7 to convolutional layers, subsample parameters from fc6 and fc7, change pool5 from $2 \times 2 - s2$ to $3 \times 3 - s1$, and use the *à trous* algorithm [18] to fill the "holes". We remove all the dropout layers and the fc8 layer. We fine-tune the resulting

Single Shot MultiBox Detector

Predictor的实现

Convolutional predictors for detection Each added feature layer (or optionally an existing feature layer from the base network) can produce a fixed set of detection predictions using a set of convolutional filters. These are indicated on top of the SSD network architecture in Fig. 2. For a feature layer of size $m \times n$ with p channels, the basic element for predicting parameters of a potential detection is a $3 \times 3 \times p$ *small kernel* that produces either a score for a category, or a shape offset relative to the default box coordinates. At each of the $m \times n$ locations where the kernel is applied, it produces an output value. The bounding box offset output values are measured relative to a default

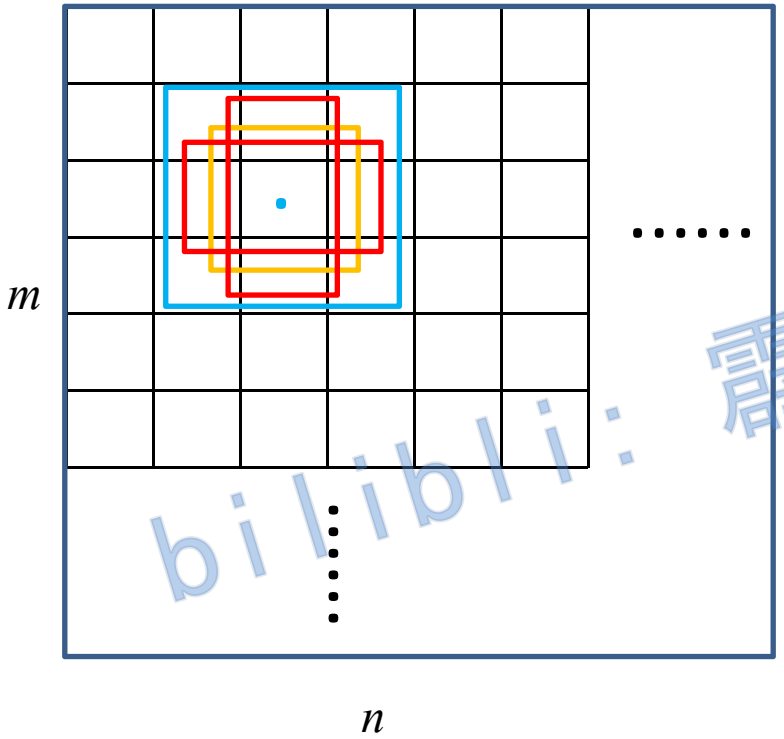
Single Shot MultiBox Detector

Predictor的实现

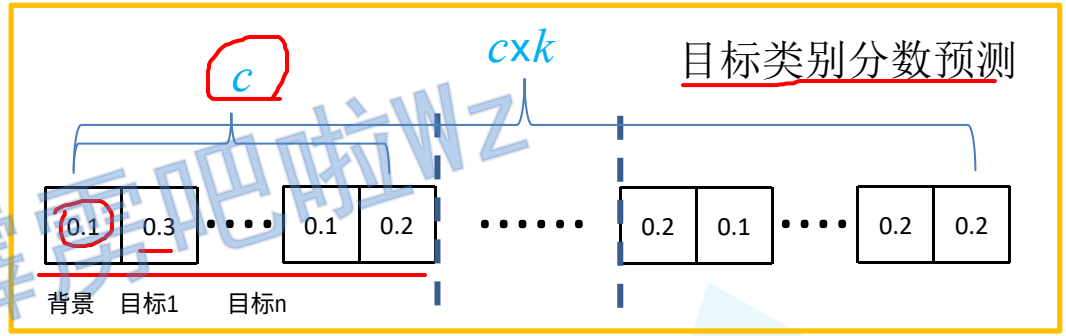
Default boxes and aspect ratios We associate a set of default bounding boxes with each feature map cell, for multiple feature maps at the top of the network. The default boxes tile the feature map in a convolutional manner, so that the position of each box relative to its corresponding cell is fixed. At each feature map cell, we predict the offsets relative to the default box shapes in the cell, as well as the per-class scores that indicate the presence of a class instance in each of those boxes. Specifically, for each box out of k at a given location, we compute c class scores and the 4 offsets relative to the original default box shape. This results in a total of $(c+4)k$ filters that are applied around each location in the feature map, yielding $(c+4)kmn$ outputs for a $m \times n$ feature map. For an illustration of default boxes, please refer to Fig. 1. Our default boxes are similar to the *anchor boxes* used in Faster R-CNN [2], however we apply them to several feature maps of different resolutions. Allowing different default box shapes in several feature maps let us efficiently discretize the space of possible output box shapes.

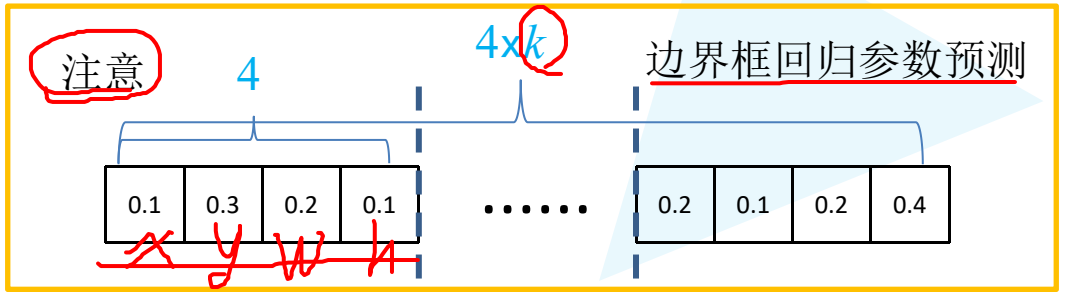
Single Shot MultiBox Detector

Predictor的实现



$$(c+4)\times k = c\times k + 4\times k$$





Single Shot MultiBox Detector

正负样本的选取

正样本

Matching strategy During training we need to determine which default boxes correspond to a ground truth detection and train the network accordingly. For each ground truth box we are selecting from default boxes that vary over location, aspect ratio, and scale. We begin by matching each ground truth box to the default box with the best jaccard overlap (as in MultiBox [7]). Unlike MultiBox, we then 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.

Single Shot MultiBox Detector

正负样本的选取

负样本

Hard negative mining After the matching step, most of the default boxes are negatives, especially when the number of possible default boxes is large. This introduces a significant imbalance between the positive and negative training examples. 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. We found that this leads to faster optimization and a more stable training.

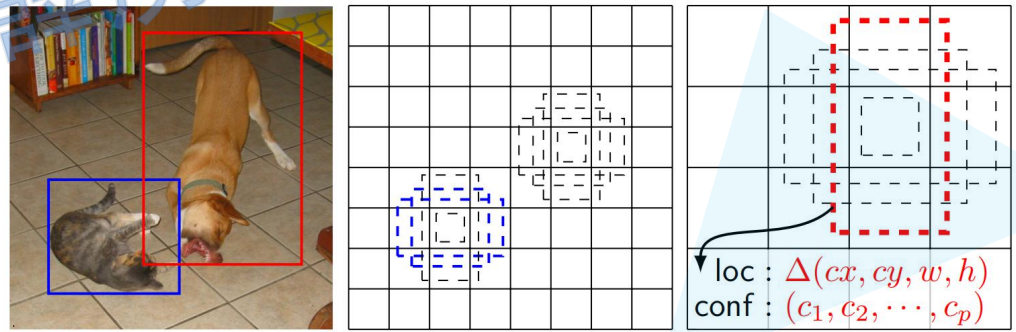
Single Shot MultiBox Detector

损失的计算

类别损失　　定位损失

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

其中N为匹配到的正样本个数，α为1



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

Single Shot MultiBox Detector

损失的计算

类别损失

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

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

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

$\hat{c}_i^p$为预测的第i个default box对应GT box(类别是P)的类别概率

$x_{ij}^p = \{0,1\}$为第i个default box匹配到的第j个GT box(类别是P)

Single Shot MultiBox Detector

损失的计算

定位损失

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

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

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

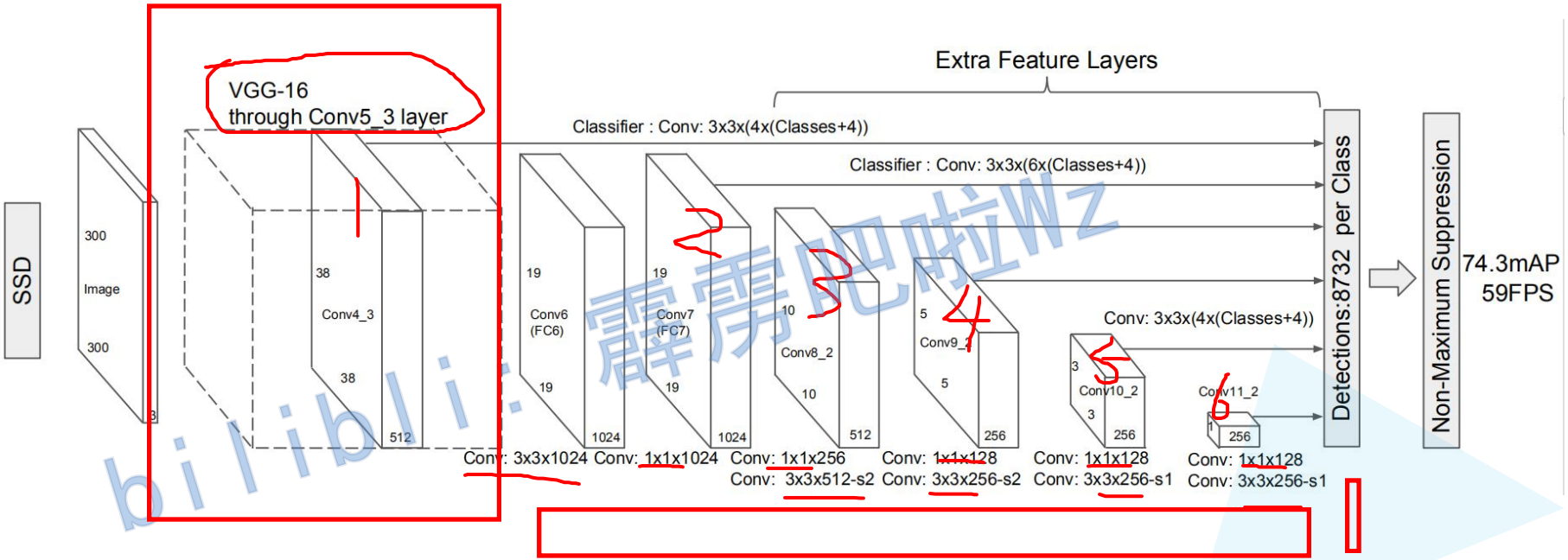
$$\hat{g}_j^w=\log\left(\frac{g_j^w}{d_i^w}\right) \qquad \hat{g}_j^h=\log\left(\frac{g_j^h}{d_i^h}\right)$$

l_i^m为预测对应第i个正样本回归参数

$\hat{g}_j^m$为正样本i匹配的第j个$GT\ box$的回归参数

$$smooth_{L_1}(x)=\begin{cases}0.5x^2 & \text{if } |x|<1\\ |x|-0.5 & \text{otherwise}\end{cases}$$

Single Shot MultiBox Detector



沟通方式

1.github

https://github.com/WZMIAOMIAO/deep-learning-for-image-processing

2.CSDN

https://blog.csdn.net/qq_37541097/article/details/103482003

3.bilibili

https://space.bilibili.com/18161609/channel/index

尽可能每周更新