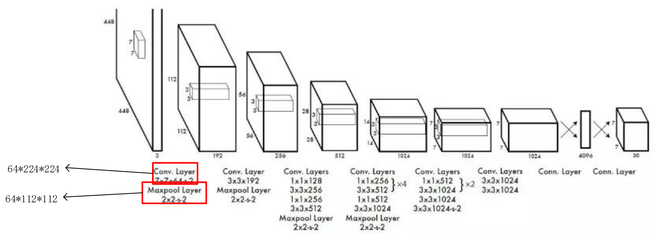


448*448->7*7 每个pixel都要回归预测B=2(2个框)以及两个框分别的conf,以及分类的prob(voc中cls=20)
7*7 * (2*(4+1) + 20) = 7*7 * 30
坐标回归和分类的loss不会一样重要,不稳定可能发散

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

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

坐标预测

判断第i个网格中的第j个box是否负责这个object

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

含object的box的confidence预测

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

不含object的box的confidence预测

判断是否有object中心落在网格i中

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

类别预测

Yolo v2 base net

Type	Filters	Size/Stride	Output
Convolutional	32	3 × 3	224 × 224
Maxpool		2 × 2/2	112 × 112
Convolutional	64	3 × 3	112 × 112
Maxpool		2 × 2/2	56 × 56
Convolutional	128	3 × 3	56 × 56
Convolutional	64	1 × 1	56 × 56
Convolutional	128	3 × 3	56 × 56
Maxpool		2 × 2/2	28 × 28
Convolutional	256	3 × 3	28 × 28
Convolutional	128	1 × 1	28 × 28
Convolutional	256	3 × 3	28 × 28
Maxpool		2 × 2/2	14 × 14
Convolutional	512	3 × 3	14 × 14
Convolutional	256	1 × 1	14 × 14
Convolutional	512	3 × 3	14 × 14
Convolutional	256	1 × 1	14 × 14
Convolutional	512	3 × 3	14 × 14
Maxpool		2 × 2/2	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	512	1 × 1	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	512	1 × 1	7 × 7
Convolutional	1024	3 × 3	7 × 7
Convolutional	1000	1 × 1	7 × 7
Avgpool		Global	1000
Softmax			

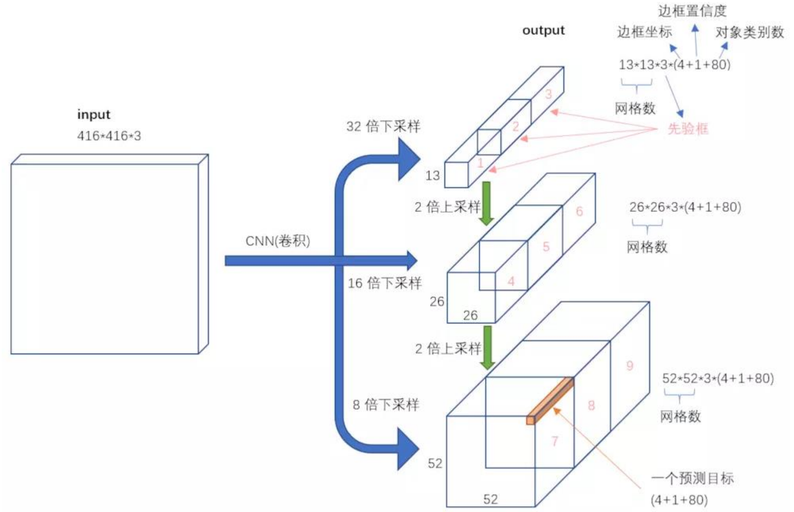
Table 6: Darknet-19.

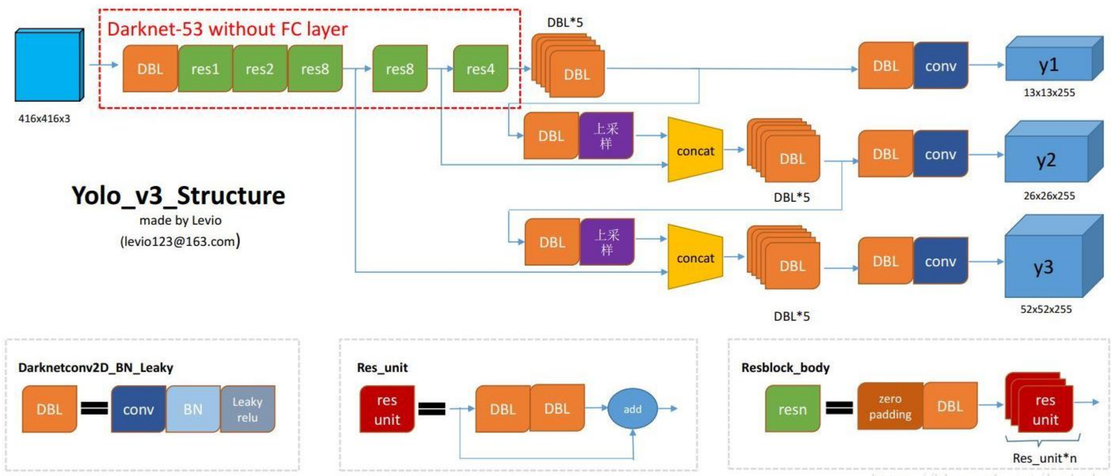
Yolo v3 base net

Type	Filters	Size	Output
Convolutional	32	3 × 3	256 × 256
Convolutional	64	3 × 3 / 2	128 × 128
1× Convolutional	32	1 × 1	
Convolutional	64	3 × 3	
Residual			128 × 128
Convolutional	128	3 × 3 / 2	64 × 64
2× Convolutional	64	1 × 1	
Convolutional	128	3 × 3	
Residual			64 × 64
Convolutional	256	3 × 3 / 2	32 × 32
8× Convolutional	128	1 × 1	
Convolutional	256	3 × 3	
Residual			32 × 32
Convolutional	512	3 × 3 / 2	16 × 16
8× Convolutional	256	1 × 1	
Convolutional	512	3 × 3	
Residual			16 × 16
Convolutional	1024	3 × 3 / 2	8 × 8
4× Convolutional	512	1 × 1	
Convolutional	1024	3 × 3	
Residual			8 × 8
Avgpool		Global	
Connected		1000	
Softmax			

Table 1. **Darknet-53.**

张量的尺寸变换是通过改变卷积核的步长来实现的





418*418的图像最后感受野是32倍,所以feature map是13*13
上采样的方法来实现这种多尺度的feature map=>13*13的yolo 26*26的yolo 52*52的yolo
992*992的图像最后感受野是32倍,所以feature map是31*13
上采样的方法来实现这种多尺度的feature map=>31*31的yolo 62*62的yolo 124*124的yolo
若每个yolo层预测的anchor(3)个数*(cls_prob(coco:80)+bbox(4)+conf(1))=3*85=255
13*13 yolo层的每个pixel*255 = 13*13 * 255 若是31*31的yolo就是31*31 * 255
26*26 yolo层的每个pixel*255 = 26*26 * 255
52*52 yolo层的每个pixel*255 = 52*52 * 255
若每个yolo层预测的anchor(2)个数*(cls_prob(2)+bbox(4)+conf(1))=2*7=14
13*13 yolo层的每个pixel*14= 13*13 * 14 若是31*31的yolo就是31*31 * 14
26*26 yolo层的每个pixel*14= 26*26 * 14
52*52 yolo层的每个pixel*14= 52*52 * 14
其中每一层yolo的anchor的size是自己取的(通过聚类),采用多尺度来对不同size的目标进行检测,越精细的grid cell就可以检测出越精细的物体
就是最后越小的feature map适用最大的anchor,因为最小的feature map感受野最大
可以自己换backbone(densenet等) 或是根据实际需要 修正darknet-53
anchor(3)个数就意味着418*418的每张图会生成13*13*3+26*26*3+52*52*3 = 10647个bbox, 992*992的每张图会生成31*31*3+62*62*3+124*124*3 = 60543个bbox
一张图中真正的object是少数,所以真正能预测object的bbox只占极少数,大部分bbox都是BG,造成了数据极大的不平衡
one-stage不如two-stage(faster rcnn)的部分就是two-stage有个rpn层会提前过滤大多数无用的BG部分,并保证正例和负例数据的平衡
所有就有后来的RetinaNet提出的Focal Loss