# 研究背景

基于

# 可参考主流框架

## 语义分割类：

### Deep Q Learning Driven CT Pancreas Segmentation with Geometry-Aware U-Net[1]

这篇文章基本继承了Active Object Localization with Deep Reinforcement Learning的思路

#### 主要思路：

在医学CT图片中，由于胰脏的占比范围变化大，且形状奇怪、占图像总比例小（0.5%左右），样本不平衡现象明显，对于图像的语义分割难度较大。使用Deep-Q来作context-adaptive localization policy（初步定位），以对于胰脏进行初步的标定，画出Bounding box, 再使用deformable U-Net进行进一步划分

#### 网络结构

先使用DQN来作定位，圈出大致位置后在进行U-net语义分割

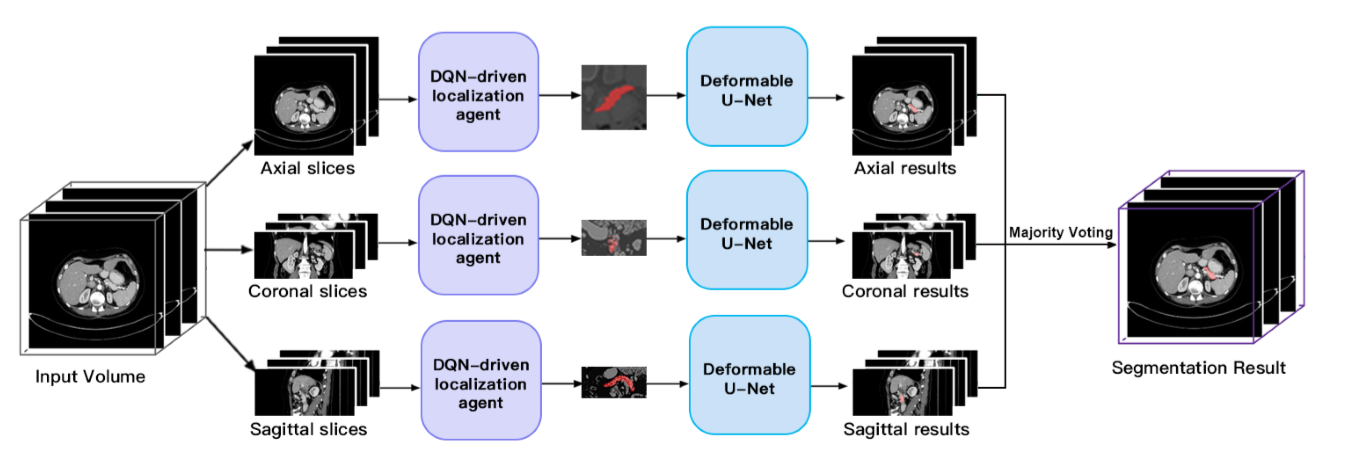


图 1：网络结构

#### 强化学习问题定义

agent可以有十种**action**，action定义为5中（1/4）偏离的zoom操作和4种1/4 的shift操作，最后一项当准确率足够大或者超出探索总步长时被触发 （trigger action）

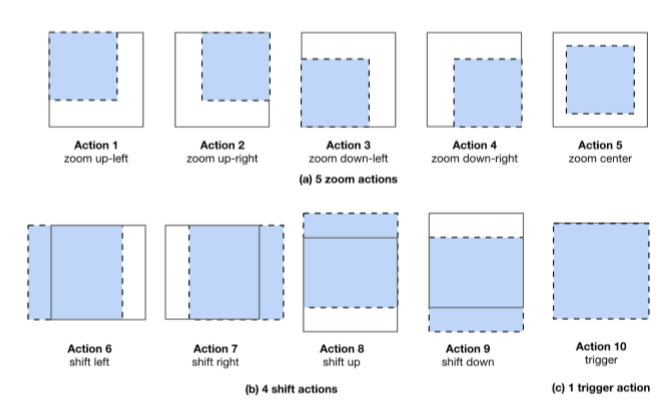


图 2：DQN圈选bounding box时，agent的10中action

**State**：定义为当前的region descriptor（即bounding box）以及一个memory vector以记录最后10个action，这些数据共同作为当前的state

**Reward:** 对于moving action和trigger action采用不同的reward定义，对于一个moving action

G代表ground truth，但是作者使用真正的胰脏mask（sematic annotation）而不是bounding box作为ground truth，因此需要注意的是这个IoU永远不可能接近1，w’是当前的bounding box，w是上一步的bounding box。对于trigger action我们定义：

和都是两个threshold作为超参数，IoU用来找到目标，而recall用来保证包含了BBX主要部分

#### 数据集和实验结果

实验使用数据集：NIH pancreas segmentation dataset（引用Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation）

性能比较：

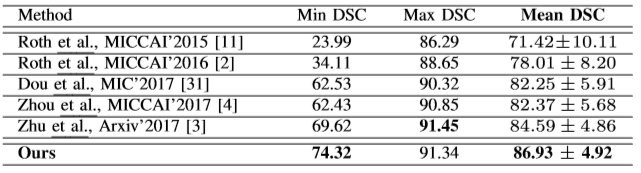
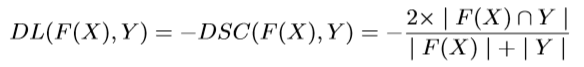


表 1：性能比较

Dice similarity coefﬁcient定义为：



### DeepOutline[2]

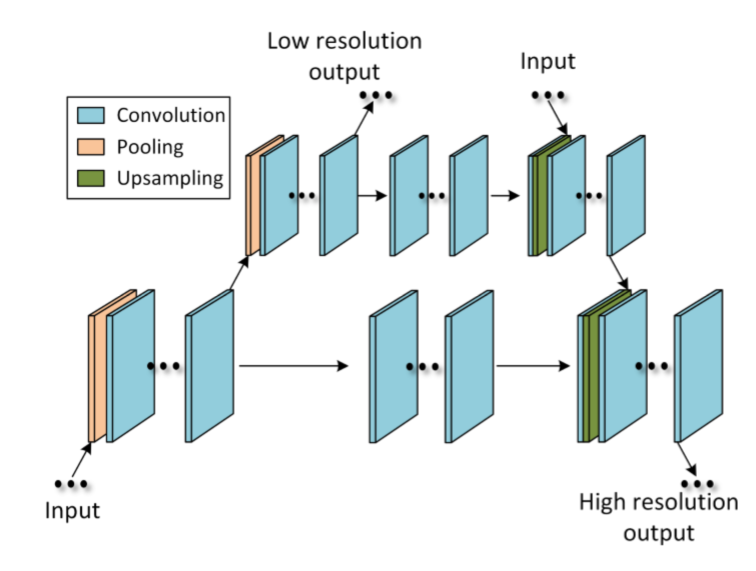
将每个物体定义为一种多边形，一张图可以有多个多边形边界

#### 主要思路：

使用类似人的思维，逐步一个点、一个点地画出物体轮廓

#### 网络结构：

采用了类似U-net的网络结构，不过在concatenate不同层的feature map之前，会把之前的feature map再过一遍卷积层，因此他们称之为delta-map

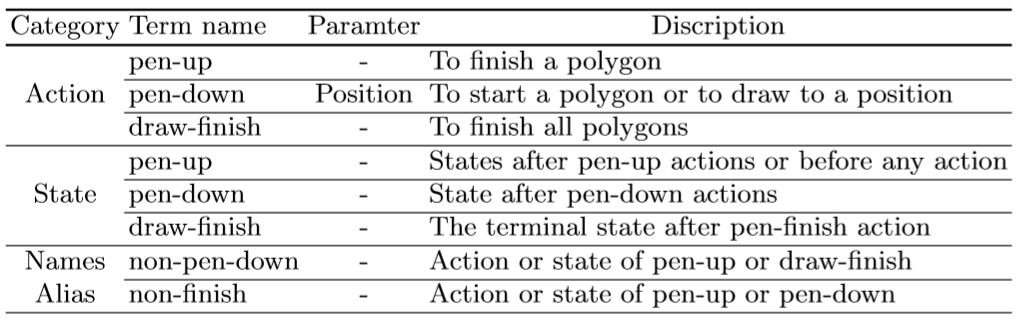


压缩至最小的feature map再经过FC输出3个动作的概率，还原的大图被当做pen-down的position map.

#### 强化学习问题定义:

使用了Q-learning作为主要的强化算法

**Action:**



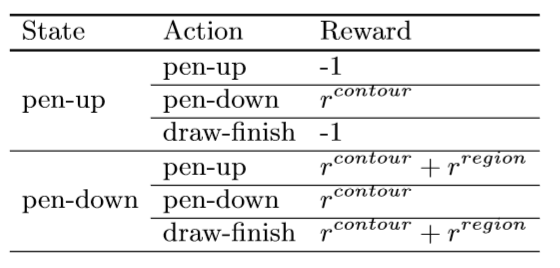
pen-up指完成一个多边形勾画的动作，pen-down是指开始勾画一个新的多变形或者寻找下一个位置，draw finish勾画过程结束

**State**：1. 两个与图像相同大小State map, 初始值为0。第一张用来表征画完或者正在画的多边形，正在画的用1来填充，已经画完的用0.5来填充。第二张state map用来储存每一个多边形最后一笔和第一笔的位置。如果最后一笔是pen-up动作，那么所有的entry全部设置为0。

**Reward：**两种reward来衡量边界画的是否符合以及最终的勾画区域是否吻合ground truth。第一种为contour reward，从上一个state到下一个state。B(x,y)为经过高斯模糊的边界。

第二种为区域奖励, m为所有找到的多边形，n是所有ground truth多边形， ：

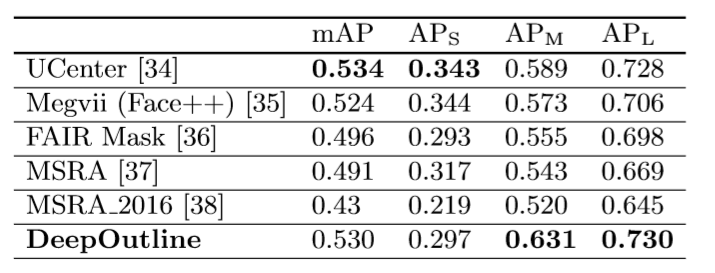
对于不同的状态做出的不同的动作，我们给出不同的回报值：



#### 数据集和实验结果：

使用了COCO2017数据集

表现结果如下图所示：对于Middle和Large size的图像，效果相对较为显著。



## 目标检测、定位类

### Active Object Localization with Deep Reinforcement Learning[3]

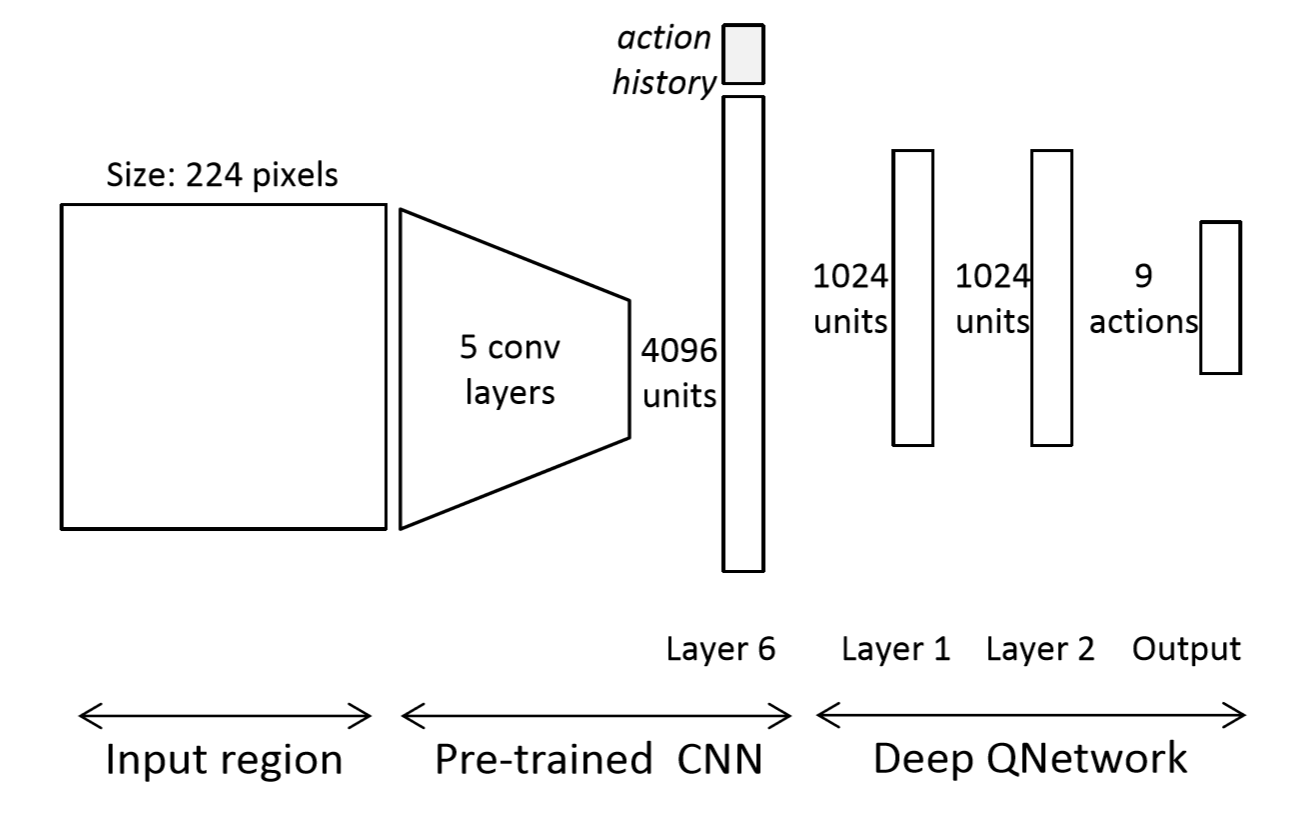
此方法作为强化学习在object localization的首次成功尝试，被iccv2015收录其中

#### 主要思路：

使用动态注意力机制，通过强化学习将物体逐步圈出

#### 网络结构

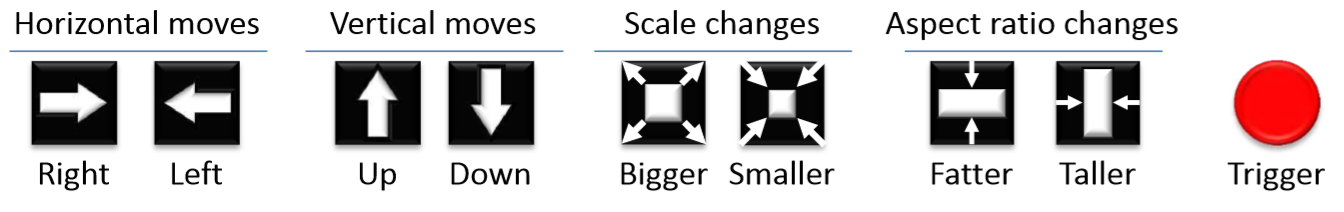
Pre-trained CNN在训练中参数不变，只训练deep-Q中的参数



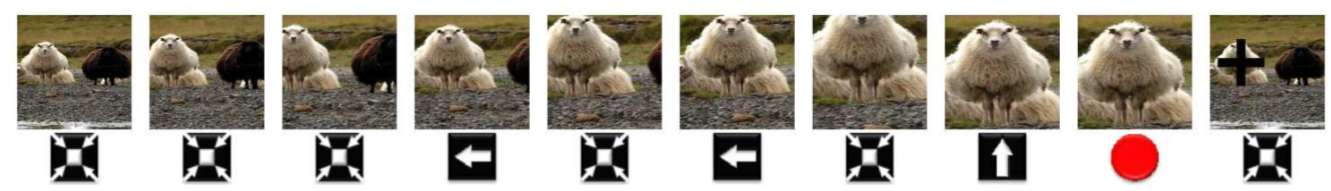
#### 强化学习问题定义

将每次bounding box的变化认为是一种MDP

**Action：**八个关于box的变形，1个action来终结此次标定，通过实验，每次移动的比例大概在0.2个图片的大小比较合适（比如，如果我们向左移动，那么就沿着横向方向的边长移动20%）



当agent做出trigger动作时候，同时会改变图像，给标定区域打叉从而避免该区域被再次注意（inhibition-of-return）



**State：**由特征向量和历史动作（o和h）构成，特征向量由attended region变为224\*224图像后经过卷积层变为4096的vector。而h是历史动作，每一条历史动作是一个长度为9的one hot向量，实验中作者采用10条历史记录，以避免循环出现的动作，提高准确率。

**Reward:** 对于moving action和trigger action采用不同的reward定义，对于一个moving action

g代表ground truth，这个回报值表征IoU每一步的变化

设置为3.0，。

虽然文中采用了-greedy，但是随机动作的选择并不是完全随机的，而是从一些可以获得positive reward的动作中随机选择的（如果没有positive的动作那么就完全随机）。也就是说限定了随机动作选择的范围。

#### 数据集和实验结果

数据集采用Pascal VOC 2007 dataset

实验效果虽然比不上R-CNN但是依旧超过很多baseline的表现

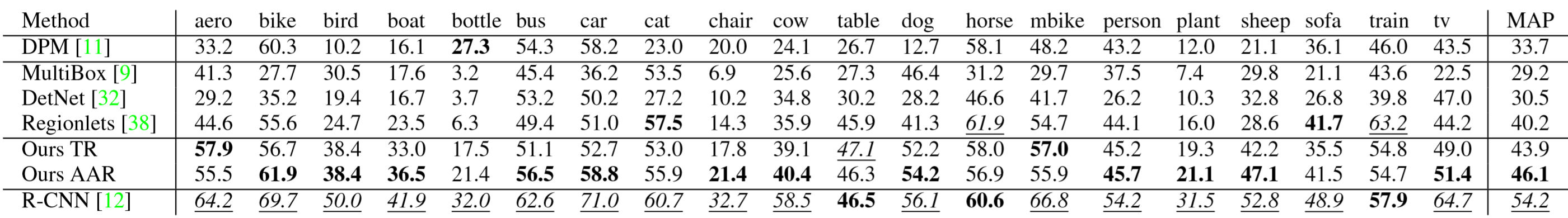


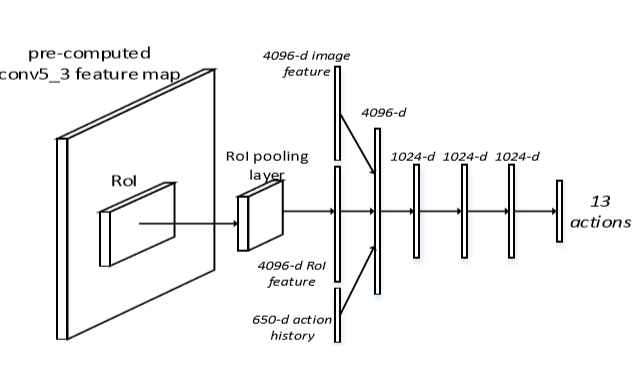
表 2：Active object localization with RL性能表现

### Tree-Structured Reinforcement Learning for Sequential Object Localization[4]

#### 主要思路：

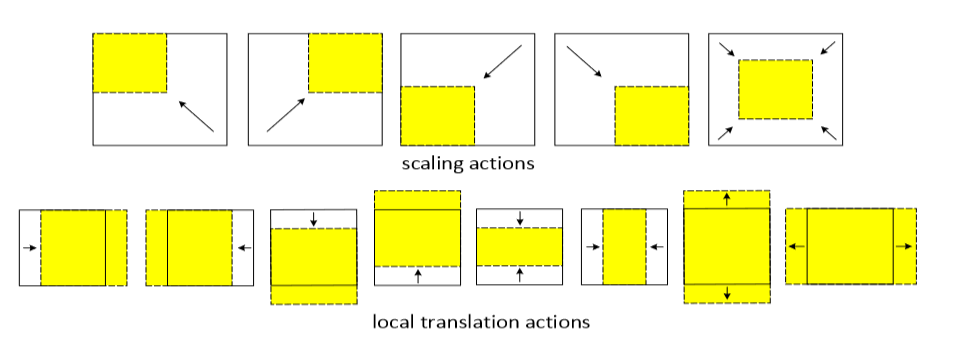
采取树结构进行多物体的定位检测，而对于树分支的方法，采用了基于强化的scale和translation的逐步优化的方法。

#### 网络结构,



#### 强化学习问题定义

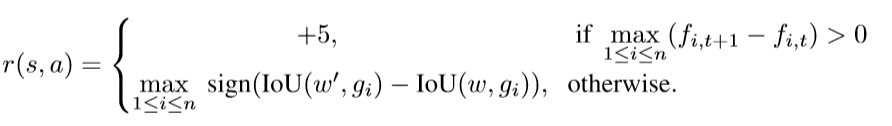
**Action：**一共有13种动作，前五种采用缩小，长款均缩小至原来的55%，而translation每次变化为边长的25%



**State：**原图片经过VGG16输出4096作为state部分1，直接使用算好的feature map找出对应的bbx对应区域，经过ROI pooling等输出的4096d的feature map，之前50步的历史记录，通过为13\*50 = 650维 的vector。将这三者叠加起来

**Reward：**

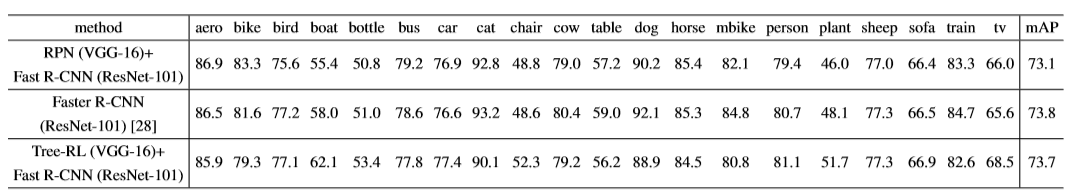
代表ground truth的bounding box，采取max的方式也就是找当前proposal和所有的bbx iou最大的一个。这里的用来停止优化bbx的，如果之前任何一个proposal的IOU大于50%，那么flag 就为1否则为-1。用这样的方法可以明确标定那个第一次超越50%的action。



#### 数据集及性能分析

数据集采用Pascal VOC 2007、2012dataset

与Faster RCNN比较，在VOC2012数据集上略逊一筹，但是与RPN等相比，在相对少的region proposal中可以获得更好的结果。

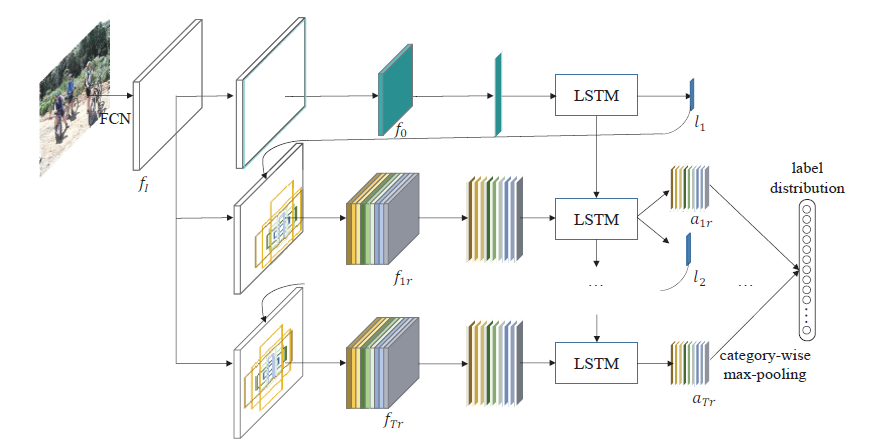


### Recurrent Attentional Reinforcement Learning for Multi-label Image Recognition

#### 主要思路：

应对多标签的分类问题，目前主流使用proposal region随后对每个proposal进行分类，文章中采用逐步提取proposal而后经过reinforcement learning方式对下一步proposal进行预测，并通过一个lstm记录之前预测过的proposal。最终将所有的proposal经过分类器得到一个全图分类。

#### 网络结构

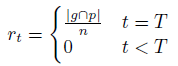


#### 强化学习问题定义

**Action：**每一步，agent做出两种动作，首先对提出的ROI进行分类，随后再找下一个可能的proposal中心位置。Action space是feature map上所有的像素，从中选取一个anchor point。

**State：**此时刻的所有的proposal regions和上一步的LSTM的hidden unit

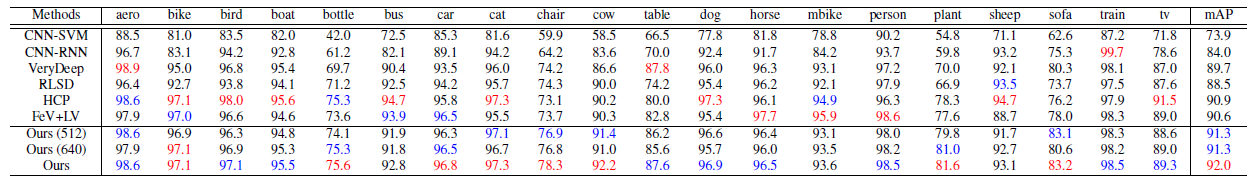
**Reward：**g是所有的ground truth label, 比如有n个而p是之前所有proposal region经过分类器后得到的前n名类别，reward是在最后一步才给出的，即所有的ground truth和prediction交集的数量除以所有总的该分类的数量。



#### 数据集及性能

使用Pascal VOC2007和MS-COCO进行训练

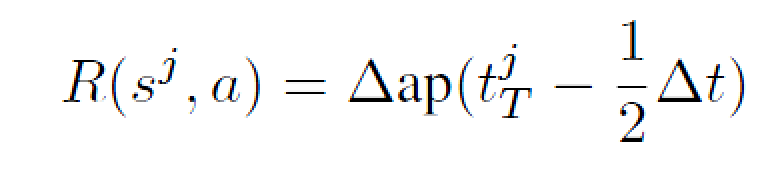
注，此处的结果对比是在multi-class中进行的，和前面的略有不同。



### Timely Object Recognition

#### 主要思路：

文章相对比较有意义的点在于reward通过AP来定义（average precision），表示每一个step的下一步对最终识别造成的影响表示每一步所用的时间表示距离termination的所剩余时间。意味着可以用一个end2end的思路进行训练网络。



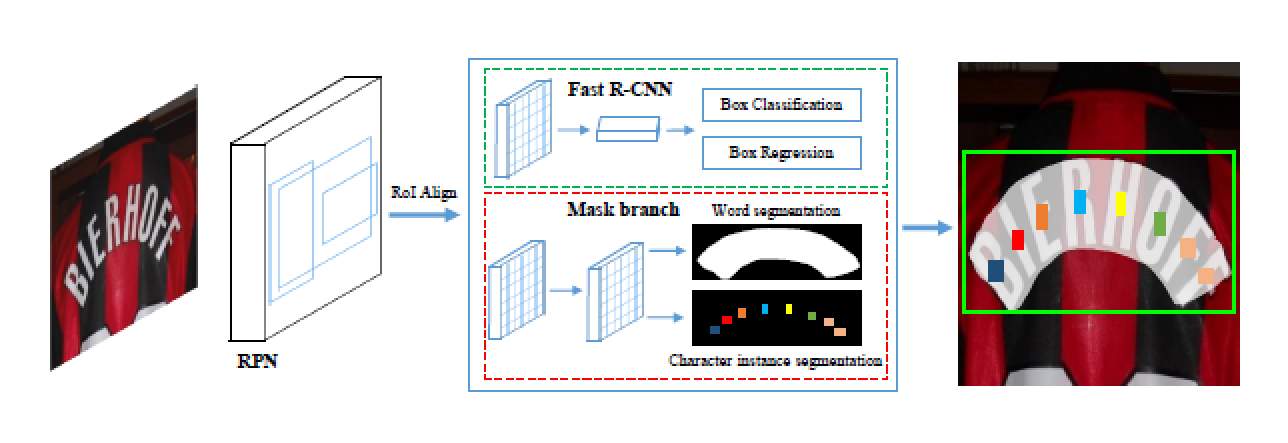
## 文字识别类

### Mask TextSpotter: An End-to-End Trainable Neural Network for Spotting Text with Arbitrary Shapes

#### 主要思路：

采用fast rcnn和mask将场景中的文字圈出来并进行语义分割，对36个类（26字母+10数字）进行character map的构造，配合mask的pooling，最终再一个edit distance构成最终的文字

#### 网络结构



#### 数据集和结果

貌似可以不限制长度的进行文字检测，毕竟只要有character map就可以一直加文字上去

# 想法与总结

* 目前的RL segmentation和localization问题在训练过程中基本都可以直接和ground truth互动获得最直接的reward（例如现在的bounding box和gt的区别，或者IoU的变化增减作为reward信号），但是我们的划分可能涉及到一个很长的end-to-end训练，不知是否可行
* 2.2.3 提出一种end to end的训练，但是对其训练效果持疑
* 划分的时候每次划分多少个字母，是否为定值，最终到底选取多少个proposal
* 为什么不用Faster RCNN一类的做region proposal，而使用RNN来做localization
* 有的text recognition的论文是直接从大图片上找，有的是直接crop

# 参考文献

[1] Y. Man, Y. Huang, J. Feng, X. Li, and F. Wu, “Deep Q Learning Driven CT Pancreas Segmentation with Geometry-Aware U-Net,” *IEEE Trans. Med. Imaging*, vol. 38, no. 8, pp. 1971–1980, 2019, doi: 10.1109/TMI.2019.2911588.

[2] Z. Wang, S. Sarcar, J. Liu, Y. Zheng, and X. Ren, “Outline Objects using Deep Reinforcement Learning,” 2018.

[3] J. C. Caicedo and S. Lazebnik, “Active object localization with deep reinforcement learning,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2015 Inter, pp. 2488–2496, 2015, doi: 10.1109/ICCV.2015.286.

[4] Z. Jie, X. Liang, J. Feng, X. Jin, W. F. Lu, and S. Yan, “Tree-structured reinforcement learning for sequential object localization,” *Adv. Neural Inf. Process. Syst.*, no. Nips, pp. 127–135, 2016.