Fast R-CNN

By Ross .B. Girshick

（这个人之前还写了R-CNN,SPP net，现在最新的用fast R-CNN，也就是王立威老师用的方法）

开源：https://github.com/rbgirshick/fast-rcnn

Abstract：

FastRCNN是RCNN和SPPnet的改进。

Introduce:

RCNN和SPPnet缺点

1. Training is a multi-stage pipeline

2. Training is expensive in space and time

3. Object detection is slow

Fast RCNN的优点

1. Higher detection quality (mAP) than R-CNN, SPPnet

2. Training is single-stage, using a multi-task loss

3. Training can update all network layers

4. No disk storage is required for feature caching

Fast R-CNN architecture and training

整个流程的概述

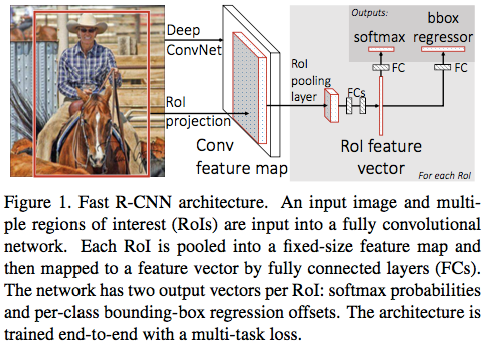
1. processes the whole **image** with several convolutional (conv) and max pooling layers to produce a **conv feature map**

2. for each object proposal a **region of interest** (RoI) pooling layer extracts a fixed-length **feature vector** from the feature map.

3.Each feature vector is fed into a sequence of fully connected (fc) layers that finally branch into **two sibling output layers**:

①produces softmax probability estimates over **K object classes** plus a **catch-all “background” class**②outputs four real-valued numbers for each of the K object classes.

(Each set of 4 values encodes **refined bounding-box positions** for one of the K classes)



1. RoI pooling layer

**convert** the features inside valid ROI into feature map with a fixed spatial extent of H × W

**divide** the *h × w* RoI window into an *H × W* grid of sub-windows of approximate size *h/H × w/W* **max-pooling** the values in each sub-window into the corresponding output grid cell

RoI pooling layer是一个SPP的简化版，a Pyramid level

(理解为feature map经过RoI pooling layer得到固定大小（变小了）的feature map)

2.Initializing from pre-trained networks

转变三个预训练好的网络结构（Alexnet、VGG\_CNN\_M\_1024、VGG16)

①最后的pooling层改为固定大小的RoI Pooling层

②将分类和检测放到一起，全连接后面分出来两个sibling分支分别做分类和检测

③输入变为一幅图+ RoI列表

3.Fine-tuning for detection

hierarchical sampling and uses a streamlined training process with one fine-tuning stage that jointly optimizes a softmax classifier and bounding-box regressors

①Multi-task loss

分类和回归的loss

总的L=(p,u,tu,v)=Lcls(p,u)+λ[u≥1]Lloc(tu,v)L=(p,u,tu,v)=Lcls(p,u)+λ[u≥1]Lloc(tu,v)

②Mini-batch sampling

训练中每个batch中有两张图，有50%概率进行翻转，随机抽样64个RoI，其中正负样本的选取方式为：

其中25%的RoI中包含ground truth的50%（也就是有25%是正样本），lable值u>=1。

剩余的包含ground truth比例为[0.1,0.5]标记为负样本（u=0）

③Back-propagation through RoI pooling layers

前向传播没有问题，但反向传播出现多个RoI在pooling层冲突

如果是max pooling，不冲突时选择max pooling来源反向传播，冲突就直接累加。

4.Scale invariance

brute force （single scale）(简单认为object不需预先resize到类似的scale再传入网络，直接将image定死为某种scale，直接输入网络来训练就好了，然后期望网络自己能够学习到scale-invariance的表达)

image pyramids （multi scale）(生成一个金字塔，然后对于object，在金字塔上找到一个大小比较接近227x227的投影版本，然后用这个版本去训练网络)。

2好但慢，本文最终选了1

检测与结果