读的论文的题目：
R-CNN:Rich feature hierarchies for accurate object detection and semantic segmentation
Fast R-CNN:Fast R-CNN
Faster R-CNN:Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks
SPP:Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition
Mimick:Mimicking Very Efficient Network for Object Detection

下面主要写，这五种方法的简述，优点，缺点，性能的比较,以及对Mimic方#法改进的思路

R-CNN

R-CNN的简述：

1. 用Selective search产生类无关的候选域
2. 用大的CNN网络，从候选域中提取出固定长度的特征
3. 用一系列的SVM分类

训练过程： 有监督的预训练+特定领域的微调
有监督的预训练：使用ILSVRC 2012数据集预训练CNN网络
特定领域的微调：使用VOC数据集对CNN网络进行微调，优化SVM,bounding-box回归

R-CNN的优点

相对与以前基于HOG和SIFT的目标检测方法，精确率提高了很多，mAP提高了大概20%

R-CNN的缺点

1. 训练需要分为几个阶段：预训练，微调CNN,优化SVM,训练bounding-box 回归
2. 训练耗费空间大：需要1.8GB存储特征
3. 物体检测速度慢：7层网络结构每张图片需要10s,16层网络结构每张图片需要53s

Fast R-CNN

Fast R-CNN的简述：

1. 输入整张图片和一些物体候选域，通过一系列卷积层和池化层产生整张图片的feature map
2. 为每个RoI,池化层从feature map中提取出固定长度的特征向量
3. 特征输入到一系列全连接层，输出：k+1类的概率，每个类的一个四元组，用于bounding box回归

对R-CNN的改进：
1. 来自同一图片的RoI共享前向和后向传播的计算和内存
2. 使用了Multi-task loss:实现了end-to-end的训练

Fast R-CNN的优点

1. 训练是单阶段的
2. 特征不需要缓存到磁盘中去

Fast R-CNN的缺点

产生物体的候选域存在时间瓶颈

Faster R-CNN

Faster R-CNN简述

候选域的产生使用RPN，物体检测使用Fast R-CNN

Faster R-CNN的优点

检测速度快，精确率高

SPP

SPP简述

计算整张图片的Feature map一次，使用SPP为每个候选域提取固定长度的特征

SPP的优点

1. 接收可变大小的输入
2. 各个patch块之间共享卷积计算

SPP的缺点

训练个测试都分为几个阶段，所以速度比较慢
无法更新SPP层后面的层，反向传播是高度无效的

Mimick

Mimick简述

用Faster R-CNN网络和真实数据监督训练一个小的网络
训练：使用模仿的方法训练RPN,产生候选域，然后用RPN产生的候选域微调Faster-RCNN

Mimick的优点

在使用比大网络少的参数和时间的情况下，达到跟大网络相近的检测精确度

Mimick的缺点

目标检测的精确度没有提高

五种方法性能比较

测试时间比较

方法	网络结构	设备	时间（s）
R-CNN	T-Net(7层)	Nvidia Titan Black GPU	10
R-CNN	T-Net(7层)	CPU	53
Fast R-CNN	VGG16(16层)	Nvidia K40 GPU	0.32
Faster R-CNN	VGG16(16层)	K40GPU	0.178
Faster R-CNN	ZF(7层)	K40GPU	0.059
SPP（1尺度）	ZF(7层)	GeForce GTX Titan GPU(6GB)	0.142
SPP（5尺度）	ZF(7层)	GeForce GTX Titan GPU(6GB)	14.46
1/2-Mimic-finetune	VGG16(16层)	TITANX	0.0228

mAP比较

训练集：VOC 2012,测试集： VOC 2010

方法	网络结构	mAP(%)
R-CNN	T-Net(7层)	53.7
R-CNN	O-Net(16层)	62.9
Fast R-CNN	VGG16(16层)	66.1

训练集：VOC 2007,测试集：VOC 2007

方法	网络结构	mAP(%)
R-CNN	T-Net(7层)	58.5
R-CNN	O-Net(16层)	66.0
Fast R-CNN	VGG16(16层)	66.9
SPP(1尺度)	ZF(7层)	58
SPP(5尺度)	ZF(7层)	59.2
Faster R-CNN	ZF(7层)	59.9

方法	网络结构	mAP(%)
Faster R-CNN	VGG16(16层)	69.9

训练集：VOC 07+12(07的训练验证集加12的训练验证集),测试集：VOC 2007

方法	网络结构	mAP(%)
Fast R-CNN	VGG16(16层)	70.0
Faster R-CNN	ZF(7层)	59.9
Faster R-CNN	VGG16(16层)	**73.2**
1/2-Mimic-finetune	VGG16(16层)	**72.79**
1/4-Mimic-finetune	VGG16(16层)	65.76
1/4-two-Mimic	VGG16(16层)	67.66
1/8-two-Mimic	VGG16(16层)	56.14

其他训练集和测试集下的mAP,网络结构都是VGG16(16层)

方法	训练集	测试集	mAP(%)
Fast R-CNN	07\diff	07	68.1
Fast R-CNN	07++12	10	68.8
Fast R-CNN	07++12	12	68.4
Fast R-CNN	12	12	65.7
Faster R-CNN	COCO+07+12	07	**78.8**
Faster R-CNN	12	12	67.0
Faster R-CNN	07++12	12	**70.4**
Faster R-CNN	COCO+07++12	12	**75.9**

Mimic方法改进

原来的网络结构为：

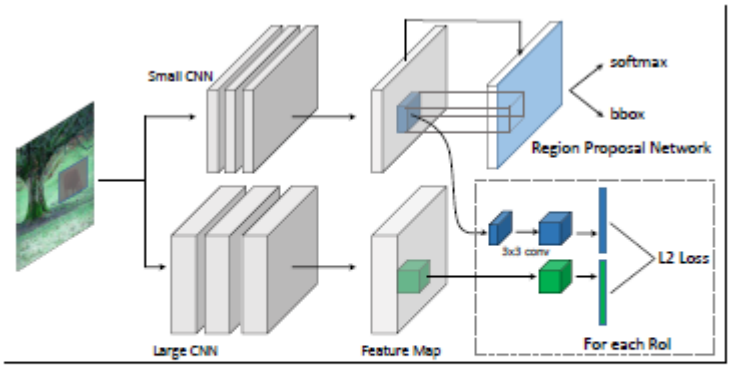
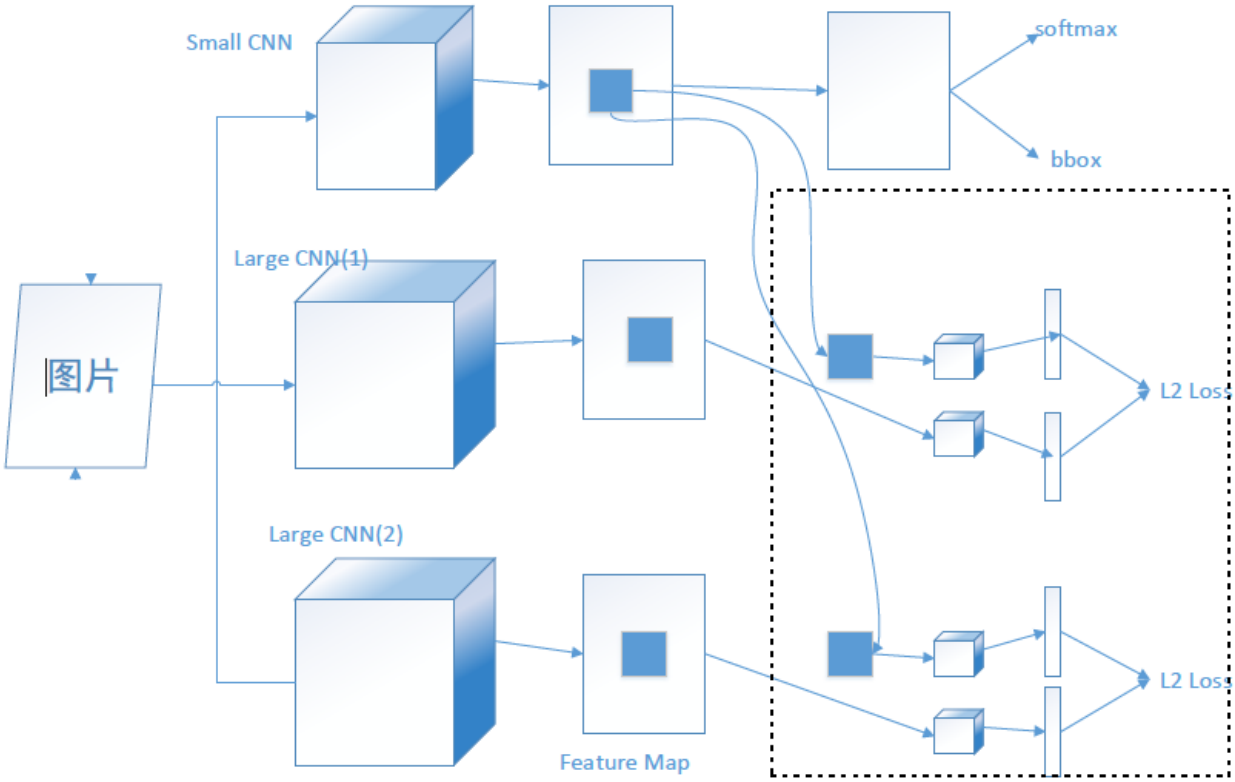


Figure 1: Overall architecture of feature mimic by proposal sampling. A Region Proposal Network generates candidate RoIs, which then used to extract features from the feature maps.

改进后的网络结构为：



原来的损失函数为：

$$L(W) = \lambda_1 L_m(W) + L_{gt}(W)$$

$$L_m(W) = \frac{1}{2N} \sum_i \frac{1}{m_i} ||u^{(i)} - r(v^{(i)})||_2^2$$

$$L_{gt} = L_{cls}(W) + \lambda_2 L_{reg}(W)$$

修改后的损失函数为：

$$L1_m(W) = \frac{1}{2N}\sum_i \frac{1}{m_i}\,||u_1^{(i)} - r(v^{(i)})||_2^2$$

$$L2_m(W) = \frac{1}{2N}\sum_i \frac{1}{m_i}\,||u_2^{(i)} - r(v^{(i)})||_2^2$$

$$L_m(W) = x_1 L1_m + x_2 L2_m$$

如果大网络1检测到了物体$x_1 = 1$否则等于0
如果大网络2检测到了物体$x_2 = 1$否则等于0
损失函数可以理解为：如果那个大网络检测到了物体则可以理解为，这个网络对这张图片提取的feature map比较好，应该用这个网络来监督小网络训练，如果大网络没有检测到物体，那么说明这个大网络对这张图片提取的feature map不好，不应该用来误导小网络
使用的两个大网络要仔细选择，计算特征的方法不同，但性能相差不多的网络，比如选fast rcnn和faster rcnn就不好，因为这两个网络计算的特征应该是比较像的，而且faster明显优于fast，这样的话fast的贡献就太少了