YOLO

**创新：**

YOLO将物体检测作为回归问题求解。基于一个单独的end-to-end网络，完成从原始图像的输入到物体位置和类别的输出。从网络设计上，YOLO与rcnn、fast rcnn及faster rcnn的区别如下：

（1）YOLO训练和检测均是在一个单独网络中进行。YOLO没有显示地求取region proposal的过程。而rcnn/fast rcnn 采用分离的模块（独立于网络之外的selective search方法）求取候选框（可能会包含物体的矩形区域），训练过程因此也是分成多个模块进行。Faster rcnn使用RPN（region proposal network）卷积网络替代rcnn/fast rcnn的selective search模块，将RPN集成到fast rcnn检测网络中，得到一个统一的检测网络。

（2）YOLO将物体检测作为一个回归问题进行求解，输入图像经过一次inference，便能得到图像中所有物体的位置和其所属类别及相应的置信概率。而rcnn/fast rcnn/faster rcnn将检测结果分为两部分求解：物体类别（分类问题），物体位置即bounding box（回归问题）。

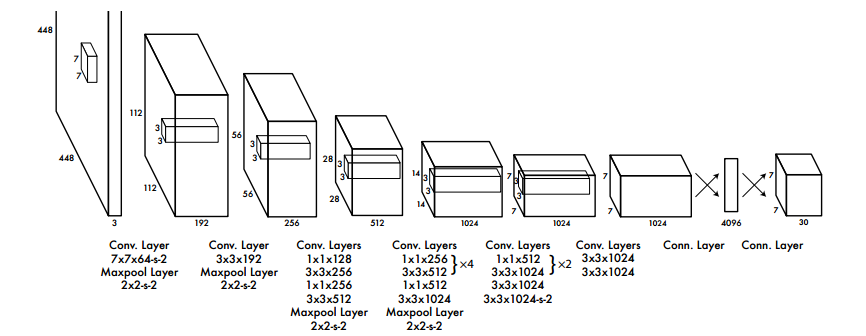
**核心思想：**

“物体落在哪个cell，哪个cell就负责预测这个物体”， 要分两个阶段来看，包括训练和测试。

训练阶段。在训练阶段，如果物体中心落在这个cell，那么就给这个cell打上这个物体的label（包括xywh和类别）。也就是说我们是通过这种方式来设置训练的label的。换言之，我们在训练阶段，就教会cell要预测图像中的哪个物体。

测试阶段。因为你在训练阶段已经教会了cell去预测中心落在该cell中的物体，那么cell自然也会这么做。

**检测网络：**

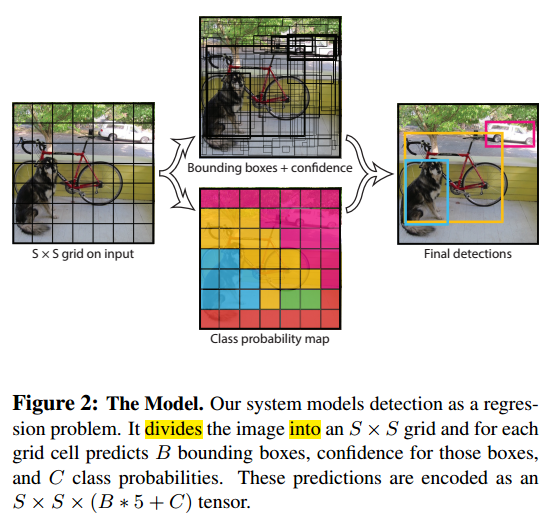


YOLO检测网络包括24个卷积层和2个全连接层，如上图所示。其中，卷积层用来提取图像特征，全连接层用来预测图像位置和类别概率值。

YOLO网络借鉴了GoogLeNet分类网络结构。不同的是，YOLO未使用inception module，而是使用1x1卷积层（此处1x1卷积层的存在是为了跨通道信息整合）+3x3卷积层简单替代。

Unified Detection

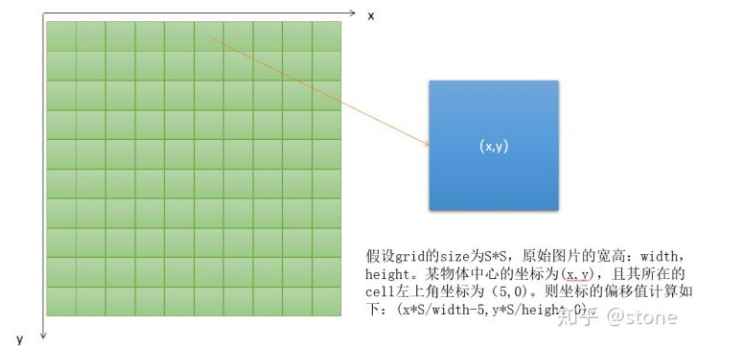
YOLO将输入图像分成SxS个格子，每个格子负责检测‘落入’该格子的物体。若某个物体的中心位置的坐标落入到某个格子，那么这个格子就负责检测出这个物体。



每个格子输出B个bounding box（包含物体的矩形区域）信息，以及C个物体属于某种类别的概率信息。

Bounding box信息包含5个数据值，分别是x,y,w,h,和confidence。其中x,y是指当前格子预测得到的物体的bounding box的中心位置的坐标。w,h是bounding box的宽度和高度。注意：实际训练过程中，w和h的值使用图像的宽度和高度进行归一化到[0,1]区间内；x，y是bounding box中心位置相对于当前格子位置的偏移值，并且被归一化到[0,1]。

偏移的计算方法如下：



confidence反映当前bounding box是否包含物体以及物体位置的准确性，计算方式如下：

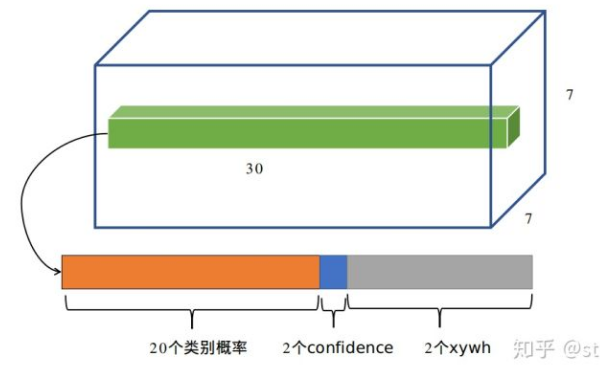


其中，若bounding box包含物体，则Pr(object) =1,confidence=IOU；否则Pr(object) = 0,confidence=0.

在测试时，将条件分类概率与与置信度相乘来作为分类概率：



因此，YOLO网络最终的全连接层的输出维度是 S\*S\*(B\*5 + C)。YOLO论文中，作者训练采用的输入图像分辨率是448\*448(预训练时224\*224)，S=7，B=2；采用VOC 20类标注物体作为训练数据，C=20。因此输出向量为7\*7\*(20 + 2\*5)=1470维。作者开源出的YOLO代码中，全连接层输出特征向量各维度对应内容如下：



**损失函数**

[1] 位置相关误差（坐标、IOU）与分类误差对网络loss的贡献值是不同的，因此YOLO在计算loss时，使用修正。

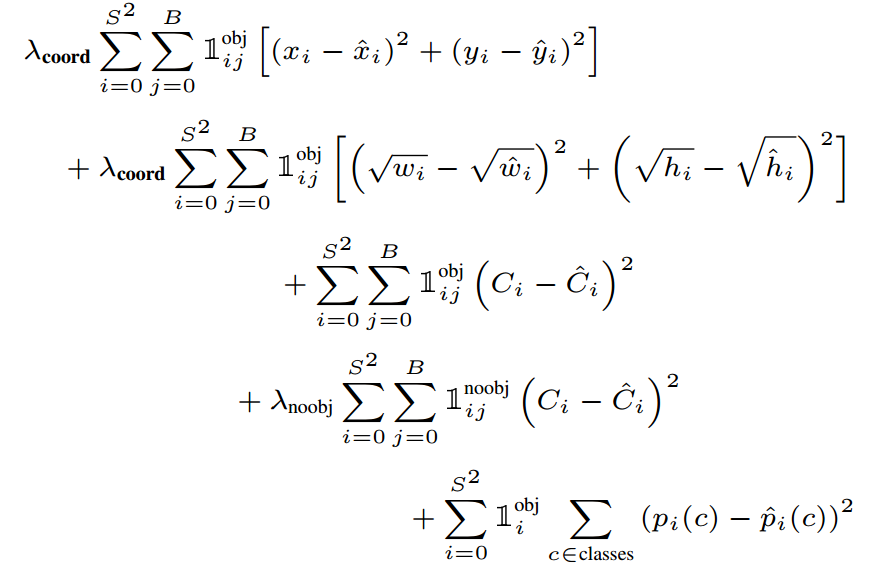
[2] 在计算IOU误差时，包含物体的格子与不包含物体的格子，二者的IOU误差对网络loss的贡献值是不同的。若采用相同的权值，那么不包含物体的格子的confidence值近似为0（这样的话不包含目标时就不对IOU进行调整），变相放大了包含物体的格子的confidence误差在计算网络参数梯度时的影响。为解决这个问题，YOLO 使用修正。（*注此处的‘包含’是指存在一个物体，它的中心坐标落入到格子内*）。  
 [3]对于相等的误差值，大物体误差对检测的影响应小于小物体误差对检测的影响（实际中应强调后者，把后者变大）。这是因为，相同的位置偏差占大物体的比例远小于同等偏差占小物体的比例。YOLO将物体大小的信息项（w和h）进行求平方根来改进这个问题。（*注：这个方法并不能完全解决这个问题*）。

（平方根计算可扩大偏差在小框中对于检测的影响）

Example：偏差为5

大物体：

小物体： （扩大了对小物体偏差的影响）



其中，为网络预测值，为标注值。表示物体落入格子i中，和分别表示物体落入与未落入格子i的第j个bounding box内。（每个格子都要打标签）

**训练集标签**

**Confidence预测**

对于训练阶段来说，我们要给每个bounding box的confidence打label，那么这个label怎么算? 其实很简单，如果一个物体中心没有落在cell之内，那么每个bounding box的  ，IOU就没有算的必要了，因此confidence的label就直接设置为0。如果物体的中心落在了这个cell之内，这个时候  ，因此confidence变成了  ，lable设置为1，预测 IOU是在训练过程中不断计算得出。

对于预测阶段，网络只输出一个confidence值，它实际上隐含地包含了

**类别预测**

对于训练阶段，也就是打label阶段，怎么打label呢？对于一个cell，如果物体的中心落在了这个cell，那么我们给它打上这个物体的类别label，并设置概 率为1。换句话说，这个概率是存在一个条件的，这个条件就是cell存在物体。

对于测试阶段来说，网络直接输出，就已经可以代表有物体存在的条件下类别概率。但是在测试阶段，作者还把这个概率乘上了confidence。



**训练**

YOLO模型训练分为两步：

1）预训练。使用ImageNet1000类数据训练YOLO网络的前20个卷积层+1个average池化层+1个全连接层。训练图像分辨率224x224。

2）用步骤1）得到的前20个卷积层网络参数来初始化YOLO模型前20个卷积层的网络参数，然后用VOC 20类标注数据进行YOLO模型训练。为提高图像精度，在训练检测模型时，将输入图像分辨率resize到448x448。