

**人工智能与机器学习课程报告**

**2020年春季学期**

**考 核 科 目： 人工智能与机器学习**

**学生所在院系： 汽车工程师学院**

**授 课 教 师： 龚家元 陈安庆**

**报 告 题 目： 目标检测算法**

**姓 名： 王 欣**

**学 号： 201702052**

目录

[**基于深度学习的目标检测算法概述 - 2 -**](#_Toc32482)

[**1 深度学习目标检测算法 - 3 -**](#_Toc657)

[**2 基于分类的检测算法 - 4 -**](#_Toc15634)

[2.1 OverFeat - 4 -](#_Toc6544)

[2.2 Ｒ-CNN - 4 -](#_Toc2561)

[2.3 SPP-Net - 5 -](#_Toc26910)

[2.4 FastR-CNN - 6 -](#_Toc13287)

[2.5 FasterR-CNN - 7 -](#_Toc28377)

[**3 基于回归的检测算法 - 8 -**](#_Toc12559)

[3.1 DNN检测模型 - 8 -](#_Toc636)

[3.2 可扩展的目标检测模型 - 8 -](#_Toc12918)

[3.3 基于记忆的检测模型 - 9 -](#_Toc12411)

[3.4 YOLO - 9 -](#_Toc23682)

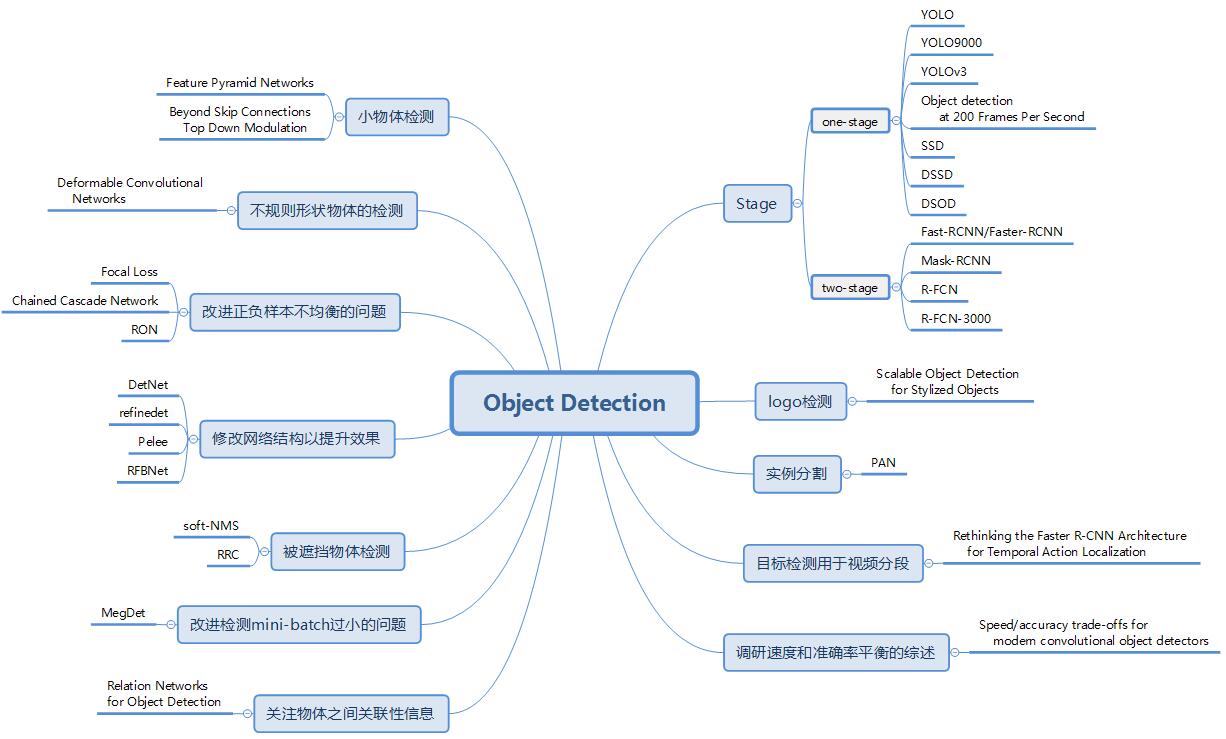
[3.5 SSD - 10 -](#_Toc13914)

[3.6 YOLOv2 - 10 -](#_Toc24186)

[3.7 YOLOv3 - 11 -](#_Toc5839)

[**4 总结 - 11 -**](#_Toc9887)

**基于深度学习的目标检测算法概述**



图一 图为基于深度学习的目标检测算法的思维导图

目标检测主要是对输入图像中的物体类别和位置进行判断，实质上是图像分类和目标定位的结合，是计算机视觉领域中的一个重要研究方向。目标检测在人脸识别、动作检测、无人驾驶、智慧农业、智慧医疗等领域取得了广泛的应用。在深度学习兴起之前，传统的目标检测算法大多采用人工特征来进行检测，人工特征主要针对目标的某个具体特征进行检测。

其中，Sobel、Prewitt、Roberts、Canny和LOG等描述子主要用来做边缘检测;Harris、FAST、CSS和DOG主要用于角点检测等。人工特征的好坏对目标检测的精度起到了决定性作用。深度学习使用非线性模型将原始输入数据逐层转化为更高层次、更抽象的特征，具有强大的学习和表征能力，能够自动发现分类或检测任务所需要的特征。深度学习的发展极大地推动了目标检测，基于深度学习的目标检测获得了广泛的关注，成为了当今计算机视觉领域的研究热点之一。

1 深度学习目标检测算法

## 目标检测的任务是找出图像或视频中的感兴趣物体，同时检测出它们的位置和大小，是机器视觉领域的核心问题之一。目标检测过程中有很多不确定因素，如图像中物体数量不确定，物体有不同的外观、形状、姿态，加之物体成像时会有光照、遮挡等因素的干扰，导致检测算法有一定的难度。进入深度学习时代以来，物体检测发展主要集中在两个方向：

## two stage算法如R-CNN系列。

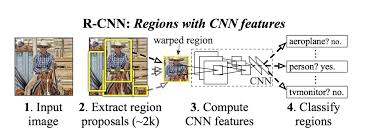


图2 R-CNN算法

## one stage算法如YOLO、SSD等。

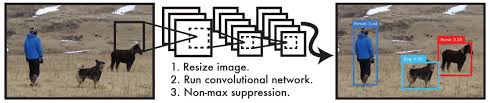


图3 YOLO算法

两者的主要区别在于two stage算法需要先生成proposal（一个有可能包含待检物体的预选框），然后进行细粒度的物体检测。而one stage算法会直接在网络中提取特征来预测物体分类和位置。

2 基于分类的检测算法

基于分类的目标检测算法首先利用Selective Search、Edge Boxes等算法生成可能包含待检测目标的候选区域(Region proposal)，再对这些候选区域进一步进行分类以及位置校准得到最终的检测结果。

## 2.1 OverFeat

OverFeat是最先将深度学习应用到目标检测中的算法之一。OverFeat利用卷积神经网络提取特征，可同时实现图像分类、定位以及目标检测任务，不同任务之间共享特征提取层，在ILSVRC2013数据集上的MAP达到了24.3%。虽然OverFeat相对传统算法有显著改进，但共享特征层的表达能力不强，对小目标的检测效果不好，整体效果仍然差强人意。

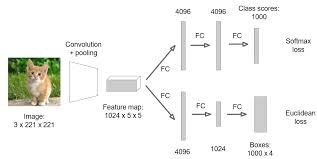


图4 OverFeat算法

## 2.2 Ｒ-CNN

R-CNN算法首先采用Selective Search方法，将输入图像分割成多个模块，再基于颜色、纹理等相似度对这些模块进行合并，最终得到大约2000个不同大小的候选区域，并将这些区域归一化到固定大小;然后使用卷积神经网络对这些候选区域分别进行特征提取，采用多个SVM对这些提取到的特征进行分类;最后对这些区域进行位置校准。Ｒ-CNN在VOC数据集上获得了58.5%的准确率，但是R-CNN存在以下2个问题:

1.对每个候选区域进行特征提取会产生大量重复运算，无法做到实时更新;

2.由于卷积神经网络的全连接层要求所有输入大小相同，R-CNN将图片缩放到相同大小，破坏了物体的长宽比，导致部分有用信息丢失，影响检测效果。

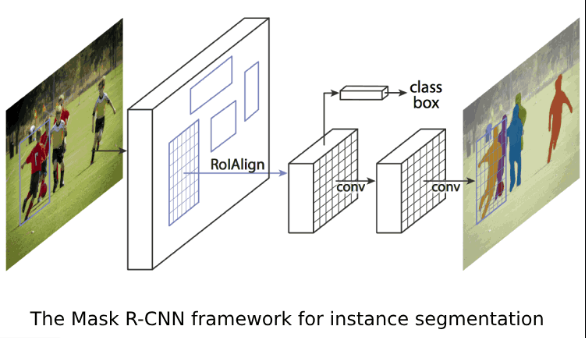


图5 R-CNN算法

## 2.3 SPP-Net

为了解决R-CNN只能检测固定大小图像的问题，提出了SPP-Net算法。

SPP-Net只需要对输入图像进行一次卷积得到整张图片的特征图(featuremap)，然后将不同大小的候选区域映射到特征图上。为了满足全连接层对输入大小一致的要求，SPP-Net采用空间金字塔池化层(Spatial Pyramid Pooling，SPP)，利用不同大小的池化窗口将这些映射到特征图上的候选区域统一到相同维度，最后利用SVM进行分类，同时利用边框回归来微调候选框的位置。

空间金字塔池化层使得SPP-Net能够处理不同大小的图片，并且完整保存了输入图片的信息。同时由于SPP-Net只需要对图像进行一次卷积，减少了运算时间，其检测速度比R-CNN快38～102倍，但同R-CNN一样，在SPP-Net中，确定候选框、特征提取、训练SVM、训练边框回归模型这些步骤依然是单独进行的，导致训练时间较长;在训练时将所有的样本存储到磁盘上会占用大量空间;SPP-Net只能对空间金字塔池化层后的全连接层进行微调，无法同时更新前面的卷积层。

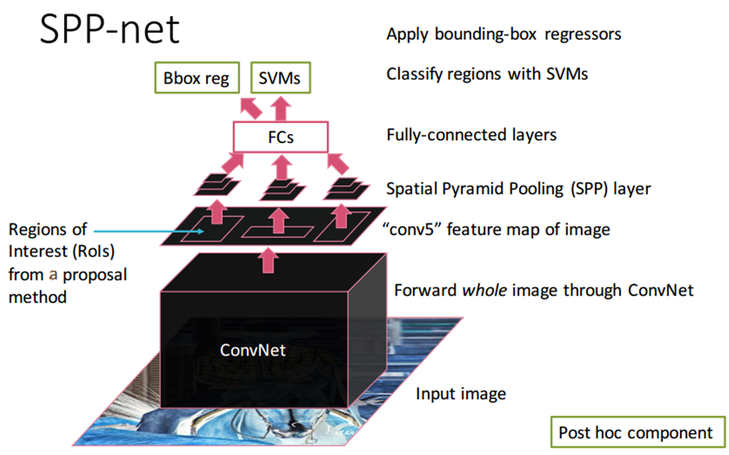


图6 SPP-Net算法

## 2.4 FastＲ-CNN

Fast R-CNN继承了SPP-Net能够处理不同输入大小图片的优点，但是将SPP-Net多尺度空间金字塔池化层简化为池化窗口大小，可以任意设定单尺度的ROlPooling;同时引入了多任务学习模式，将多个步骤简化到一个卷积神经网络中;对卷积神经网络的全连接层输出做SVD分解得到2个向量，分别用于softmax分类以及边框回归。Fast R-CNN检测时所有的特征都存放在显存中，释放了大量磁盘空间。

FastR -CNN主要分为以下4个步骤:

1.利用卷积神经网络提取图像特征。

2.利用区域生成算法提取候选区域，并将这些区域映射到特征图上。

3.利用ROl Pooling将候选区域池化到固定大小的特征表示。

4.分别应用soft max做多分类目标识别，应用边框回归对边框位置进行微调。 Fast R-CNN极大提升了检测速度，在精度上也有了一定提高，但是对候选区域的提取依然采用Selective Search方法，消耗了大量时间，仍然无法达到实时性的要求。

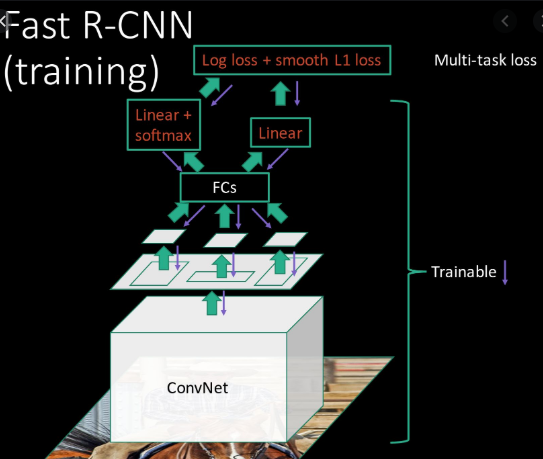


图7 Fast RCNN网络

## 2.5 FasterＲ-CNN

FasterR-CNN利用RPN网络提取候选区域，并将ＲPN和FastＲ-CNN融合到一个网络中共享卷积特征，实现了真正意义上的端到端训练。RPN是一个全卷积网络，其思想是通过在特征图上做窗口滑动，将锚点框 (anchor)映射到原图，得到不同大小和比例的候选区域。RPN主要有2个输出:

1.分类支路采用softmax得到anchor属于物体或背景的概率;

2.边界框回归支路对anchor的位置进行校准得到4个坐标值。

FasterR-CNN的主要步骤为:

1.利用卷积神经网络对输入图片提取特征。

2.利用RPN生成k个不同大小和比例的候选区域。

3.ROIPooling利用ＲPN生成的候选区域和卷积神经网络提取到的共享特征图得到固定大小的候选区域特征图(proposalfeaturemap)。

4.利用softmax对proposals进行具体类别的分类，利用边框回归获得物体的精确位置。FasterR-CNN利用RPN进一步提升了检测速度，同时提高了检测精度。

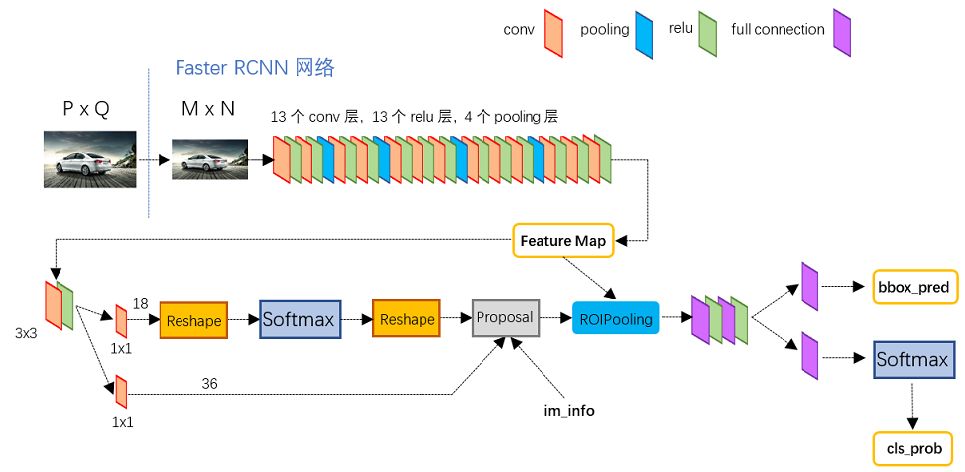


图8 Faster RCNN网络

3 基于回归的检测算法

## 3.1 Deep neural network for object detection

利用深度神经网络(Deep Neural Network，DNN)将目标检测当做回归问题处理以此进一步提高精度。将AlexNet最后的多分类层改为回归层，对每个目标输出二进制掩模(binarymask)，如果某个像素在给定物体的边界框内则置为1，否则为0。为了检测多个距离相近的目标，神经网络需要输出多个二进制掩模。为了精确定位，将二进制掩模的像素放大到与输入图像一致，同时使用定位器进一步提高定位准确度。

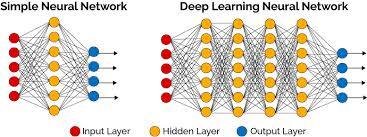


图9 DNN网络示意图

## 3.2 可扩展的目标检测

神经网络模型对每个盒子预测一组边界框以及一个分数，表示这个盒子包含物体的可能性。在训练过程中采用聚类方法来得到物体的典型位置以便于与真实框进行匹配，随后的YOLOv2也采用了类似的思想来得到用于训练的anchor。

## 3.3 基于记忆的检测模型

通过模拟大脑的记忆和预测机制设计了一种检测模型，首先在静态图上滑动固定大小的窗口以生成序列，接着利用卷积神经网络来提取这些序列的特征，最后长短期记忆以适当的顺序接收这些序列特征。基于记忆的检测模型集成了特征学习和序列学习，可应用于预测监督场景中对象潜在的位置。

## 3.4 YOLO

YOLO的网络结构延续了GoogleNet的核心思想，利用卷积层来提取图片特征，利用全连接层来得到最后的检测结果。YOLO将输入图像分成S×S个网格，每个网格负责检测中心落在该网格的物体。YOLO的端到端训练和预测使得其具有较快的检测速度，可以达到45f/s，然而YOLO的检测精度却只有63．4%;在YOLO算法中虽然每个网格预测B个边框，但是只预测一组类别概率值，并且最后只选择IOU最高的边框作为输出结果。如果一个网格中包含多个物体，YOLO只能检测出其中一个物体。

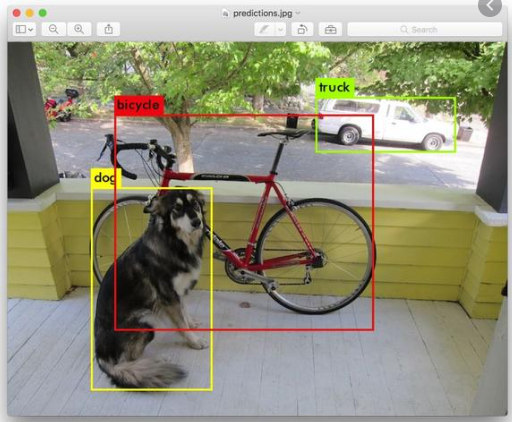


图10 YOLO目标检测

## 3.5 SSD

SSD借鉴了FasterR-CNN中anchor的机制，在特征图上生成一系列不同大小和长宽比的默认框，同时采用困难样本挖掘方法对负样本进行采样使得训练过程中正负样本的比例维持在1:3。

SSD对每一个默认框输出相应的位置偏移以及类别置信度。值得注意的是，SSD将背景类别也做了一个特殊的类别，也就是如果检测的类别为C类，那么每一个默认框需要预测C+1个类别置信度值，最后选择置信度最高的类别作为默认框所属的类别。SSD同时采用了特征金字塔的方式来做多尺度融合预测。SSD的检测速度达到了59f/s，MAP为73．9%，相对YOLO来说有了较大提升，但是对小物体的检测效果仍然不理想。

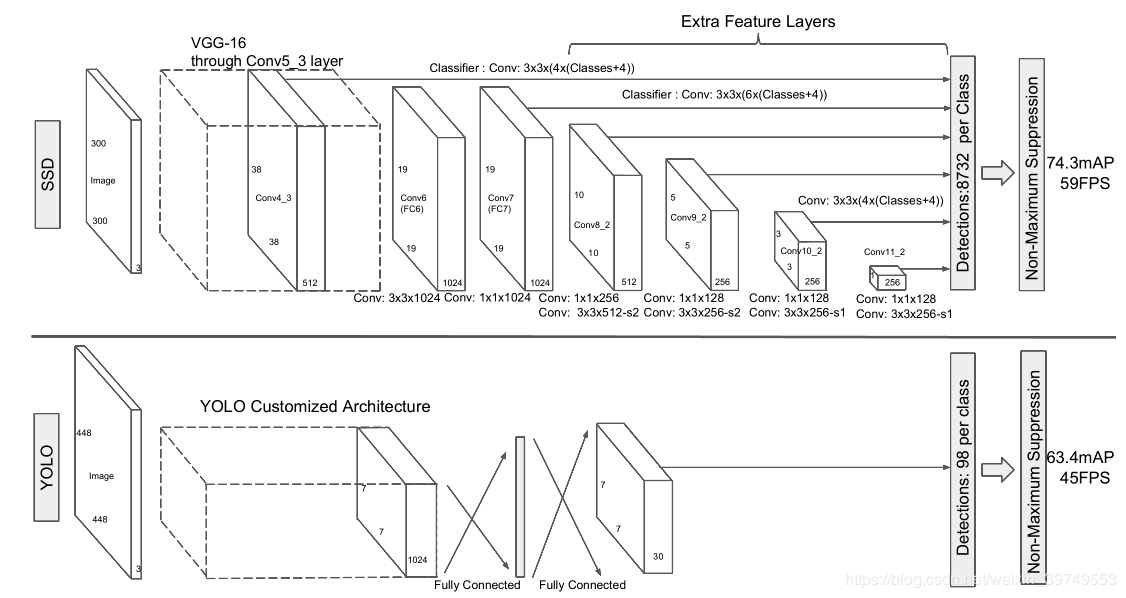


图11 YOLO目标检测算法与SDD检测算法的比较

## 3.6 YOLOv2

为了得到更快、更准确的检测效果，YOLOv2在YOLO的基础上做了许多方面的改进:

1.在每一层卷积层后加入了BN层加快整个网络的收敛速度;

2.使用高分辨率图像来训练网络模型。目前许多目标检测算法采用在ImageNet数据集上得到的预训练模型作为基础特征提取网络，但许多预训练模型的输入图像大小低于256×256。比如YOLO采用224×224的图像来训练分类网络，但检测时采用448×448的图像，导致模型需要适应分辨率的变化。YOLOv2首先利用224×224的图像来预训练分类网络，再用448×448的图像来微调网络模型，最后检测时采用448×448的图像，这样网络就不需要花时间适应不同的分辨率。除此之外，YOLOv2还用了K-means聚类方法来确定anchor的大小、利用Darknet-19作为基础的特征提取网络等。与YOLOv2在同一时期提出的YOLO9000结合检测和分类数据集，在2类数据集上进行训练，可实现9000多类物体的检测。

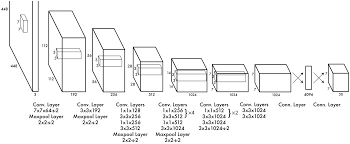


图 12 YOLOv2检测方法示意图

## 3.7 YOLOv3

YOLOv3在YOLO9000的基础上主要做了如下改进:

1.借鉴FasterＲ-CNN的思想。对每个候选框增加了“物体性”的预测，判断其属于前景或者背景的概率，不同的是，YOLOv3对每个真实框只分配一个先验边界框。 2.采用多尺度融合预测。YOLOv3借鉴FPN的思想，利用多尺度特征图，提取细粒度的特征，增加对小物体的检测效果。

3.采用Darknet-53作为主干网络。Darknet-53综合了YOLOv2的主干网络Darknet-19和ResNet的优点，在ImageNet上的Top-5准确率可媲美ＲesNet-152，同时减少了计算量，进一步提高了速度。YOLOv3虽然相对YOLO9000提升了对小物体的检测效果，但对大物体的检测效果却略有下降。

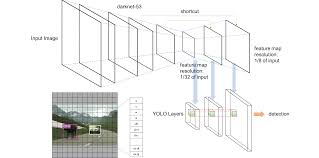


图 13 YOLOv3目标检测算法示意图

# 4 总结

近年来，目标检测算法蓬勃发展，基于深度学习的目标检测框架仍是目标检测任务的主流研究方向。同时，目标检测是计算机视觉和数字图像处理的一个热门方向，广泛应用于机器人导航、智能视频监控、工业检测、航空航天等诸多领域，通过计算机视觉减少对人力资本的消耗，具有重要的现实意义。因此，目标检测也就成为了近年来理论和应用的研究热点，它是图像处理和计算机视觉学科的重要分支，也是智能监控系统的核心部分，同时目标检测也是泛身份识别领域的一个基础性的算法，对人脸识别、步态识别、人群计数、实例分割等任务起着至关重要的作用。由于深度学习的广泛运用，目标检测算法得到了较为快速的发展。

目前大部分目标检测算法采用anchor机制，在特征图上产生大量密集的anchor，再对这些anchor进行进一步的分类和位置校准。但大量anchor容易产生正负样本不平衡的问题，同时针对不同数据集需要设置不同尺度与长宽比的anchor。如何设计合适的anchor还需要进一步研究。一些研究人员尝试使anchor-free的方法来检测目标，但这一类方法对主干网络的要求相对较高。

除此之外，如何优化目标检测框架，设计更好的损失函数、更适合目标检测任务的网络来更好地提取特征仍旧是未来目标检测算法的重点研究方向。虽然One-Stage检测算法和Two-Stage检测算法都取得了很好的效果，但是对于真实场景下的应用还存在一定差距，目标检测这一基本任务仍然是非常具有挑战性的课题，存在很大的提升潜力和空间。