学校代码：10730

分类号：　　密级：

兰州大学

硕士学位论文

（专业学位）

论文题目（中文）	YOLOv5 目标检测算法多阶段改进
论文题目（外文）	Multi-Stage Improvement of YOLOv5 Object Detection Algorithm
作者姓名	黎戈
类型领域	应用统计
研究方向	机器学习与数据挖掘
教育类型	学历教育
指导教师	白建明
合作导师	
论文工作时段	2019 年 10 月 至 2021 年 3 月
论文答辩日期	

校址：甘肃省兰州市城关区天水南路 222 号

学　　院：数学与统计学院　　学　　号：220180920421

学生姓名：黎戈　　导师姓名：白建明

学科名称：应用统计·应用统计

论文题目：YOLOv5 目标检测算法多阶段改进

原 创 性 声 明

本人郑重声明：本人所呈交的学位论文，是在导师的指导下独立进行研究所取得的成果。学位论文中凡引用他人已经发表或未发表的成果、数据、观点等，均已明确注明出处。除文中已经注明引用的内容外，不包含任何其他个人或集体已经发表或撰写过的科研成果。对本文的研究成果做出重要贡献的个人和集体，均已在文中以明确方式标明。

本声明的法律责任由本人承担。

论文作者签名：黎戈　　日　期：2021.5.19

关于学位论文使用授权的声明

本人在导师指导下所完成的论文及相关的职务作品，知识产权归属兰州大学。本人完全了解兰州大学有关保存、使用学位论文的规定，同意学校保存或向国家有关部门或机构送交论文的纸质版和电子版，允许论文被查阅和借阅；本人授权兰州大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用任何复制手段保存和汇编本学位论文。本人离校后发表、使用学位论文或与该论文直接相关的学术论文或成果时，第一署名单位仍然为兰州大学。

本学位论文研究内容：

☑可以公开

□不宜公开，已在学位办公室办理保密申请，解密后适用本授权书。

（请在以上选项内选择其中一项打“√”）

论文作者签名：黎戈　　导师签名：

日　　期：2021.5.19　　日　　期：2021.5.19

YOLOv5 目标检测算法多阶段改进[1]

中文摘要

目标检测作为现今计算机视觉的基础任务之一，近年来受到了人们广泛的关注，诸如图像标注、动作识别、人脸识别和视频分割等领域都对其十分依赖。现如今，一种以卷积神经网络作为特征提取方法的目标检测算法逐渐成为了当今的主流算法。其中，YOLOv5 算法更是凭借其出色的表现被大家所一致认可。然而现实场景复杂多变，YOLOv5 在一些场景下也会存在误检，导致准确率下降。因此，设计一种检测性能更好的模型成为了一项挑战。

基于上述背景，本论文对 YOLOv5 算法进行改进，结合 YOLOv5 算法的几个阶段，提出将动态锚框和注意力机制添加到 YOLOv5 网络结构之中，并且在预测框筛选阶段提出使用目标框加权融合算法。具体改进包括三个方面：

第一，为解决 YOLOv5 算法中锚框的先验信息不准确的缺点，提出加入动态锚框机制。首先通过 K-Means 聚类算法为训练数据集生成锚框，然后在模型中加入动态锚框模块，最后在网络的训练过程中动态地对锚框的大小以及位置进行更新。通过对比实验证明加入动态锚框机制的确能够提升模型的平均精度。

第二，为解决 YOLOv5 中不同尺度下的特征图的不平衡问题提出添加注意力机制。通过对原有网络结构同时添加通道注意力和空间注意力，使检测网络能够更显著地提取特征，增加了网络的检测能力。同时将改进后的网络与原网络进行对比实验。结果表明，添加注意力机制后的模型确实在检测精确度上有明显提高。

第三，YOLOv5 中使用非极大值抑制选择最终的预测边框，该方法会直接丢弃掉得分低的预测值，没有完全利用有效信息。本论文提出使用目标框加权融合代替非极大值抑制，充分利用网络的特征信息来筛选预测的目标框，并通过对比实验验证其有效性。

最后，论文利用 Udacity 自动驾驶数据集上将改进后的模型和 YOLOv5 进行对照实验，实验结果表明改进后的模型比原有模型在平均精度上提高了 3.1%，

[1] 本论文获得以下项目资助：
1. 车险欺诈的识别、检查与控制策略研究；中央高校基本科研业务费专项资金项目（16LZUJBWZD002），2016.01-2017.12；主持人：白建明
2. 智慧城市、金融科技、工业大数据中的数据挖掘、机器学习与数据建模；技术咨询合同项目（071200225），2019.12-2021.12；主持人：白建明

因此可以证明本论文所改进的模型在检测精度上具有一定的效果。

关键词：目标检测，YOLOv5，注意力机制，动态锚框，目标框加权融合

Multi-Stage Improvement of YOLOv5 Object Detection Algorithm

Abstract

As one of the basic tasks of computer vision today, object detection has received extensive attention in recent years. Fields such as image labeling, action recognition, face recognition, and video segmentation all rely heavily on it. Nowadays, a target detection algorithm using convolutional neural network as a feature extraction method has gradually become the mainstream. Among them, the YOLOv5 algorithm is unanimously recognized by everyone with its outstanding performance. However, the real scene is complex and changeable, and YOLOv5 will also have misdetection in some scenes, resulting in a decrease in accuracy. Therefore, designing a model with better detection performance has become a challenge.

Based on the above background, this paper improves the YOLOv5 algorithm in its several stages by adding dynamic anchor frames and attention mechanisms to its network structure, and using the weighted target frame fusion algorithm in screening stage of the prediction frame. Specific improvements include three aspects.

First, a dynamic anchor frame mechanism is proposed to overcome the shortcoming of the inaccurate prior information of the anchor frame in the YOLOv5 algorithm. To begin with, the K-Means clustering algorithm is used to generate anchor frames for the training data set, then the dynamic anchor frame is added to the model, and finally the size and position of the anchor frames are dynamically updated during the network training process. The comparative experiments demonstrate that adding a dynamic anchor frame mechanism can indeed improve the average accuracy of the model.

Second, we propose to add to an additional attention mechanism to solve the imbalance problem of feature maps at different scales in YOLOv5 algorithm. By adding channel attention and spatial attention to the original network structure at the same time, the detection network can extract features more significantly and perform much better. Compared with the original one, the model after adding the attention mechanism does have a significant improvement in detection accuracy.

Third, YOLOv5 algorithm uses non-maximum value suppression to select the final prediction frame. This method directly discards the low-scoring prediction value, and does not fully utilize effective information. This paper proposes to use the weighted fusion of target frames instead of non-maximum suppression, making full use of the characteristic information of the network to screen the predicted target frames, and verifying its effectiveness through comparative experiments.

Finally, we conduct a control experiment on the improved model using the Udacity autopilot data set. The results show that it has an average accuracy of 3.1% higher than the original one. Therefore, it can be proved that the improved model in this paper has a certain effect on detection accuracy improvement.

Keywords: object detection, YOLOv5, attention mechanism, dynamic anchor, weighted-boxes-fusion

目 录

中文摘要 I
Abstract III
第一章 绪论 1
1.1 研究背景 1
1.2 研究思路与论文结构 2
1.2.1 研究思路 2
1.2.2 论文结构 3
1.3 研究创新与意义 3
第二章 文献综述 5
2.1 目标检测 5
2.2 锚框 7
2.3 注意力机制 8
2.4 目标框加权融合 9
第三章 目标检测及其相关理论 11
3.1 深度学习概述 11
3.2 神经网络概述 11
3.2.1 多层感知机 11
3.2.2 激活函数 12
3.2.3 优化方法 12
3.2.4 反向传播 13
3.2.5 深度学习的一般步骤 13
3.3 YOLOv5 14
3.3.1 一些目标检测算法 14
3.3.2 YOLOv5 16

3.4 目标检测评价指标 17
3.4.1 精确率与召回率 17
3.4.2 IoU 18
3.4.3 平均精度 18
第四章 YOLOv5 算法多阶段改进 20
4.1 输入阶段改进：基于动态锚框 21
4.1.1 问题的提出 21
4.1.2 模型构建 21
4.1.3 对比实验 23
4.2 骨干网络改进：基于注意力机制 24
4.2.1 SENet 25
4.2.2 模型构建 26
4.2.3 对比实验 28
4.3 回归框筛选阶段改进：基于目标框加权融合 30
4.3.1 问题的提出 30
4.3.2 模型构建 30
4.3.3 对比实验 31
第五章 实验结果与分析 32
5.1 数据集介绍 32
5.2 对比实验结果 32
5.3 实验结果分析 34
第六章 结论与展望 36
6.1 主要结论 36
6.2 研究展望 36
参考文献 38
致 谢 41

第一章 绪论

1.1 研究背景

早在上世纪 50 年代，被誉为“计算机科学之父”的图灵就开始了对早期人工智能的探索与研究。半个多世纪以来，经过许多科学家们的不断努力，随着深度神经网络的诞生，一种由数据驱动的能够进行自我迭代更新参数的深度学习模型使得人工智能领域取得了重大的突破和飞速的发展。目前，诸如自动翻译、机器视觉、人脸识别、智能搜索、语音识别等各行各业都靠深度学习与人工智能取得了长足发展。作为计算机视觉四大任务之一的检测任务，随着深度学习的发展突破了它的自身瓶颈。依靠卷积神经网络(CNN)，目标检测从一个较为冷门的科学变得越来越受到人们的关注，例如身份识别、交通运输检测、医学影像识别等各个领域都离不开目标检测的身影。

数十年来，目标检测都是计算机视觉领域中的一个很关键的技术，吸引了人们很多的重视。它的目的是检测图片中令人们感兴趣的目标，例如人类、动物以及汽车等等。与此同时，目标检测作为其他任务（如实例分割、图像描述生成、物体追踪等）的基础，也起着至关重要的作用。随着 Krizhevsky 提出名为 AlexNet(Alex et al, 2012)的新结构，全世界见证了卷积神经网络的诞生，以这一年为时间节点，研究者们自然地将目标检测分成了两种——基于数字图像处理的目标检测，基于深度学习的目标检测。

基于数字图像处理的目标检测大致分为：特征提取和分类两步。由于传统方法是人工提取特征，没有针对性的终点，而且没有应对变化的能力，模型缺乏鲁棒性，就会存在“准确度差、效率低”的缺点。基于深度学习的目标检测利用 CNN 进行特征的提取，把特征提取和分类放到一个网络中去完成。并且随着网络层数的不断加深，卷积神经网络能够在不同的尺度下提取深度特征，使其能够提取更深程度的特征信息。基于深度学习的目标检测又可分为“两阶段”和“单阶段”，顾名思义，“两阶段”指的是将提取特征和检测分两个步骤，而“单阶段”则是将这两个步骤合并为一步。“两阶段”目标检测以 R-CNN(Girshick, 2014)、Fast-RCNN(Girshick, 2015)、Faster-RCNN(He et al, 2015)为代表，为先进性区域选取，再进行分类的网络结构；“单阶段”目标检测以 SSD(Liu et al, 2016)、YOLO(Redmon et al, 2016)系列为代表，将区域选取和分类整合到同一个网络结构中，构建一个“分类+回归”的多任务学习模型结构。

其中，YOLO 系列算法经过若干代的发展，YOLOv5(Jocher et al, 2020)以其较高的精确度和较为简单的模型等特点成为目前目标检测领域中备受大家欢迎的模型之一。例如虽然 YOLOv5 的作者在本论文撰写阶段并没有发表文章，但是在 github 的目标检测相关代码库中，YOLOv5 得到了最多的关注；在目标检测的相关竞赛中，YOLOv5 也作为被使用最多的算法取得了优秀的成绩。尽管如此，YOLOv5 仍然存在许多待改进之处，诸如在锚框的生成上没有考虑不同大小的目标的分布问题，在回归框的选取上没有考虑更多的特征信息等，这有可能会导致该算法在某些场景下达不到令人满意的检测效果。

针对上述问题，本论文在 YOLOv5 的研究基础上，对模型加以改善。在原有的基础上分别基于动态锚框、注意力机制和目标框加权融合对模型进行多阶段的改进。主要分为以下几个步骤：第一，增加动态锚框结构作为先验框的生成方法，以便更好地提取网络特征结构，且动态锚框具有自学习能力，能够通过自身迭代得出适合网络的参数。第二，增加注意力模块来使得网络在处理图像上能够更加关注于感兴趣的区域，更少关注无用区域，以增加模型的性能。第三，使用目标框加权融合作为预测框的选取方法，更为充分地利用网络的特征信息，且更为准确。

1.2 研究思路与论文结构

1.2.1 研究思路

本论文主要对 YOLOv5 算法进行改进。YOLOv5 算法在目前的实际应用上已经取得了优异的成绩，如今的很多公司在他们的检测算法上都会选择 YOLOv5 作为目标检测算法，且该算法也取得了不错的效果。本论文站在提升检测精确度的角度对模型进行多阶段的改进，即网络的输入阶段、训练阶段和回归预测阶段共三个阶段。

在网络训练之前的阶段本论文考虑到通常单阶段目标检测算法会使用的锚框作为网络的先验信息进行训练，因此本论文从改进锚框的角度入手，分析 YOLOv5 模型中锚框生成的不足之处，并且提出将动态锚框这种可以随着训练过程自我更新的机制加入到 YOLOv5 的网络结构之中，并且在 Pascal VOC 数据集上做对比实验来验证其有效性。

在训练阶段本论文考虑到视觉注意力机制是提升 CNN 性能的一个不错的方法，因此在 YOLOv5 的主干网络中添加视觉注意力机制，通过空间注意力和通道注意力的混合使用，使得原有模型能够更好的学习到提取重要特征的能力，为

模型带来更为优秀的检测性能。

在模型的回归检测阶段本论文考虑到原始模型中使用的非极大抑制方法的缺陷，从而将一种更为高效、更为准确的方法目标框加权融合应用到网络结构的回归预测阶段以代替原有方法。同样地，本阶段也使用 Pascal VOC 做对比实验以验证改进后的效果。

最后，本论文将多阶段改进的新模型与原有模型及目标检测经典模型在 Udacity 自动驾驶数据集上做对照实验，通过一个贴近现实的数据集纵向对比本论文所改进的方法与其他方法的性能，证实本论文所改进的方法在现实中的应用价值。

1.2.2 论文结构

论文结构大体如下：

第一章为绪论，总体说明本论文的背景及意义，展示深度学习和人工智能在生活中的应用，也阐述目标检测的多学科多领域交叉结合的背景。

第二章为文献综述，分别从目标检测、锚框、注意力机制和非极大值抑制的方向介绍国内外研究者的研究成果。

第三章为基础理论概述，介绍目标检测的背景、基础理论、常见算法和评价指标。并介绍本论文研究的基础——YOLOv5 算法。

第四章为本论文所做的工作，较为详细地阐述本论文对 YOLOv5 算法所做出的若干点改进。分别从动态锚框、注意力机制、目标框加权融合三个方面对 YOLOv5 模型进行改进。并且在每个改进点的基础上进行对比实验以验证其有效性。

第五章的主要内容是设计对照实验以及根据实验结果做出相应的分析。首先介绍本论文的改进算法所用到的数据集。然后设计对照实验，将本论文所改进的算法与目标检测的一些经典算法对比。最后根据对比实验的结果进行分析。

第六章是结论与展望，先对全文做出总结，并讨论未来可能的研究主题与方向。

1.3 研究创新与意义

本论文的创新点和意义可以总结为以下几个方面：

一、动态锚框结构的使用

原始 YOLOv5 网络使用 K-Means 算法(MacQueen, 1967)在三种不同尺度下生成锚框。虽然该方法可以在不同的训练集下自动地生成相应的锚框，但实际情

况下目标往往分布不均，难以均匀地分布在三种不同大小的锚框上，导致误检的概率大大提高。因此本论文提出动态锚框的概念来缓解这一问题。通过类似ARM(Zhang, 2018)的结构对锚框进行尺度和位置上的动态更新，进而影响回归效果，使模型总体精确度更高。

二、添加注意力机制

YOLOv5 的主干网络深度较大，再加上特征金字塔结构，使得网络非常复杂。即使采用类似短路连接的模块，复杂的网络结构还是会导致提取特征的不平衡。本论文通过加入视觉注意力机制，强调更为重要的信息，抑制不太重要的信息。同时在 Li (2018)、Xiao(2020)等论文的思路下，使用基于全局平均池化(GAP)和全局最大池化(GMP)的混合注意力模块，进一步提升模型检测性能。

三、使用目标框加权融合

原有 YOLOv5 网络结构中在选择特征框的阶段使用非极大抑制的方法，该方法虽然效率高，但是精确度却难以达到令人满意的程度。本论文在目标框加权融合(WBF; Solovyev et al, 2019)的启发下，使用 WBF 替代原方法中的 NMS 方法作为回归框筛选的方法。通过 WBF 方法能够使模型利用到更多的特征信息从而得出更为精确的结果。

第二章 文献综述

根据本论文研究的需要，本章主要从目标检测、锚框及动态锚框、注意力机制和预测框筛选四个方面对相关文献进行综述。

2.1 目标检测

目标检测是一种与计算机视觉和图像处理有关的计算机技术，常用于检测一张图片或一段视频中人们所关注的对象（例如建筑物或汽车）的实例。现今生活中很多领域都用到了目标检测技术，如过安检时验证身份信息的人脸识别技术以及在智慧交通系统中对机动车和行人做检测和跟踪的技术等等。

目标检测问题可表述如下：假设有一个感兴趣的对象类，对于一张图像或者一段视频而言，其中包含我们感兴趣的对象类，目标检测的任务就是设计一个算法或系统来找到具体对象实例以及其在图像上的位置。因此，目标检测从本质上来说可以理解为一个多任务学习，即分类和位置。换句话说，第一任务是找到用于描述图像区域的信息表示，即特征向量或描述符。第二个任务是在特征提取后，应用机器学习的分类算法对上一个阶段找到的区域进行分类。

在深度学习还没有被大家所普遍使用之前，Lowe(1999)提出尺度不变特征变换(SIFT)来检测图像特征。SIFT 方法通过提取一组图像中的关键点，并与数据库中已有的特征进行比较，根据特征向量之间的 Euclidean 距离找到与其相似的特征，据此来检测图片。Viola and Jones(2001)提出 Haar-like 特征作为人类面部特征的提取方式。方向梯度直方图(HOG)是 Navneet et al(2006)提出的一种方法，它能够更有效的提取特征，以至于被很多人使用。Felzenszwalb(2008)提出了可变形部件模型(DPM)，它在 HOG 的基础上进行改进，解决了 HOG 难处理遮挡物体的问题。

随着 2012 年深度学习和卷积神经网络的提出，人们对于处理计算机视觉问题有了新的工具。由于深度卷积神经网络能够学习图像的高级且鲁棒的特征表示，因此不少研究者将其应用于目标检测。

人们尝试先进行区域选择再利用 CNN 做特征筛选，该方法被成为“两阶段目标检测”(Two-stage Object Detection)。Girshick et al(2014)是第一个结合目标检测和 CNN 的，他们在其研究的基础上提出了 R-CNN。R-CNN 以选择性搜索(Selective Search, Uijlings et al, 2013)作为主要思想。通过选择性搜索提取一组对象候选框，然后将每个框缩放到相同大小，运用 CNN 模型（如 AlexNet）进行特

征提取，最后用线性 SVM 分类器(Cortes et al, 1995)进行类别识别。He et al(2014)提出了空间金字塔池化网络(SPPNet)，在该网络被提出之前，输入都是固定大小的，而何恺明等人通过 SPPNet 来去除这个限制，对于候选区域的选择不需要重新缩放到相同大小。考虑到 R-CNN 中提取特征操作的冗余，Girshick(2015)提出 Fast R-CNN，并在特征提取阶段加入 RoI Pooling 层来固定特征图尺度。随后，He(2015)提出 Faster R-CNN，在其网络结构中使用 RPN 来选择候选区域(Region Proposal)。

随着时代发展，计算机的计算能力也得到了一定的增强。由于“两阶段”目标检测器需要先划分候选区域再进行特征筛选，因此不能达到实时检测的效果。为了模型的简洁以及运算速度的提升，人们相对于“两阶段”目标检测提出了“单阶段”目标检测。深度学习时代第一个被提出来的单阶段目标检测方法是 YOLO，它由 Redmon et al(2016)提出，作者放弃了两阶段“区域检测+验证”的思想，取而代之的是将单个神经网络作用于整个完整图像。YOLO 的运行速度非常快，但缺点是检测精确度不如两阶段的目标检测。之后，Redmon 对 YOLO 的网络结构进行改进，并分别于 2017 年和 2018 年提出了 YOLO 的后续版本 YOLOv2 和 YOLOv3，它们在提高检测精度的同时保持了很高的检测速度。Alexey(2020)在 YOLOv3 的基础上加以改进，提出了新的版本 YOLOv4；同年，来自 Ultralytics 团队的 Jocher(2020)在 YOLOv3 的基础上提出 YOLOv5 模型。该模型运行准确度高于以往的两阶段目标检测模型，且模型运行速度快，可以很好的应用于嵌入式设备和移动端进行检测，因此，YOLOv5 成为目前目标检测表现最好网络模型之一。SSD 由 Liu et al(2016)提出，它的主要贡献是引入了新的检测技术。Lin et al(2017)找出了单阶段目标检测器准确率低的原因并提出了 RetinaNet。Tan et al(2020)提出了 EfficientDet，该网络基于 FPN 提出用 BiFPN 和混合缩放去融合不同尺度下的特征，做到不同尺度下的权值共享，在网络参数量、检测精度和运行速度上都得到了较大的提升。

可以看出，目标检测的发展历程经历了从传统数字图像信号到深度学习的转变。在传统目标检测中，人们的研究着重于数字图像自身的特征，然而现实生活中的图像中目标形态各异、光照强度不均、背景变化多样，导致传统放方法难以提取鲁棒的特征，进而导致检测效果的不准确。而基于卷积神经网络的算法就得以很好的解决这个问题。而在深度学习阶段，目标检测又经历了从两阶段到单阶段的转变。正如其名称所说，两阶段目标检测算法将目标检测大致分为两个阶段，即提取特征和分类，但由于两阶段的网络结构导致模型收敛难度大大上升，因此两阶段目标检测随着研究的深入慢慢被单阶段目标检测取代。单阶段目标检测算法仅仅通过一个深度神经网络结构就完成了这两个阶段，经若干年的研究，现已

发展到了可以兼顾精度与速度的程度。随着研究的深入，越来越多复杂的网络结构被提出，这也导致了目标检测算法性能的完善。尽管如此，在目标检测公开数据集 MSCOCO 上，各算法的表现仍然没有达到完美的效果。因此，对目标检测算法的进一步研究是有必要的。

2.2 锚框

锚框(Anchor Boxes)是一组具有一定高度和宽度的预定义边界框。研究者们为了检测目标对象的长宽比以及纵横比常常通过训练集中的对象来定义这些锚框。在检测期间，预定义的锚框会平铺在图像上。网络会预测概率和其他属性，例如每个平铺锚框的交并比(IoU)和偏移量(Offset)，这些预测用于完善每个单独的锚框。在训练过程中，我们也可以为模型定义多个锚框，每个锚框用于检测不同大小的对象。锚框是固定的初始边界框猜测，网络不会直接预测边界框，而是会预测与平铺的锚点框相对应的概率和细化度。网络为定义的每个锚框返回一组唯一的预测。最终特征图表示每个类的对象检测。锚框的使用使网络能够检测多个对象，不同比例的对象以及重叠的对象。

两阶段网络使用候选区域(Region Proposal)来达到类似锚框的先验效果。在 R-CNN 中，作者使用类似于滑动窗口的方法过滤掉负样本区域得到候选区域。

而在单阶段网络中，人们往往使用锚框来定义边界框的先验信息。目前锚框的生成方式大致分为三种：人为经验规定、聚类生成和作为超参数学习。

在 SSD 中，对每个锚点而言，锚框的生成通过定义一组锚框大小$s_1, s_2, \dots, s_n$和宽高比$r_1, r_2, \dots, r_m$得到$(s_1, r_1), (s_2, r_2), \dots, (s_n, r_m)$共$(\mathrm{n}+\mathrm{m}-1)$个不同大小的锚框。Zhang et al(2018)通过对 SSD 算法的改进提出 RefineDet 网络结构，在该网络中提出使用锚框细化模块(ARM)来调整默认锚框的初始化方法，从而为检测器带来更好的先验信息。在 YOLO 中，模型提出用聚类的方法如 K-Means 生成锚框。通过预定义大中小三种不同的尺度，并在每个尺度下生成三个锚框，共九个锚框作为训练阶段的先验框。

Li et al(2019)在 ARM 模块的基础上提出动态的锚框模型，即 DAFS 模型。该模型在 ARM 的基础上提出了动态特征选择操作，用于为每个经 ARM 细化过后的锚框选择新的像素点，使这些像素点的感受野能更加适合锚框区域。DAFS 模型的提出使得检测精度大为提升，尤其是对于高 IoU 阈值下精度的提升。Qi et al(2020)对锚框进行研究，将动态锚框应用于旋转目标检测的研究，提出了动态锚点学习模型(DAL)，该模型通过新定义的匹配度来执行更加有效的标签分配，使得模型自身可以动态地选择高质量的锚点进行精确检测。

总结来看，锚框的使用相较于不使用锚框有着许多方面的优势，如减小计算量、可根据具体任务做调整、通过计算偏移量来降低优化难度等。例如 YOLO 系列的作者 Joseph Redmon 曾在 YOLOv1 结构中放弃使用锚框，后发现效果不如预期，又在 YOLOv2 结构中选择使用锚框，提高了模型的检测精度。动态锚框的提出更是将锚框的生成并入训练过程中，使其能够更好的生成与训练数据集更加贴切的先验框，为网络结构带来更为合理的先验信息。尽管动态锚框的使用增加了模型的复杂程度，导致了计算量的上升，但是却为模型提供了符合数据本身的先验信息，进一步为模型带来更好的检测性能。

2.3 注意力机制

科学家们对人类视觉感知的研究从而将关注重点放在注意力机制上。由于人类认知的局限性，人们在处理视觉、听觉等各种信息时总是会将更多的关注度放在自己感兴趣的信息上。近年来，学者们将注意力机制应用于深度学习，并证实注意力机制的确能有效提高模型的性能。在计算机视觉任务中，人们发现：通过训练，深度神经网络能够了解每个新图像中需要注意的区域，从而形成注意力。随着研究的深入，人们发现可以将注意力分为软注意力和硬注意力。梯度下降算法是实现软注意力机制的方法，它的权重可以通过深度学习中的前向传播和反向传播进行学习，而硬注意力则需要通过更为复杂的方式实现。

在目标检测领域中，注意力机制常常被使用。Jaderberg et al(2015)提出了空间变换网络(STL)，通过将可微的 STL 模块插入 CNN 中，使得 CNN 自身能够通过特征图的特征对自身做出空间变换。CNN 中每个特征图都是多通道的，Hu et al(2017)提出了压缩和激励网络(SENet)，人们开始对通道注意力的研究加以关注。该网络结构通过压缩模块对输入特征图的通道方向进行压缩，通过激励模块对每个特征通道生成权重，并根据特定任务学习该权重。Li et al(2020)基于 SENet，通过结合全局平均池化(GAP)和全局最大池化(GMP)设计新的结构，在混合注意力的基础上提出了自适应域注意力。Wang et al(2017)提出了非本地神经网络（Non-local Neural Networks），使用 Non-local 层捕捉较远像素点之间的相关关系，通过自注意力使模型在训练和预测过程中实现全局参考。

综上所述，将注意力机制添加到卷积神经网络中就能够使网络更加关注于更为有效的特征，从而获得更好的性能。如 Xiao(2020)在其文章中提到了将视觉注意力机制应用于计算机视觉领域对图像生成、场景分类、目标检测有极大的益处。同时，Yohanandan et al(2018)通过他们的研究表明视觉注意力机制其实更接近于人类自身的视觉认知。近年来，越来越多的研究者们通过引入视觉注意力机制来

改善目标检测网络的提取关键特征的能力，事实证明，注意力机制的确是一个用来提升模型性能的优秀选择。

2.4 目标框加权融合

非极大值抑制(NMS)是一种回归框选择的方法。通常，一张图像中我们感兴趣的目标往往是不同形状的，其大小以及宽高比都大不相同。因此目标检测算法往往会对同一个目标预测出许多不同的预测值（回归框）。NMS 作为目标检测算法的最后一步，通常是用来在模型所预测出来的众多回归框中选择出最适合的那一个。

NMS 最早由 Neubeck et al(2006)提出。其核心思想是“抑制”那些不太可能成为真实值的预测值，保留最有可能成为真实值的那一个预测值。其具体算法如图 2.1 所示。

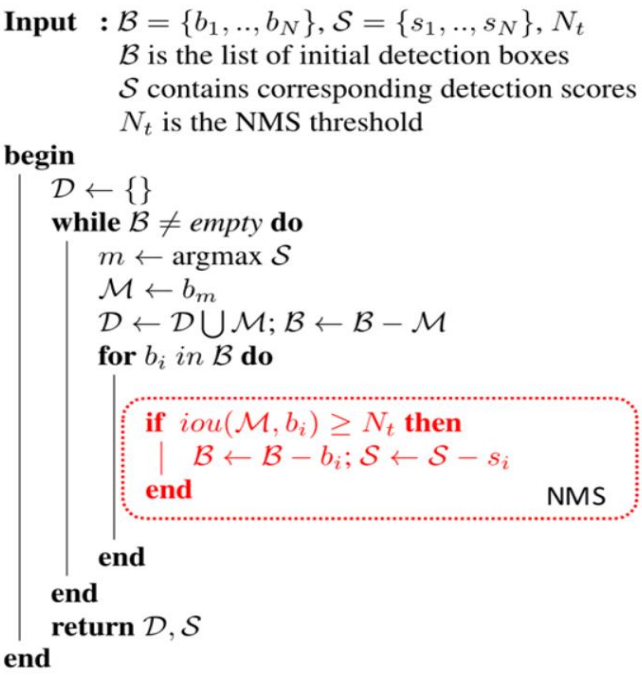


图 2.1 NMS 算法

Navaneeth et al(2017)在 NMS 的基础上提出了 Soft-NMS，Soft-NMS 对 NMS 进行了简单的改动，它不再将得分低的预测框直接删除，而是设置一个权值函数来降低其权重。Liu et al(2019)在 Soft-NMS 的基础上提出了 Adaptive-NMS 的算法，该算法将 Soft-NMS 中的阈值N_t设置为根据目标密度而自适应的阈值。

Solovyev et al(2019)在 NMS 的基础上提出了目标框加权融合(Weighted Boxes Fusion, WBF)，该方法的主要思想为：为每个预测的边界框设置不同的权

重，通过加权融合计算出一个结果作为最终融合的结果。其具体步骤如表 2.1 所示。

表 2.1 WBF 算法步骤

1. 将所有预测框放入一个列表 B 中，将其对应得分放入列表 C 中

2. 建立两个空列表 L 和 F，L 用来存放适合的边界框，F 中只含有一个融合后的边界框

3. 设置一个阈值 THR，遍历 B，找到所有与 F 的 IoU 值大于 THR 的边界框

4. 若步骤 3 中没有找到边界框，则将该边界框加入 B 和 F 的尾部

5. 若步骤 3 中找到了边界框，则将该边界框加入 B 的尾部，且按下列公式更新 F 中的边界框：

$$\mathbf{C} = \frac{\sum_{i=1}^{\mathbf{T}} \mathbf{C}_i}{T} \tag{1}$$

$$\mathbf{X1,2} = \frac{\sum_{i=1}^{T} \mathbf{C}_i * \mathbf{X1,2}_i}{\sum_{i=1}^{\mathbf{T}} \mathbf{C}_i} \tag{2}$$

$$\mathbf{Y1,2} = \frac{\sum_{i=1}^{T} \mathbf{C}_i * \mathbf{Y1,2_i}}{\sum_{i=1}^{\mathbf{T}} \mathbf{C}_i} \tag{3}$$

6. 遍历完 B 之后再按如下公式对 F 中框的置信度做一次调整：

$$\mathbf{C} = \mathbf{C} * \frac{min(T,N)}{N} \tag{4}$$

NMS、Soft-NMS、Adaptive-NMS、WBF 都是在模型的预测阶段对预测框进行筛选从而选择出最优结果作为目标检测的预测值的方法。NMS 的优点在于计算量小，但是由于其对得分低的预测框进行抑制的特性，往往会丢掉一部分信息，导致结果的偏差。WBF 的优点在于利用了所有预测框的信息进行加权求和，因此所得结果的精确度更高。其缺点在于算法的复杂度过高，导致 WBF 处理一张图像的平均时间会比 NMS 的更长。本论文将讨论的侧重点放在精度的提升之上，因此将 WBF 作为本论文提升检测精确度的一种方法。

第三章 目标检测及其相关理论

本章内容先介绍深度学习的相关内容，再介绍神经网络的工作原理，然后介绍本论文的研究基础——YOLOv5 网络的具体结构，最后对目标检测评价指标进行介绍。

3.1 深度学习概述

深度学习是一种机器学习方法(Deng et al, 2014)，从学习任务结构的角度，学习可以被分为有监督的，半监督的或无监督的(Bengio et al, 2013; Schmidhuber , 2015; Hinton et al, 2015）。

深度学习中的形容词“深度”是指在网络中使用多层。早期的工作表明，线性感知机不能成为通用分类器，但是具有非多项式激活函数和一个无界宽度的隐藏层的网络可以。深度学习是一种现代变体，它涉及无限大小的无限制层数，这允许实际应用和优化实现，同时在温和条件下保留理论通用性。在深度学习中，出于效率，可培训性和可理解性的考虑，还允许各层是异类的，并且与生物学上有据可依的连接主义模型大相径庭，因此是“结构化”部分。

3.2 神经网络概述

3.2.1 多层感知机

多层感知机(MLP; Trevor et al, 2009)是前馈神经网络的补充。如图 3.1 所示。输入层接收要处理的输入信号。所需的任务（例如预测和分类）由输出层执行。放置在输入和输出层之间的任意数量的隐藏层是多层感知机的真正计算引擎。类似于多层感知机中的前馈网络，数据在前向方向上从输入层流到输出层。多层感知机中的神经元通过反向传播学习算法进行训练。多层感知机设计为近似任何连续函数，可以解决不可线性分离的问题。多层感知机的主要用例是模式分类、识别、预测和近似。

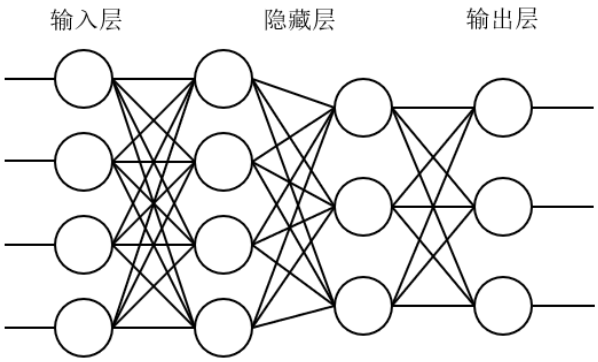


图 3.1 多层感知机

3.2.2 激活函数

激活函数是在人工神经网络中使用的函数，对于较小的输入，该函数输出较小的值，如果其输入超过阈值，则输出较大的值。如果输入足够大，则激活功能将“触发”，否则将不执行任何操作。换句话说，激活功能就像一个门，用于检查输入值是否大于临界值。

在如今的深度学习，特别是 CNN 中，常常使用一种类似于斜坡型的函数来作为激活函数，它被称为整流线性单元(Hahnloser et al, 2000)，简称 ReLU。ReLU 函数的表达式为：

$$f(x) = max(0, x)$$

ReLU 函数的优点在于它的形式上的简洁性非常有利于计算和区分，特别是在反向传播中的计算。尽管该函数在函数值为 0 的点不可导，但是在实际运用中却不影响，因为在实际情况中函数值恰好为 0 的点非常少，因此在运用中可以将该点的导数手动设置为 0。ReLU 函数以其出色的性能成为深度学习中最流行的激活函数之一。

3.2.3 优化方法

优化方法是更新参数的算法，它指引参数进行自我更新，使得更新参数后的模型的损失函数更小。

梯度下降是一种迭代优化算法，最初由柯西在 1847 年提出的。该方法主要是被人们用来寻找某个函数的局部最小值的一种有效方法。通过梯度下降算法，模型可以查找函数参数（系数）的值，这些参数将成本函数尽可能最小化。具体步骤如下所示。

算法 3.1（梯度下降算法）

设$\boldsymbol{\theta}$为模型参数（权重），$J(\boldsymbol{\theta})$时关于参数的代价函数，α为学习率，代表了能够在

代价函数下降最大的方向上做出更新时的步长。

则对于$j = 1,2,\dots,N$，重复以下步骤直至参数$\boldsymbol{\theta}$收敛：

1.计算梯度$\frac{\partial J}{\partial \theta}$

2.更新参数 $\theta_j \coloneqq \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\boldsymbol{\theta})$

除梯度下降之外，学者们通过研究提出了很多其他优化算法。在深度学习的发展中，人们经常使用的优化方法有：随机梯度下降(Bottou et al, 2012)、小批量梯度下降(Li et al, 2014)、NAG(Botev et al, 2016)、AdaGrad(John et al, 2011)、RMSProp(Hinton, 2012)、Adam(Diederik et al, 2014)等等。最常被使用到的算法是 SGD 和 Adam，但是由于神经网络的结构各不相同，而且选取什么样的优化算法很大程度上取决于模型所选取的损失函数，因此要考虑到多方面的因素进而最终决定优化方法的选取更大。

3.2.4 反向传播

早在 1970 年代，反向传播的概念就被提出，但是直到 16 年以后，它的重要性才得到充分认识。现今，神经网络训练是通过反向传播实现的。模型的误差值可以通过朝着某一方向使用反向传播算法而得到下降，进而使得模型的准确性得以提升。概括地来说，经过前向传播后，该算法根据权值和偏差进行后向传递，调整模型的参数。反向传播的算法如下所示。

算法 3.2(反向传播算法)

1. 输入x：设输入层相应的激活函数为a^1
2. 前向传播：对于$l = 2,3,\dots,L$计算$z^l = w^l a^{(l-1)} + b^l$以及$a^l = \sigma(z^l)$
3. 输出误差δ^L：计算向量$\delta^L = \nabla_a \mathrm{C} \odot \sigma'(\mathrm{z^L})$
4. 反向传播：对于$l = L-1, L-2, \dots, 2$计算$\delta^l = \left(\left(w^{(l+1)^T}\right)\sigma^{((l+1))}\right) \odot \sigma'\left(\mathrm{z^l}\right)$
5. 输出：代价函数的梯度为$\frac{\partial \mathrm{C}}{\partial \mathrm{w_{jk}^l}} = \mathrm{a_k^{l-1}}\delta_j^l$以及$\frac{\partial C}{\partial b_j^l} = \delta_j^l$

3.2.5 深度学习的一般步骤

由上述内容可知，深度学习是作为以神经网络和多层感知机为基本单位，由很多层的网络结构堆叠而成的一种特殊的模型。作为机器学习算法的一种，深度学习的一般步骤可概括为如下所示。

算法 3.3（深度学习的一般步骤）

输入：训练集 $T=\{(\boldsymbol{x_1},y_1),(\boldsymbol{x_2},y_2),\ldots,(\boldsymbol{x_n},y_n)\}$ 其中 $x\in X=R^n$，$y\in Y=\{y_1,y_2,\ldots,y_m\}$,$n=1,2,\ldots,N$;$m=1,2,\ldots,M$

输出：深度学习模型f

1. 初始化模型参数$w=w_0,b=b_0$

2. 定义损失函数L

3. 定义优化方法O

4. 重复以下步骤直至模型收敛：

(1) 根据 w,b 计算出预测值 $\hat{y}=f(x)=w*x+b$

(2) 根据 L 计算损失 $loss=L(y,\hat{y})$

(3) 根据优化器 O 及 $loss$ 更新模型 f

3.3 YOLOv5

3.3.1 一些目标检测算法

(1) R-CNN

在 2012 年，Krizhevsky 等使用 CNN 进行一般图像分类任务取得了可喜的结果。 2013 年，Girshick 等发表了一种方法，将这些结果推广到目标检测。这种方法称为 R-CNN(CNN with region proposals)。

R-CNN 正向计算有几个阶段，如图 3.2 所示。首先，生成感兴趣的区域。RoI 是与类别无关的边界框，很有可能包含一个感兴趣的对象。在本论文中，称为选择性搜索的单独方法用于生成这些方法，但也可以使用其他区域生成方法。接下来，使用卷积网络从每个区域建议中提取特征。边界框中包含的子图像将进行扭曲以匹配 CNN 的输入大小，然后馈送到网络。最后通过 SVM 进行分类。

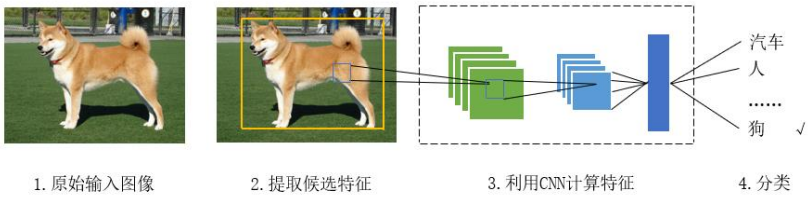


图 3.2 R-CNN 的结构示意图

（2）Fast R-CNN

2015 年，先前论文 R-CNN 的同一作者 Girshick 发布了一种更实用的对象识别方法，以构建更快的目标检测算法，并将其命名为 Fast R-CNN。 该方法类似于 R-CNN 算法。主要思想是对整个图像执行 CNN 的前向传递，而不是对每个 RoI 分别执行。

Fast R-CNN 的一般结构如图 3.3 所示。该方法接收图像以及从图像计算出的感兴趣区域作为输入，并通过使用 RoI Pooling 层固定特征图的大小，与 R-CNN 中一样，RoI 是使用外部方法生成的。

正如其名称所显示，Fast R-CNN 的运行速度比 R-CNN 快，其原因正是因为每个图像只进行一次卷积运算，并从中生成特征图。

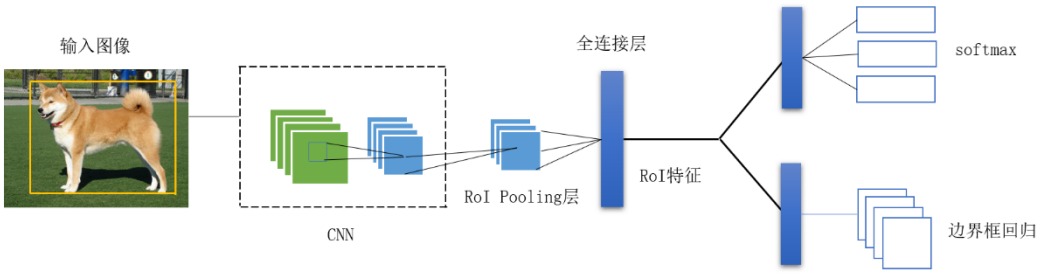


图 3.3 Fast-RCNN 结构示意图

（3）SSD

SSD(Single Shot MultiBox Detector)进一步提高了集成检测能力。该方法根本不会生成候选区域，也不会涉及图像片段的任何重采样。它使用一次卷积网络生成对象检测。该算法有点类似于滑动窗口方法，从一组默认的边界框开始。 这些包括不同的纵横比和比例。为这些框计算的对象预测包括偏移量参数，这些参数预测对象周围的正确边界框与默认框有多少不同。该算法使用不同的特征图（即较大和较小的特征图）作为分类器的输入来处理不同的比例。 由于该方法生成密集的边界框集，因此分类器之后是非最大抑制阶段，该阶段会消除低于某个置信度阈值的大多数框。

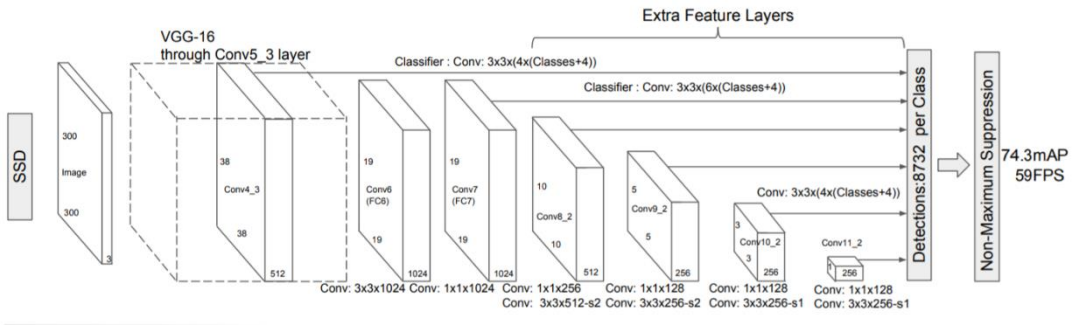


图 3.4　SSD 模型结构

3.3.2 YOLOv5

随着目标检测的发展，Joseph Redmon 在提出 YOLO 模型之后接着更新了 YOLOv2 和 YOLOv3，并成为了当时最新的目标检测方法。2020 年 6 月，来自 Ultralytics 团队的 Glenn Jocher 在 YOLOv3 的基础上提出 YOLOv5 模型。YOLOv5 的初始版本非常快速，高性能且易于使用。 尽管 YOLOv5 尚未对 YOLO 模型提出新颖的模型体系和结构改进，但是 YOLOv5 还是改善了目标检测方法的最新水平。并且它引入了新的 PyTorch（Paszke et al, 2019）训练和部署框架，使得自定义模型的训练变得更加方便。图 3.5 为 YOLOv5 的结构示意图。

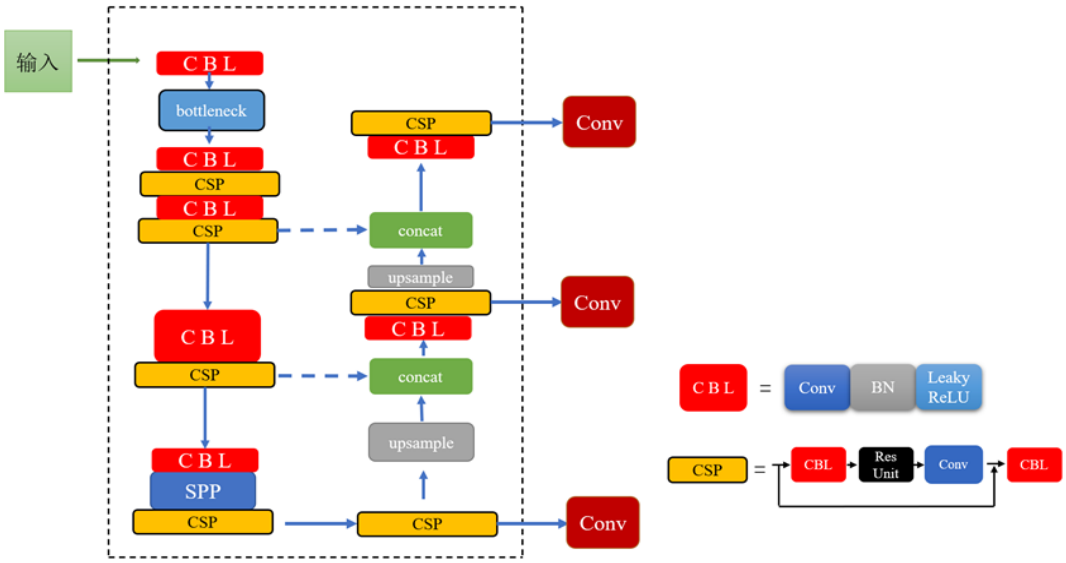


图 3.5　YOLOv5 结构示意图

YOLOv5 从输入、主干、输出等各个方位对原有网络做出了升级和更新，下面本论文主要从两个创新点来对原有网络进行改善：CSPNet（Wang et al, 2019）和 PANet（Wang et al, 2018）。

CSPNet(Cross Stage Partial Network)的作者于 2019 年基于 DenseNet 提出 CSPNet，其目的包括：（1）减小计算量（2）提升网络的计算能力（3）减小内存消耗。CSPNet 的网络结构的设计基于 DenseNet，先将输入分为两个部分，一部分按照常规的卷积网络进行计算，另一部分构建类似 DenseNet 的短路连接模块，这样做的好处是可以缓解梯度消失问题。最后通过 Partial Transition 层进行融合，通过不同的梯度流以避免梯度的重复计算。YOLOv5 的作者将 CSPNet 与 YOLOv3 的主干网络结合，形成了 CSPDarknet53 作为 YOLOv5 的主干网络。

一般的 CNN 中，在图像不断地通过卷积层时，特征的复杂度增大，同时图像空间的分辨率降低，因此无法准确识别高级特征。PANet(Path Aggregation Network)在 FPN 的基础上，使用自底向上的结构，并且使用底层到顶层的横向连接，有效缩短路程。

3.4 目标检测评价指标

3.4.1 精确率与召回率

精确率(Precision)与召回率(Recall)是一对机器学习中用来衡量分类器精确程度的度量。在二分类问题中，常用混淆矩阵(Confusion Matrix)来表示样本预测值的正负与样本真实值的正负之间的关系。表 3.4 表示了一个混淆矩阵。

表 3.4 混淆矩阵

		预测值	
		正	负
真实值	正	TP（将正样本预测为正）	FN（将正样本预测为负）
	负	FP（将负样本预测为正）	TN（将负样本预测为负）

精确率和召回率的定义如下：

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

在实际情况中，往往需要把二者结合在一起，即 P-R 曲线。

3.4.2 IoU

交并比(IoU)常常被用来比较两个边框之间的重叠程度。在目标检测领域中，IoU 所表达的含义为真实框(Ground-Truth Box)和预测框(Predicted Box)之间的重叠量。在某些情况下，我们使用 IoU 时预定义了它的阈值（如 0.5）。具体如图 3.6 所示。

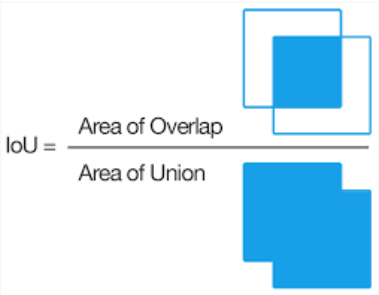


图 3.6 IoU 示例

3.4.3 平均精度

平均精度(Average Precision)是目标检测领域中最常用的评价模型性能的指标，它的取值范围是 0 到 1。下面具体解释 AP 的计算方法。

目标检测结果常常会预设一个阈值（如 0.5），在该阈值下，依次求得所有预测框和真实框的 IoU，并将这些预测结果按 IoU 的大小从大到小进行排序。将分类正确且 IoU 大于阈值的样本记为 TP，同理求出 FP、FN、TN 并根据所求 TP、FP、FN、TN 即可求出在该阈值下的精确率和召回率。

改变阈值（如改成 0.6），重复上述步骤即可求出在另一阈值下的精确率和召回率，并可画出 P-R 曲线。平均精度即为 P-R 曲线下方的面积。通常情况下，取 0，0.1，0.2，…，0.9，1.0 这 11 个值作为 IoU 预定义阈值，求出每个阈值下的 P 和 R，根据如下公式求出 AP。

$$AP = \frac{1}{11} \sum_{r = \{0.0,\ldots,1.0\}} AP_r$$

$$= \frac{1}{11} \sum_{r=0.0,\ldots,1.0} \max_{\tilde{r} \geq r} p(\tilde{r})$$

mAP(Mean Average Precision)是 AP 的平均值。通常情况下，当数据集仅有

一类样本时，mAP 与 AP 的含义相同；当数据集含有多类样本时，mAP 即为每个样本 AP 的平均值。但在一些文章中也有作者将 AP 写作 mAP，用 mmAP 表示 AP 在每一个类别上的平均值。本论文统一使用 mAP 作为目标检测的评价指标。

第四章　YOLOv5 算法多阶段改进

目标检测主要用于描述和检测图像中的物体相关的计算机视觉任务。经过数十年的发展，目标检测由传统的数字图像处理方法转变为基于深度学习的方法。近年来研究者们对于目标检测的算法大致从以下几个方面入手：第一，从网络结构自身，如 SSD 选取效率更高的 Resnet 作为主干网络，YOLOv4 和 YOLOv5 通过向主干网络中加入 CSP 模块提升模型性能等。第二，从训练数据集上，例如在 YOLOv5 模型中，作者通过 Mosaic 方法进行数据增强。第三，从一些小技巧上，如交叉验证、非极大抑制、TTA 等。

本论文的研究内容分别在锚框生成阶段、YOLOv5 的主干网络阶段和筛选预测边框阶段对原有模型做动态锚框、注意力机制和目标框加权融合三个方面的改进。

因此，本章的结构也大致分为以下三个部分。首先，在锚框生成阶段提出向网络结构中加入动态锚框机制，通过将锚框的选择并入网络训练可动态地更新锚框的大小、位置、宽高比等信息，使得模型能够以获得更加合理的先验信息。其次，在 YOLOv5 的主干网络中加入视觉注意力机制。原有网络结构中并未添加注意力模块，导致网络不能很好的将注意力集中在重要特征上。本论文通过加入空间注意力和通道注意力，使得模型能够更有效率地进行重要特征的挖掘。最后，在预测回归框筛选阶段使用目标框加权融合(WBF)方法进行选择，以牺牲一定运行速度作为代价得到模型检测精度的提升。

同时，在每个小节的末尾都使用 Pascal VOC 数据集对改进前后的模型做对照实验，并做出分析。

本论文所提出的改进模型的大致结构如图 4.1 所示。

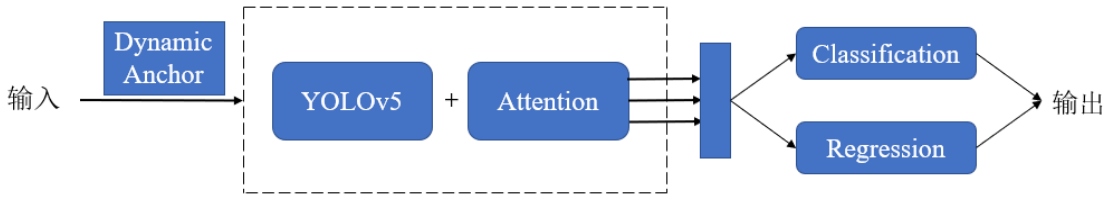


图 4.1　本论文模型结构示意图

4.1 输入阶段改进：基于动态锚框

4.1.1 问题的提出

单阶段目标检测中一般会设置锚框(Anchor Box)来达到两阶段目标检测中选择侯选区域的作用。而锚框的生成大致可分为：人为经验设定、通过聚类得到和通过训练学习三种方法，具体见第二章第二节。

YOLO 使用 K-Means 聚类生成锚框，以训练数据集的边界框作为基准，通过 FPN 网络在三种不同大小的特征图下分别设定三种锚框。在 YOLOv5 中作者仍然使用该方法，然而使用 K-Means 方法生成的锚框具有以下几点问题：第一，K-Means 算法自身具有一定的局限性。K-Means 算法容易受到初始设定值和离群点的影响，会引起聚类结果的不稳定。第二，由于三种不同大小的锚框是人为设定的，而现实情况下所要检测的目标并不会按照这三种大小均匀分布，就会导致一定程度上的错误，例如将原本是小物体的目标分入检测中物体的锚框中，或是将原本是中物体的目标分入检测大物体的锚框中。由此会导致在之后的边界框回归训练过程中具有较大的损失，增大模型优化的难度。第三，生成锚框的阶段在训练之前，而在训练阶段将锚框视作常量。假如锚框的设置不是很合理，则在训练阶段会增加模型损失和收敛难度。

4.1.2 模型构建

针对上一小节所提出的问题，本论文在构建模型的锚框生成阶段提出加入动态锚框的概念。通过在 YOLOv5 模型的基础上加入 DAFS 结构来弥补上一小节中所提到的缺点。

DAFS 模型(Dynamic Anchor Feature Selection)的概念由 Li et al 在 2019 年所提出，该模型在 ARM 模块的基础上提出。作者指出在 RefineDet 中，对于任意一个特征图上的点，使用 ARM 模块优化过的锚框作为 ODM 的输入会导致该点的感受野和锚框的不匹配问题，反而在一些情况下会降低模型的检测能力。因此作者提出在网络的检测部分(ODM)上根据细化过的锚框的形状大小动态地调整特征图上的点，以减小这种不匹配的问题。同时，作者还提出用双向特征融合(Bidirectional Feature Fusion, BFF)代替 TCB。BFF 能融合自顶向下和自底向上的双向路径，能够使每层接受到来自上下不同层之间的信息。DAFS 模型如图 4.2 所示。

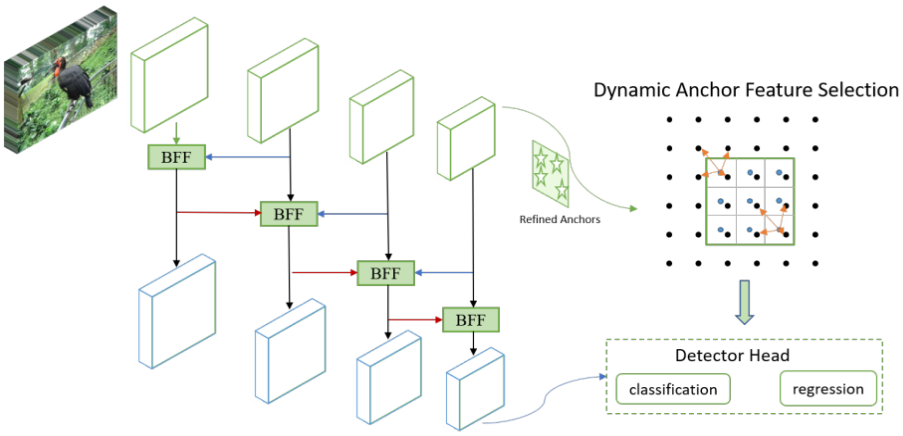


图 4.2 DAFS 模型结构

本论文在 YOLOv5 的基础上加入动态锚框的概念。首先和原方法一样，通过 K-Means 算法在三个不同的尺度上聚类生成九个锚框作为模型的初始锚框。然后在模型中添加 ARM 模块，然后通过 BFF 将 ARM 模块和 YOLOv5 主干网络所连接，最后得出动态的细化锚框作为模型训练先验框。具体模型结构如图 4.3 所示。

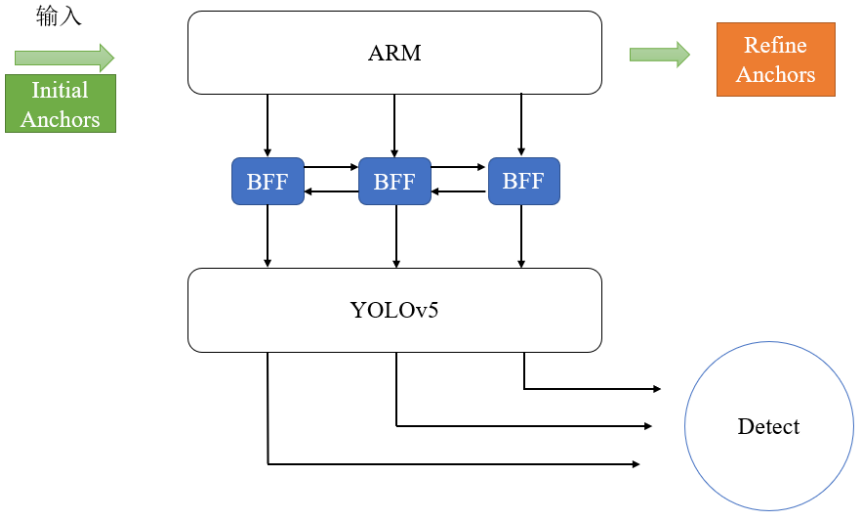


图 4.3 本论文模型结构(YOLOv5 + DA)

由图 4.3 可以看出，模型首先采用 YOLOv5 中使用的初始锚框，然后通过 ARM 模块过滤掉初始锚框的负样本（在 RefineDet 中，一个锚框被定义为负样本当且仅当该锚框和真实框的 IoU 小于 0.5，在本论文中仍然应用这个定义）以及根据真实值对初始锚框进行位置上的微调，以便作为网络的先验信息。然后通过 BFF 模块连接至 YOLOv5 主干网络中，对特征图上对应点做出微调。由于 BFF 是双向连接模块且连接了不同大小的相邻特征图，因此在下一次迭代中 ARM 又

可以根据特征图的更新对锚框做进一步更新。最后，网络在三个不同尺度上做预测再融合，得到最终的结果。

4.1.3 对比实验

本小节使用上面提到的 YOLOv5+DA 的模型和原模型做对照实验。数据集使用 Pascal VOC 数据集。

Pascal VOC 数据集来自于 Pascal VOC 挑战赛，本论文只关注检测任务。数据集共含 20 个小类。

本论文使用 VOC07+12 作为训练集， VOC07 检测集。实验中使用的评价指标为 mAP。具体实验结果如下所示。

表 4.1 实验结果

方法	mAP	运行时间(ms)
YOLOv5	83.4%	30
YOLOv5+DA	**84.5%**	41

由表 4.1 可以看出，在 VOC07 的测试数据集上，“YOLOv5 + 动态锚框”的方法较原有方法在 mAP 上增加了 1.1%，主要原因可以认为市本论文所提出的“YOLOv5 + 动态锚框”的方法改善了锚框的生成，获得了更加有效的先验信息，从而在该数据集上有着更好的检测效果。同时，该方法比原始 YOLOv5 方法在检测一张图片所消耗的时间上慢了 11ms，由此可见本论文所提出的改进方法更适合于使用在对检测速度要求不太高的离线系统上。

图 4.4 为实验的可视化结果。总体上来看，无论是改进前的 YOLOv5 算法还是改进后的加入动态锚框的算法，其平均精度都达到了 80%以上，可以认为两种方法都基本符合现实中的需求。但是根据图 4.4 的测试结果来看，还是可以看出原算法 YOLOv5 会有一些漏检及误检情况的出现，而改进之后的 YOLOv5+DA 模型在一定程度上缓解了漏检误检的情况，从这个角度也可以认为模型的改进起到了一定的效果。

图 4.4 实验可视化结果：（左）YOLOv5 + DA （右）YOLOv5

由图 4.4 中可以看出，第一行的图片中原算法存在误检情况，第二行的图片中原算法存在误检和漏检的情况，第三行的图片中原方法存在漏检的情况。改进后的算法中误检及漏检的情况得以解决。

4.2 骨干网络改进：基于注意力机制

注意力机制可以说是目前深度学习领域最为强大的概念之一，它源自于一种人类的直觉——人们在处理大量信息时，往往会对某个部分更加“注意”。

给定一幅图像，因为我们看到了狗的鼻子，右边的耳朵，以及眼睛（红色方框中的东西），所以我们希望看到尖尖的耳朵（黄色方框中的东西）。但是，类似围巾、毛衣和雪地等背景并不会像其他特征那样引人关注。

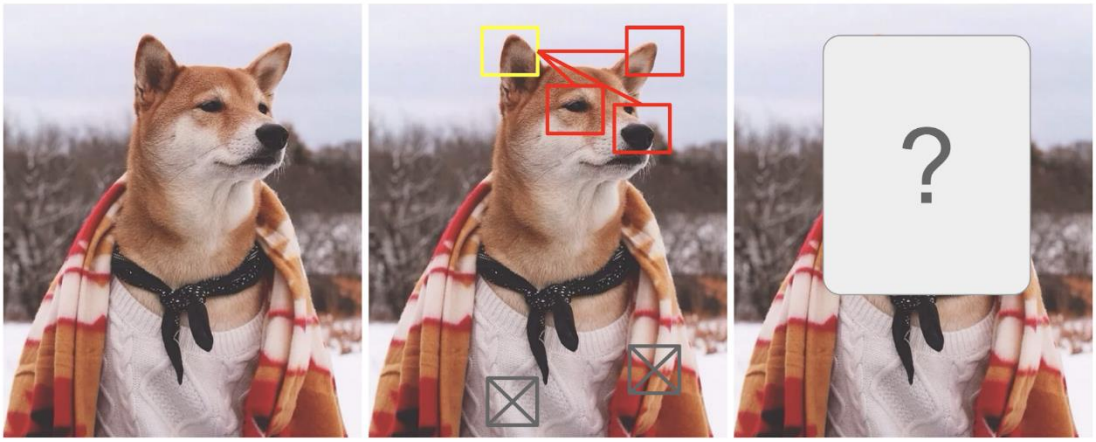


图 4.5 人的视觉注意力机制

4.2.1 SENet

SENet 是 Hu et al 于 2017 年所研究的将通道注意力和空间注意力所融合而得到的网络结构，其具体结构如图 4.6 所示。

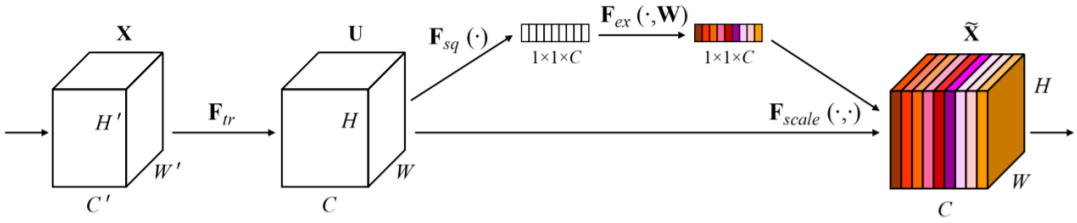


图 4.6 SE 模块的结构

在 SENet 中，其作者通过提出挤压和激励两个步骤来处理全局信息。其具体操作如下。

$$z_c = \mathbf{F}_{sq}(\mathbf{u}_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j).$$

上式表示了 Squeeze 的操作，其中z是由U朝着$H * W$的空间维度压缩而生成，z_c表示z的第c个元素，H，W代表高宽，$u_c(i,j)$代表u的第c个通道的第$(i，j)$个元素。在 SE 模块中，作者使用了最为简单的聚合方法：全局平均池化来表达全局特征。

Excitation 的操作如下所示。

$$\mathbf{s} = \mathbf{F}_{ex}(\mathbf{z}, \mathbf{W}) = \sigma(g(\mathbf{z}, \mathbf{W})) = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{z})),$$

$$\widetilde{\mathbf{x}}_c = \mathbf{F}_{scale}(\mathbf{u}_c, s_c) = s_c\,\mathbf{u}_c,$$

最终的输出加上了尺度(Scale)操作，本质是在不同的通道上赋予不同的权重，来加强重要的通道抑制不重要的通道。

4.2.2 模型构建

研究表明，通过向卷积神经网络中添加视觉注意力机制，能够使得网络自身有选择性地忽略一些无关信息从而提升网络的总体表现。因此本小节提出将注意力机制应用于 YOLOv5 主干网络之中。

在 SENet 中，Hu et al 提出将 SENet 应用于 Inception 和 ResNet 中，均取得了不错的效果。

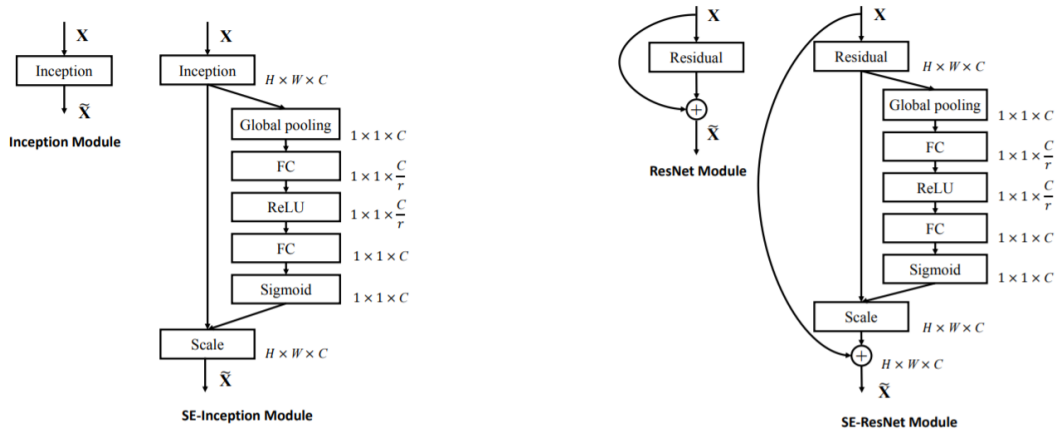


图 4.7 SE-Inception 模块和 SE-ResNet 模块

Hu et al 在 SE 模块中使用 GAP 作为全局池化操作。GAP 往往倾向于识别目标的大致范围，而 GMP 则可以通过识别全局最大点来指示检测目标的特征信息，特别是目标较小且在前向传播中特征图比例相较于空间维度大幅缩小时，GMP 往往更加有效。不同于 SE 模块中作者在通道注意力模块中仅考虑 GAP，本论文考虑在通道注意力模块中同时使用 GAP 和 GMP。具体结构如图 4.8 所示。

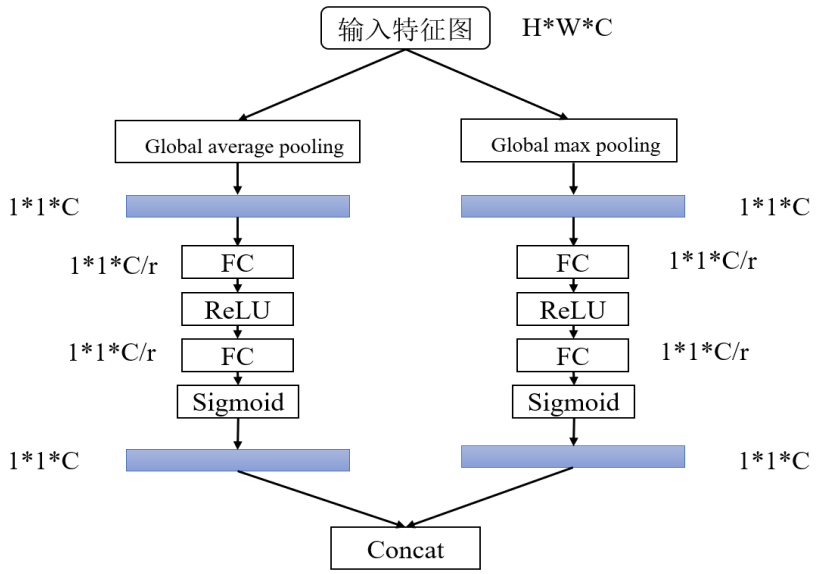


图 4.8 本论文使用的通道注意力模块

本论文在 YOLOv5 的网络结构的较底层添加混合通道注意力和空间注意力的注意力模块。具体结构如图 4.9 所示。

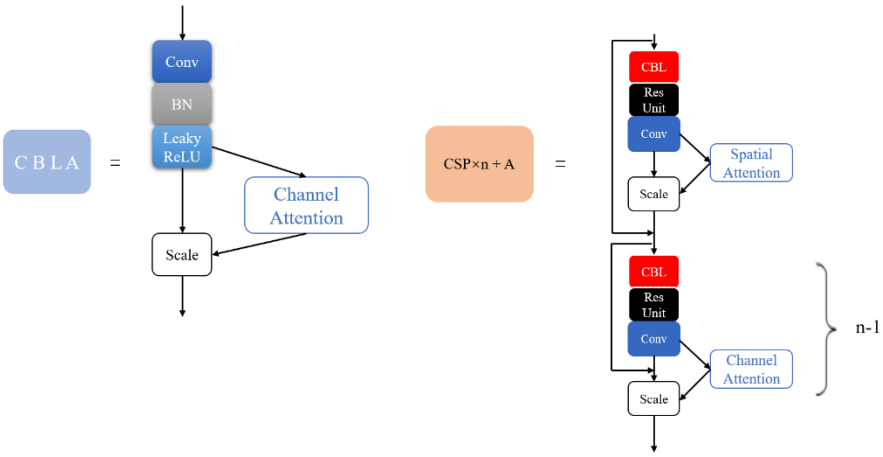


图 4.9 本论文使用的两个注意力模块

综合上述内容，本论文所使用的添加注意力机制的整体网络结构如图 4.10 所示。

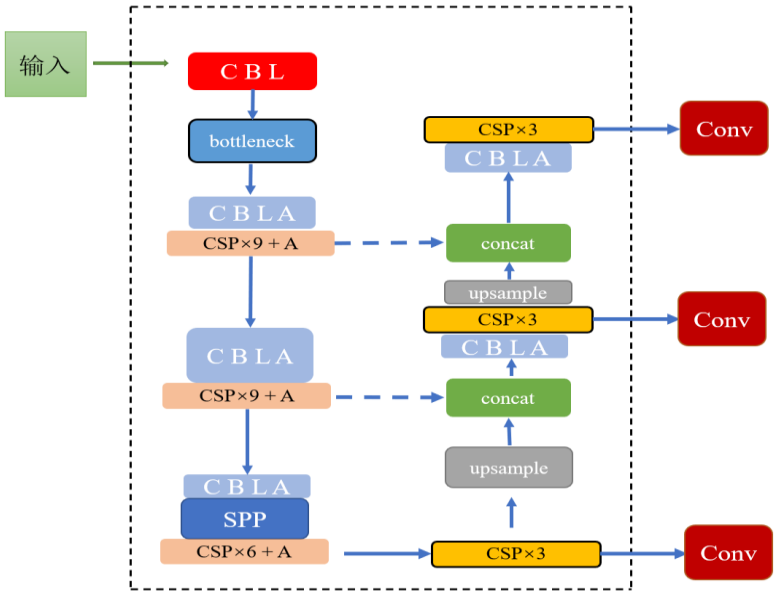


图 4.10 YOLOv5 + Attention 结构示意图

由图 4.10 所示，整体的网络结构遵循了 YOLOv5 的结构，用 CSPdarknet53 作为模型的主干网络进行训练。如上文所述，通过在“Conv+BN+LeakyReLU”模块之后添加通道注意力，在三个不同尺度的特征图经过卷积操作之后加入“CSP×n+A”向模型中添加混合注意力。通过两种注意力的混合使用提高模型对重要特征的提取能力，从而获得更好的效果。

4.2.3 对比实验

同上一小节一样，本小节将添加注意力机制的 YOLOv5 模型和原有模型在 Pascal VOC 数据集上做对照实验，同时选取 mAP 作为评价指标。

在训练阶段，使用 VOC12+VOC07 训练集训练网络模型，batch_size 设为 32，选用 SGD + Momentum 优化方法进行训练，在测试阶段，使用 VOC07 test 作为测试集。表 4.2 为实验结果。

表 4.2 实验结果

方法	mAP	运行时间(ms)
YOLOv5	84.3%	26
YOLOv5+Attention	**86.2%**	34

如表 4.2 所示，在添加了注意力模块之后的 YOLOv5 模型相较于原始的

YOLOv5 模型就检测能力而言有了一定的提升。从表中可以看出添加注意力机制后，对图片的检测时间增加了 8ms，而 mAP 增加了 1.9%。相较于模型性能的提升，运行速度的变慢在可以接受的范围之内。图 4.11 为对比实验的可视化结果，图中左侧部分为添加注意力机制模型的检测结果，右侧部分为原模型检测结果。本论文选取了若干组原算法存在误检漏检的对比图片，从图中大致可以看出改进后模型的检测精确度要好于改进前的模型。

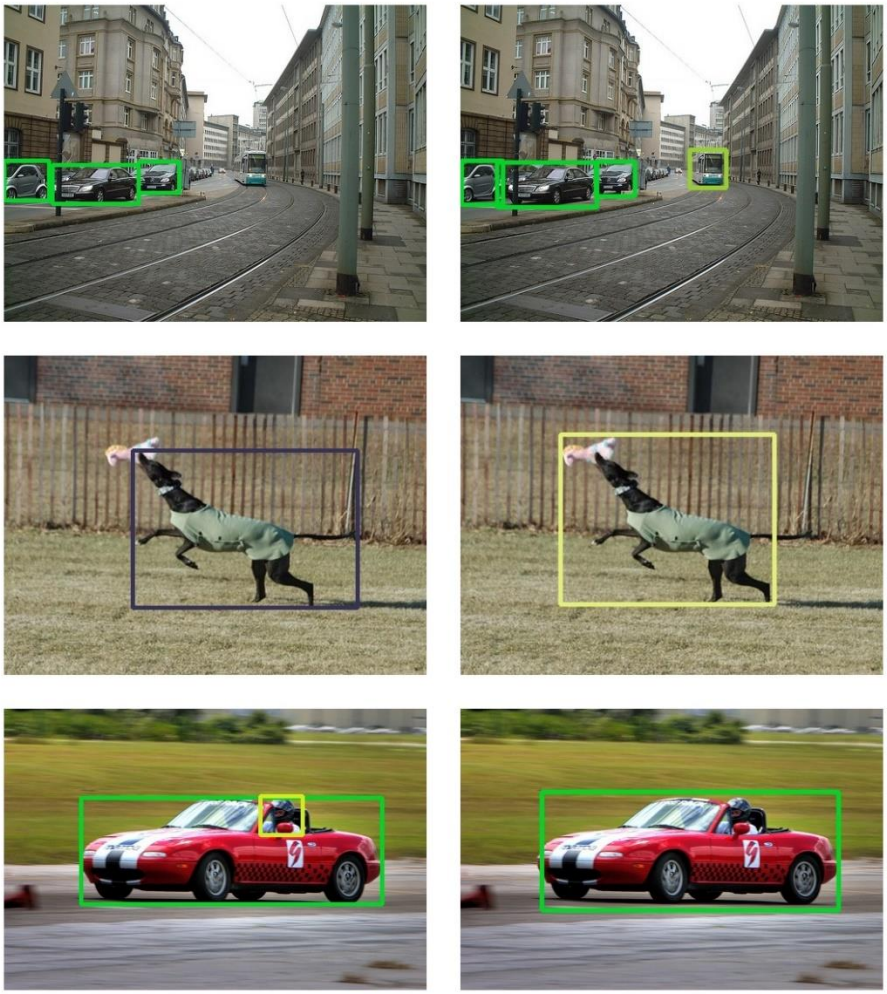


图 4.11 实验可视化结果：（左）YOLOv5 + Attention （右）YOLOv5

由图 4.11 中可以看出，第一行图片中原算法出现误检，将 bus 误分类为 car；第二行图片中原算法出现误检，将 dog 误分类为 bird；第三行图片中出现漏检现象。改进后的算法有效地解决了误检以及漏检的一些情况，可认为将视觉注意力机制加入算法中的改进达到了一定的效果。

4.3 回归框筛选阶段改进：基于目标框加权融合

4.3.1 问题的提出

非极大抑制(NMS)和目标框加权融合(WBF)为目标检测中选择预测值的常用方法，具体见第二章第四节。

在 YOLOv5 网络结构中，Glenn Jocher 使用非极大抑制(NMS)作为选择最终预测的边界框的方法。该方法具有以下两个缺点：

第一，NMS 仅保留得分最大的预测边界框而丢弃得分小的预测边界框，但得分较小的边界框也同样包含一定的特征信息，直接丢弃相当于没有完全利用全部信息。第二，在一些情况下得分最高的预测边界框也不能很好的拟合真实的目标框，直接选用该预测框作为最终预测值具有较高的损失。

4.3.2 模型构建

基于上述两点，本论文提出使用 WBF 代替 NMS 作为边界框回归中选择合适预测值的方法。具体网络结构如图 4.12 所示。

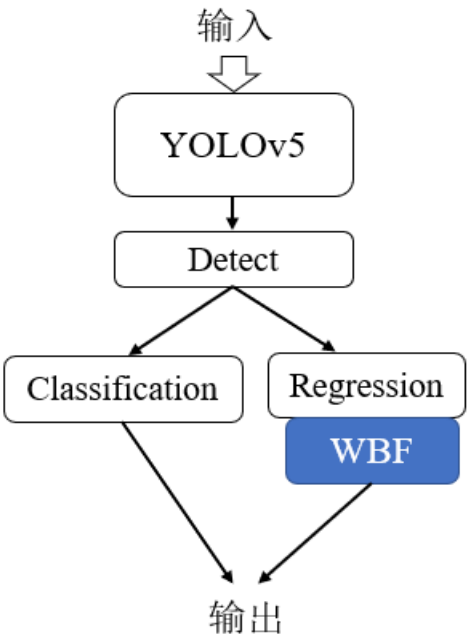


图 4.12 YOLOv5+WBF 网络结构示意图

为做对比实验，本小节仅将 YOLOv5 模型中的 NMS 替换为 WBF，保持其他结构的一致，实验结果见下一小节。

4.3.3 对比实验

本章仍用 VOC07+12 作为训练集，使用 VOC07 作为测试集做对照实验。表 4.3 为实验结果。

表 4.3 实验结果

方法	mAP	运行时间(ms)
YOLOv5	83.6%	23
YOLOv5+WBF	**86.9%**	71

如表 4.3 所示，仅仅将 NMS 替换为 WBF 可以使网络的 mAP 提升 3 个百分点。因此可以证明 WBF 为一个有效的提升准确率的方法。

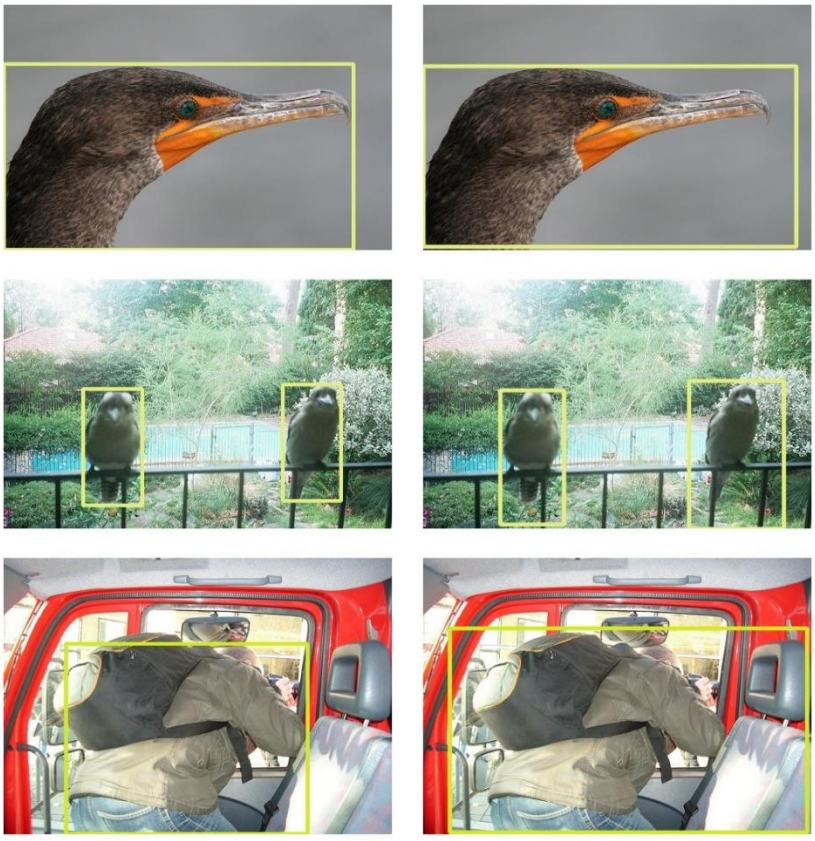


图 4.13 实验可视化结果：（左）YOLOv5+WBF （右）YOLOv5+NMS

图 4.13 为实验的可视化结果，本论文在回归框筛选阶段仅将 NMS 替换为 WBF，而保持网络中其他结构的不变，因此由图 4.13 中可以看出改进前后模型的区别仅体现在回归框的不同上，可以比较明显的看出改进后的模型的回归框更加贴合真实情况，因此通过加入 WBF 的改进可以使得模型获得更高的 mAP 值。

第五章 实验结果与分析

上一章分别从输入、骨干网络和回归框筛选三个阶段对原有 YOLOv5 模型做了不同的改进，并且通过三个实验分别证明了其有效性。本章研究的主要内容就是将这三个阶段的改进融合成一个整体的模型，通过实验去验证其是否仍然有效。本章先介绍实验所使用的数据集，然后将改进的模型和目标检测的几个经典模型做对比实验，最后根据实验结果做一些简单的分析。

5.1 数据集介绍

为验证改进后的模型的检测效果，本章使用 Udacity 自动驾驶数据集训练模型，并在测试集上验证结果。

Udacity 自动驾驶数据集是一个开源数据集。该数据集取自自动驾驶车辆的视角，图像都是在行驶的车辆上拍摄的视频截取而成，并且进行 2D 标注。数据集一共有 15000 张图像，主要分为行人、汽车、自行车、信号灯等 5 个类别。图像的大小都是 1920*1200，为方便训练和测试，本论文统一将图像大小放缩为 960*600。

5.2 对比实验结果

本论文实验环境如表 5.1 所示。

表 5.1 实验环境配置

操作系统	Ubuntu 16.04.7 LTS
处理器	Intel Core i7-9700
内存	16GB
GPU	NVIDIA GeForce GTX 1080Ti
显存	11GB
程序	Python 3.7
深度学习框架	PyTorch 1.5

本论文使用 PyTorch 框架进行训练。将标注转化为<classes, x, y, w, h>格式，使用 SGD + Momentum 优化方法，Momentum 设置为 0.9，学习率为 0.01，batch_size

为 64，训练 100 个 epochs。本论文为研究 YOLOv5+DA+Attention+WBF 的结果，分别与 Faster R-CNN、SSD 和 YOLOv5 进行对照，并得出实验结果。表 5.2 和图 5.3 分别显示了实验的结果。

表 5.2 实验结果

模型	主干网络	mAP	运行时间(ms)
Faster R-CNN	Resnet101	83.7%	101
SSD	Vgg16	75.6%	69
Yolov5	CSPDarknet53	87.4%	43
Yolov5+wbf	CSPDarknet53	89.8%	106
Yolov5+DA	CSPDarknet53	88.3%	51
Yolov5+Attention	CSPDarknet53	86.5%	54
YOLOv5+DA+A+WBF	CSPDarknet53	**90.5%**	117

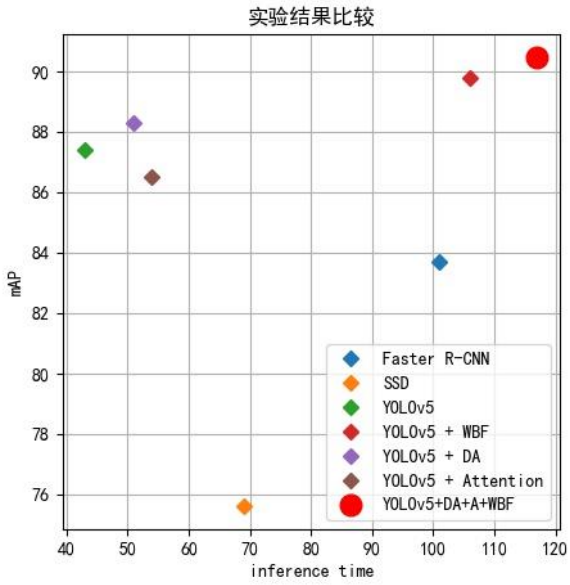
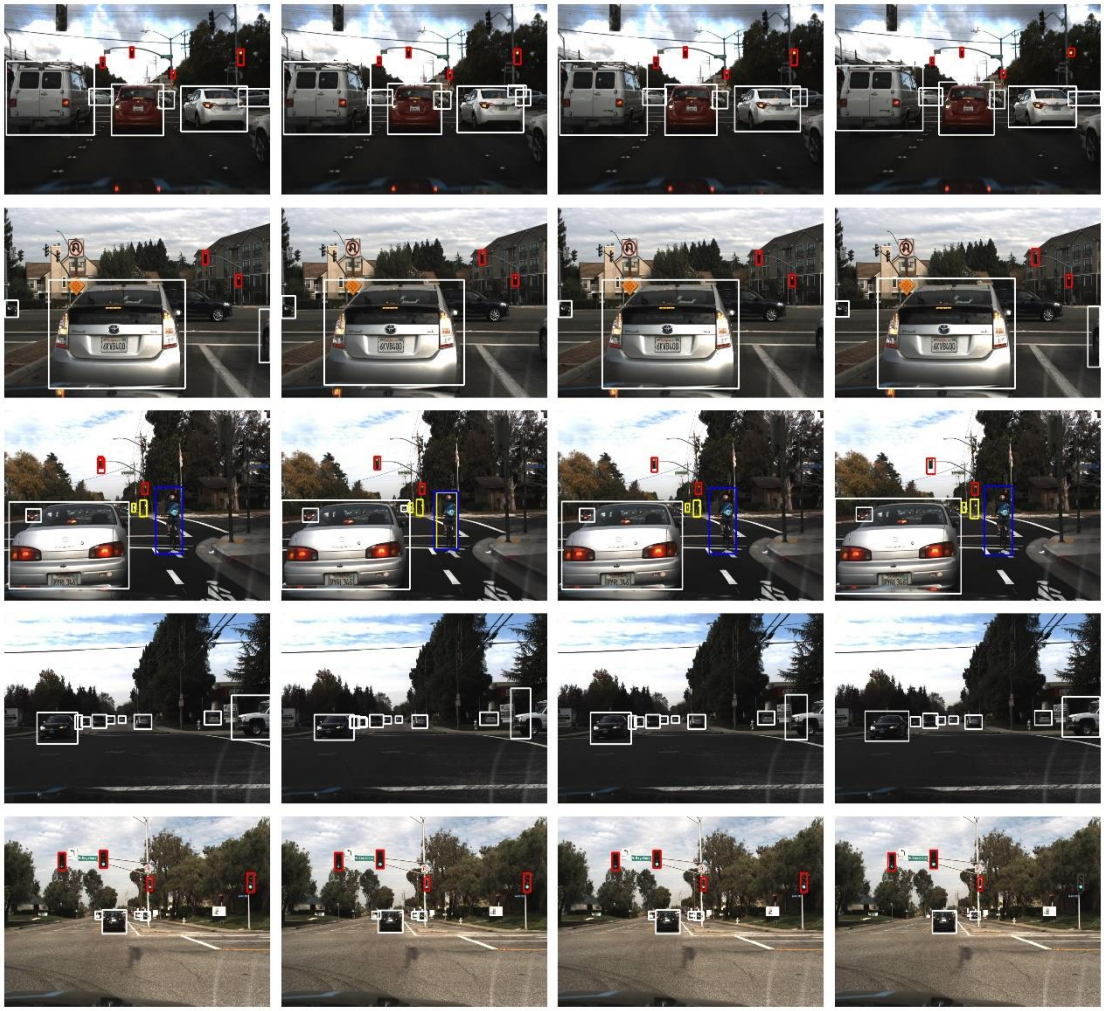


图 5.3 实验结果比较图

5.3 实验结果分析

由表 5.2 的结果可知，“YOLOv5+DA+Attention+WBF”算法在 Udacity 数据集的测试集上的 mAP 高达 90.5%，相比于未改进之前的 YOLOv5 算法的 87.4%高出了 3.1%。图 5.3 显示了各个算法之间的 mAP 和运行时间的二维信息，从图中可以看出本论文所改进的算法有着精度上的优势，尽管运行时间较慢，但也在可接受的范围内。同样地，可以看出本论文提出的算法得到的预测值和真实值之间的差距很小。这说明了本论文所提出的改进方法完全可以应用于现实的场景中去，并且比原有的 YOLOv5 方法准确度更高。



(a)FastRCNN (b)SSD (c)YOLOv5 (d)论文改进算法

图 5.4 本论文提出的方法与传统目标检测算法效果比较

图 5.4 为本论文提出的算法与传统的几种目标检测算法的效果对比图，可以看出，在第一行、第三行和第四行的图片中，存在许多分辨率小的物体，无论是 FastRCNN、SSD 还是未改进的 YOLOv5 算法，均存在一定程度上的误检和漏检，

如未能正确的检测出自行车和行人、存在对车辆的误检等等；而本论文所提出的算法由于在输入、网络和边框筛选三个阶段分别提出了改进，进而使模型具有更加强大的检测能力，因此在图中能够得到更加准确的检测结果。在第二行和第五行的检测结果中，由于物体相对较少，场景相对单一，各个算法均能达到较为准确结果。因此由图 5.4 可以看出本论文所提出的算法对于复杂场景下的目标检测具有较好的应对能力。本论文在 YOLOv5 的基础上改进了算法的多个阶段，相较于其他几种算法具有较高的准确度，并且预测值和真实值之间几乎没有肉眼可见的偏差，这说明将本论文所改进的算法运用到实际的场景之中将会达到一个不错的效果。综上所述，相较于目前流行的目标检测算法，本论文提出的算法在性能上有着一定的优势，同时也可以较为方便地运用到真实的情况之下，使现实生活中的目标检测更为智能化。

第六章 结论与展望

6.1 主要结论

本论文提出了一种基于 YOLOv5 的新型网络结构，该方法较原始 YOLOv5 而言有了一定的提升。本论文主要从三个不同的角度上对原有模型进行优化。

第一个角度是对锚框的优化。目标检测本质上是一个分类加回归的多任务学习。无论是两阶段目标检测任务还是单阶段目标检测任务，都需要做边界框回归。而边界框回归是基于锚框（又称先验框）做位置和形状上的微调。因此锚框（即先验框）的选取成为了影响检测结果的很重要的一个因素。原有 YOLOv5 结构生成锚框之后再训练过程中将其固定，本论文借鉴 RefineDet 和 DAFS 的思想，不再将锚框作为固定值，而是使其参与训练，得到更为准确的结果。

第二个角度是注意力机制的添加。添加注意力的主要目的是为了在有限的资源中，将资源分配给更重要的任务。在卷积神经网络中，若原始模型未添加注意力机制，很可能导致在网络的很多中间层中不能有效地抓住主要特征。本论文结合 SENet 和 CBAM 的思想，向网络中同时加入通道注意力和空间注意力，有效的增强了网络提取主要特征的能力。

第三个角度是边界框回归时对边界框的选取。原文使用 NMS 方法保留得分最高的预测框丢弃其他的预测框，这种做法是有缺陷的：第一得分最高的预测值并不一定能代表所有的预测值，第二直接丢弃剩余预测框相当于丢弃了一部分有效信息，未能利用全部有效信息。本论文提出使用 WBF 方法代替 NMS，通过加权融合的方法生成最终的预测值，弥补了 NMS 方法做造成的缺陷，提升了模型最终的性能。

经过三个不同阶段的改进，模型在检测精度上取得了一定程度的提升，并且通过在自动驾驶数据集上的实验验证了改进后的模型在现实的场景下也有一定价值。通过融合三个阶段的改进，能够使模型达到最好的效果。

6.2 研究展望

目标检测技术发展到现在已越来越完善，如今基于目标检测的更多问题，如 3D 目标检测、物体分割等，正被大家所关注。本论文基于当下比较流行的目标检测算法 YOLOv5 出发，分别讨论了添加动态锚框模块和注意力机制之后模型性能的改变。通过实验结果可以证明，本论文所提出的改进后模型比原有模型在

mAP 上提升了 3.1%，但仍然存在着诸多问题需进一步研究。综合全文，未来的研究还可以从以下两个方向上去着手：

(1)从网络本身出发，寻找更好的结构。本论文提出的方法的主干网络是 Darknet 结构，该结构中使用大量“Conv + BN + Leaky ReLU”的结构。未来的工作可以将注意力放在优化自身结构上，如卷积层使用空洞卷积、膨胀卷积等。

(2)从模型简化、提升运行速度的角度出发。现如今深度学习中许多的方法较之前的方法做出精度上的提升都是以运行速度作为代价，本论文所提出的方法也不例外。而现今由于移动端和嵌入式端设备的需求，模型的简化或将成为重点。未来研究可朝着该方向发展，如减少网络层数，提出类似 MobileNet(Sandler et al, 2018)的结构等。

参考文献

[1] Alex Krizhevsky, Ilya Sutskever, Geoffrey Hinton. ImageNet Classification with Deep Convolutional Neural Networks[J]. Advances in neural information processing systems, 2012, 25: 1097-1105.

[2] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation[C]. IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA, 2014.

[3] Ross Girshick. Fast R-CNN[C]. IEEE International Conference on Computer Vision (ICCV). Santiago, Chile, 2015.

[4] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, 39(6): 1137-1149.

[5] Wei Liu, Dragomir Anguelov, Dumitru Erhan, et al. SSD: Single Shot MultiBox Detector[C]. ECCV: European Conference on Computer Vision. Amsterdam, The Netherlands, 2016.

[6] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA, 2016.

[7] Tony Lindeberg. Scale Invariant Feature Transform[C]. Fifth International Conference on Signal and Image Processing. Bangalore, India, 2014.

[8] Paul Viola, Michael Jones. Rapid Object Detection using a Boosted Cascade of Simple Features[C]. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR. Kauai, HI, USA, 2001.

[9] Navneet Dalal, Bill Triggs. Histograms of Oriented Gradients for Human Detection[C]. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). San Diego, CA, USA, 2005.

[10] Ross Girshick, Forrest Iandola, Trevor Darrell, Jitendra Malik. Deformable Part Models are Convolutional Neural Networks[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA, 2015.

[11] Corinna Cortes, Vladimir Vapnik. Support-Vector Networks[J]. Machine Learning, 1995, 20(3): 273-297.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2014, 37(9): 1904-16.

[13] Joseph Redmon, Ali Farhadi. YOLO9000: Better, Faster, Stronger[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, USA, 2017.

[14] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollár. Focal Loss for Dense Object Detection[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2018, 42(2): 318-327.

[15] Tsung-Yi Lin, Michael Maire, Serge Belongie, et al. Microsoft COCO: Common Objects in Context[C]. ECCV: European Conference on Computer Vision, Zurich, Switzerland, 2014.

[16] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, Enhua Wu. Squeeze-and-Excitation Networks[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA, 2017.

[17] Jan Hosang, Rodrigo Benenson, Bernt Schiele. Learning non-maximum suppression[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, USA, 2017.

[18] Roman Solovyev, Weimin Wang, Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models[J]. Image and Vision Computing, 2019, 107:104-117.

[19] MacQueen J. Some methods for classification and analysis of multivariate observations[C]. 5-th Berkeley Symposium on Mathematical Statistics and Probability. Los Angeles, Ca, 1967.

[20] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, Stan Z. Li. Single-Shot Refinement Neural Network for Object Detection[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA, 2018.

[21] Yoshua Bengio, Aaron Courville, Pascal Vincent. Representation Learning: A Review and New Perspectives[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2013, 35(8): 1798-1828.

[22] Juergen Schmidhuber. Deep Learning in Neural Networks: An Overview[J]. Neural Netw, 2015, 61:85-117.

[23] Junyan Hu, Hanlin Niu, Joaquin Carrasco, et al. Voronoi-Based Multi-Robot Autonomous Exploration in Unknown Environments via Deep Reinforcement Learning[J]. IEEE Transactions on Vehicular Technology, 2020, 69(12): 14413 - 14423.

[24] Yann LeCun, Yoshua Bengio, Geoffrey Hinton. Deep learning[J]. Nature, 2015,521: 436-444.

[25] Haskell B. Curry, The method of steepest descent for nonlinear minimization problems[J]. Quart.appl.math, 1994, 2(3): 258-261..

[26] Aleksandar Botev, Guy Lever, David Barber. Nesterov's Accelerated Gradient and Momentum as approximations to Regularised Update Descent[C]. International Joint Conference on Neural Networks (IJCNN). Anchorage, AK, USA, 2017.

[27] John Duchi, Elad Hazan, Yoram Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization[J]. Journal of Machine Learning Research, 2011, 12(61): 2121−2159.

[28] Diederik P. Kingma, Jimmy Ba. Adam: A Method for Stochastic Optimization[C].3rd International Conference on Learning Representations (ICLR). San Diego, CA, USA, 2015.

[29] David E. Rumelhart, Geoffrey Hinton, Ronald J. Williams. Learning Representations by Back Propagating Errors[J]. Nature, 1986, 323(6088): 533-536.

[30] Valueva M V, Nagornov N N, Lyakhov P A, et al. Application of the residue number system to reduce hardware costs of the convolutional neural network implementation[J]. Mathematics and Computers in Simulation (MATCOM), 2020, 177(C): 232-243.

[31] Ronan Collobert, Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning[C]. Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008). Helsinki, Finland, 2008.

[32] Hamed Habibi Aghdam, Elnaz Jahani Heravi. Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification[M]. New York, NY, United

States: Springer Publishing Company, Incorporated, 2017.

[33] Hiroaki Sakoe, Ryosuke Isotani, Kazunaga Yoshida, Ken-ichi Iso, Takao Watanabe. Speaker-independent word recognition using dynamic programming neural networks [J]. Readings in Speech Recognition, 1990: 439-442.

[34] Adam Paszke, Sam Gross, Francisco Massa, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library[C]. 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada, 2019.

[35] Chien-Yao Wang, Hong-Yuan Mark Liao, I-Hau Yeh, et al. CSPNet: A New Backbone that can Enhance Learning Capability of CNN[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Seattle, WA, USA, 2020.

[36] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, Jiashi Feng. PANet: Few-Shot Image Semantic Segmentation with Prototype Alignment[C]. IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South), 2019.

[37] Shuai Li, Lingxiao Yang, Jianqiang Huang, Xian-Sheng Hua, Lei Zhang. Dynamic Anchor Feature Selection for Single-Shot Object Detection[C]. IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South), 2019.

[38] Sanghyun Woo, Jongchan Park, Joon-Young Lee, In So Kweon. CBAM: Convolutional Block Attention Module[C]. ECCV: European Conference on Computer Vision. Munich, Germany, 2018.

[39] Jongchan Park, Sanghyun Woo, Joon-Young Lee, In So Kweon. BAM: Bottleneck Attention Module[C]. British Machine Vision Conference (BMVC). Newcastle, UK, 2018.

[40] Yang Xiao. An Overview of the Attention Mechanisms in Computer Vision[J]. Journal of Physics: Conference Series. 2020, 1693(1): 1-7.

[41] Wei Li, Kai Liu, Lizhe Zhang, Fei Cheng. Object detection based on an adaptive attention mechanism[J]. Scientific Reports, 2020, 10(1): 1-13.

[42] Shivanthan Yohanandan, Andy Song, Adrian G. Dyer, Dacheng Tao. Saliency Preservation in Low-Resolution Grayscale Images[C]. ECCV: European Conference on Computer Vision. Munich, Germany, 2018.

[43] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen. MobileNetV2:Inverted Residuals and Linear Bottlenecks[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, UT, USA, 2018.

致 谢

在兰州大学三年的硕士研究生的学习生涯即将结束，经过三年的学习，我的知识水平得到了加强，能力得到了提升，视野得到了拓展。回顾这些年的时光，我衷心地向帮助过我的人们表示感谢。

首先我要感谢的是我的硕士研究生导师白建明老师。白老师专心科研，认真教学，不仅在学术上带领我取得成绩，也在生活中教会我许多为人处世的道理。在此，我由衷地感谢白建明老师的辛勤付出。

感谢数学与统计学院的赵学靖、严定琪等老师的教学，通过他们的课程，使我受益匪浅。

感谢我的研究生宿舍的室友们，通过向他们学习我进步了很多。

感谢我的家人对我的支持，让我能够顺利地完成学业。

感谢资助本论文的项目对本论文研究的支持。

最后，感谢本论文的评审老师以及参加答辩的专家、老师们，感谢你们对本论文提出的宝贵意见。

黎戈

2021 年 3 月 10 日