# 基于深度学习的目标检测算法综述

**作者：SIGAI**

**2018.04.24**

## 原创声明




### 导言

目标检测的任务是找出图像中所有感兴趣的目标（物体），确定它们的位置和大小，是机器视觉领域的核心问题之一。由于各类物体有不同的外观，形状，姿态，加上成像时光照，遮挡等因素的干扰，目标检测一直是机器视觉领域最具有挑战性的问题。本文将针对目标检测（Object Detection）这个机器视觉中的经典任务进行解析，抛砖引玉。如对文中的内容持不同观点，欢迎到 SIGAI 公众号发消息给我们，一起探讨！

### 什么是目标检测？

目标检测的任务是找出图像中所有感兴趣的目标（物体），确定它们的位置和大小，是机器视觉领域的核心问题之一。由于各类物体有不同的外观，形状，姿态，加上成像时光照，遮挡等因素的干扰，目标检测一直是机器视觉领域最具有挑战性的问题。

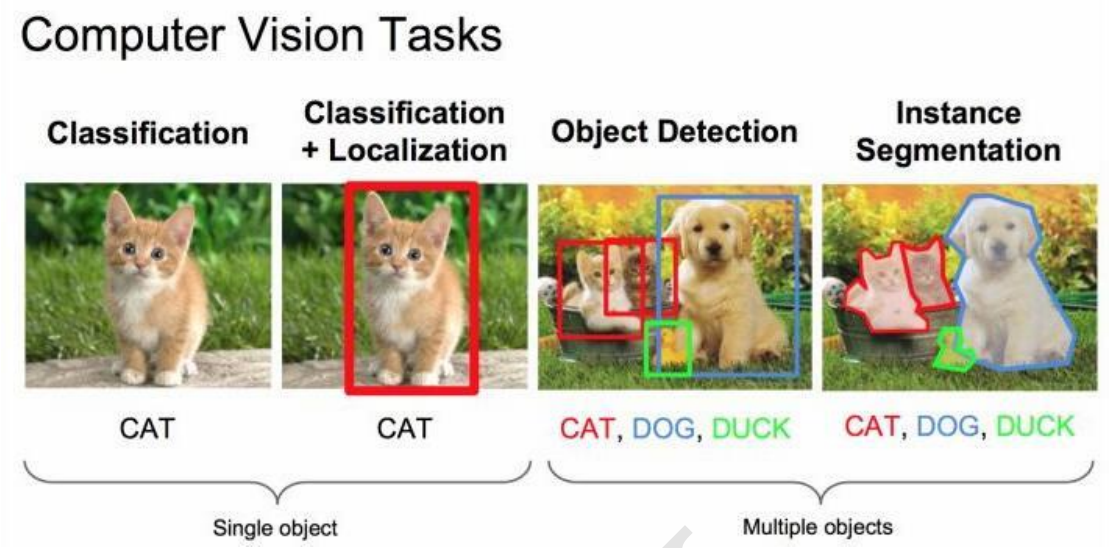
计算机视觉中关于图像识别有四大类任务：

分类-Classification：解决“是什么？”的问题，即给定一张图片或一段视频判断里面包含什么类别的目标。

定位-Location：解决“在哪里？”的问题，即定位出这个目标的的位置。

检测-Detection：解决“是什么？在哪里？”的问题，即定位出这个目标的的位置并且知道目标物是什么。

分割-Segmentation：分为实例的分割（Instance-level）和场景分割（Scene-level），解决“每一个像素属于哪个目标物或场景”的问题。




除了图像分类之外，目标检测要解决的核心问题是：

1.目标可能出现在图像的任何位置。

2.目标有各种不同的大小。

3.目标可能有各种不同的形状。

如果用矩形框来定义目标，则矩形有不同的宽高比。由于目标的宽高比不同，因此采用经典的滑动窗口+图像缩放的方案解决通用目标检测问题的成本太高。

目标检测的应用

目标检测在很多领域都有应用需求。其中被广为研究的是人脸检测，行人检测，车辆检测等重要目标的检测。人脸检测在 SIGAI 上一篇文章“人脸识别算法演化史”中已经简单介绍，后面我们会针对这个问题撰写综述文章。

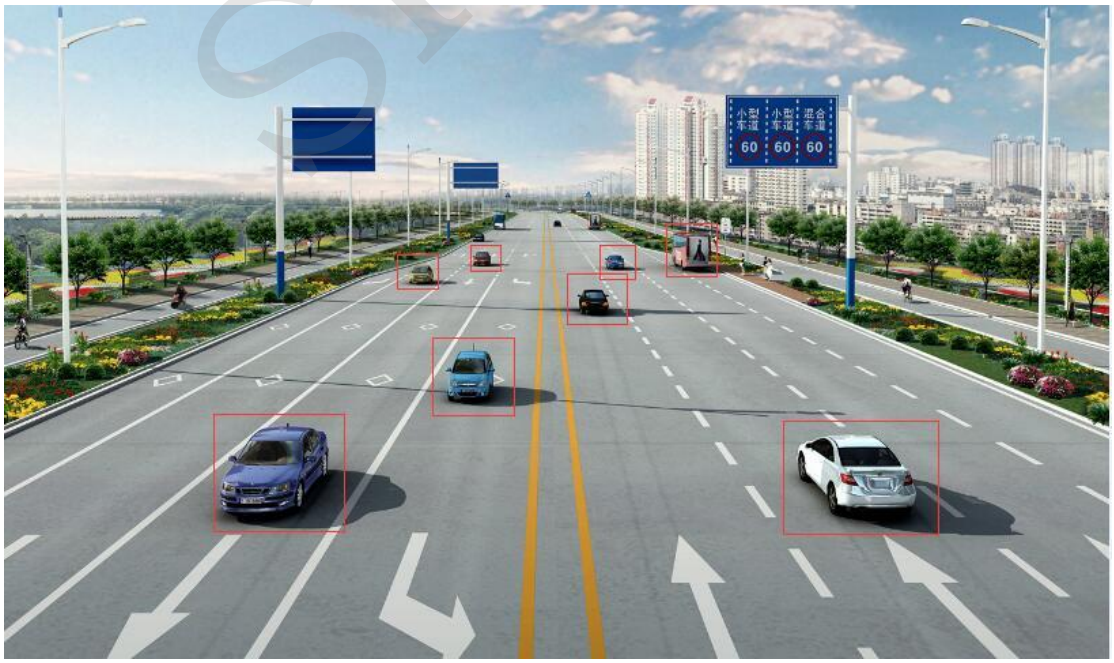
行人检测

行人检测在视频监控，人流量统计，自动驾驶中都有重要的地位，后续也会有相关综述文章。

行人检测

车辆检测

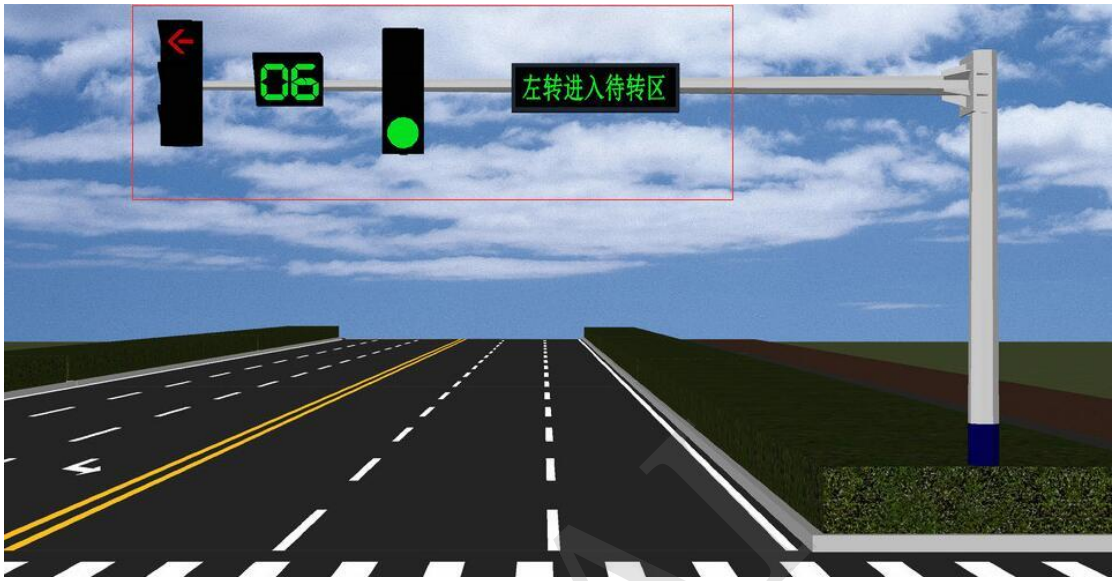
车辆检测在智能交通，视频监控，自动驾驶中有重要的地位。车流量统计，车辆违章的自动分析等都离不开它，在自动驾驶中，首先要解决的问题就是确定道路在哪里，周围有哪些车、人或障碍物。




车辆检测

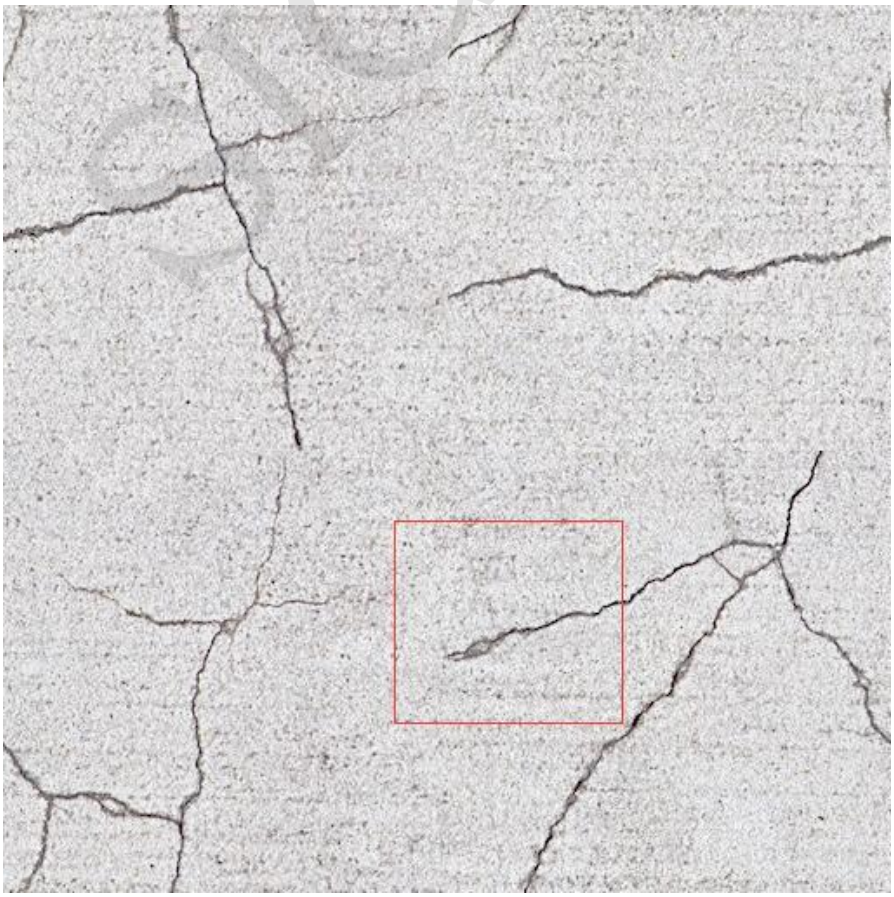
其他应用

交通标志如交通灯、行驶规则标志的识别对于自动驾驶也非常重要，我们需要根据红绿灯状态，是否允许左右转、掉头等标志确定车辆的行为。



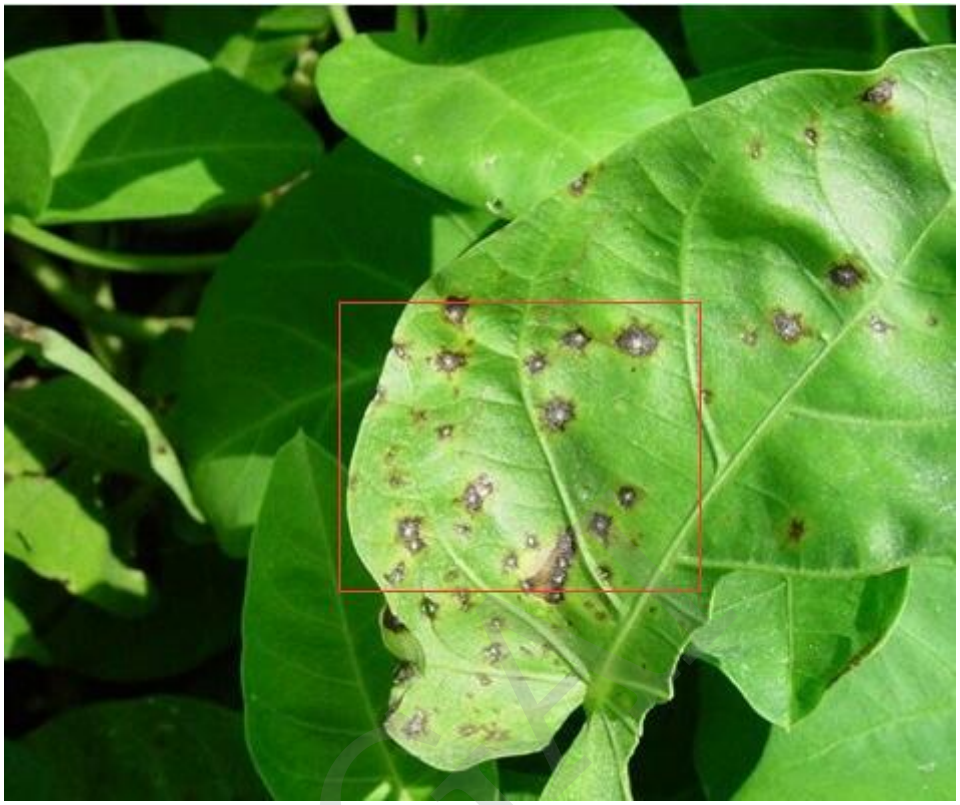

交通标志检测

除了这些常见目标的检测之外，很多领域里也需要检测自己感兴趣的目标。比如工业中材质表面的缺陷检测，硬刷电路板表面的缺陷检测等。



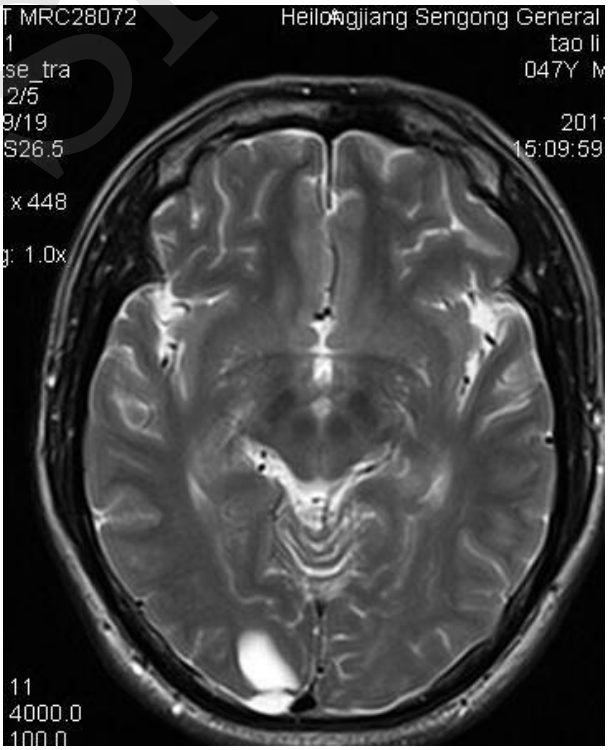
材质表面缺陷检测

农业中农作物表面的病虫害识别也需要用到目标检测技术：



农作物病虫害检测

人工智能在医学中的应用目前是一个热门的话题，医学影像图像如 MRI 的肿瘤等病变部位检测和识别对于诊断的自动化，提供优质的治疗具有重要的意义。



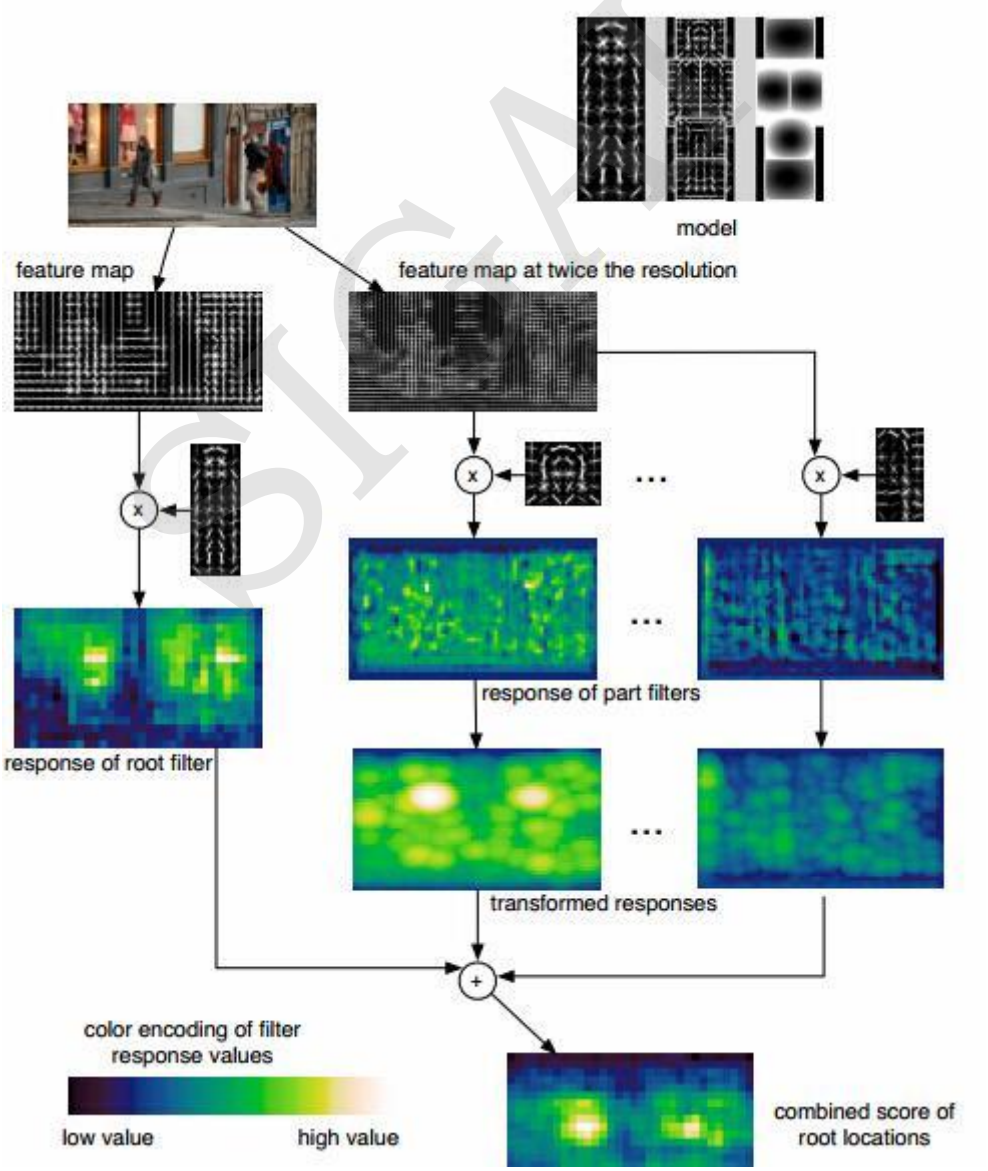

肿瘤检测

# 目标检测相关算法

## DPM 算法

与人脸、行人等特定类型的目标检测不同，通用目标检测要同时检测出图像中的多类目标，难度更大。处理这一问题的经典方法是 DPM（Deformable Part Model），正如其名，这是可变形的组件模型，是一种基于组件的检测算法。该模型由 Felzenszwalb 在 2008 年提出，并发表了一系列的 CVPR，NIPS 文章，蝉联三届 PASCAL VOC 目标检测冠军，拿下了 2010 年 PASCAL VOC 的“终身成就奖”。

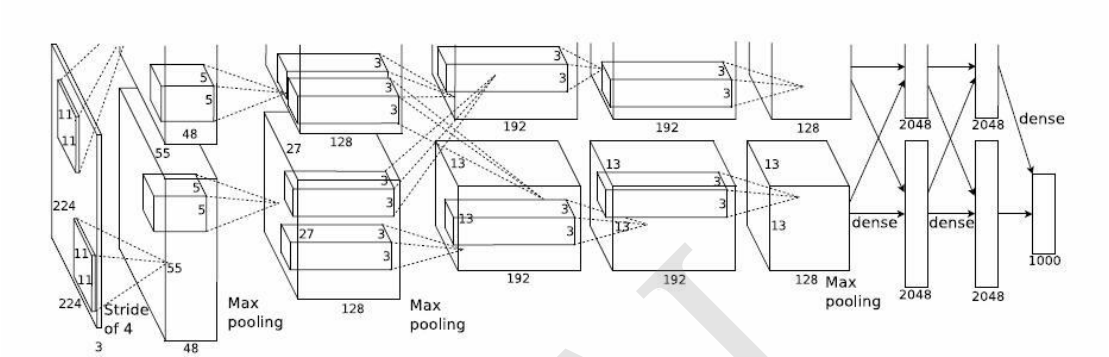
在深度卷积神经网络（DCNN）出现之前，DPM 算法一直是目标检测领域最优秀的算法，它的基本思想是先提取 DPM 人工特征（如下图所示），再用 latentSVM 分类。这种特征提取方式存在明显的局限性，首先，DPM 特征计算复杂，计算速度慢；其次，人工特征对于旋转、拉伸、视角变化的物体检测效果差。这些弊端很大程度上限制了算法的应用场景。




DPM 目标检测流程

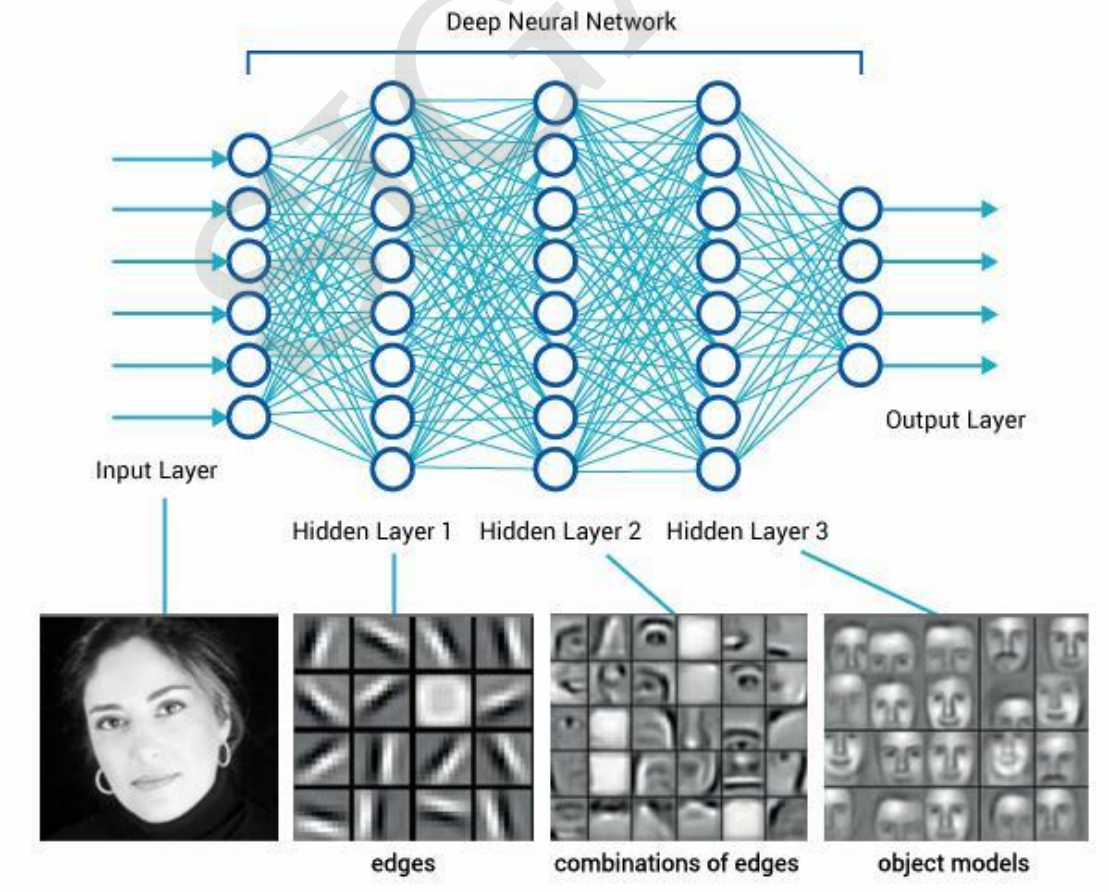
Alexnet

现代深度神经网络的想法早在 2006 年就被 Geoffrey Hinton 提出，直到 2012 年，Alex Krizhevsky 凭借著名的 Alexnet 卷积神经网络模型以领先第二名 10%的成绩夺得 ILSVRC2012 图像分类比赛冠军，深度学习技术才真正走进主流学术界和工业界的视野。深度神经网络的出现颠覆了传统的特征提取方式，凭借其强大的表达能力，通过丰富的训练数据和充分的训练能够自主学习有用的特征。这相比传统的人工发现特征并根据特征设计算法的方式是质的飞跃。




AlexNet 网络结构

通过卷积神经网络可以学到物体在各个层次的抽象表达（关于卷积神经网络的原理以及为什么会有效，SIGAI 会在接下来的文章中介绍）：




深度学习得到的层次特征表达

OverFeat

2013 年纽约大学 Yann LeCun 团队中 Zhang xiang 等提出的 OverFeat 在 ILSVRC2013 比赛中获得了多项第一，他们改进了 Alexnet，提出了使用同一个卷积网络完成了多个任务的方法。该方法充分利用了卷积神经网络的特征提取功能，它把分类过程中提取到的特征同时又用于定位检测等各种任务，只需要改变网络的最后几层，就可以实现不同的任务，而不需要从头开始训练整个网络的参数。这充分体现和发掘了 CNN 特征共享的优点。

该文主要的亮点是：

1.共享卷基层用于多任务学习。

2.全卷积网络思想。

3.在特征层进行滑窗操作（sliding window）避免大量重复运算，这也是后来的系列算法不断沿用和改进的经典做法。

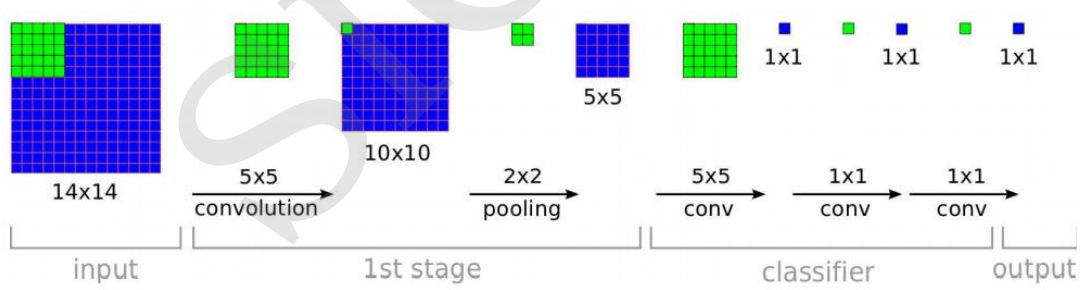
OverFeat 几个明显的缺陷：

1.采用了多尺度贪婪的划窗策略，导致计算量还是很大 。

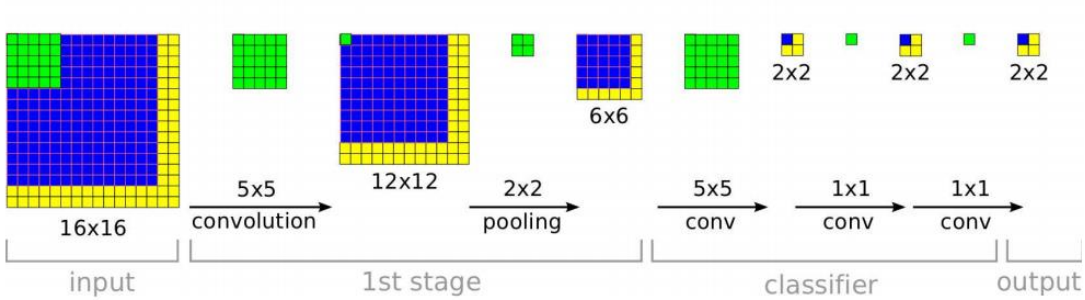
2.由于当时并没有太优秀的 backbone 网络，共享特征层的表征能力不是太强，没有考虑多尺度特征融合，对小目标效果差，整体的检测效果不尽如人意。ILSVRC 2013 数据集上的 mAP（可以简单的理解为检测准确率）为 24.3%。

经典的卷积神经网络有一个问题是它只能接受固定大小的输入图像，这是因为第一个权全连接层和它之前的卷积层之间的权重矩阵大小是固定的，而卷积层、全连接层本身对输入图像的大小并没有限制。而在做目标检测时，卷积网络面临的输入候选区域图像大小尺寸是不固定的。

解决这个问题的方法有多种。下面用一个例子说明怎么让一个已经设计好的 DCNN 模型，可以支持任意大小图片输入，第一种方法是全卷积网络（FCN），即去掉所有全连接层，全部由卷积层来替代：



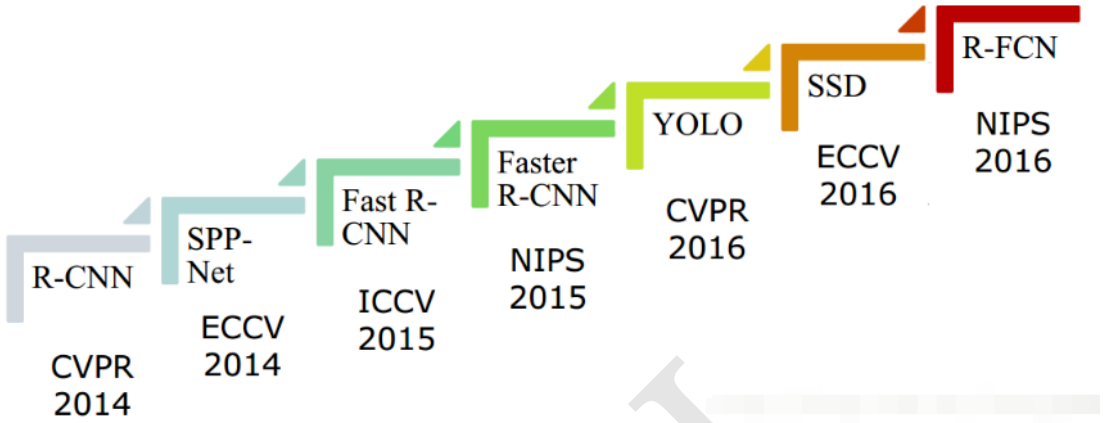

FCN 并不是把 5×5 的图片展平成一维向量再进行计算，而是直接采用 5×5 的卷积核对一整张图片进行卷积运算。比如 16×16 大小的特征图片，那么会是什么样的结果？请看下面的示意图：




这个时候就会发现，网络最后的输出是一张 2×2 大小的特征图片。可以发现采用 FCN 网络，可以输入任意大小的图片。需要注意的是网络最后输出的特征图片大小不再总是 1×1 而是一个与输入图片大小相关。

OverFeat 有很多创新，但是不能算是目标检测典型的 Pipeline，所以我们单独提了出来。下面将从 R-CNN 开始介绍目前基于 DCNN 物体检测发展脉络。

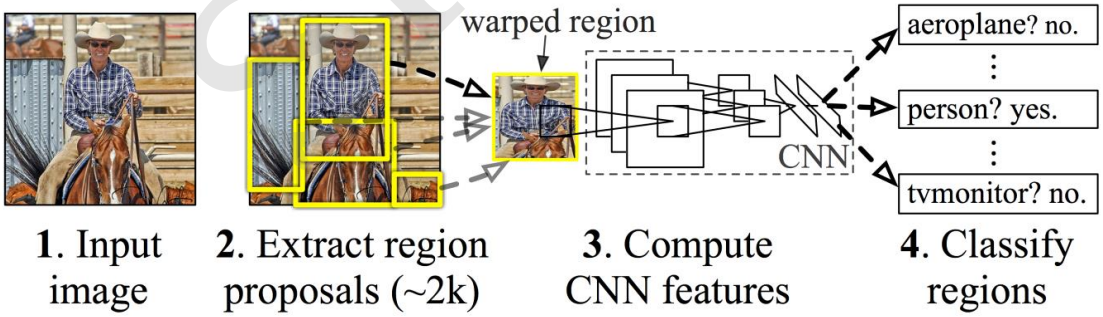
卷积神经网络用于目标检测之后，进展神速，在短期内大幅度的提高了算法的精度，推动这一技术走向实用。




基于 DCNN 的目标检测算法发展路线图

R-CNN

Region CNN(简称 R-CNN)由 Ross Girshick（江湖人称 RBG 大神，Felzenszwalb 的学生）提出，是利用深度学习进行目标检测的里程碑之作，奠定了这个子领域的基础。这篇文章思路清奇，在 DPM 方法经历多年瓶颈期后，显著提升了检测率（ILSVRC 2013 数据集上的 mAP 为 31.4%）。RBG 是这个领域神一样的存在，后续的一些改进方法如 Fast R-CNN、Faster R-CNN、YOLO 等相关工作都和他有关。




R-CNN 检测框架

R-CNN 检测时的主要步骤为：

1.使用 Selective Search 算法从待检测图像中提取 2000 个左右的区域候选框，这些候选框可能包含要检测的目标。

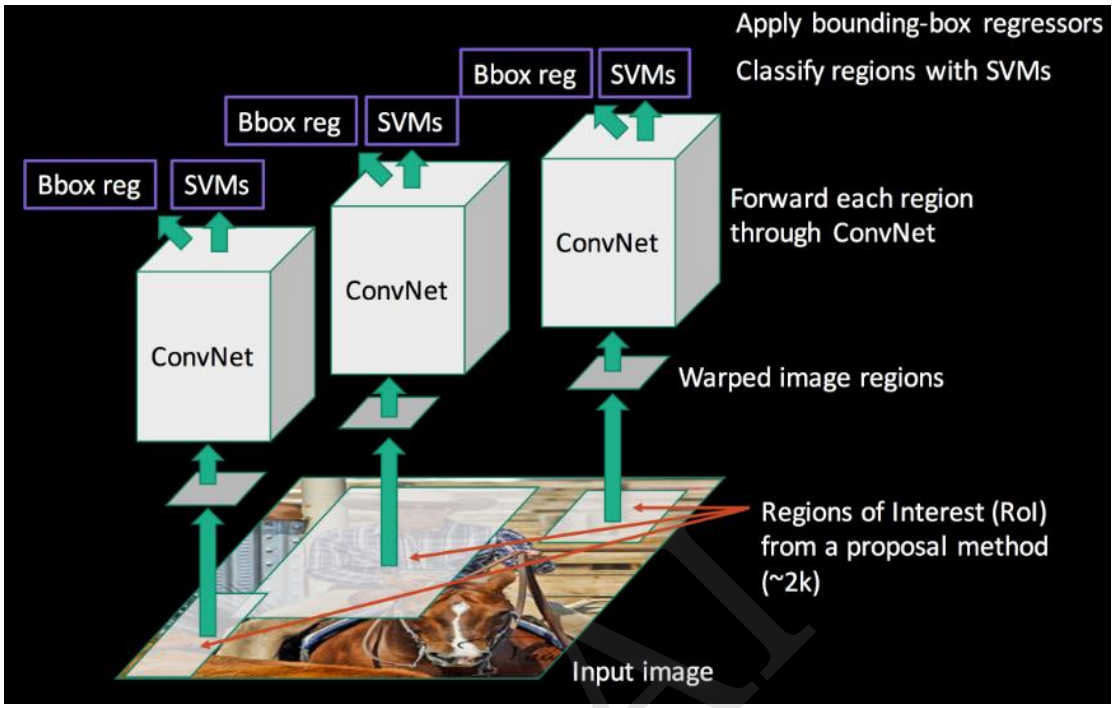
2.把所有侯选框缩放成固定大小（原文采用 227×227）。

3.用 DCNN 提取每个候选框的特征，得到固定长度的特征向量。

4.把特征向量送入 SVM 进行分类得到类别信息，送入全连接网络进行回归得到对应位置坐标信息。

R-CNN 不采用滑动窗口方案的原因一是计算成本高，会产生大量的待分类窗口；另外不同类型目标的矩形框有不同的宽高比，无法使用统一尺寸的窗口对图像进行扫描。用于提取特征的卷积网络有 5 个卷积层和 2 个全连接层，其输入是固定大小的 RGB 图像，输出为 4096 维特征向量。对候选区域的分类采用线性支持向量机，对每一张待检测图像计算所有

候选区域的特征向量，送入支持向量机中进行分类；同时送入全连接网络进行坐标位置回归。




R-CNN 虽然设计巧妙，但仍存在很多缺点：

1.重复计算。R-CNN 虽然不再是穷举，但通过 Proposal（Selective Search）的方案依然有两千个左右的候选框，这些候选框都需要单独经过 backbone 网络提取特征，计算量依然很大，候选框之间会有重叠，因此有不少其实是重复计算。

2.训练测试不简洁。候选区域提取、特征提取、分类、回归都是分开操作，中间数据还需要单独保存。

3.速度慢。前面的缺点最终导致 R-CNN 出奇的慢，GPU 上处理一张图片需要十几秒，CPU 上则需要更长时间。

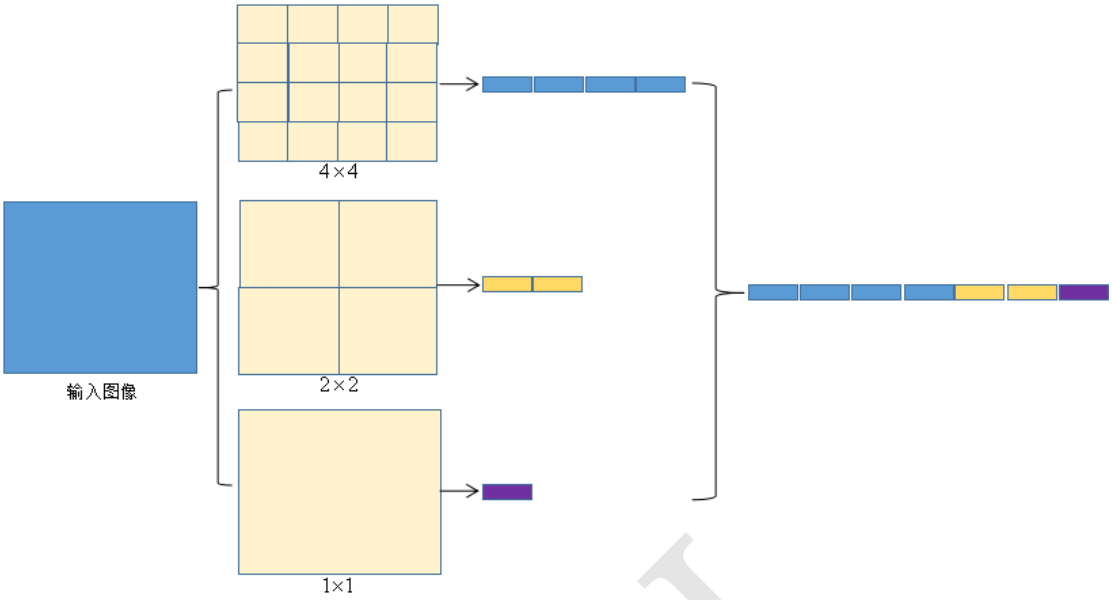
4.输入的图片 Patch 必须强制缩放成固定大小（原文采用 227×227），会造成物体形变，导致检测性能下降。

## SPPNet

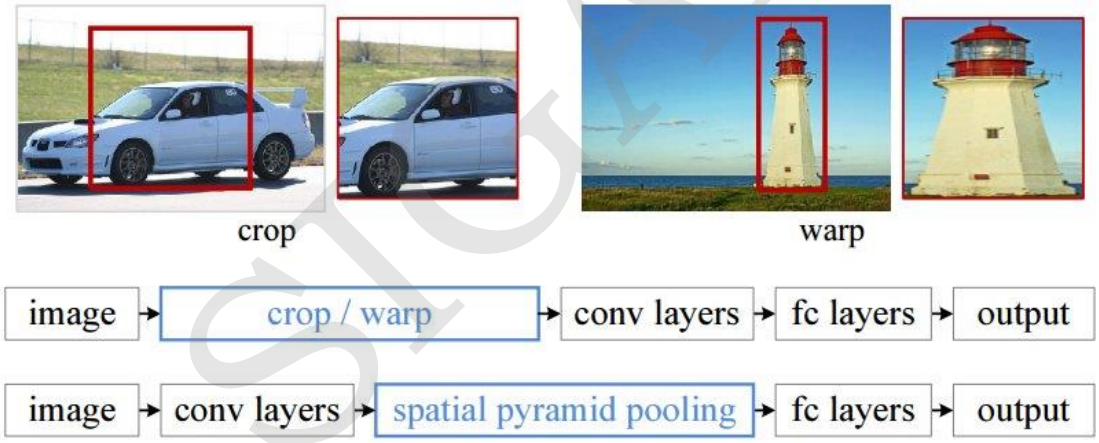
此后 MSRA 的 Kaiming He 等人在 R-CNN 的基础上提出了 SPPNet，该方法虽然还依赖候选框的生成，但将提取候选框特征向量的操作转移到卷积后的特征图上进行，将 R-CNN 中的多次卷积变为一次卷积，大大降低了计算量（这一点参考了 OverFeat）。

R-CNN 的卷积网络只能接受固定大小的输入图像。为了适应这个图像尺寸，要么截取这个尺寸的图像区域，这将导致图像未覆盖整个目标；要么对图像进行缩放，这会产生扭曲。在卷积神经网络中，卷积层并不要求输入图像的尺寸固定，只有第一个全连接层需要固定尺寸的输入，因为它和前一层之间的权重矩阵是固定大小的，其他的全连接层也不要求图像的尺寸固定。如果在最后一个卷积层和第一个全连接层之间做一些处理，将不同大小的图像变为固定大小的全连接层输入就可以解决问题。

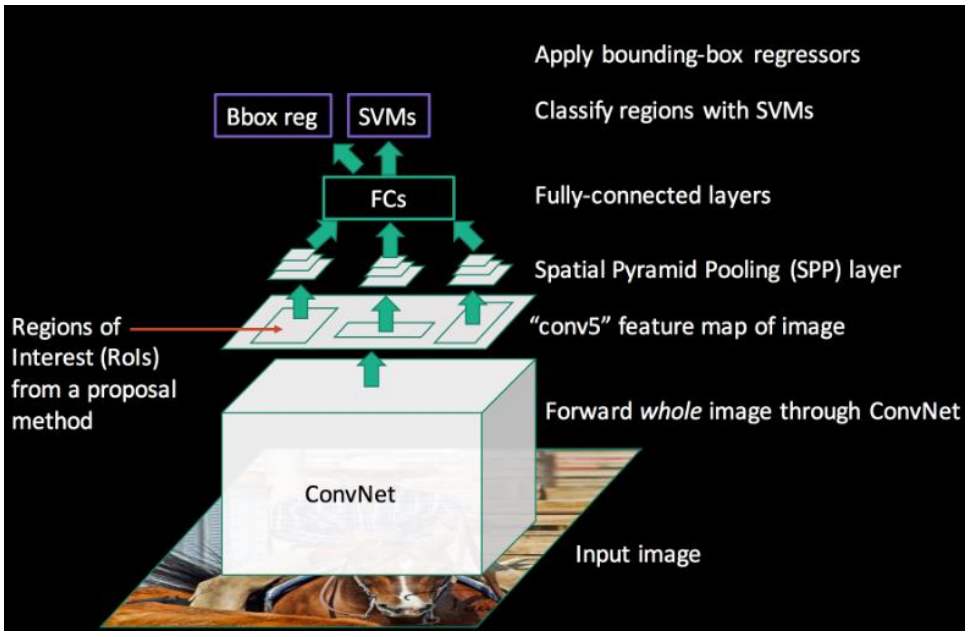
SPPNet 引入了 Spatial Pyramid pooling 层，对卷积特征图像进行空间金字塔采样获得固定长度的输出，可对特征层任意长宽比和尺度区域进行特征提取。具体做法是对特征图像区域进行固定数量的网格划分，对不同宽高的图像，每个网格的高度和宽度是不规定的，对划分的每个网格进行池化，这样就可以得到固定长度的输出。下图是 SPP 操作示意图：




相比 R-CNN，SPPNet 的检测速度提升了 24 ~ 102 倍。下图是 R-CNN 和 SPPNet 检测流程的比较：




下图是 SPPNet 的原理：

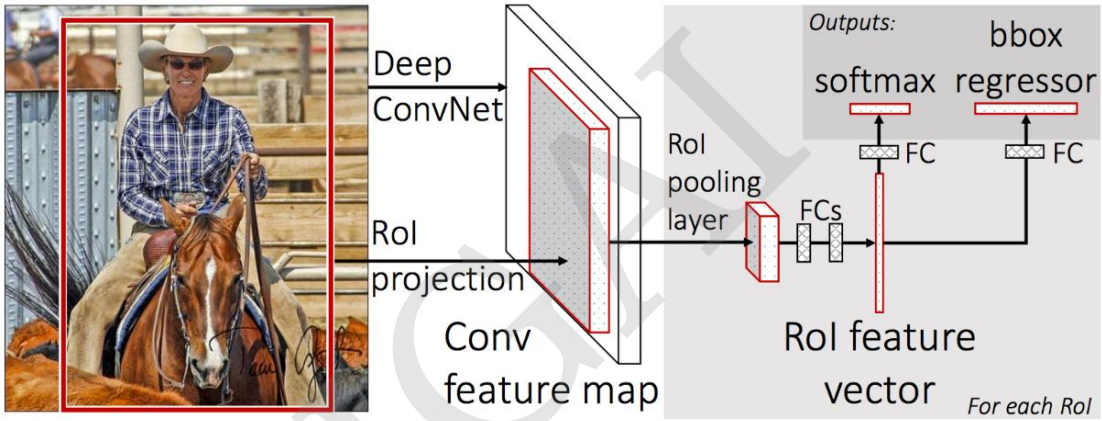


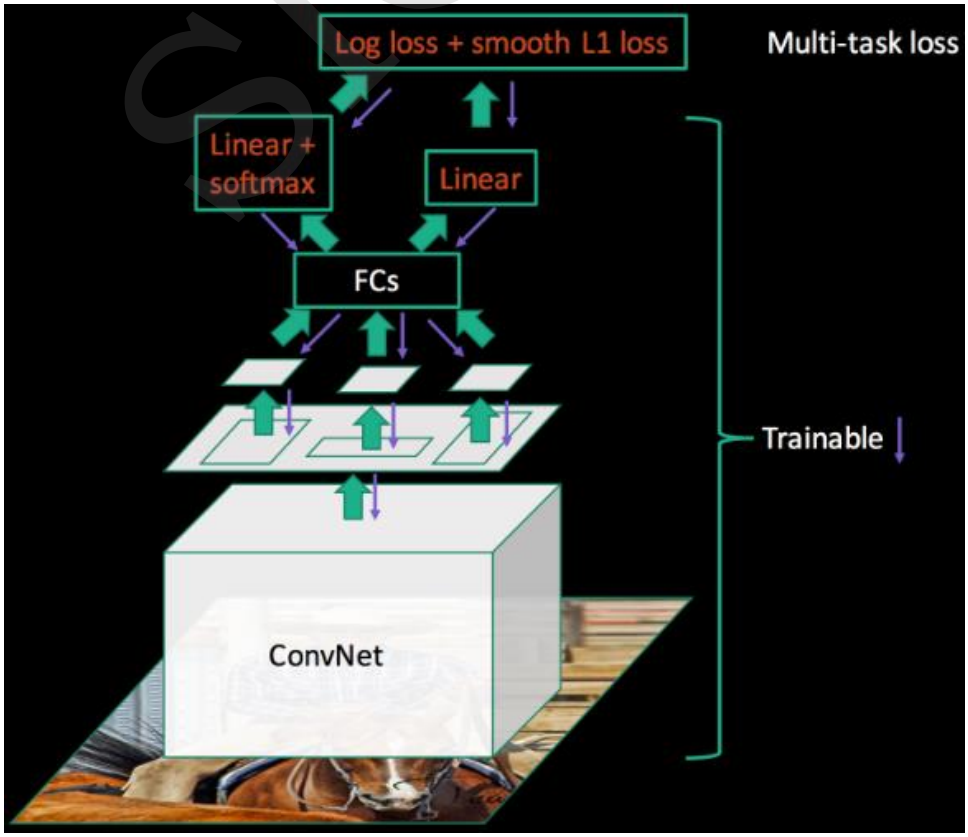
SPPNet 检测框架图

SPPNet 和 R-CNN 一样，它的训练要经过多个阶段，中间特征也要进行存储；backbone 网络参数沿用了分类网络的初始参数，没有针对检测问题进行优化。

Fast RCNN

Ross Girshick 针对 SPPNet 做了进一步改进提出的 FRCNN ，其主要创新是 RoI Pooling 层，它将不同大小候选框的卷积特征图统一采样成固定大小的特征。ROI 池化层的做法和 SPP 层类似，但只使用一个尺度进行网格划分和池化。该层可以直接求导，训练时直接将梯度传导到 backbone 网络进行优化。FRCNN 针对 R-CNN 和 SPPNet 在训练时是多阶段的和训练的过程中很耗费时间空间的问题进行改进。将深度网络和后面的 SVM 分类两个阶段整合到一起，使用一个新的网络直接做分类和回归。使得网络在 Pascal VOC 上的训练时间从 R-CNN 的 84 小时缩短到 9.5 小时，检测时间更是从 45 秒缩短到 0.32 秒。



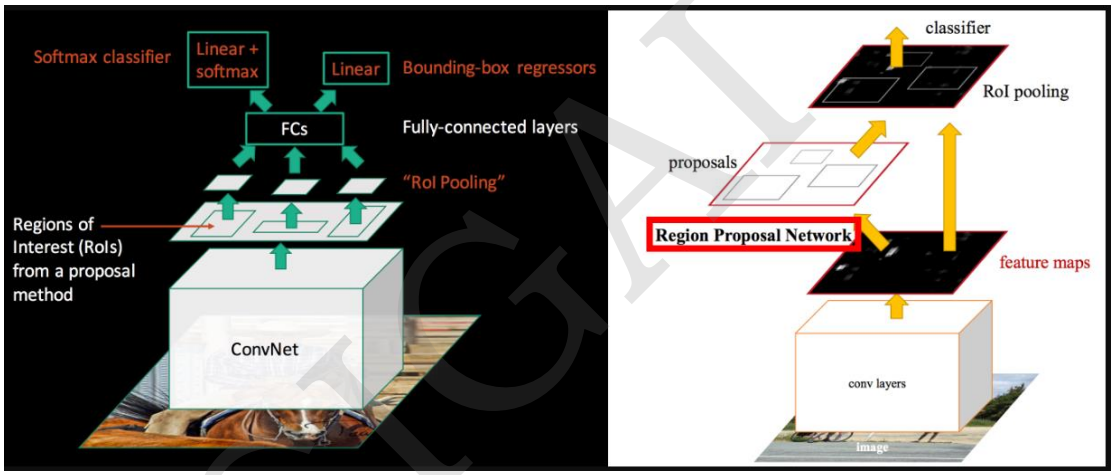




重要的是 Fast RCNN 的 backbone 网络也可以参与训练了！！！

Faster RCNN

SPPNet 和 Faster RCNN 都需要独立的候选区域生成模块，这个模块计算量很大，而且不易用 GPU 加速。针对这个问题，Shaoqin Ren 等人在 Faster RCNN 基础上提出 Faster R-CNN ，在主干网络中增加了 RPN （Region Proposal Network）网络，通过一定规则设置不同尺度的锚点（Anchor）在 RPN 的卷积特征层提取候选框来代替 Selective Search 等传统的候选框生成方法，实现了网络的端到端训练。候选区域生成、候选区域特征提取、框回归和分类全过程一气呵成，在训练过程中模型各部分不仅学习如何完成自己的任务，还自主学习如何相互配合。这也是第一个真正意义上的深度学习目标检测算法。

注：Shaoqin Ren 实现的 matlab 版本中 RPN 阶段和 FRCNN 阶段是分开训练的，但是在实际的实践中（RBG 实现的 Python 版本）发现二者可以一起优化训练，而且精度没有损失，可以说 Faster RCNN 真正实现了端到端的训练。




Fast RCNN（左） 和 Faster RCNN（右）框架结构对比

R-FCN

由于现在的主流网络层数越来越多，基于 Faster RCNN 检测框架的方法的计算量受到了 3 个因素的影响：

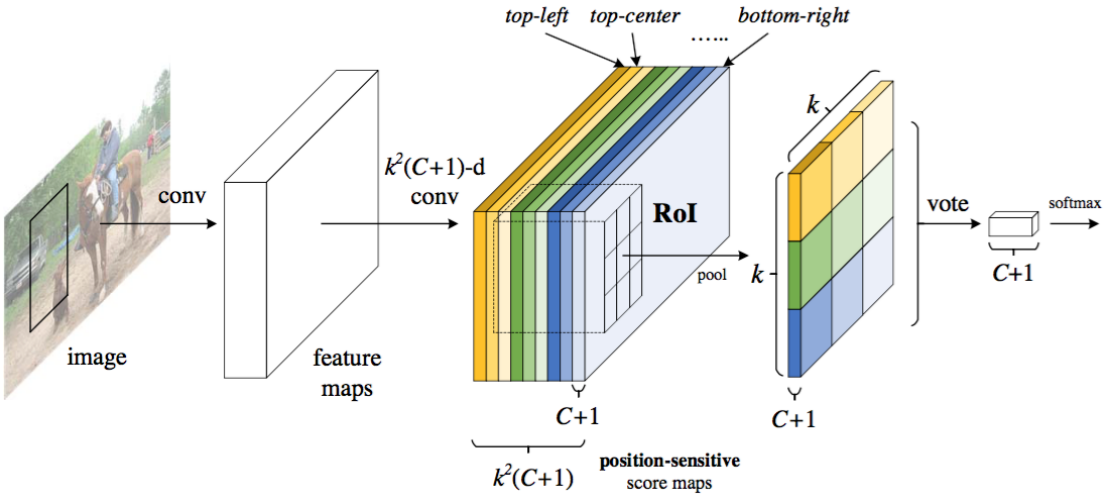
1.基础网络的复杂度

2.候选框数量的多少

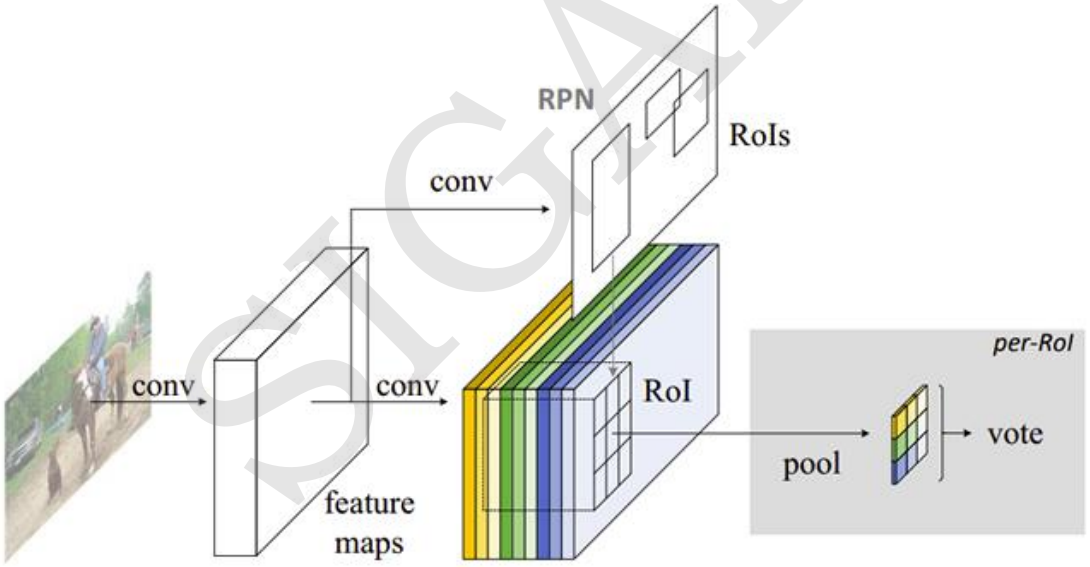
3.分类和位置回归子网络的复杂度（每个候选框的 box 都会独立进行前向计算）。

一般来说直接优化前两点性价比不太高。如果直接优化 RoI-wise subnetwork 是否可行呢,将子网络的深度尽可能减少？分类是要增加物体的平移不变性（不同的位置都是同一个物体）；目标检测时减少物体的平移变化（目标检测需要得到物体所在的位置）。通常我们所用的网络都是 ImageNet 的分类任务训练得到的，在目标检测的时候进行 Finetune。由于得到的初始模型基于分类任务，那么会偏向于平移不变性，这和目标检测就出现了矛盾。

MSRA 的 Jifeng Dai 等人提出了 R-FCN，通过 position-positive score maps（位置敏感得分图）来解决这个矛盾。位置敏感得分图通过预测 RoI 中不同部位的类别投票表决产生该 RoI 的类别预测。引用原文中的例子，“如果我们的算法要识别婴儿，那么把一个目标区域分成九宫格，其中算法认为其中五个格子中的区域分别像婴儿的头、四肢和躯干，那么根据投票机制，就认为这个目标区域里的是一个婴儿。这很符合我们人类的判断逻辑。”




R-FCN 沿用了 Faster RCNN 的框架结构，不同的是在 Faster R-CNN 的基础上通过引入位置敏感得分图，将 RoI-wise subnetwork 消灭了，直接在位置敏感得分图上利用 ROI Pooling 进行信息采样融合分类和位置信息。



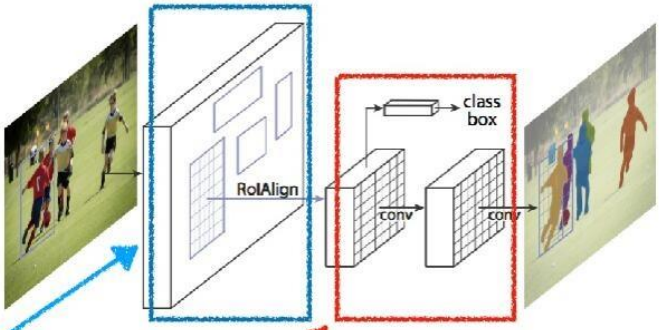

R-FCN 网络框架结构

| | R-CNN [7] | Faster R-CNN [19, 9] | R-FCN [ours] |
|---|---|---|---|
| depth of shared convolutional subnetwork | 0 | 91 | 101 |
| depth of RoI-wise subnetwork | 101 | 10 | **0** |

ResNet101 为例，不同检测框架复用卷积网络层数

Mask R-CNN

2017 年 Kaiming He 等提出了 Mask R-CNN ，并获得 ICCV2017 Best Paper Award。作者指出，Faster R-CNN 在做下采样和 RoI Pooling 时都对特征图大小做了取整操作，这种做法对于分类任务基本没有影响，但对检测任务会有一定影响，对语义分割这种像素级任务的精度影响则更为严重。为此，作者对网络中涉及特征图尺寸变化的环节都不使用取整操作，而是通过双线性差值填补非整数位置的像素。这使得下游特征图向上游映射时没有位置误差，不仅提升了目标检测效果，还使得算法能满足语义分割任务的精度要求。




The **Mask R-CNN** framework for instance segmentation.

1. To fix the misalignment, we propose a simple, quantization-free layer, called RoIAlign, that faithfully preserves exact spatial locations.

2. Adding a branch for predicting segmentation masks on each Region of Interest (RoI), in parallel with the existing branch for classification and bounding box regression.

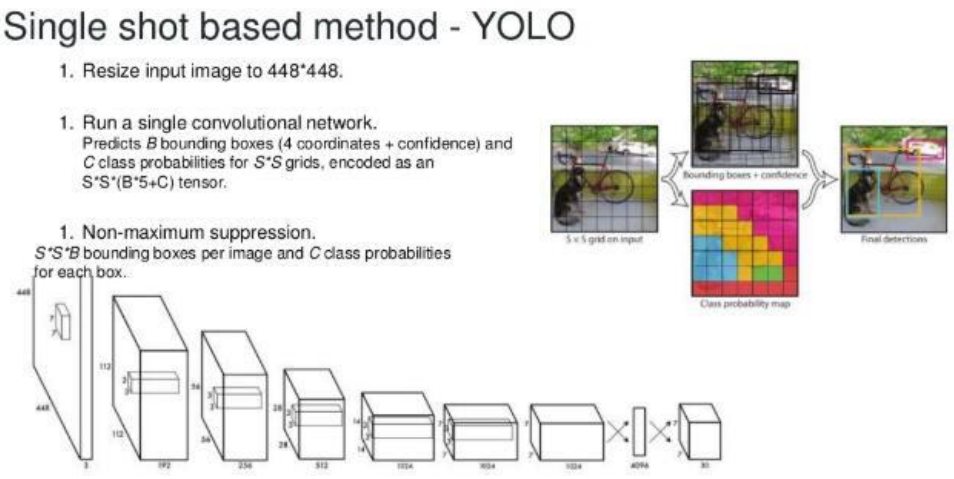
以上介绍的检测方法都属于 two-stage 的方案，即分为候选区域生成和区域分类两步，接下来我们将介绍几种 single-stage 的经典方法。

YOLO 系列

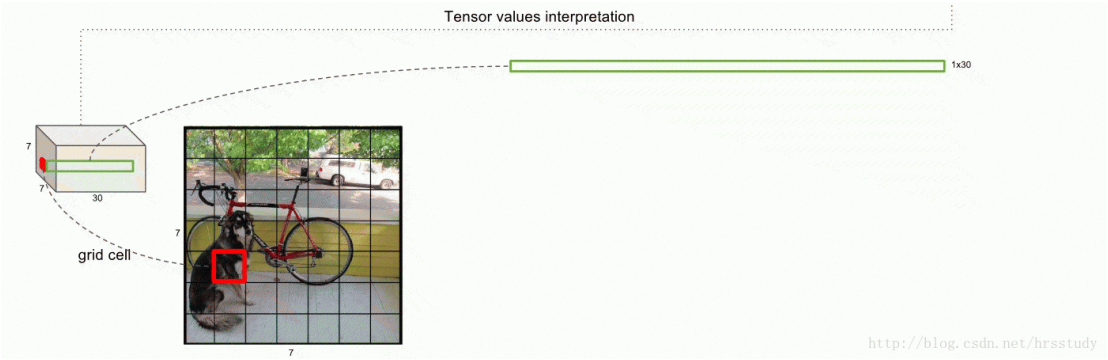
2015 年，随着 YOLO 算法的出现，深度学习目标检测算法开始有了两步（two-stage）和单步（single-stage）之分。区别于 R-CNN 系列为代表的两步检测算法，YOLO 舍去了候选框提取分支（Proposal 阶段），直接将特征提取、候选框回归和分类在同一个无分支的卷积网络中完成，使得网络结构变得简单，检测速度较 Faster R-CNN 也有近 10 倍的提升。这使得深度学习目标检测算法在当时的计算能力下开始能够满足实时检测任务的需求。

算法将待检测图像缩放到统一尺寸，为了检测不同位置的目标，将图像等分成 $S \times S$ 的网格，如果某个目标的中心落在一个网格单元中，此网格单元就负责预测该目标。

YOLOv1 只针对最后 7x7 的特征图进行分析，使得它对小目标的检测效果不佳，当多个目标出现在一个 Grid Cell 时不容易区分。




YOLOv1 原理图




YOLOv1 在 7X7 特征图上对每个 Grid cell 进行操作

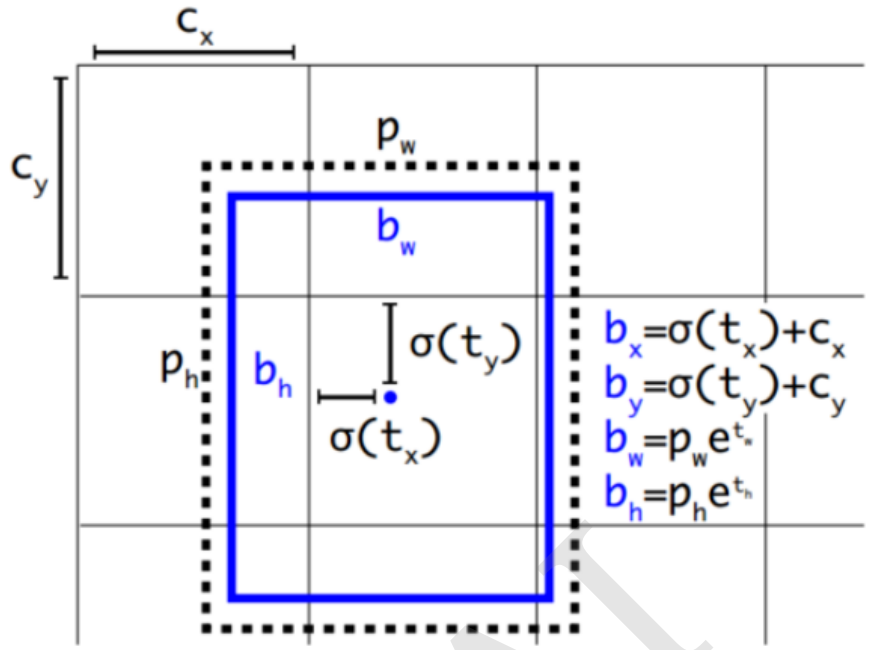
YOLOv2 改进了 YOLOv1 的网络结构，除加入时下热门的批量归一化层以及残差网络结构以外，还针对性的训练了一个高分辨率的分类网络（448x448）然后利用该网络训练检测网络，单单通过提升输入的分辨率，mAP 获得了 4%的提升。YOLOv1 利用单个 grid cell 拼接成全连接层完成边框的预测，导致丢失较多的空间信息，定位不准,作者在这一版本中进行了优化改进：

1.借鉴了 Faster R-CNN 中的 Anchor 思想，但是作者在实践中发现用基于规则选择的 Anchor 效果并没有得到提升，实验中作者对 Pascal VOC 和 COCO 的数据集进行了统计分析（聚类分析）选择针对行的 Anchor 的尺寸性能的到了明显提升。

| Box Generation | # | Avg IOU |
|---|---|---|
| Cluster SSE | 5 | 58.7 |
| Cluster IOU | 5 | 61.0 |
| Anchor Boxes [15] | 9 | 60.9 |
| Cluster IOU | 9 | 67.2 |

**Table 1: Average IOU of boxes to closest priors on VOC 2007.** The average IOU of objects on VOC 2007 to their closest, unmodified prior using different generation methods. Clustering gives much better results than using hand-picked priors.

2.作者在使用 anchor boxes 时发现模型收敛不稳定，尤其是在早期迭代的时候。大部分的不稳定现象出现在预测 box 的 (x,y) 坐标的优化上。因此作者就没有采用直接预测 offset 的方法，而使用了预测相对于 grid cell 的坐标位置的办法，利用 logistic 函数把 ground truth 归一化到 0 到 1 之间，坐标值被归一化后，模型优化会更稳定。

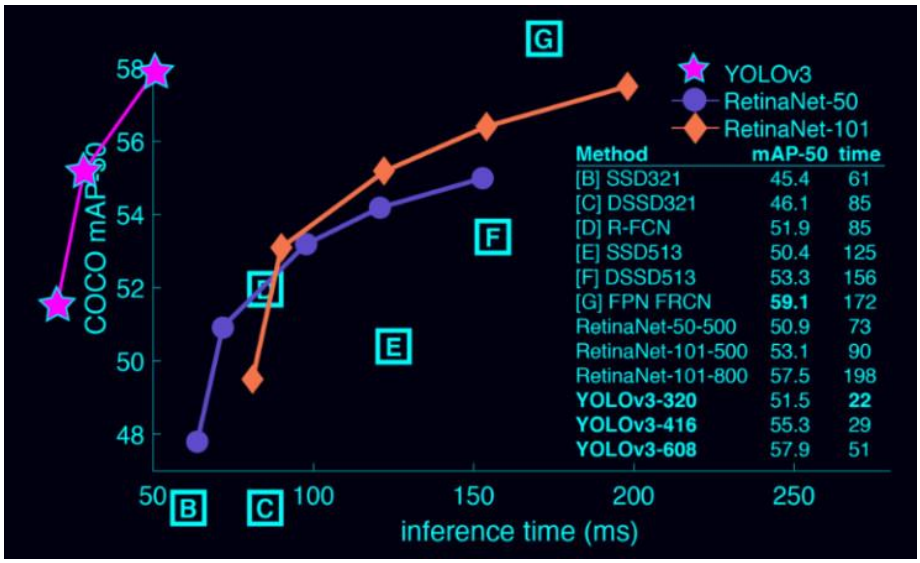



YOLOv3 在 YOLOv2 的基础上使用了全新设计的 Darknet53 残差网络并结合 FPN 网络结构，在网络后两个特征图上采样后于网络前期相应尺寸的特征图聚合再经过卷积网络后得到预测结果。这些改进使得 YOLOv3 用三分之一的时间达到与 SSD 相当的精确度。在 COCO test-dev 上 mAP@0.5 达到 57.9%，与 RetinaNet（FocalLoss 论文所提出的单阶段网络）的结果相近，但速度快 4 倍.

YOLOv3 的模型比之前的版本复杂了不少，可以通过改变模型结构的大小来权衡速度与精度。

YOLOv3 的改进点：

1. 多尺度预测（FPN）
2. 更好的 Backbone 网络（Darknet53 残差网络）
3. 分类损失采用 binary cross-entropy 损失函数替换 Softmax 损失函数（Softmax 会选择分数最高的类别判定为当前框所属的类别，而现实中一个目标可能属于多个类别标签）

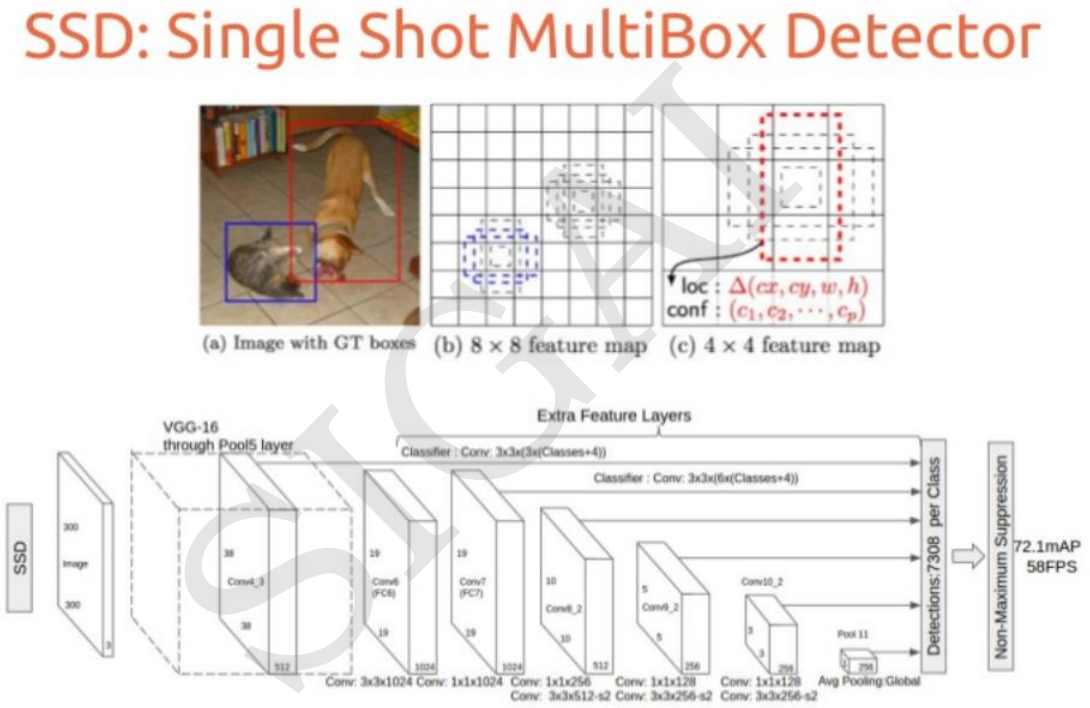


SSD

SSD 对 YOLO 进行了改进，达到了和两阶段方法相当的精度，同时又保持了较快的运行速度。SSD 也采用了网格划分的思想，和 Faster RCNN 不同的是它将所有的操作整合在一个卷积网络中完成。为了检测不同尺度的目标，SSD 对不同卷积层的特征图像进行滑窗扫描；在前面的卷积层输出的特征图像中检测小的目标，在后面的卷积层输出的特征图像中检测大的目标。它的主要特点是：

1.基于多尺度特征图像的检测：在多个尺度的卷积特征图上进行预测，以检测不同大小的目标，一定程度上提升了小目标物体的检测精度。

2.借鉴了 Faster R-CNN 中的 Anchor boxes 思想，在不同尺度的特征图上采样候选区域，一定程度上提升了检测的召回率以及小目标的检测效果。下图是 SSD 的原理：




FPN

FPN（Feature Pyramid Network）方法同时利用低层特征高分辨率和高层特征的高语义信息，通过融合这些不同层的特征达到提升预测的效果的作用。FPN 中预测是在每个融合后的特征层上单独进行的，这和常规的特征融合方式有所不同。

FPN 网络结构如下图 d（其中 YOLO 使用 b 结构，SSD 使用 c 结构）所示，它的结构具有相当的灵活性，可以和各种特征提取网络结合作为检测算法的基础网络。在后文中会看到，目前大多数 state-of-art 的模型都采用了这种结构。其中 RetinaNet 在 FPN 的基础上使用了 ResNet 网络提取特征，并用 Focal Loss 损失改善单步目标检测算法中普遍存在的前景类和背景类损失不均衡的问题。这些基于 FPN 结构的检测算法能够在增加网络深度、获取更丰富语义信息的同时从浅层特征图中获取更丰富且高分辨率的图像特征，这使得这种网络结构在实际应用中表现出优异的性能。

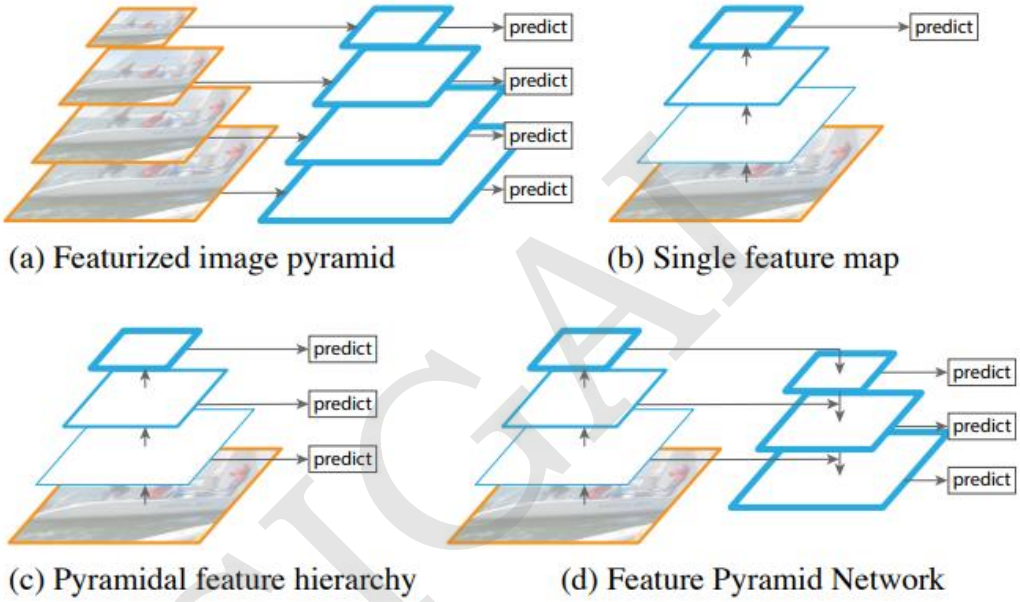
目前主流检测框架有 4 种使用特征的形式：

a.图像金字塔。即将图像缩放到不同的大小，然后不同大小的图像生成对应的特征。这种方法的缺点是增加了时间成本。有些算法会在检测时采用这种图像金字塔的方案。

b.单一尺度特征层。SPPNet，Fast RCNN，Faster RCNN 采用这种方式，即仅采用网络最后一层卷积层的特征。

c.SSD 采用这种多尺度特征融合的方式，但是没有上采样过程，即从网络不同层抽取不同尺度的特征做预测，这种方式不会增加额外的计算量。SSD 算法中没有用到足够低层的特征（在 SSD 中，最低层的特征是 VGG 网络的 conv4_3），而足够低层的特征对于检测小物体是很有帮助的。

d.FPN 采用 bottom-up 与 top-down 的结构，实现了低层特征和高层语义特征的融合，提高了**特征映射的信息密度和分辨率**，提高了小目标物体的检测效果；区别于 SSD，FPN 每层都是独立预测的。






COCO2017 排行榜

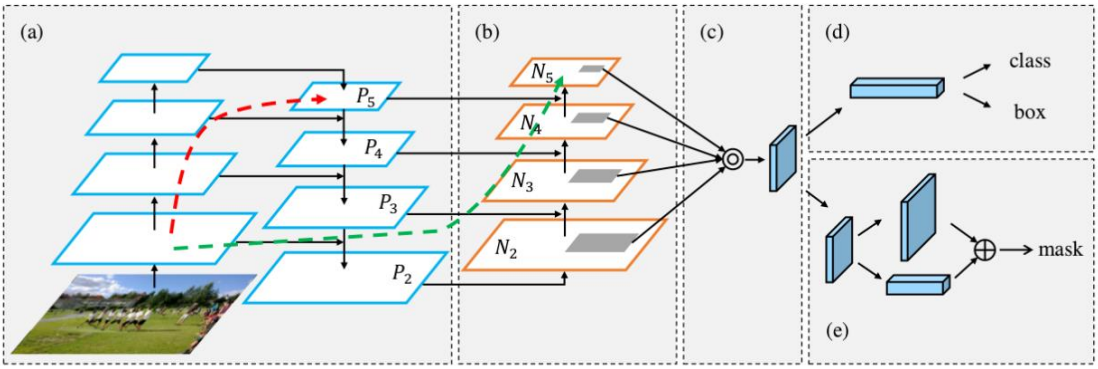
最后我们来看通用目标检测算法的最新进展。下图是 MSCOCO 2017 年目标检测竞赛的领先算法：

| | AP | $AP^{50}$ | $AP^{75}$ | $AP^{S}$ | $AP^{M}$ | $AP^{L}$ | $AR^{1}$ | $AR^{10}$ | $AR^{100}$ | $AR^{S}$ | $AR^{M}$ | $AR^{L}$ | date |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Megvii (Face++) | 0.526 | 0.730 | 0.585 | 0.343 | 0.556 | 0.660 | 0.391 | 0.645 | 0.689 | 0.513 | 0.727 | 0.827 | 2017-10-05 |
| UCenter | 0.510 | 0.705 | 0.558 | 0.326 | 0.539 | 0.648 | 0.392 | 0.640 | 0.678 | 0.497 | 0.720 | 0.829 | 2017-10-05 |
| MSRA | 0.507 | 0.717 | 0.566 | 0.343 | 0.529 | 0.627 | 0.379 | 0.638 | 0.690 | 0.524 | 0.720 | 0.824 | 2017-10-05 |
| FAIR Mask R-CNN | 0.503 | 0.720 | 0.558 | 0.328 | 0.537 | 0.627 | 0.380 | 0.622 | 0.659 | 0.485 | 0.704 | 0.800 | 2017-10-05 |
| Trimps-Soushen+QINIU | 0.482 | 0.681 | 0.534 | 0.310 | 0.512 | 0.610 | 0.373 | 0.611 | 0.652 | 0.466 | 0.688 | 0.801 | 2017-10-05 |
| bharat_umd | 0.482 | 0.694 | 0.536 | 0.312 | 0.514 | 0.606 | 0.365 | 0.605 | 0.647 | 0.456 | 0.696 | 0.793 | 2017-10-05 |
| DANet | 0.459 | 0.676 | 0.509 | 0.283 | 0.483 | 0.591 | 0.358 | 0.587 | 0.625 | 0.427 | 0.664 | 0.783 | 2017-10-05 |

其中排名第一的模型为旷视科技（face++）提交的 MegDet。他们的方案没有在检测算法方面做过多优化（采用的是 ResNet50+FPN），而是在并行训练规模上做了优化。训练硬件环境是由 128 个 GPU 组成的集群，通过改进跨 GPU 批量归一化算法和学习率变化策略，将 batch size 增大到 256 张，这使得批量归一化层中使用的批均值和方差更能够反应总体特

征，有效提升了归一化效果，从而大幅提升训练速度并且得到了非常好的结果。

排名第二的方案 PAN 改进了 FPN 算法，如下图所示。它在 FPN 的基础上不仅增加了一个降采样网络（b），还聚合使用了多个不同尺度特征图上的预测候选框（c）。该模型不仅在这一届的 COCO 目标检测竞赛中名列第二，而且取得了语义分割任务的冠军。




第三名的模型出自 MSRA 之手，他们同样没有对检测算法本身做过多改进，在 FPN 基础上使用了 Xception 网络结构和 SoftNMS，但与以往不同的是使用了可变卷积层 DCN（deformable convnet）替代了传统卷积，使得卷积层能够根据图片的语义信息调整卷积核感受点的位置，达到提升网络特征提取能力的目的。下图是可变卷积层的原理：

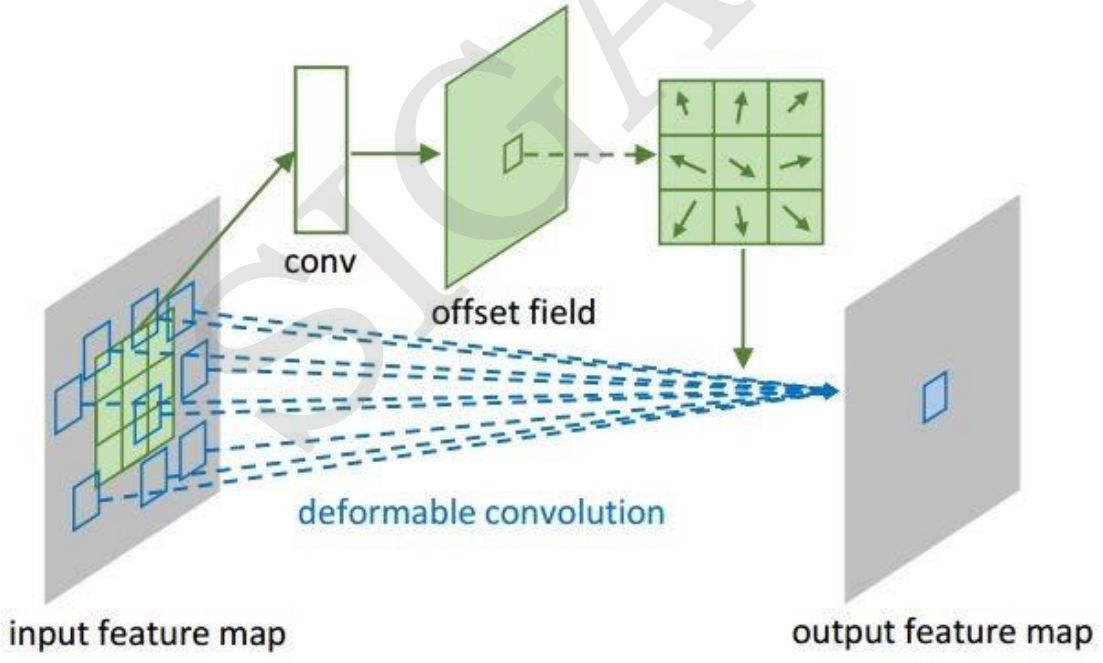



Illustration of $3 \times 3$ deformable convolution

排名第四的结果是用以 FPN+ResNeXt 网络为作为基础网络的 Mask R-CNN 算法得到的。后面大多数成绩优异的模型都是 R-FCN、FPN、Faster-RCNN 等经典模型结合 DCN、Attention 机制、先进分类网络结构和模型融合等技术而形成的算法。

## 推荐文章

[1] 机器学习-波澜壮阔 40 年 SIGAI 2018.4.13.

[2] 学好机器学习需要哪些数学知识？ SIGAI 2018.4.17.

[3] 人脸识别算法演化史 SIGAI 2018.4.20.

## 原创声明