

分类号: TP242.6

密 级: 公开

学校代码: 10363

学 号: 2210330146



安徽工程大学

Anhui Polytechnic University

硕士学位论文

题目 基于深度学习的机械臂
垃圾分拣方法研究

论文作者 易方旭

校内导师 王正刚（副教授）

校外导师 毛德平

专业学位类别 电子信息（085400）

研究方向（领域） 人工智能与系统集成

论文提交日期: 2024 年 5 月 31 日

分类号: TP242.6

单位代码: 10363

密 级: 公 开

学 号: 2210330146

题 目 基于深度学习的机械臂垃圾分拣方法研究

题 目 Research on the Garbage Sorting Method for the
Robotic Arm Based on Deep Learning

学生姓名: 易方旭

校内导师: 王正刚 (副教授)

校外导师: 毛德平

专 业: 电子信息

研究方向: 人工智能与系统集成

论文答辩日期: 2024.05.15

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名：易方旭

日期：2024年5月31日

安徽工程大学学位论文版权使用授权书

学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程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，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 ☐，在 _____ 年解密后适用本版权书。

本学位论文属于

不保密 ☒。

学位论文作者签名：易方旭

指导教师签名：王正明

日期：2024 年 5 月 31 日

日期：2024 年 5 月 31 日

基于深度学习的机械臂垃圾分拣方法研究

摘 要

随着我国经济的迅猛发展以及居民生活质量的显著提高,垃圾的产量也在逐年上升,这为垃圾的分类与处理带来了前所未有的挑战。机器人技术和计算机视觉领域的进展,为解决这一问题提供了新的技术手段。通过深度学习与机械臂的结合,可以实现对垃圾的自动识别和分类,提升垃圾处理的效率和准确性。本文对基于深度学习的机械臂垃圾分拣方法中涉及的目标检测与路径规划展开研究,论文主要工作内容如下:

(1)针对复杂背景下垃圾的识别和定位问题,提出了基于改进YOLOv5s垃圾目标检测算法。首先,为改善复杂背景下YOLOv5s目标检测算法出现漏检、误检的问题,在主干网络中使用融合密集连接与Ghost模块的C3_DG模块,在提升主干的特征提取能力的同时,不增加过多的参数量。其次,为提高算法对不同尺寸目标识别的精度,通过添加I-BiC模块对多尺度特征融合网络进行优化。使用Ghost Bottleneck替换颈部C3模块内的瓶颈结构,从而构建新的C3Ghost模块,减少模型对计算资源的消耗。同时,引入SimAM注意力机制用以抑制背景噪声影响。最后,在自制数据集上与Trashnet数据集上,改进后算法都表现出了较原算法更好的性能。

(2)在垃圾分拣过程中,路径规划算法对机械臂至关重要,其能够

使机械臂高效地到达目标位置，并避免碰撞，提升机械臂作业的效率与安全性。针对当前机械臂领域广泛使用的 RRT-Connect 路径规划算法，存在搜索时间长、节点生长随机、路径转折多且不平滑等问题，提出了一种融合动态步长与人工势场的路径规划算法。为改善 RRT-Connect 算法采样点过于分散的问题，引入目标偏置策略。针对使用固定步长造成扩展缓慢的问题，引入动态步长策略对步长进行动态调整。为减少扩展过程中的随机性，融合人工势场法，使新节点的生长受到随机采样函数分量、目标引力分量和障碍物斥力分量的约束。最后，对规划后的路径进行修剪并使用三次 B 样条进行平滑处理，改善路径中的冗余点与转折，减少机械臂运动中抖动的发生。在相同环境中，改进后算法较 RRT-Connect 算法路径长度减少 15.4%，规划时间下降了 49.2%。

(3)机械臂垃圾分拣方法应用与实现。将 Mirobot 六轴机械臂与深度相机结合，采用眼在手外的方式搭建了机械臂垃圾分拣实验平台。通过垃圾分拣实验，验证了本文提出的基于深度学习的机械臂垃圾分拣方法的可行性。

关键词：深度学习，垃圾分拣，机械臂，目标检测，路径规划

RESEARCH ON THE GARBAGE SORTING METHOD FOR THE ROBOTIC ARM BASED ON DEEP LEARNING

ABSTRACT

With the rapid development of China's economy and the significant improvement in the quality of life of its citizens, the volume of garbage generated has been increasing year by year, posing new challenges for waste classification and treatment. Advancements in robotics technology and computer vision provide advanced technologies for improving waste classification processes. By combining robotic arms with deep learning, automatic recognition and classification of garbage can be achieved, improving the efficiency and accuracy of garbage disposal. In this paper, the object detection and path planning of the robotic arm involved in the garbage sorting method of the robotic arm based on deep learning are studied. The main work of the paper is as follows:

(1) A garbage object detection algorithm based on improved YOLOv5s is proposed for the recognition and localization of garbage in complex backgrounds. Firstly, to improve the problem of missed and false detections in the YOLOv5s object detection algorithm under complex backgrounds, a C3-DG module that integrates dense connections with the

Ghost module is used in the backbone. This improves the feature extraction ability of the backbone without adding too many parameters. Secondly, to improve the accuracy of the algorithm in recognizing targets of different scales, the multi-scale feature fusion network is optimized by adding an I-BiC module. Ghost Bottleneck is used to replace the bottleneck structure in the neck C3 module to construct a new C3Ghost module, reducing the consumption of computational resources by the model. Finally, SimAM attention mechanism is introduced to suppress the influence of background noise. On both the self-made dataset and the Trashnet dataset, the improved algorithm showed better performance than the original algorithm.

(2) In the garbage sorting process, the path planning algorithm is crucial for the robotic arm, allowing it to efficiently reach the target position while avoiding collisions, thus enhancing the efficiency and safety of the robotic arm's operation. For the RRT-Connect path planning algorithm widely used in the field of robotic arms currently has problems such as long search time, random node growth, multiple and uneven path transitions. Therefore, a new path planning algorithm that integrates dynamic step size and artificial potential field is proposed. To solve the problem of scattered sampling points in the RRT Connect algorithm, a target bias strategy is introduced. To address the problem of slow expansion and increased search time caused by using fixed step sizes, a dynamic step size strategy is introduced to dynamically adjust the step size. To reduce

randomness in the expansion process, the artificial potential field method is integrated to constrain the growth of new nodes by the random sampling function component, target gravity component, and obstacle repulsion component. Finally, the planned path is pruned and smoothed using cubic B-splines to reduce redundant points and turns in the path and the occurrence of shaking in the motion of a robotic arm. Compared with the original algorithm, the improved algorithm reduces the path length by 15.4% and the planning time by 49.2% in the same space.

(3) Application and realization of garbage sorting method with the robotic arm. Combining the Mirobot six-axis robotic arm with a depth camera, a robotic arm garbage sorting experimental platform is built with the Eye-to-Hand. The feasibility of the deep learning based garbage sorting method with the robotic arm proposed in this paper is verified through garbage sorting experiments.

KEY WORDS : deep learning, garbage sorting, robotic arm, object detection, path planning

目 录

摘 要.....	I
ABSTRACT.....	III
第 1 章 绪论.....	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	2
1.2.1 垃圾分拣方法研究现状.....	2
1.2.2 基于深度学习的垃圾目标检测.....	3
1.2.3 路径规划研究现状.....	4
1.3 论文主要研究内容与章节安排.....	6
第 2 章 卷积神经网络理论基础与机械臂运动学建模.....	8
2.1 引言.....	8
2.2 卷积神经网络.....	8
2.2.1 卷积层.....	9
2.2.2 激活函数.....	10
2.2.3 池化层.....	11
2.2.4 全连接层.....	11
2.3 机械臂建模与运动学分析.....	11
2.3.1 机械臂位姿描述与坐标变换.....	12
2.3.2 机械臂 D-H 参数模型	15
2.3.3 机械臂正运动学分析.....	17
2.3.4 机械臂逆运动学分析.....	18
2.4 本章小结.....	19
第 3 章 基于深度学习的垃圾检测算法研究.....	20
3.1 引言.....	20
3.2 目标检测算法.....	20
3.2.1 两阶段目标检测算法.....	20

3.2.2 单阶段目标检测算法.....	22
3.3 基于改进 YOLOv5s 的垃圾目标检测算法.....	26
3.3.1 C3_DG 与 C3Ghost 构建.....	26
3.3.2 多尺度特征融合优化.....	29
3.3.3 注意力机制.....	31
3.3.4 改进后模型结构.....	33
3.4 实验结果与分析.....	34
3.4.1 数据集.....	34
3.4.2 实验环境.....	35
3.4.3 算法评价指标.....	36
3.4.4 消融对比试验.....	36
3.4.5 注意力机制对比.....	38
3.4.6 检测效果对比.....	39
3.4.7 不同数据集对比实验.....	40
3.4.8 不同算法对比试验.....	41
3.5 本章小结.....	41
第 4 章 机械臂路径规划算法.....	42
4.1 引言.....	42
4.2 RRT 与 RRT-Connect 算法基本原理.....	42
4.3 基于 DAPF-RRT 算法的路径规划.....	45
4.3.1 目标偏置.....	45
4.3.2 动态扩展步长.....	45
4.3.3 节点扩展策略优化.....	46
4.3.4 路径修剪.....	50
4.3.5 轨迹平滑.....	51
4.3.6 DAPF-RRT 算法流程.....	52
4.4 三维空间仿真实验及分析.....	53
4.5 机械臂路径规划仿真.....	57
4.6 本章小结.....	60

第 5 章 机械臂垃圾分拣方法应用与实现.....	61
5.1 引言.....	61
5.2 实验平台介绍.....	61
5.3 相机标定及手眼标定.....	62
5.3.1 D435i 深度相机标定.....	62
5.3.2 手眼标定.....	63
5.4 目标检测算法部署与 ROS 环境下机械臂路径规划实验.....	66
5.4.1 目标检测算法部署.....	66
5.4.2 ROS 环境下机械臂路径规划实验.....	67
5.5 机械臂垃圾分拣实验.....	70
5.5.1 机械臂工作空间中无障碍物.....	71
5.5.2 机械臂工作空间中有障碍物.....	72
5.6 本章小结.....	73
第 6 章 总结与展望.....	74
6.1 总结.....	74
6.2 展望.....	75
参考文献.....	76
攻读硕士学位期间主要研究成果.....	82
致 谢.....	83

第 1 章 绪论

1.1 研究背景及意义

随着我国经济水平的不断提高和城市化进程的加快,垃圾的产量随之急剧增加。据统计,2019 年全国 196 个大、中城市生活垃圾产生量达到 23560.2 万吨,处置量为 23487.2 万吨,垃圾治理迫在眉睫,垃圾分类及资源化利用受到空前重视^[1]。然而,我国的垃圾处理方式相对落后,主要依赖于填埋和焚烧^[2]。填埋不仅占用大量土地,还可能污染土壤和地下水;焚烧过程可能释放有毒有害物质,对人体健康造成严重影响,焚烧产生的底渣含有重金属和有毒物质,如处理不当,也会造成严重污染^[3]。垃圾分类处理是实现循环利用的过程,可以实现“变废为宝”^[4]。因此,国家对垃圾分类处理问题高度重视,相继颁布了一系列相关政策。例如,2017 年国家相关部委发布了《生活垃圾分类制度实施方案》,要求 46 个城市实行垃圾强制分类,并对垃圾的可回收率提出要求^[5]。2019 年 6 月,九部门联合发布《关于在全国地级及以上城市全面开展生活垃圾分类工作的通知》,推动全国范围内的垃圾分类实践^[6]。

目前,我国的垃圾分类回收工作大多还依赖于人工完成。手动分拣垃圾成本高、效率低,大量未分类的垃圾给工作人员带来了巨大的工作量。长期在恶劣工作环境中作业还可能危害工作人员的健康,因此,急需采用机器人技术代替人工进行垃圾分拣^[7]。传统机械臂分拣方法通常应用于单一重复的抓取任务。通过示教模式获取抓取点和放置点的位置信息,然后基于这些预定义的位置信息编写运动控制代码来执行抓取和放置动作。该类系统对目标物体的位置姿态有着严格的约束条件,并且由于缺乏视觉识别,无法对目标物体的类型和位置进行判断,从而限制了系统的适用范围和灵活性,只能完成简单的分拣功能。随着深度学习和机器人技术的飞速进步,将这些前沿技术应用于智能化垃圾分拣系统成为可能。通过将深度学习与机械臂应用于垃圾分拣,可以代替人类更高效和便捷地完成大量重复性和基础服务性工作,极大的解放劳动力^[8]。本文将深度学习与路径规划算法应用于垃圾分拣中,实现对未知物体进行正确识别、定位与机械臂运动路径的快速规划。在完成垃圾的自动分拣的同时,能够使机械臂在工作环境存在障碍

物的情况下完成避障。

1.2 国内外研究现状

1.2.1 垃圾分拣方法研究现状

发达国家垃圾处理产业起步较早，在自动化垃圾处理领域技术较为领先^[9]。2012 年，芬兰的 Zen Robotics 公司设计了一种机器人识别和分类系统^[10]。该系统使用机器视觉、3D 扫描和光谱信息来挑选垃圾中的木材、石头、金属和其他物体，Zen Robotics 系统最高分拣速度可达 4000 次/小时。

AMP Robotics 公司设计的 AMP 分拣机器人以其出色的处理速度而著称^[11]。分拣系统主要是由一个并联机器人在传送带上工作，其高度和伸展范围都可调节，如图 1-1 所示。该系统能够以每分钟 80 件的速度进行分拣，速度远超手工分拣。该系统应用对象广泛，能够处理各种可回收材料，包括金属、电线、纸板等。AMP 分拣机器人通过计算机视觉和一系列传感器来扫描并区分物体的颜色、纹理、形状、大小和材料。机器人使用吸盘附件从传送带上抓取物品，并将它们投放到指定的回收箱中。截至 2021 年 4 月，AMP 平台已经在美国 20 多个州处理了超过 10 亿件可回收物品。AMP 分拣机器人提高了垃圾回收效率和准确率，是人工智能结合机器人技术在垃圾分拣方面的代表性应用。



图 1-1 AMP 垃圾分拣系统

相比于国外，国内在机器人、机器视觉以及人工智能等领域的起步比较晚，但发展较为迅速。2020 年广东弓叶科技有限公司推出了一款名为“Picking AI”的先进垃圾分拣机器人。该机器人融合计算机视觉技术，通过机械臂拾取目标垃圾，将其准确地放置到指定的容器中。这款机器人适用于处理已分类的干垃圾，能够

有效地从中分离出金属、塑料、纸张等可回收物料，其分拣精度超过 95%，显著提高了垃圾分拣的效率和准确性。杭州市江干区部署了“环卫士”垃圾分拣机器人，这些机器人集成了视觉识别和路径规划等尖端技术。“环卫士”垃圾分拣机器人垃圾识别准确率达 98%，能够连续工作 8 小时，每天可拾取并分类大约 17.5 公斤垃圾。江干区在钱江新城的四条主要街道的人行道和慢车道上投入了两台这垃圾分拣机器人。这些机器人的使用，将预计取代四到六名环卫工人的日常清洁工作，提高垃圾分类的效率和质量。



图 1-2 “环卫士”垃圾分拣机器人

1.2.2 基于深度学习的垃圾目标检测

目标检测是计算机视觉、深度学习、人工智能等领域的基础研究方向，广泛应用于机器人、无人机、航空航天等领域。目标检测的主要任务是从图像中定位出感兴趣的目标，确定其类别并给出每个目标的边界框^[12]。

传统目标检测算法使用不同尺度滑动窗口得到候选区域，使用 SIFT(Scale Invariant Feature)算子^[13]等提取候选区域的特征点，并利用 SVM 算法^[14]对候选区域类别进行预测。Mindy Yang 等^[15]使用 SIFT 算法对垃圾进行提取特征，利用 SVM 分类实现对垃圾特征分类，分类精度达到 63%。然而，传统目标检测算法相对复杂，面对检测目标背景复杂时提取能力弱，泛化能力不强。

目前基于深度学习的目标检测算法大体可以分为两阶段检测算法和单阶段检测算法。两阶段检测算法以 RCNN 系列^[16-18]为代表。单阶段检测算法以 SSD(Single Shot MultiBox Detector)算法^[19]以及 YOLO(You Only Look Once)系列等为代表^[20-23]。

Wang 等^[24]通过 Faster RCNN 实现垃圾自主检测，提出一种数据融合和增强

策略,并验证了该方法具有较好的泛化能力和较高的检测精度。两阶段检测算法虽然在精度方面有优势,但在检测速度、模型大小以及训练难度方面与单阶段模型有一定差距。Ma 等^[25]提出了一种用于垃圾检测的 L-SSD 算法。L-SSD 算法提出了一个新的特征金字塔,该结构在各个尺度上都具有更强的语义;使用 ResNet-101 更换 VGG16 骨干网络以提高模型特征提取能力;最后引入 Focal Loss 损失函数,以调节正负样本比例失衡的问题。2016 年单阶段目标检测模型 YOLO 系列算法的提出使目标检测的速度和精度得到了极大的提高。目前,国内外已经对利用 YOLO 算法解决垃圾分类问题展开了一定的研究。Sun 等^[26]提出 YOLOv5-OCDS 的垃圾检测算法。YOLOv5-OCDS 算法在颈部使用 ODConv 替换部分普通卷积,捕捉目标在不同尺度的特征表达;构建 C3DCN 模块,增强模型的特征提取能力;将耦合头替换为解耦头,使模型更专注于学习每个类别的特征和位置信息;使用 Soft-NMS 算法替换原始的 NMS 算法,提高对小目标的检测能力。Chen 等^[27]基于 YOLOv5 架构开发了一个新型轻量级垃圾识别算法。该算法主干使用 Shufflenet v2 与 SE 注意力机制,用以缩减模型体积;颈部使用更高效的 GSConv 模块替代普通卷积,同时 C3 模块也被新的 VoVGSCSP 模块所取代,进一步轻量化模型并提升其识别精度。罗安能等^[28]提出基于 YOLOv5s 改进的可回收垃圾目标检测算法,结合 ShuffleNetV2 与深度可分离卷积,在自制垃圾数据集上模型精度达到 94.01%,并在检测速度上有较大提升。魏铖磊等^[29]针对密集堆放的垃圾,通过引入多尺度感受野模块与通道注意力机制对 YOLOv3 算法进行了改进,使模型在面对密集堆放目标时有较好性能。

1.2.3 路径规划研究现状

路径规划问题是机械臂完成自主垃圾分拣的核心问题。路径规划是指从起点到目标点规划一条可行的路径。同时规划过程中要避免规划路径与障碍之间的发生碰撞,且要符合约束条件,例如机械臂各关节角活动限制等。路径规划方法大体可以分为三类:基于图搜索的算法、随机采样法的路径规划、基于仿生的路径规划等。

基于图论的经典路径搜索算法有 Dijkstra 算法^[30]、A*算法^[31]等。Dijkstra 算法是一种广度优先搜索算法,其优势在于能获得一条最短路径,但运算量大,效

率较低。Hart 等人于 1968 年融合 Dijkstra 算法和启发函数,提出了 A*算法,该算法能够寻找到最短路径,同时搜索速度有一定程度提升。A*在效率上相较于 Dijkstra 算法有了显著提升,但是仍存在搜索时间长,搜索节点多的问题。

基于仿生的路径规划是一种借鉴自然界生物行为和策略设计的路径规划方法,该类型算法自适应性强、鲁棒性高等特点。经典的基于仿生的路径规划算法有遗传算法^[32]、蚁群算法^[33]和粒子群算法等^[34]。遗传算法通过选择、交叉、变异等行为模拟生物进化过程,逐代进化出最佳路径。但算法接近最优解时,算法收敛速度缓慢,通常用于全局路径规划。蚁群算法搜索策略参考了蚁群觅食过程蚂蚁在移动过程中释放信息素,其他蚂蚁通过感知信息素来选择路径,但该算法收敛时间较长。

随着探索空间从二维增加到三维,路径规划计算的复杂性呈指数级增加,以上算法并不适合解决机械臂在复杂构型空间中的路径规划问题^[35]。针对机械臂工作的高维空间环境,以 RRT(Rapidly-exploring Random Tree)算法^[36]和 PRM(Probabilistic Roadmap Method)算法^[37]为代表的基于采样的路径规划方法逐渐成为研究热点。由于 RRT 算法简单高效,在机械臂的路径规划中得到了广泛的应用。然而,RRT 算法也有一些不足,RRT 算法的高随机性虽然有助于避免陷入局部最优解,但也导致了搜索的盲目性,规划的路径往往并非最短或最优。近些年来国内外学者在 RRT 的基础上进行了许多优化,产生了很多 RRT 算法的变体用以加快收敛率和提高搜索效率。

Ding 等人^[38]提出 EP-RRT*算法,首先使用 RRT-Connect 生成初始可行路径。然后对算法的初始路径进行扩展,得到扩展区域,并在扩展区域内进行迭代启发式采样。EP-RRT*算法与 RRT-Connect 相结合,提高了 RRT*算法在特殊环境下的搜索效率。Li J 等^[39]提出一种改进的 RRT-Connect 算法,通过设置自适应步长策略,从起点、终点和定点构建四棵随机树进行搜索,解决扩展缓慢、加速收敛的问题。实验结果表明,改进算法迭代次数减少 18.11%,路径节点数减少 23.02%。尹庆文等^[40]提出一种基于 GB-APFB-RRT*算法的机械臂路径规划,在 RRT*算法的基础上对算法初始化过程进行了改进,从起点与终点同时扩展;引入目标偏置策略,并重选父节点重布线,使每次迭代点都为路径代价最小;新节点的生长方式上引入人工势场思想,并对规划后的路径进行修剪与平滑处理。结果表明所提

算法的搜索效率平均提升了 95%，路径长度平均缩短了 20%。江韩等^[41]提出改进 RRT 算法的机械臂路径规划，该算法结合双向 RRT 和目标偏置的优点，在算法扩展过程中设置了虚拟目标点，使用动态步长提高搜索效率，并对规划后的路径进行了修剪与平滑。

1.3 论文主要研究内容与章节安排

本文着重讨论了基于深度学习的机械臂垃圾分拣方法中涉及的目标检测算法与机械臂路径规划算法。通过结合改善目标检测算法与路径规划算法的性能，实现垃圾的自动分拣。全文共 6 章，章节安排如图 1-3 所示。

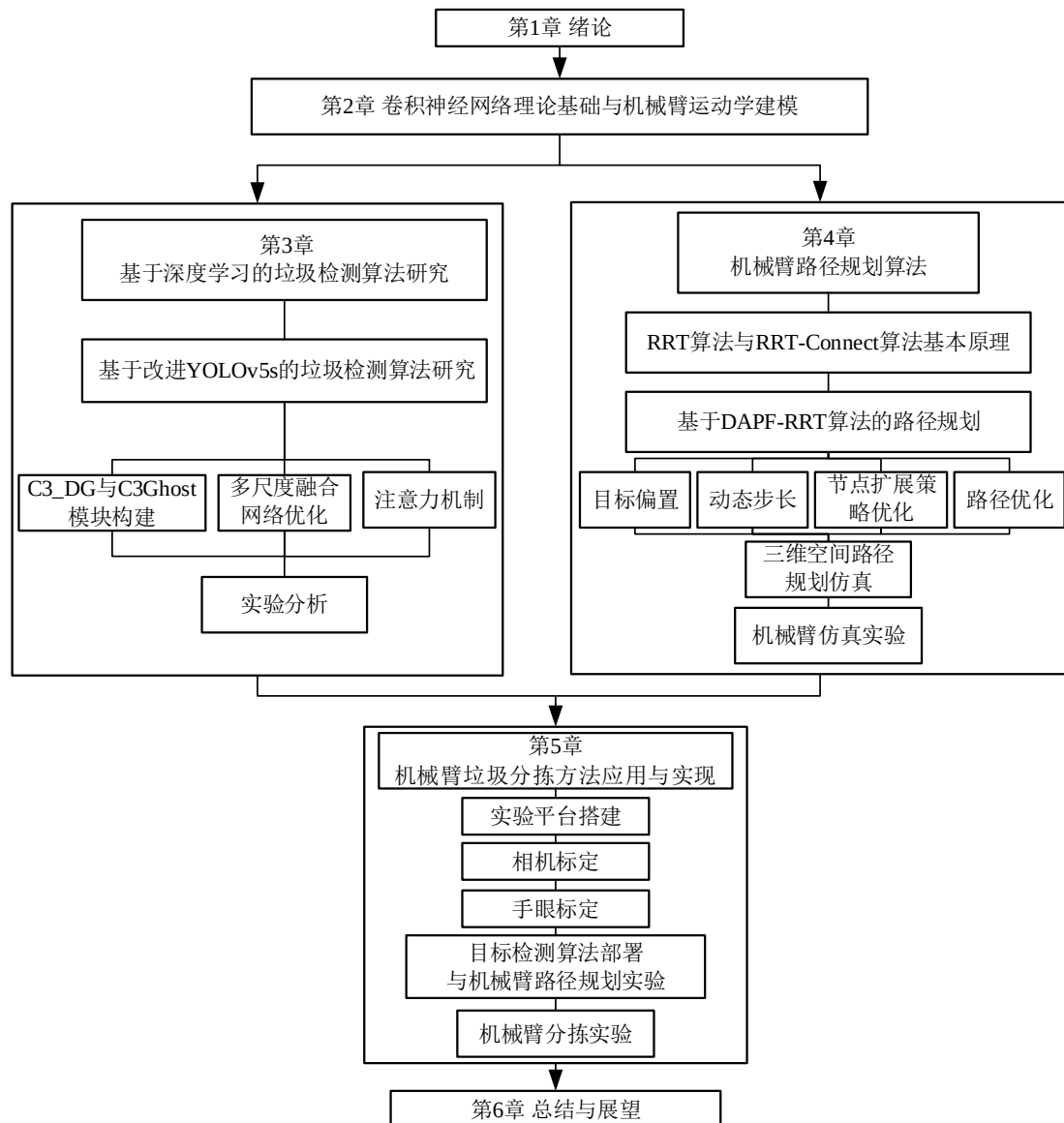


图 1-3 章节安排

各章节主要内容如下：

第 1 章，绪论。概述了基于深度学习的机械臂垃圾分拣方法的研究背景与意义，简述了目前国内外垃圾分拣方法、目标检测以及机械臂路径规划的研究现状。

第 2 章，卷积神经网络和机械臂运动学分析的理论基础。概述卷积神经网络的理论知识，包括卷积层、激活函数、池化层以及全连接层。针对 Mirobot 机械臂的结构使用改进 D-H 参数法进行建模，并对机械臂进行了正逆运动学分析。这两部分的理论为后文的垃圾目标检测算法与机械臂路径规划研究奠定了基础。

第 3 章，基于深度学习的垃圾目标检测算法研究。分析了两阶段与单阶段目标检测算法的结构与不足，最后选择比较成熟的 YOLOv5s 单阶段目标检测算法进行改进，提出了基于 YOLOv5s 的垃圾目标检测算法。针对目前垃圾数据集存在的问题，在 2020 年“华为云人工智能大赛-垃圾分类挑战杯”垃圾像数据集与 TACO 数据集的基础上自制了数据集。最后设计实验，在自制数据集与 Trashnet 数据集上验证了改进后算法的优越性。

第 4 章，机械臂路径规划。分析了 RRT 算法和 RRT-Connect 算法的原理。针对 RRT-Connect 算法的不足做出了以下改进：添加目标偏置策略，改善目标点过于分散的问题；引入动态步长，加快算法搜索；将人工势场融合到 RRT-Connect 算法的新节点扩展过程中，减少节点扩展的随机性；最后，对规划后的路径进行修剪以及平滑，通过仿真实验了算法的有效性。

第 5 章，改进后的目标检测算法与路径规划算法在机械臂分拣任务上的应用。搭建的机械臂垃圾分拣平台，并通过融合改进后的目标检测算法与 DAPF-RRT 算法完成垃圾的自动分拣。

第 6 章，基于深度学习的机械臂垃圾分拣方法的总结与展望。阐述了本文的主要工作内容，并进行了展望。

第 2 章 卷积神经网络理论基础与机械臂运动学建模

2.1 引言

卷积神经网络理论基础与机械臂运动学建模为后文垃圾目标检测算法与机械臂路径规划提供理论支持。本节分别对卷积神经网络理论基础与机械臂运动学建模进行介绍与分析。

2.2 卷积神经网络

卷积神经网络(Convolutional Neural Network, CNN)作为一种自主特征提取器在图像处理、目标检测、图像分类等领域展现了卓越的性能。与传统的机器学习方法相比, CNN 不需要手动提取特征^[42], 能够自主地从数据中学习和提取有价值的特征, 省去了繁琐的手动特征设计过程, 极大简化了模型的开发和训练过程。最早提出的卷积神经网络为 LeCun 提出的 LeNet^[43], 模型结构如图 2-1 所示。CNN 架构有诸多不同变体, 但是基本结构非常相似, 典型的 CNN 架构通常是卷积层和池化层的交替。它包括了输入层(Input)、卷积层(Conv Layer)、池化层(Pooling layer)和全连接层(Fully Connected Layers)。数据首先通过输入层进入网络, 然后使用卷积层对数据进行特征提取, 接着池化层通过下采样减少数据维度, 最后全连接层对特征进行分析并完成分类任务。

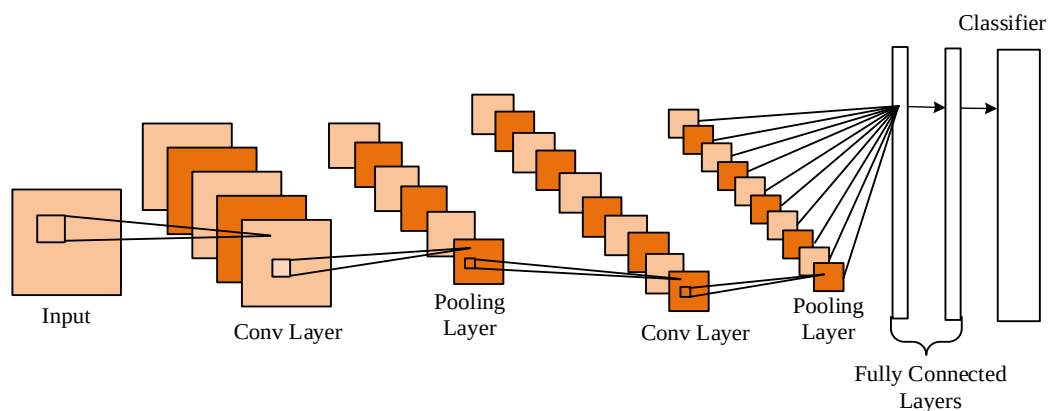


图 2-1 LeNet 网络结构图

2.2.1 卷积层

在卷积神经网络中，卷积层通过一系列可学习卷积核，对输入图像进行卷积运算，提取输入图像中的特征。卷积层具有多个卷积核，一般是由长宽相等的权重矩阵构成。这些权重通过优化损失函数，在反向传播中得到更新。卷积核的作用随网络深度的增加而变化。在浅层中通常用于提取纹理、边缘等特征，较高层通过处理较低级别的特征来提取更复杂和更高级别的特征^[44]。

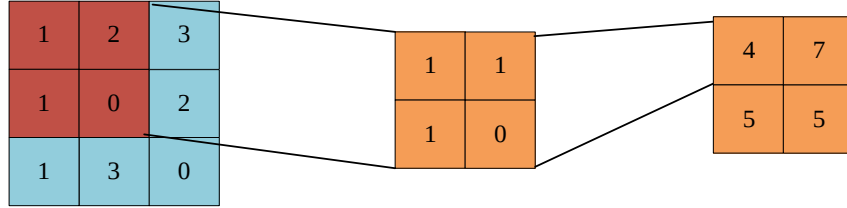


图 2-2 卷积计算过程

在卷积运算中，卷积核通过滑动方式从左上角开始根据设定的步长(stride)逐行遍历整个图像，过程中卷积核与其覆盖区域先进行线性乘法，再求和。图 2-2 为输入特征图与大小为 2×2 卷积核的进行卷积的过程，图中设定步长为 1，输入特征图的通道数须与卷积核保持一致。

由于特征图的尺寸经过多层卷积操作会不断缩小，尺寸变小会造成特征图中心被多次计算，而特征图边界上可能存在的信息被忽视。为解决以上问题通常使用填充操作(Padding)增大特征图的尺寸。常用的填充方法有重复边界填充和 0 填充，图 2-3 为边界 0 填充操作。

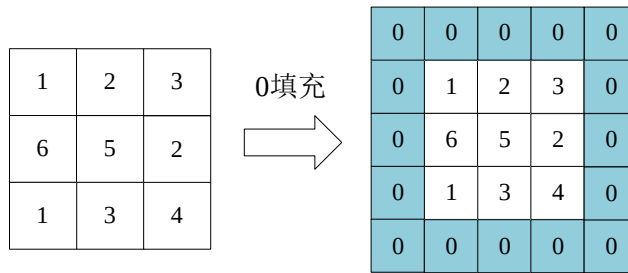


图 2-3 边界 0 填充操作

通过卷积操作后输出特征图宽 W_o 和高 H_o 可由式(2-1)表示：

$$\begin{aligned} W_o &= \frac{W + 2p - k}{s} + 1 \\ H_o &= \frac{H + 2p - k}{s} + 1 \end{aligned} \quad (2-1)$$

式中, W 和 H 为输入特征图的宽和高, k 为卷积核的尺寸, s 为步长, P 是 Padding 数量。

2.2.2 激活函数

卷积操作产生的输出是输入特征图与卷积核的加权和。如果不通过非线性变换, 那么无论网络多深, 其能力仍然受限于线性函数的表达范围, 这将使得网络难以捕捉复杂特征。因此, 为了提升神经网络的非线性拟合能力并增强其整体的表达力, 必须在网络中引入非线性元素, 即激活函数^[45]。CNN 中常用的激活函数包括 Sigmoid 函数和 Tanh 函数以及 ReLU 函数, 激活函数如表 2-1。

表 2-1 激活函数表达式

激活函数	表达式	激活函数图像
Sigmoid	$\text{Sigmoid}(z) = \frac{1}{1 + e^{-z}}$	
Tanh	$\text{Tanh}(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	
ReLU	$\text{ReLU}(z) = \begin{cases} z, & z > 0 \\ 0, & z \leq 0 \end{cases}$	

Sigmoid 函数定义如表, 其可以将任意实数映射到 0 到 1 之间的值, 常用于二分类网络中。但是 Sigmoid 函数输出不以 0 为中心, 这可能会在梯度下降期间导致梯度更新的振荡, 从而减缓训练过程。

Tanh 函数, 也称为双曲正切函数, 是 Sigmoid 函数的变种, 可以将实数映射到 -1 和 1 之间。其主要是在某种程度上解决了 Sigmoid 函数输出不是零中心的问题。然而, Tanh 函数也存在梯度消失问题, 特别是当 x 的值趋近于负无穷小或正无穷大时, 梯度可能会趋近于零, 导致学习过程中的收敛速度减慢。

ReLU 是使用最广泛的激活函数, 能够有效减少训练过程中的消失梯度问题, 同时加快学习速度。由 ReLU 函数形式可知, 与 Sigmoid 函数和 Tanh 函数相比, ReLU 函数的导数为常数, 而 Sigmoid 和 Tanh 在导数计算时不仅涉及指数运算

还需要除法运算。当 x 小于 0，ReLU 的导数为 0，当 x 大于 0，导数为 1，求导过程不涉及复杂计算，有助于网络模型梯度更新。

2.2.3 池化层

池化层一般位于卷积层之后，其主要目的是对输入图像进行下采样，以减少卷积层输出的特征图的空间维度，从而减轻网络的计算负担，并有助于防止过拟合^[46]。平均池化和最大池化是两种广泛使用的池化方法，图 2-4 为步长为 2 时两种池化方法得出的结果。最大池化选取区域中最大值作为代表，以此提取图像中明确的特征细节。平均池化则将每个区域平均值，可能会丢失图像中相关特征。后文中 YOLOv5s 的网络模型中使用的是最大池化层，将不同的最大池化层一起使用，实现识别不同大小的目标，同时增大感受野，提升网络模型的检测能力。

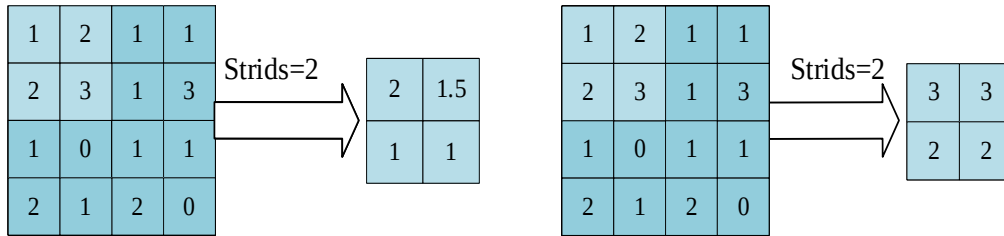


图 2-4 最大池化与平均池化

2.2.4 全连接层

输入图像经卷积层和池化层处理后产生的一系列特征图，这些特征图包含了输入图像的重要信息，但仍然是高维。全连接层的任务就是将这些高维的特征图展平，转换成一个一维的特征向量。特征向量随后传递给 Softmax 层，由全连接层输出的特征向量转化成概率值，每个概率对应每个类别的预测概率。全连接层和 Softmax 层共同构成 CNN 的分类器，实现从特征到分类标签的转换。

2.3 机械臂建模与运动学分析

机械臂是一种复杂的开链结构，其建模与运动学分析不仅十分必要，也是确保机械臂能够沿规划路径进行运动的基础。机械臂的运动学分析主要包括正运动学和逆运动学^[47]。正运动学通过给定的一组机械臂关节空间的关节角度，求解末端执行器在笛卡尔空间的位姿；逆运动学则基于末端执行器的期望位姿，求解出

达成该位姿所需的各关节在关节空间中的旋转角度。

2.3.1 机械臂位姿描述与坐标变换

位姿包括了位置与姿态。在机械臂运动学分析中，不考虑机械臂本身的物理性质，主要关注机械臂的几何特性和运动规律，将机械臂的各个部分视为刚体，简化问题。为描述刚体姿态，描绘刚体位姿的常规做法为设定一个固定在基座上的参考坐标系 $\{A\}$ ，也称为世界坐标系，以及在刚体上定义一个随之移动的坐标系 $\{B\}$ 。刚体的空间位姿则通过确定这个移动坐标系相对于固定坐标系的具体位置和方向来获得。

(1)位置描述

在直角坐标系 $\{A\}$ 中，可将任意一点 ${}^A Q$ 表示为一个 3×1 的列向量如式(2-2)。上标 A 表示选定坐标系，列向量中三个元素表示该点在对应坐标轴上的投影。 ${}^A Q$ 在空间中位置示意图如图 2-5 所示。

$${}^A Q = \begin{bmatrix} Q_x \\ Q_y \\ Q_z \end{bmatrix} \quad (2-2)$$

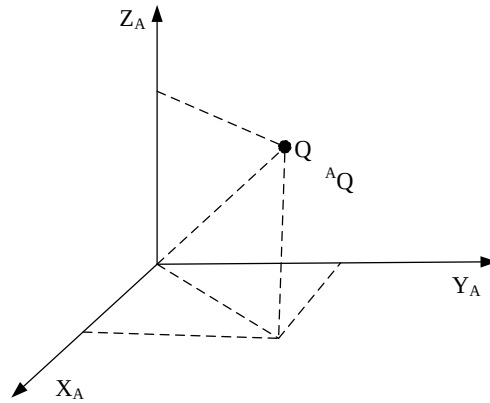


图 2-5 位置表示

(2)姿态描述

${}^A Q$ 描述了物体在空间的位置，但还需获得物体的姿态才能描述出物体的位姿。在图 2-6 中，设一直角坐标系 $\{B\}$ 与刚体固接，坐标系原点为刚体的质心。

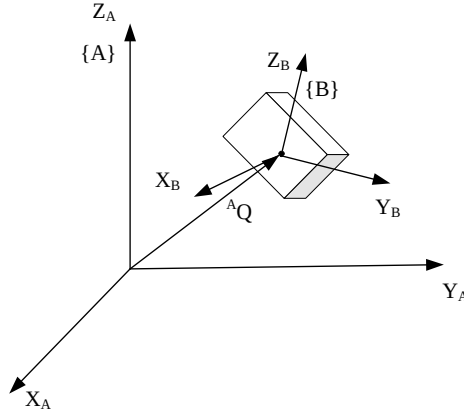


图 2-6 姿态表示

使用 X_B 、 Y_B 、 Z_B 表示坐标系的单位向量，这三个单位向量在坐标系 $\{A\}$ 中的方向余弦组成的 3×3 矩阵可以表示刚体 Q 相对于坐标系 $\{A\}$ 的姿态。这种由方向余弦构成的矩阵被称作旋转矩阵。形式如公式(2-3)所示，其中 ${}^A_B R$ 的上标 A 与下标 B 分别表示参考坐标系 $\{A\}$ 和 $\{B\}$ 。

$${}^A_B R = \begin{bmatrix} {}^A x_B & {}^A y_B & {}^A z_B \end{bmatrix} \quad (2-3)$$

绕 x 、 y 、 z 轴旋转 θ 角时，旋转矩阵分别如式 2-4 至 2-6，式中 c 表示余弦函数， s 表示正弦函数：

$$R(x, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c & -s \\ 0 & s & c \end{bmatrix} \quad (2-4)$$

$$R(y, \theta) = \begin{bmatrix} c & 0 & s \\ 0 & 1 & 0 \\ -s & 0 & c \end{bmatrix} \quad (2-5)$$

$$R(z, \theta) = \begin{bmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2-6)$$

通过位置描述与姿态描述即可获得刚体 Q 相对参考坐标系 $\{A\}$ 的位姿，刚体 Q 的位姿可表示为式(2-7)：

$$\{Q\} = \left\{ {}^A_B R \quad {}^A Q_B \right\} \quad (2-7)$$

(3) 平移坐标变换

设坐标系 $\{A\}$ 和 $\{B\}$ 的 X 、 Y 、 Z 三轴相互平行，两坐标原点不重合，如图 2-

7 所示。由上一小节可知可用 ${}^A Q$ 表示点 Q 相对于坐标系 $\{A\}$ 的位置， ${}^B Q$ 表示点 Q 相对于坐标系 $\{B\}$ 的位置。使用 ${}^A Q_B$ 表示坐标系 $\{B\}$ 相对于坐标系 $\{A\}$ 的平移矢量。 ${}^A Q$ 可由矢量相加得出，式(2-8)称为坐标平移方程。

$${}^A Q = {}^B Q + {}^A Q_B \quad (2-8)$$

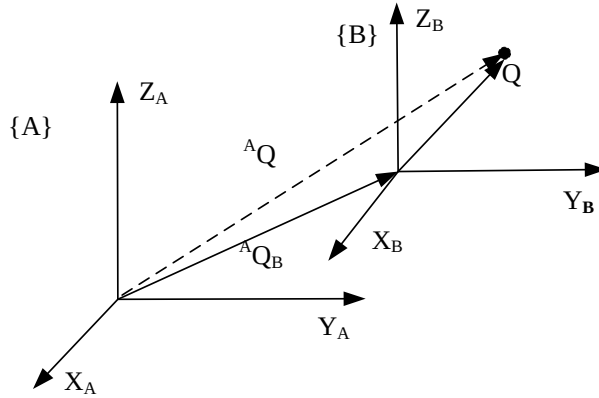


图 2-7 平移坐标变换

(4) 旋转坐标变换

如图 2-8 所示，坐标系 $\{A\}$ 和 $\{B\}$ 仅原点重合，点 Q 在两个坐标系下的向量表示方法为 ${}^A Q$ 和 ${}^B Q$ 。

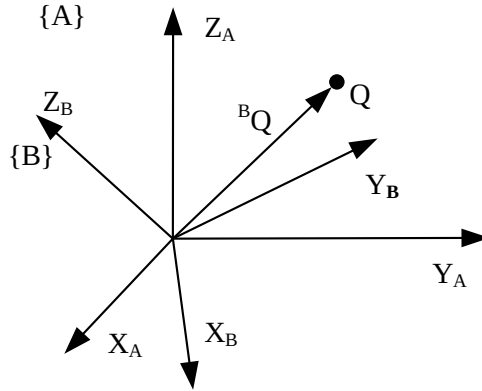


图 2-8 旋转坐标变换

可使用上一节中的旋转矩阵 ${}^A R_B$ 描述坐标系 $\{B\}$ 相对于坐标系 $\{A\}$ 的姿态，则点 Q 在 $\{A\}$ 坐标系下的向量 ${}^A Q$ 可表示为如式(2-9)：

$${}^A Q = {}^A R_B {}^B Q \quad (2-9)$$

(5) 复合坐标变换

如图 2-9 所示坐标系 $\{A\}$ 和 $\{B\}$ 原点与三个坐标轴都不重合，空间任意点 Q 在两个坐标系下的向量表示方法为 ${}^A Q$ 和 ${}^B Q$ ，点 Q 在坐标系 $\{A\}$ 和坐标系 $\{B\}$ 中的描述 ${}^A Q$ 和 ${}^B Q$ 有以下变换关系：

$${}^A Q = {}^A_B R {}^B Q + {}^A Q_B \quad (2-10)$$

式(2-10)可以看作坐标系平移与旋转的复合。假设有过渡坐标系 $\{C\}$ ，通过平移变换得到坐标系 $\{C\}$ ，坐标系 $\{B\}$ 可以看作坐标系 $\{C\}$ 通过旋转变换，变换过程如式(2-11)到(2-13)。

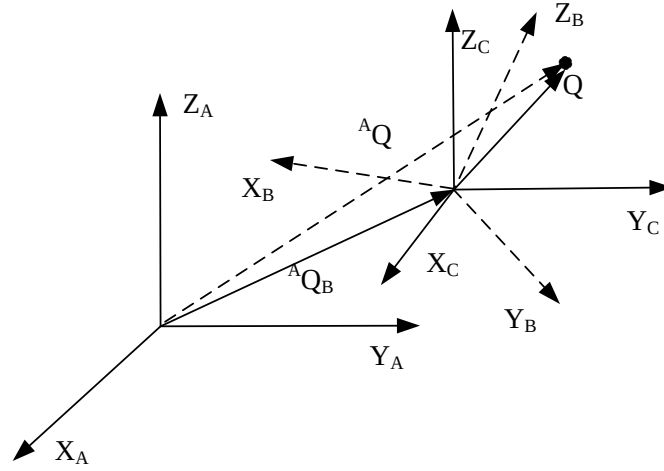


图 2-9 复合坐标变换

$${}^A Q = {}^C Q + {}^A Q_C \quad (2-11)$$

$${}^C Q = {}^C_B R {}^B Q = {}^A_B R {}^B Q \quad (2-12)$$

$${}^A Q = {}^C Q + {}^A Q_C = {}^A_B R {}^B Q + {}^A Q_B \quad (2-13)$$

2.3.2 机械臂 D-H 参数模型

在机械臂的运动学分析中，获取其末端执行器的精确位置和方向至关重要。为此，Denavit-Hartenberg (D-H) 参数法^[48]提供了一种系统性的方法来定义并解析机械臂各连杆的相对位置和姿态。这种方法通过在机械臂的每个连杆上建立坐标系，并运用齐次变换矩阵来描述坐标系之间的转换。D-H 参数法分为标准 D-H 参数法和改进 D-H 参数法，两种方法无太大差异。标准 D-H 参数法适用于开放链条类型的机器人模型，不适用于描述树状或闭链结构。相比之下，改进 D-H 参数法则无限制，因此本文使用改进 D-H 参数法。通过定义连杆长度 α_{n-1} 、连杆

扭角 α_{n-1} 、连杆偏移 d_n 和关节角 θ_n 四个参数，可以方便并准确地描绘出机器人连杆之间的空间关系，最终求解出末端执行器的确切位姿。

本文使用 Mirobot^[49] 六轴桌面小型机械臂完成分拣任务，机械臂外形与关节如图 2-10 所示。根据改进 D-H 参数法进行建模，各连杆坐标系如图 2-11 所示，D-H 参数如表 2-2 所示。

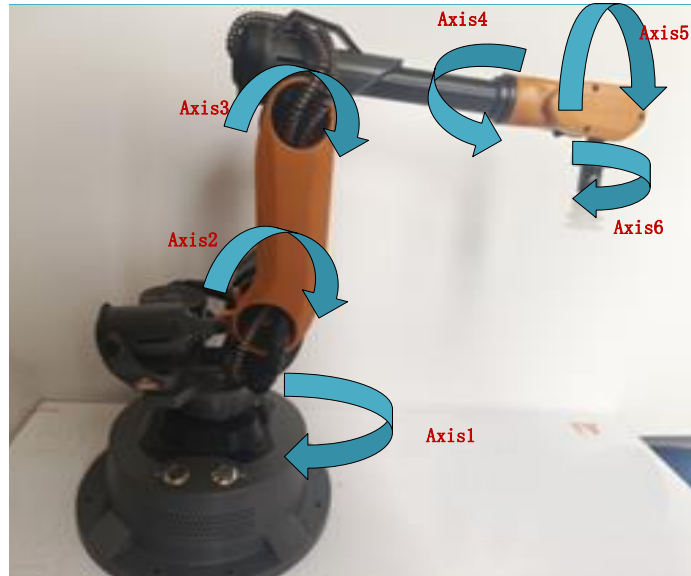


图 2-10 Mirobot 六轴机械臂

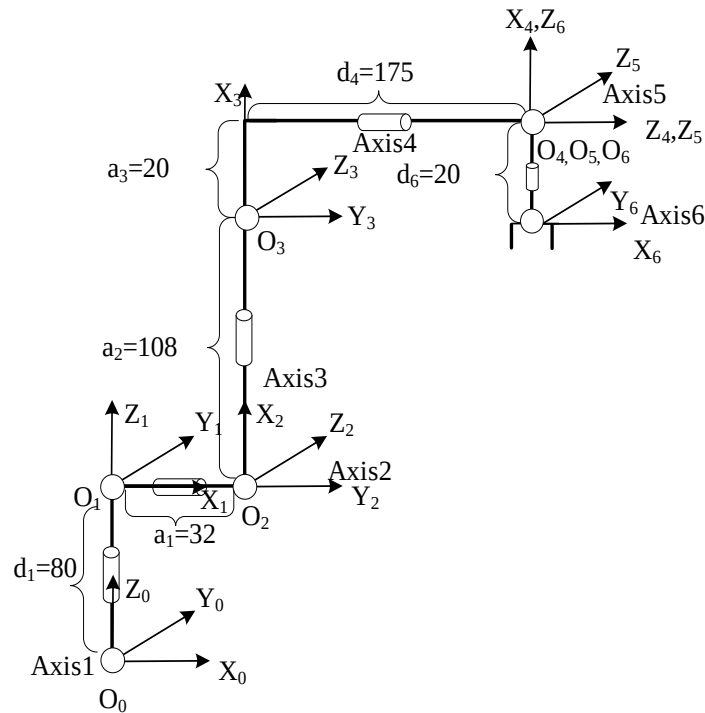


图 2-11 关节坐标系

表 2-2 改进 D-H 参数表

i	α_{n-1}	a_{n-1}	d_n	θ_n
1	0	0	d_1	θ_1
2	$-\pi/2$	a_1	0	θ_2
3	0	a_2	0	θ_3
4	$-\pi/2$	a_3	d_4	θ_4
5	$\pi/2$	0	0	θ_5
6	$\pi/2$	0	d_6	θ_6

根据建立连杆坐标系以及改进 D-H 参数表，可以获得表述相邻连杆之间相对变换关系的齐次变换矩阵 ${}^{n-1}_n T$ 如式(2-14)所示：

$${}^{n-1}_n T = \begin{bmatrix} \cos \theta_n & -\sin \theta_n & 0 & a_n \\ \sin \theta_n \cos \alpha_{n-1} & \cos \theta_n \sin \alpha_{n-1} & -\sin \alpha_{n-1} & -\sin \alpha_{n-1} d_n \\ \sin \theta_n \sin \alpha_{n-1} & \cos \theta_n \sin \alpha_{n-1} & \cos \alpha_{n-1} & \cos \alpha_{n-1} d_n \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-14)$$

2.3.3 机械臂正运动学分析

已知 Mirobot 的改进 D-H 参数表，将 D-H 参数分别带入(2-14)，能够分别求解出六个关节的齐次变换矩阵，如式(2-15)：

$$\begin{aligned} {}^0_1 T &= \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^1_2 T = \begin{bmatrix} c_2 & -s_2 & 0 & a_1 \\ 0 & 0 & 1 & 0 \\ -s_2 & c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^2_3 T = \begin{bmatrix} c_3 & -s_3 & 0 & a_2 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\ {}^3_4 T &= \begin{bmatrix} c_4 & s_4 & 0 & a_3 \\ 0 & 0 & 1 & d_4 \\ -s_4 & c_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^4_5 T = \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, {}^5_6 T = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ 0 & 0 & -1 & d_6 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2-15)$$

式中， c_1 为 $\cos \theta_1$ ， s_1 为 $\sin \theta_1$ ，以此类推。将上式各变换矩阵相乘，联立可得机械臂末端相对机械臂基座的矩阵为式(2-16)：

$${}^0_6 T = {}^0_1 T \cdot {}^1_2 T \cdot {}^2_3 T \cdot {}^3_4 T \cdot {}^4_5 T \cdot {}^5_6 T = \begin{bmatrix} n_x & a_x & o_x & p_x \\ n_y & a_y & o_y & p_y \\ n_z & a_z & o_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-16)$$

式中， p 表示机械臂末端的位置向量， n 、 o 、 a 分别表示机械臂末端相对于自身

坐标系的向量坐标。上式中各参数具体表达式如下所示：

$$\begin{aligned}
n_x &= -c_6(s_5(c_1c_2s_3 + c_1c_3s_2) - c_5(s_1s_4 + c_4(c_1c_2c_3 - c_1s_2s_3))) - s_6(c_4s_1 - s_4(c_1c_2c_3 - c_1s_2s_3)) \\
n_y &= s_6(c_1c_4 + s_4(c_2c_3s_1 - s_1s_2s_3)) - c_6(s_5(c_2s_1s_3 + c_3s_1s_2) + c_5(c_1s_4 - c_4(c_2c_3s_1 - s_1s_2s_3))) \\
n_z &= -c_6(s_5(c_2c_3 - s_2s_3) + c_4c_5(c_2s_3 + c_3s_2)) - s_4s_6(c_2s_3 + c_3s_2) \\
a_x &= s_6(s_5(c_1c_2s_3 + c_1c_3s_2) - c_5(s_1s_4 + c_4(c_1c_2c_3 - c_1s_2s_3))) - c_6(c_4s_1 - s_4(c_1c_2c_3 - c_1s_2s_3)) \\
a_y &= s_6(s_5(c_2s_1s_3 + c_3s_1s_2) + c_5(c_1s_4 - c_4(c_2c_3s_1 - s_1s_2s_3))) + c_6(c_1c_4 + s_4(c_2c_3s_1 - s_1s_2s_3)) \\
a_z &= s_6(s_5(c_2c_3 - s_2s_3) + c_4c_5(c_2s_3 + c_3s_2)) - c_6s_4(c_2s_3 + c_3s_2) \\
o_x &= c_5(c_1c_2s_3 + c_1c_3s_2) + s_5(s_1s_4 + c_4(c_1c_2c_3 - c_1s_2s_3)) \\
o_y &= c_5(c_2s_1s_3 + c_3s_1s_2) - s_5(c_1s_4 - c_4(c_2c_3s_1 - s_1s_2s_3)) \\
o_z &= c_5(c_2c_3 - s_2s_3) - c_4s_5(c_2s_3 + c_3s_2) \\
p_x &= a_1c_1 - d_6(c_5(c_1c_2s_3 + c_1c_3s_2) + s_5(s_1s_4 + c_4(c_1c_2c_3 - c_1s_2s_3))) + a_3(c_1c_2c_3 - c_1s_2s_3) \\
&\quad - d_4(c_1c_2s_3 + c_1c_3s_2) + a_2c_1c_2 \\
p_y &= a_1s_1 - d_6(c_5(c_2s_1s_3 + c_3s_1s_2) - s_5(c_1s_4 - c_4(c_2c_3s_1 - s_1s_2s_3))) + a_3(c_2c_3s_1 - s_1s_2s_3) \\
&\quad - d_4(c_2s_1s_3 + c_3s_1s_2) + a_2c_2s_1 \\
p_z &= d_1 - a_2s_2 - a_3(c_2s_3 + c_3s_2) - d_4(c_2c_3 - s_2s_3) - d_6(c_5(c_2c_3 - s_2s_3) - c_4s_5(c_2s_3 + c_3s_2))
\end{aligned}$$

2.3.4 机械臂逆运动学分析

机械臂逆运动学是正运动学相反的过程，即已经知道机械臂末端执行器的位置和姿态，求计算一系列的关节角度。求解运动学逆问题可以采用两种方法：一种是解析解，另一种是数值解。解析解通用性稍差，不同类型的机械臂使用的特定的解析解，但是解析解相较于数值解法速度更快。解析解主要分为代数法和几何法。数值解法的应用不局限于某特定类型的机械臂结构，具有更高的通用性。但是对于复杂的机械臂结构，需要大量迭代才能得到结果。同时数值法对初值要求较高，需要一定时间来寻找合适的初值，也会出现求不出解的情况。数值解包括符号消去法、延拓法和迭代法等方法。

解析解的求取与机器人设计构型有关，由 Mirobot 的机械结构可知，其后三关节轴相交于一点，符合 Pieper 准则^[50]。符合 Pieper 准则机械臂逆运动学问题可以通过解析解法来求解。本文选择解析解法对机械臂运动学问题进行求解，以下是求解过程：

已知机器人末端执行器位置矩阵如式(2-17)：

$${}^0_6T = \begin{bmatrix} n_x & a_x & o_x & p_x \\ n_y & a_y & o_y & p_y \\ n_z & a_z & o_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-17)$$

可将上式转化为:

$$[{}^1_0T]^{-1} {}^0_6T = {}^1_2T {}^2_3T {}^3_4T {}^4_5T {}^5_6T \quad (2-18)$$

由式(2-18)中对应元素相等可依次求出各个关节角度所对应的表达如式(2-19)所示:

$$\begin{aligned} \theta_1 &= \arctan(p_y, p_x) - \arctan(-(d_2 + d_3 + d_4), \pm \sqrt{p_x^2 + p_y^2 - (d_2 + d_3 + d_4)^2}) \\ \theta_5 &= \arctan[s_5, s_1 o_x - c_1 o_y] \\ \theta_6 &= \arctan\left[\frac{-c_1 a_y + s_1 a_x}{s_5}, \frac{-s_1 n_x + c_1 n_y}{s_5}\right] \\ \theta_2 &= \arctan(2a_2(c_1 p_x + s_1 p_y + d_5 c_{234}), -2a_2(p_z - d_1 - d_5 c_{234})) \\ &\quad - \arctan(B, \pm \sqrt{C}) \\ \theta_2 + \theta_3 &= \arctan[D, F] \\ \theta_2 + \theta_3 + \theta_4 &= \arctan 2\left[\frac{o_z}{s_5}, \frac{c_1 o_x + s_1 o_y}{s_5}\right] \end{aligned} \quad (2-19)$$

其中 B、C、D、F 参数如式(2-20)所示:

$$\begin{aligned} B &= (c_1 p_x + s_1 p_y + d_5 s_{234})^2 + (p_z - d_1 - d_5 c_{234})^2 + a_2^2 - a_3^2 \\ C &= 4a_2^2(c_1 p_x + s_1 p_y + d_5 s_{234})^2 + 4a_3^2(p_z - d_1 - d_5 c_{234})^2 - B \\ D &= \frac{p_z - d_1 - d_5 c_{234} - a_2 s_2}{a_3} \\ F &= \frac{c_1 p_x + s_1 p_y + d_5 s_{234} - a_2 c_2}{a_3} \end{aligned} \quad (2-20)$$

2.4 本章小结

在本章中详细探讨了卷积神经网络相关理论和机械臂运动学建模。卷积神经网络部分介绍了卷积神经网络结构, 主要包括网络中的卷积层、激活函数、池化层和激活函数。机械臂运动学建模部分对机械臂的位姿描述与坐标变换进行了介绍, 通过改进 D-H 参数法对机械臂进行建模并进行正逆运动学求解。

第 3 章 基于深度学习的垃圾检测算法研究

3.1 引言

在实现垃圾自动分拣之前，首先要进行目标检测，以获得目标的类别信息与位置信息。本章首先介绍了目前主流两阶段目标检测算法与单阶段目标检测算法的基本原理。其次详细介绍了 YOLOv5s 目标检测算法的结构，并针对 YOLOv5s 算法在垃圾检测任务上的不足进行了改进。通过实验验证了改进后算法的有效性和普适性。

3.2 目标检测算法

3.2.1 两阶段目标检测算法

2014 年，Girshick 提出了 RCNN 算法，结构如图 3-1 所示。该算法首先使用 Selective Search 方法提取候选框，从每张图像中提取大约 2000 个候选区域。由于 CNN 通常接受固定尺寸的输入，并且输入图像应该具有相同的尺寸和宽高比，因此需要通过缩放处理将所有这些候选区域调整为相同的大小，以匹配 CNN 的输入。随后，使用 CNN 对每个候选区域进行特征提取，并将提取的图像特征输入到 SVM 分类器中进行分类。在每个类别上使用非极大值抑制的方法对候选框进行筛选。最后，通过边界框回归对筛选后的候选框位置进行微调。在 VOC2007 数据集上，平均准确率(mAP)达到 58.5%。尽管 RCNN 相对于传统目标检测方法取得了很大的进步，但存在目标检测速度慢、计算量大，效率低等问题。

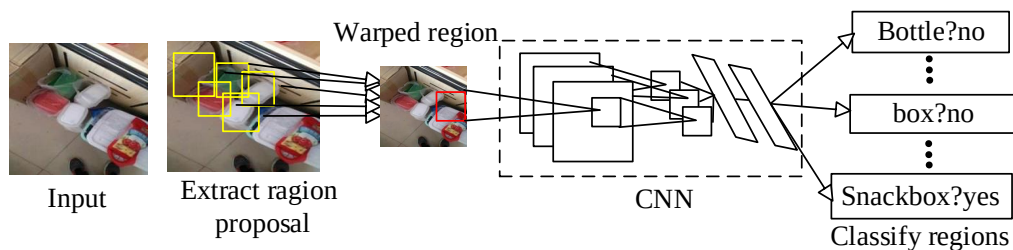


图 3-1 RCNN 网络结构图

继 RCNN 之后，Girshick 在 RCNN 的基础上提出了 Fast RCNN 算法，其结

构如图 3-2 所示。相较于 RCNN，Fast RCNN 主要有三个方面的改进。首先，解决了 RCNN 中需要对每个类别都训练一个 SVM 分类器，使用 Softmax 函数代替 SVM 进行分类。此外，引入了 ROI pooling 层，ROI pooling 层通过对每个候选区域应用池化操作，统一特征尺寸以便分类和边界框回归。同时优化了 Fast RCNN 的优化了处理流程，将分类和 bbox 回归任务结合在一起。但是 Fast RCNN 仍然不能满足实时检测的需求。

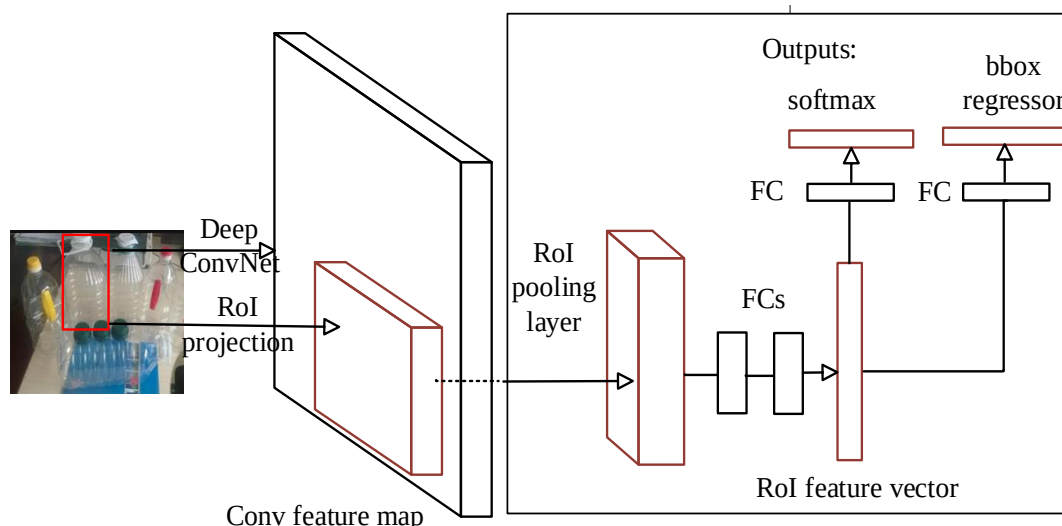


图 3-2 Fast RCNN 网络结构图

Faster RCNN 的主要创新之一是引入了区域建议网络（Region Proposal Network, RPN）。RPN 网络直接在模型中生成候选区域，解决了 Fast RCNN 中需要依赖 Selective Search 来产生候选区域的复杂和耗时问题。Faster RCNN 结构如图 3-3 所示，算法检测步骤大致可以概括为以下几步：首先，利用卷积层处理输入图像进行特征提取，产生特征图；其次，特征图被送入 RPN，RPN 负责生成初级的目标候选区域及相应的候选框信息；接着，通过 RoI pooling 对 RPN 提出的候选区域进行池化；最后，全连接层利用池化后的特征对候选区域进行分类，并进行边界框回归，以确定最终的检测框的位置和类别得分。此外，Faster RCNN 通过引入 Anchor box 概念，进一步提升了检测性能。

尽管 Faster RCNN 取得了显著的性能提升，但仍然存在一些缺点。在处理高分辨率图像时，RPN 和后续的检测网络需要更多的计算资源，导致处理速度变慢。此外，算法的效果很大程度上依赖于 Anchor box 的设计和数量。

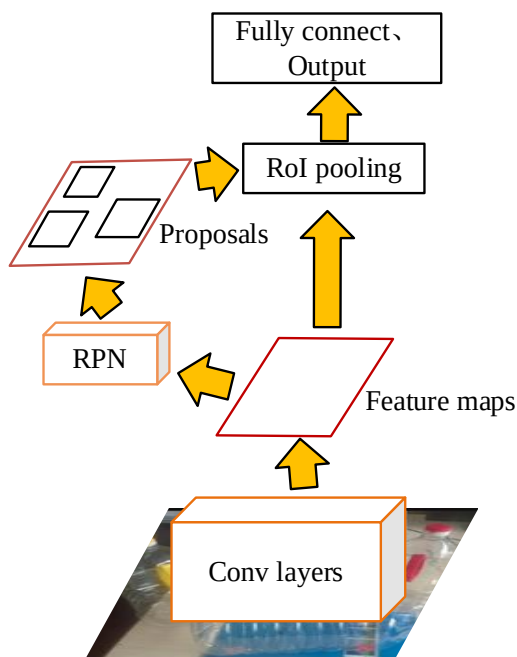


图 3-3 Faster RCNN

3.2.2 单阶段目标检测算法

SSD 算法融合了 Faster RCNN 与 YOLO 算法的特点，该模型使用了 YOLO 算法中使用的回归思想，并借鉴了 Faster RCNN 检测模型中提出的 Anchor box 概念。SSD 对不同尺度的特征图来进行检测，在大尺度特征图上检测小目标，在小尺度特征图上检测大目标，显著提高各个尺度的检测能力。SSD 应用了多种尺度比和横纵比的先验框用以提升不同形状目标的检测能力。这些特性使得 SSD 算法在目标检测任务中具有广泛的应用前景。

YOLO 算法是由 Joseph Redmon 等人提出的一种单阶段目标检测方法。YOLO 算法将目标检测问题转化为回归问题，对于输入图像只需一次网络计算，即可直接获取图像中目标的边界框和类别概率。尽管 YOLOv1 在速度上取得了显著进展，但其在小目标检测和定位精度上仍有提升空间。

后续版本 YOLOv2(又称 YOLO9000)采用 Darknet-19 作为特征提取网络。在 YOLO 原有基础上，YOLOv2 引入了 Batch Normalization 改善收敛性并防止过拟合。使用 Anchors Box 机制、聚类设置先验框等进一步提高的精度。同时通过多尺度训练策略增强了模型对不同大小目标的检测能力。YOLOv2 相较于上一代 YOLO 算法在保持实时检测速度的同时，对小目标的检测上精度有所提升，但相

较于两阶段算法仍有提升的空间。

YOLOv3 进一步强化了模型的性能，使用 DarkNet-53 用于取代 DarkNet-19 作为主干用来提升特征提取能力。YOLOv3 在三种不同尺度上预测目标，以适应多尺度目标检测，并采用逻辑回归代 Softmax 方法，允许每个边界框预测多个类别。YOLOv3 在小物体检测方面表现出显着改进，并且在检测实时性上表现非常出色。

YOLOv4 在前几代算法的基础上做出了多项改进，包括引入了更加高效的骨干网络 CSPDarknet53，使用了 Mish 激活函数，集成 SPPNet 和 PANet 作为网络的颈部结构，以及引入了 CIOU 作为损失函数。YOLOv4 算法极大地提升了检测的准确性和效率。

YOLOv5 融合了 YOLOv1 到 YOLOv4 算法的优点，能够在保持现有模型检测精度的同时大幅提高检测速度，是目标检测领域的最佳选择之一。YOLOv5 共 4 种结构，包括 YOLOv5s、YOLOv5m、YOLOv5l 和 YOLOv5x，模型参数差异如表 3-1 所示。其中 YOLOv5s 的网络深度浅，特征图宽度最小，其实时性、计算量及模型大小在这四种结构中表现最好。

表 3-1 模型参数

model	FLOPS/B	FPS	Params/M
YOLO5s	13.2	476	7.1
YOLOv5m	39.4	333	21.8
YOLOv5l	88.1	256	47.8
YOLOv5x	166.4	164	89

改进模型基础算法采用了 YOLOv5s 6.0 版本。此版本 YOLOv5 的基本框架可以分为四个部分：输入(Input)、骨干网络(Backbone)、颈部网络(Neck)以及头部(Head)模型，YOLOv5s 网络结构如图 3-4 所示。

输入部分：(1)使用 Mosaic 数据增强方法，拼接四张图片，对拼接后的图片进行随机缩放、裁剪、排列和拼接。均衡大、中、小目标的数量，对小目标的检测有一定程度帮助，提高了网络的训练效率和鲁棒性。(2)YOLOv5 会针对不同任务，不同数据集人工标注的锚框预设锚框的大小。(3)自适应图像缩放，将原始图片均匀缩放到标准尺寸，自适应添加黑边进行补充，避免因图像拉伸或压缩造成的失真。

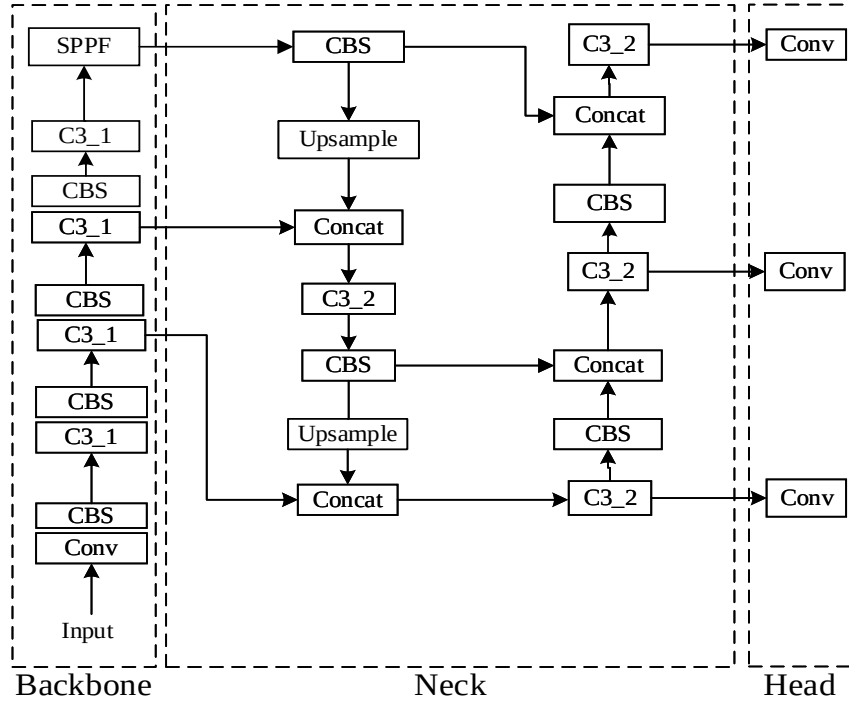


图 3-4 YOLOv5s 结构图

YOLOv5s 的主干网络主要用于对输入图像的特征提取，包括了 CBS 模块，C3 模块和 SPPF 模块，模块结构如图 3-5 所示。

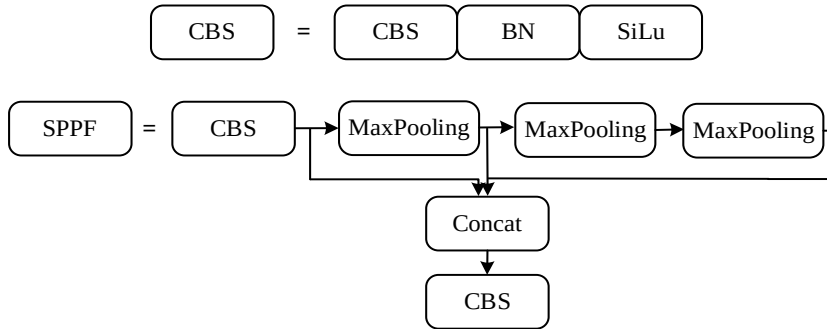


图 3-5 CBS 结构与 SPPF 结构图

CBS 模块由卷积、归一化和 SiLU 激活函数组成。SPPF 模块是从前文所提的 SPP 模块改进而来。该模块处于主干的末端，对输入的特征图连续进行三次 MaxPooling 操作，并将得到的结果拼接在一起。通过融合不同尺寸的 MaxPool，增大感受野的同时，实现局部特征和全局特征融合。

C3 模块是对输入图像进行特征提取的重要模块，结构如图 3-6(a)所示，上一层输入的特征图分为两部分，一半经由 CBS 模块后不做其他操作，一半经过相同的 CBS 模块后经过 Bottleneck 模块。将处理后的特征图进行拼接。Bottleneck 模块有两种结构，如图 3-6(b)所示，有残差边的 Bottleneck1 位于主干中，

Bottleneck2 则位于颈部网络。特征图经过 C3 结构只进行特征提取，大小与通道数不变。

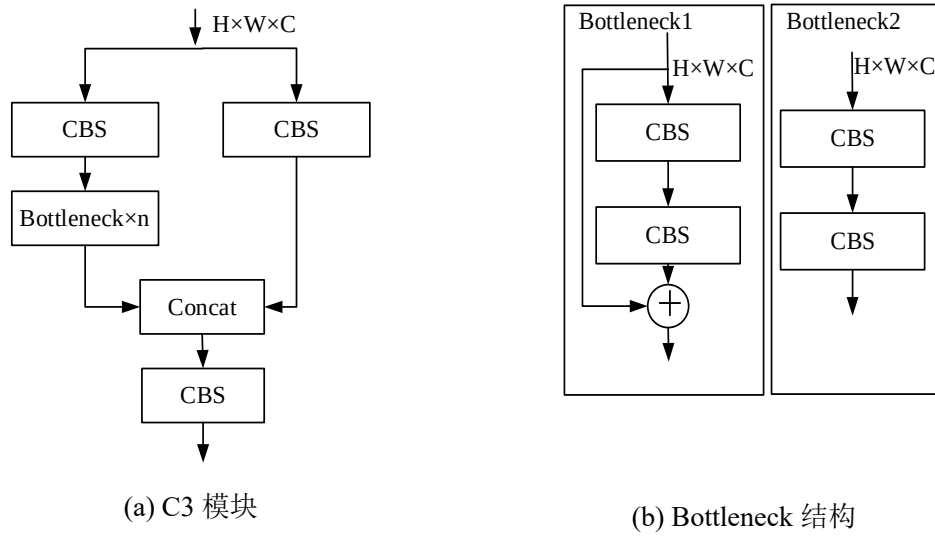


图 3-6 C3 模块及 Bottleneck 结构图

颈部网络主要用于特征融合，结构如图 3-7 所示。主干网络在特征提取中会生成 80×80 、 40×40 和 20×20 三个尺度的特征图。通过特征金字塔网络(FPN)和路径聚合网络(PANet)实现不同尺度的融合。FPN 将包含丰富语义信息的特征图从顶部向底部传递。同时，PANet 将低层包含丰富位置信息的特征从低到高传递。FPN 和 PANet 的结合增强了 Neck 网络的特征融合能力，能够获得更多的上下文信息，减少信息丢失。

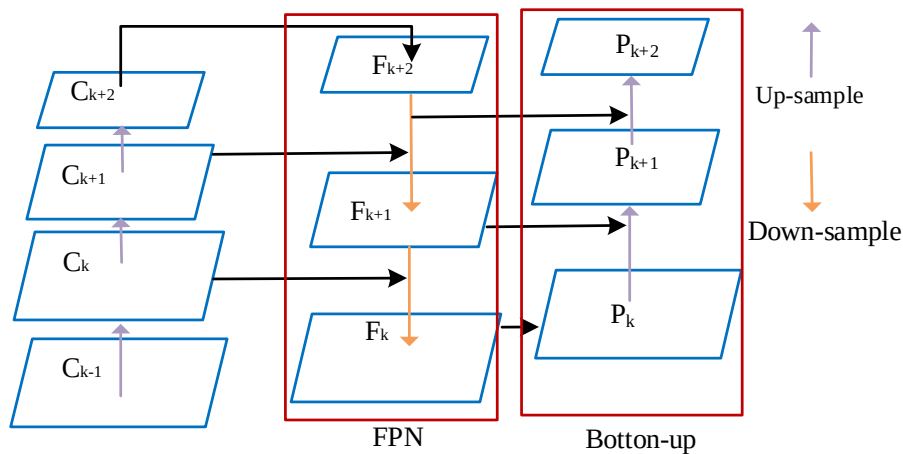


图 3-7 颈部网络结构图

YOLOv5s 头部包括 3 种规格的检测头，分别用于检测不同尺度的目标。检测头内部含有卷积层和预测层，以产生目标的位置、类别概率和中心点偏移量。

3.3 基于改进 YOLOv5s 的垃圾目标检测算法

本节在目前比较成熟单阶段 YOLOv5s 模型基础上进行优化改进。在主干使用 C3_DG 模块，提高了特征利用率，加深网络深，提高主干特征提取能力；通过添加 I-BIC 模块，对多尺度融合网络进行优化，减少小尺寸目标特征损失，改善各尺寸目标检测能力，同时构建 C3Ghost 模块，以降低模型参数量；最后引入 SimAM 注意力机制，提升模型在复杂背景下检测能力和抗干扰能力。通过实验对比表明改进后模型改善了复杂背景、目标尺寸多样条件下的检测能力，提升了模型的鲁棒性。

3.3.1 C3_DG 与 C3Ghost 构建

密集连接是 DenseNet^[51]中提出的卷积神经网络结构，图 3-8 为密集连接结构图。密集连接与其他网络结构主要区别是特征重用，特征重用意味着每一层的输入为其前面所有卷积层输出的合并，并将当前层的输出传递到后面的所有层，特征重用大大减少了参数量，提高了参数利用率。同时，密集连接使每一层都能够直接利用损失函数的梯度以及最开始的输入信息，可以从低级特征和前期产生的高级特征中受益。

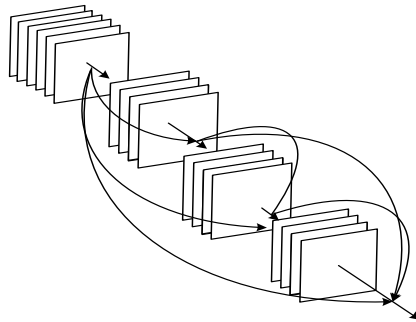


图 3-8 密集连接

GhostNet^[52]是华为团队提出的一种轻量化 CNN 网络，其中所提出的 Ghost 模块在实现网络轻量化中得到广泛应用。图 3-9 为 Ghost 模块结构图。Ghost 模块通过三步操作获得与普通卷积相同数量的特征图。首先，使用核大小为 1×1 的卷积对输入特征图进行降维处理，并得到一部分特征图；其次，通过使用例如深度可分离卷积(Depth-wise Convolution)从第一部分特征图中生成第二部分特征图；

最后，将这两部分特征图进行 Concat 操作，拼接在一起并输出。GhostConv 能够在几乎不影响检测精度的同时降低普通卷积的冗余性，减少参数量与计算量。

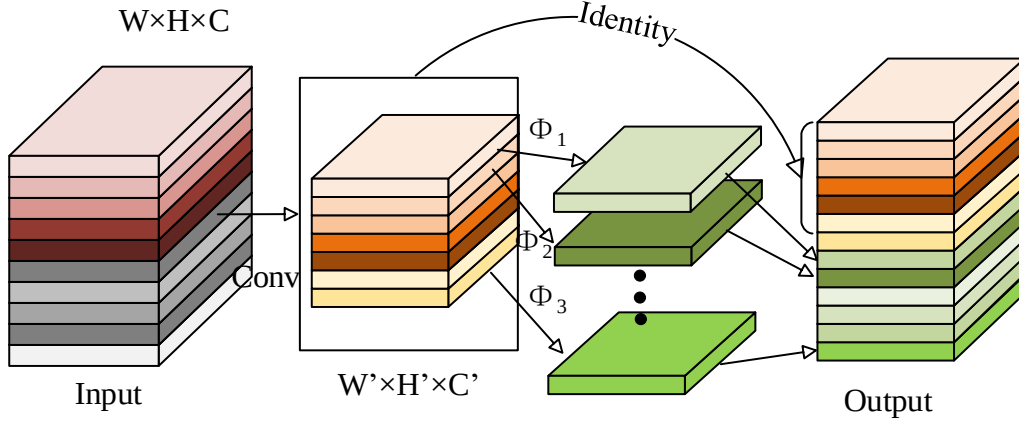


图 3-9 Ghost 模块结构图

对于高度、宽度和输入通道数分别为 h 、 w 和 c 的输入特征图 $X \in R^{h \times w \times c}$ ，通过常规卷积生成 n 个特征图的运算公式如(3-1)公式所示：

$$Y = X * f + b \quad (3-1)$$

式中， b 为偏差， $*$ 表示卷积运算， f 表示卷积核，大小为 $c \times k \times k$ ， Y 表示输出特征图。有 n 个卷积核参与计算，得到的 n 通道的特征图的 $Y \in R^{h' \times w' \times n}$ ， h' 、 w' 分别为输出特征图的高和宽。通过常规卷积操作计算量大小如公式(3-2)所示：

$$FLOPs = c \times k \times k \times n \times h' \times w' \quad (3-2)$$

特征图经过一次卷积操作，输出将会产生 n 个特征图。Ghost 模块仅选取部分 m 个特征图($m \leq n$)特征图进行标准卷积操作，运算如公式(3-3)所示：

$$Y' = X * f' + b \quad (3-3)$$

通过对上一步卷积生成的 m 个特征图进行线性变换，获得剩余新特征图。为方便计算，省略项偏差 b ，变换公式为：

$$y_{i,j} = \phi_{i,j}(y'_i) \quad \forall i = 1, \dots, m, \quad j = 1, \dots, s \quad (3-4)$$

式中， y'_i 是 Y' 中的第 i 个原始特征图， $\phi_{i,j}$ 表示线性计算， $y_{i,j}$ 代表着第 j 个线性变化特征图， s 为经过变换得到的特征图。经过变换后可获得 n 个特征图。Ghost 模块的计算量计算公式如式(3-5)所示：

$$FLOPs = c \times k \times k \times \frac{n}{s} \times h' \times w' + (s-1) \times d \times d \times \frac{n}{s} \times h' \times w' \quad (3-5)$$

Ghost 模块通过对部分特征图进行卷积同时结合线性变换，参数量显著减

少。结合式可得模型的压缩比为如式(3-6)，Ghost 模块能够在几乎不影响检测精度的同时降低普通卷积的冗余性，降低计算量。

$$R_c = \frac{c \times k \times k \times n \times h' \times w'}{c \times k \times k \times \frac{n}{s} + (s-1) \times \frac{n}{s} \times d \times d \times h' \times w'} \approx \frac{s \times c}{s \times c - 1} \approx s \quad (3-6)$$

步长为 1 的 Ghost Bottleneck 模块由 2 个 Ghost 模块与 1 个残差边组成，结构如图 3-10 所示。第一个 Ghost 模块用作扩展层，增加通道数，后面引入批量归一化(BN)和 ReLU 非线性激活；第二个 Ghost 模块减少通道数，用以与残差连接匹配，第二个 Ghost 模块之后不使用 ReLU 激活函数。

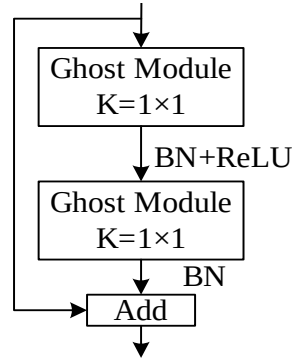


图 3-10 Ghost Bottleneck 结构图

基于密集连接网络思想与 Ghost Bottleneck 结构设计了两种瓶颈结构代替原始结构，图 3-11(a)为 C3_DG 结构。C3_DG 中主要由 4 个 Ghost 模块卷积串联组成瓶颈结构，使用密集连接方式组合 4 个 Ghost 模块。为了保留密集连接的诸多优点，简化结构，并更好的嵌入 C3 模块中，使用 Add 操作对不同层的特征图进行融合，不改变通道数。C3_DG 提高了特征利用率，加深了网络深度，提升主干特征提取能力，不过多增加参数量。如图 3-11(b)，在颈部使用步长为 1 的 Ghost Bottleneck 构建 C3Ghost，通过低成本的操作减少冗余特征图的产生，轻微降低网络性能，但能够有效降低参数量。

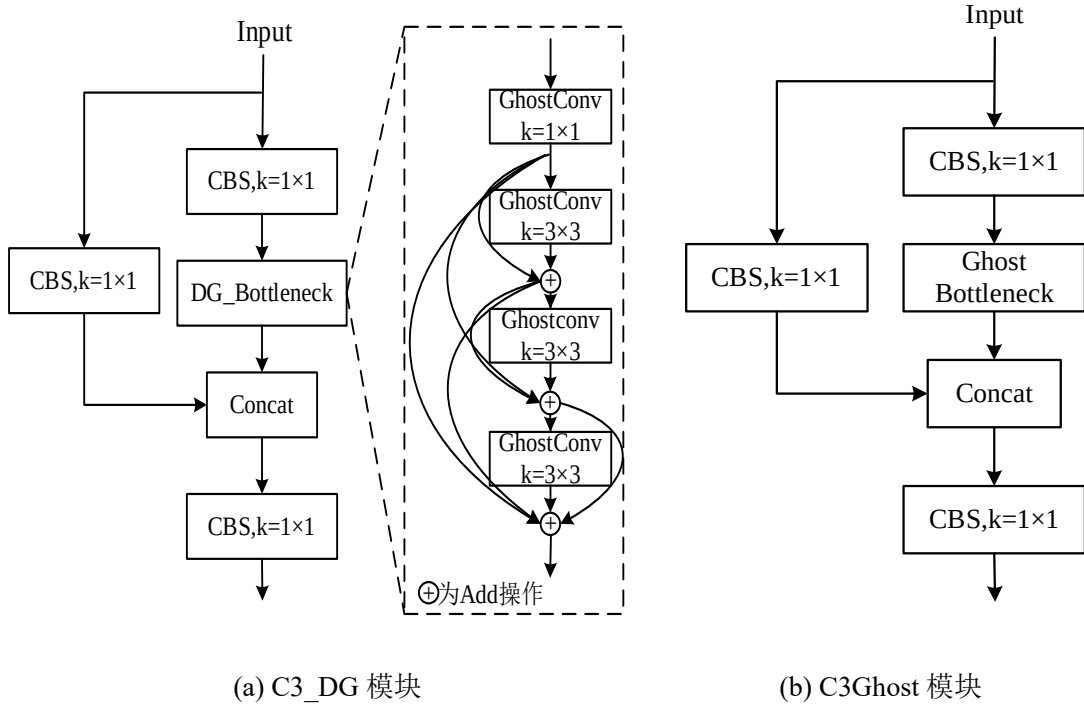


图 3-11 C3_DG 模块和 C3Ghost 模块

3.3.2 多尺度特征融合优化

在 FPN 结构中，随着网络的加深，较浅特征中包含的小尺寸目标特征和空间信息有所丢失。Li 等人^[53]提出的双向联结(Bidirectional Concatenate, BiC)模块，如图 3-12 所示，通过在自上而下的传输过程中加入浅层信息流，使浅层特征更好的参与多尺度特征融合，增强融合特征的表达能力，能够帮助保留更准确的定位信号。

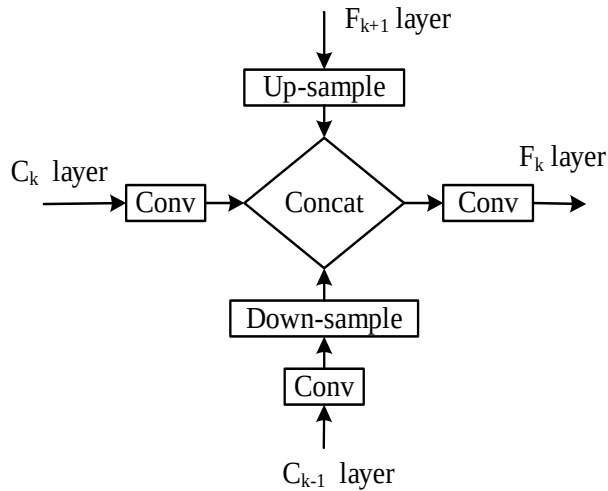


图 3-12 BiC 模块

在 BiC 模块的基础上，为了进一步减少小尺寸目标特征的丢失，引入 Involution 算子进行下采样，提出了 I-BiC(Involution- Bidirectional Concatenate)模块。Involution 算子^[54]与普通卷积特性相反，具有空间特异性(spacital-specific)和通道无关性(channel-agnostic)。Involution 算子能够自适应地将权重分配给不同位置并优先考虑具有不同信息量的特征，更好捕获空间域中特征。同时，改善了传统卷积为减少计算量而使用小卷积内核导致感受野受限的问题。Involution 算子实现过程如图 3-13，图中 $G=1$ 。

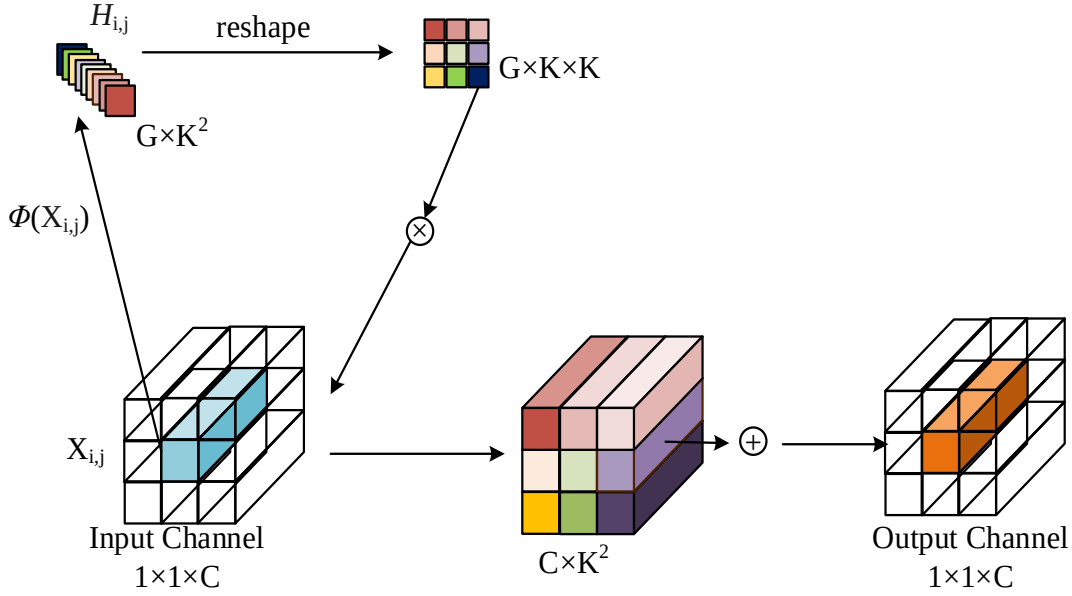


图 3-13 Involution 核生成过程

首先将输入通道数 C 分为 G 组，同一组内使用相同的 Involution 核，可将某个坐标 (i, j) 对应像素特征向量 $X_{i,j} \in R^{1 \times 1 \times C}$ 生成的 Involution 核表示为公式(3-7)：

$$H_{i,j} = \phi(X_{i,j}) = W_1 \phi(W_0 X_{i,j}) \quad (3-7)$$

其中 ϕ 为 Involution 核生成函数， $W_0 \in R^{\frac{C}{r} \times C}$ 和 $W_1 \in R^{(K \times K \times G) \times \frac{C}{r}}$ 为线性变换，通过 1×1 卷积实现， ϕ 包括 BN 层和 ReLU 函数， K 为 Involution 核的大小， r 为通道压缩比。特征图上某一像素特征向量 $X_{i,j} \in R^{1 \times 1 \times C}$ 通过核生成函数变为 $H_{i,j} \in R^{G \times K^2}$ ，再 reshape 展开成 Involution 核的形状，经过上述操作就得到了该像素点的 Involution 核。最后利用产生的 $K \times K \times G$ 的 Involution 核与输入通道进行乘法和加法运算，即可得到输出特征图。

融合 I-BiC 模块的多尺度融合网路如图 3-14 所示。I-BiC 模块可以在自上而

下的传输路径中引入自下向上的信息流,使模型在保留更多浅层小尺度目标特征与空间信息的同时,使各尺度特征充分融合,兼顾不同大小尺度目标的检测精度。I-BiC 模块实现具体步骤为: C_{k-1} , C_k 层特征图使用 1×1 卷积统一放缩到相同维度; C_{k-1} 层特征图使用 3×3 步长为 2 的 Involution 算子进行下采样,获得与 C_k 层相同尺寸的特征图;最后将经过上采样的 F_{k+1} 层与以上两个部分进行 Concat 操作。

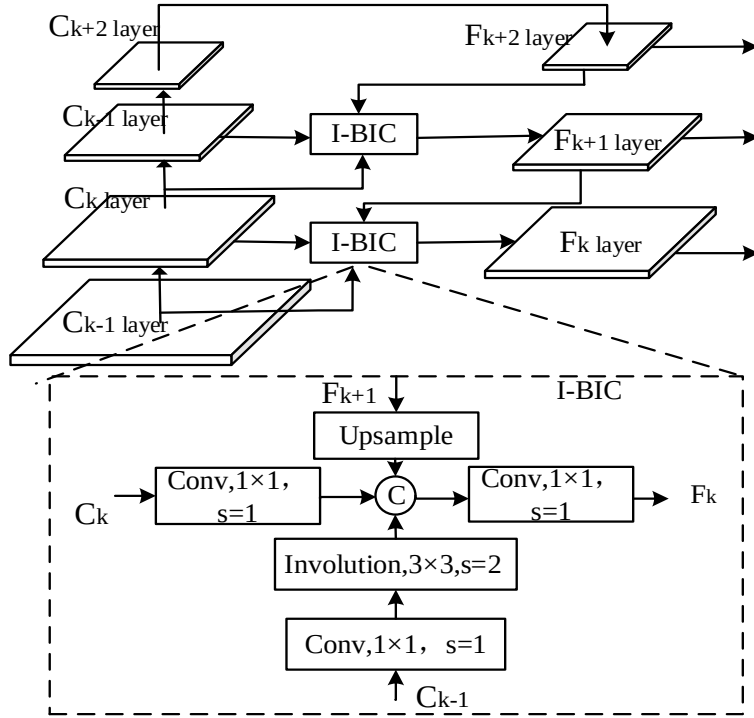


图 3-14 I-BiC 模块

3.3.3 注意力机制

注意力机制在计算机视觉中得到广泛应用,其能够明显提升目标检测、图像分类和图像分割等任务的网络模型性能。输入的图像中不仅包含检测的目标,还包含了背景噪声或遮挡,通过有选择地关注图像的某些部分,注意力机制可以帮助模型专注于相关特征,而忽略干扰信息,可以提高准确性并缩短训练时间。注意力机制大体上可以分为三类:通道注意力、空间注意力和混合注意力机制。SENet^[55]首次提出了一种通道注意力的有效机制,并在各种深度 CNN 中表现出了出色的性能。SENet 的核心概念是引入 Squeeze and Excitation(SE)的自注意力模块,结构如图 3-15。通过自动学习的方式,获得输入特征图每个通道的重要程度,将权重赋值给各通道,使神经网络关注权重高的通道。SE 模块分为两个阶

段：将这些权重与原始特征图相乘，突出最相关的通道，同时抑制信息量最小的通道。

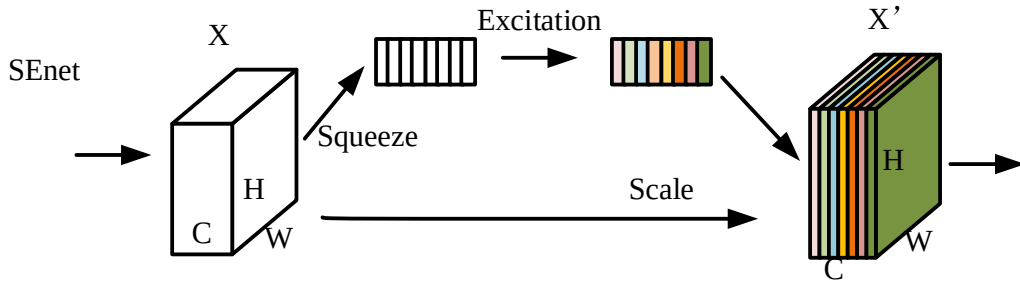


图 3-15 SENet 注意力机制

CBAM(Convolutional Block Attention Module)模块^[56]结合空间和通道注意力机制，相比只关注通道能带来更好的结果，结构如图 3-16 所示。CBAM 首先对输入特征图使用全局最大池化和全局平均池化，分别从不同角度提取全局特征。将这两种池化得到的特征图输入到一个多层感知器(MLP)中，MLP 通过两层全连接层对特征进行进一步处理。接下来，MLP 的输出在通道维度上进行融合，通过 Sigmoid 激活函数计算出每个通道的权重系数。最终，这些权重系数与原始输入特征图逐通道相乘，用于强化或弱化某些通道的特征。对于空间注意力，CBAM 先对输入特征图的每个通道执行最大池化和平均池化操作，这两种操作分别捕捉空间上的不同特征响应。将这两种池化的结果在通道维度上堆叠，得到一个融合了最大值和平均值信息的特征图。通过卷积层调整通道数，然后使用 Sigmoid 函数得到空间维度上的权重图。最后，将这个权重图与原始输入特征图相乘，实现对空间维度上细节特征的精细调整。CBAM 通过这种结合通道和空间注意力的方式，逐步精化特征图，从而有效提高模型对目标特征的表达和识别能力。

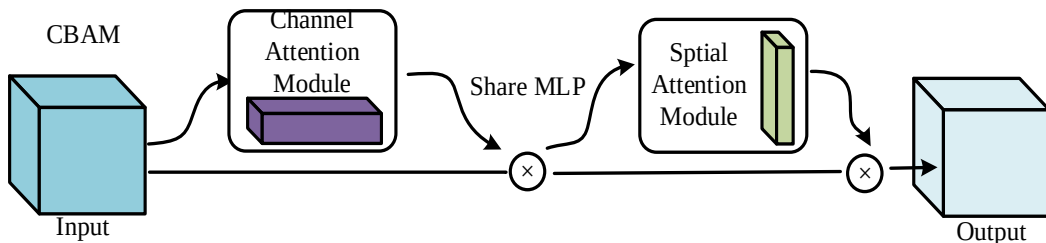


图 3-16 CBAM 注意力机制结构图

与现有的通道和空间注意力相比，SimAM^[57]直接生成了一个受人类神经科学理论启发的全三维权重注意力模块，两种机制共存，在速度和准确性方面都取得了很好的表现。SimAM 嵌入到 YOLOv5s 模型的颈部，使网络在不添加额外模

型参数的前提下，通过优化特征图的能量函数来获得 3D 特征权重，使模型更加专注于图像的重要特征信息，抑制复杂背景对网络的干扰。

SimAM 能量函数定义如式(3-8)所示：

$$e_t(w_t, b_t, y, x_i) = \frac{1}{M-1} \sum_{i=1}^{M-1} \left(-1 - (w_t x_i + b_t) \right)^2 + \left(1 - (w_t t + b_t) \right)^2 + \lambda w_t^2 \quad (3-8)$$

式中， t 和 x_i 分别表示输入特征通道中的目标神经元和其他神经元。 M 表示该通道上神经元的总数。 $w_t t + b_t$ 和 $w_t x_i + b_t$ 是 t 和 x_i 的线性变换形式， w_t 和 b_t 是线性变换的权重和。 w_t 和 b_t 可以通过公式(3-9)和(3-10)来求解。

$$w_t = -\frac{2(t - \mu_t)}{(t - \mu_t)^2 + 2\sigma_t^2 + 2\lambda} \quad (3-9)$$

$$b_t = -\frac{1}{2}(t + \mu_t)w_t \quad (3-10)$$

式中， $\mu_t = \frac{1}{M-1} \sum_{i=1}^{M-1} x_i$ 和 $\sigma_t^2 = \frac{1}{M-1} \sum_{i=1}^{M-1} (x_i - \mu_t)^2$ 分别是所选通道中除目标神经元 t 外的所有神经元的平均值和方差。可以使用公式(3-11)来获得最小能量。

$$e_t^* = \frac{4 \left(\frac{1}{M} \sum_{i=1}^M (x_i - \hat{\mu})^2 + \lambda \right)}{(t - \hat{\mu})^2 + \frac{2}{M} \sum_{i=1}^M (x_i - \hat{\mu})^2 + 2\lambda} \quad (3-11)$$

式中 $\hat{\mu} = \frac{1}{M} \sum_{i=1}^M x_i$ ，能量越低，神经元 t 和其他神经元的区别越大，代表这些神经元携带更多信息并且具有更高的重要性。

3.3.4 改进后模型结构

本文提出的改进模型包含三部分：使用 C3_DG 模块与 CBS 构建特征提取网络，由 C3Ghost 模块、I-BiC 模块以及 SimAM 注意力机制构成的多尺度特征融合网络以及头部网络组成。改进后网络结构如图 3-17 所示，图中蓝色框内为 I-BiC 结构，主干中虚线箭头部分表示省略 CBS 模块。改进后的算法在主干使用 C3_DG 模块加强特征提取能力；在颈部构建 I-BiC 模块用以减少特征损失，改善各尺寸目标检测能力，同时使用 C3Ghost 模块，替换原有的 C3 模块，降低模型参数量；最后引入 SimAM 注意力机制，提升模型在复杂背景下检测能力和抗干扰能力。

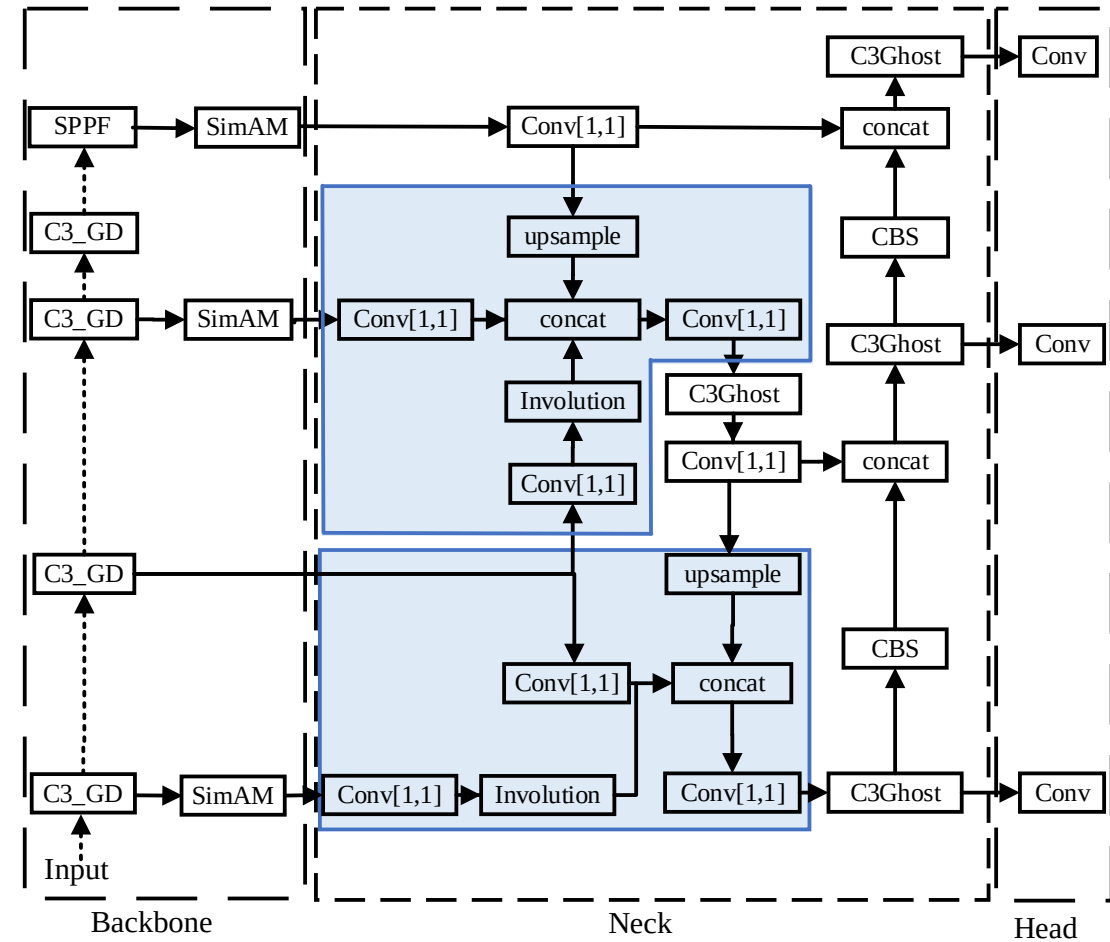


图 3-17 改进后模型结构

3.4 实验结果与分析

3.4.1 数据集

基于深度学习的垃圾检测是计算机视觉物体检测领域的热门话题之一，但垃圾目标检测的公开数据集较少，目前许多研究使用自制数据集^[58-61]。TrashNet 由斯坦福大学的 YANG 等创建，该数据集包括六类：废物、玻璃、纸张、纸板、塑料和金属，在垃圾分类检测领域使用较多，但大部分图片只包含一个实例。“华为云人工智能大赛-垃圾分类挑战杯”生活垃圾像数据集包含 4 个大类，44 个小类，15208 张图片，标注格式为 PASCAL VOC，共 17941 个标注，包含了常见的垃圾种类，但数据集有标注不准等问题。TACO 是一个用于垃圾分类和检测的数据集，包括 1500 张图像和 4784 个标注，TACO 拥有相当多的背景包括从热带海滩到伦敦街道，但数据集数量较少^[62]。

本实验数据集来源于 2020 年“华为云人工智能大赛-垃圾分类挑战杯”垃圾

像数据集与 TACO 数据集，从中选取了 14 类常见垃圾，包括快餐盒(disposable snack boxes)，充电宝(power bank)，塑料衣架(plastic hanger)，干电池(dry cell)，快递袋(express paper bag)，易拉罐(can)，化妆品瓶(toiletries)，玻璃瓶(glassware)，箱子(box)，软膏(chrisma)，过期药物(expired drug)，食用油桶(Edible oil barrel)，金属食品罐(metal food can)，瓶子(bottle)。使用 Labelimg 对数据集中部分漏标、错标进行重新标注，清洗后的数据集共 4997 张，具体各类别数量如图 3-18 所示。按照 8:1:1 将数据集随机划分为训练集、测试集和验证集。同时为验证改进后的模型的普适性与有效性，在公开的 Trashnet 数据集上进行了实验验证。

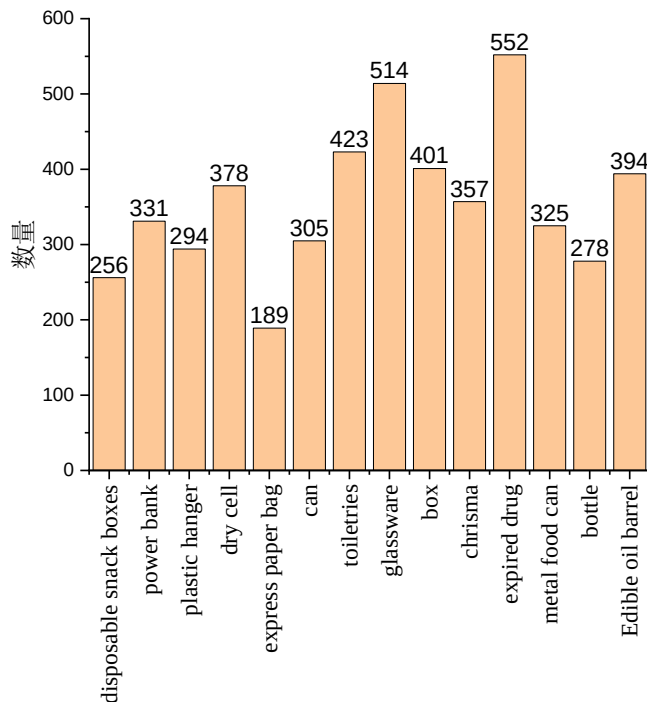


图 3-18 数据集中各类别数量

3.4.2 实验环境

实验基于 Pytorch 深度学习框架，版本为 1.12.1，Python 版本为 3.8，具体配置如表 3-2 所示。

表 3-2 实验环境

配置名称	版本及型号
操作平台	Ubuntu18.04
GPU	NVIDIA RTX A5000(24G) ×2
CPU	Intel Xeon 5218R ×2
内存	128G
CUDA&Cudnn	v11.3&v8.3.2

使用随机梯度下降(SGD)算法进行模型训练，以更新和优化网络模型的权重，

学习率使用余弦退火算法，余弦退火按照余弦函数周期性地增加和减少学习速率。训练参数设置如下：批量大小为 64，学习率为 0.01，动量为 0.937，权重衰减因子为 0.0005，训练周期为 300 轮。

3.4.3 算法评价指标

评估指标引入了准确率 P、召回率 R、平均精度 mAP@0.5、参数量 params 以及模型大小来评估改进后的模型性能。

$$P = \frac{TP}{TP + FP} \quad (3-12)$$

$$R = \frac{TP}{TP + FN} \quad (3-13)$$

$$AP = \int_0^1 P(R) dR \quad (3-14)$$

$$mAP = \frac{\sum_{i=1}^N AP_i}{C} \quad (3-15)$$

式(3-12)~(3-15)中，TP(True Positives)表示的是模型正确地将正例识别为正例的数量。在垃圾检测中，表示被正确标记为垃圾的垃圾数量。FP(False Positives)表示的是模型错误地将负例识别为正例的数量，即在垃圾检测中表示实际上不是垃圾但被错误地标记为垃圾的数量。FN(False Negatives)表示模型错误地将正例识别为负例的数量，即在垃圾检测中指的是垃圾未被检测出来的数量。这里的正例为算法关注并试图检测的类别。负例为算法不关注的类别，与正例相对，例如待检测目标背景中存在的物体。mAP 的值是通过对所有类别 AP 求平均值获得，C 表示检测到的类别总数，mAP 的值越大，一定程度上反映算法检测效果越好。

3.4.4 消融对比试验

为了验证每个改进点对网络性能改进的贡献，设计了消融对比实验。根据表 3-3 消融实验可以看出，首先 I-BiC 模块更好的保留浅层特征，兼顾大尺寸和小尺寸目标检测精度，使 mAP@0.5 较原始模型增长了 0.8%。使用 C3_DG 模块加深了网络深度，增大感受野，特征得到高效利用，可以较为明显的提升模型的特征提取能力，使模型 mAP@0.5 升高了 1.7%，但参数量有轻微增长。

C3Ghost 替换了原有的 C3 模块，使参数量有所下降，但同时模型性能有轻微下降。

表 3-3 消融实验结果对比

C3-DG	C3Ghost	SimAM	颈部	P/%	R/%	mAP@0.5/%	Params/M
-	-	-	FPN+PAN	78.4	73.2	80.9	7.1
-	-	-	I-BiC	79.8	74.1	81.7	7.2
√	-	-	FPN+PAN	81.9	73.4	82.6	7.6
√	√	-	FPN+PAN	81.5	72.7	82.3	6.6
√	√	√	FPN+PAN	82.2	74.3	83.1	6.6
√	√	√	I-BiC	83.3	75.8	83.8	6.7

在网络中添加 SimAM 注意力机制，改进模型 mAP@0.5 增长 0.8%。改进后模型较原始模型 mAP 提高了 2.9%，精度与召回率分别增长 4.9%与 2.8%，参数量有一定程度下降。

表 3-4 单个类别对比

类别	YOLOv5s	Proposed model
	AP/%	AP/%
disposable snack box	77.5	82.8
power bank	84.3	85.2
plastic hanger	86.7	90
dry cell	77.1	86.1
express bag	88.8	90.2
can	83.5	89.1
toiletries	87.7	90.3
glassware	85.1	84.3
box	72.7	76.6
chrisma	84.8	87.6
expired drug	83.8	84.9
edible oil barrel	76.3	76.5
metal food can	59.1	61.7
bottle	85.3	88.4

由表 3-4 单个类别的对比实验数据可知，改进后模型对于各个尺寸的目标检测能力有所提升。AP 指标仅在 glassware 类别上略低于原始模型。

图 3-19 为改进模型与原始模型训练过程对比，改进模型 mAP@0.5 更高，P 指标提升明显，R 指标在训练过程中波动更为平缓，train/box_loss, train/obj_loss, train/class_loss 分别为训练集的位置坐标损失、置信度损失和类别损失，三者值越小分别代表检测框的预测更准，分类正确率更高，检测准确率更高。

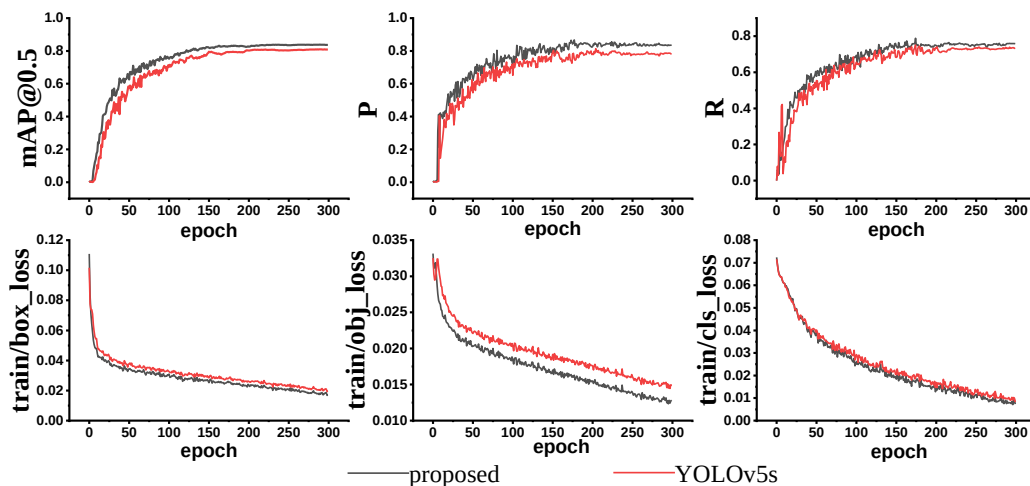


图 3-19 改进模型与原始模型训练过程对比

3.4.5 注意力机制对比

为验证添加 SimAM 注意力机制的有效性,使用 Grad-CAM 算法对生成的热力图进行可视化分析。图 3-20 为热力图可视化结果对比,热力图中的颜色深浅反映了模型对不同区域认为的重要性,颜色越深的区域表明模型认为其越关键。由图 3-20(b)与图 3-20(c)可知,原模型容易受背景噪声干扰,而在添加注意力机制之后的模型对非目标区域关注程度明显降低。

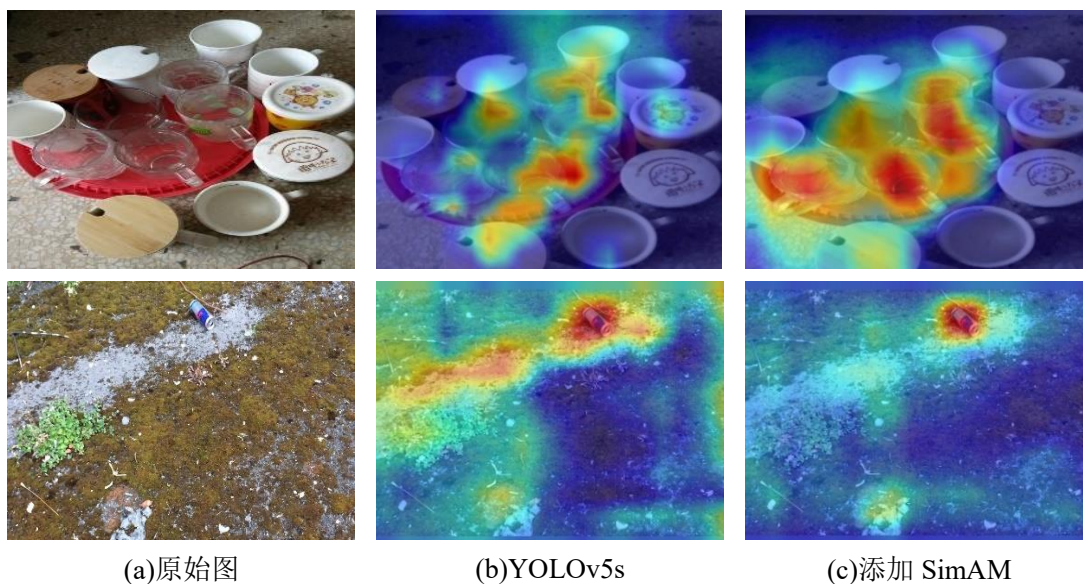


图 3-20 热力图可视化结果对比

为测试不同注意力机制对网络性能的影响,将 SimAM 注意力机制与 SE 与 CBAM 注意力机制进行了对比试验,三种注意力机制在网络中的位置相同,对比试验结果如表 3-5 所示。分别添加三种注意力机制后,模型参数量相近,但添加 SimAM 注意力机制后模型性能最好。

表 3-5 不同注意力机制对比

注意力机制	mAP@0.5/%	Params/M
SimAM	83.8	6.71
SE	83.2	6.72
CBAM	83.6	6.72

3.4.6 检测效果对比

图 3-21 为 YOLOv5s 与改进后面模型检测效果对比。图中(a)组与(d)组可以看出在背景复杂，被检测目标被遮挡的情况下，YOLOv5s 算法出现了漏检，图中的 box 和 bottle 都未检出。图中 3-21(b)组目标背景复杂，目标容易与背景混淆，YOLOv5s 出现了将背景误检的情况，而改进后的算法依然能够有效地检测出目标。综合 3-21(a)组到 3-21(d)组可以看出，改进后模型对于不同尺寸的目标展现了较原算法更好的检测能力。





图 3-21 YOLOv5s 与改进后模型检测效果对比

3.4.7 不同数据集对比实验

为进一步验证改进后算法的有效性、普适性，将原始模型与改进后模型在 TrashNet 数据集上进行了实验。原始 TrashNet 数据集并未标注，此次实验使用了文献[63]中公开的标注好的数据集，实验环境与前文相同。图 3-22 为在 Trashnet 数据集上 YOLOv5s 与改进后模型检测效果对比。综合图图 3-22 (a)与图图 3-22 (b)可以看出，改进后的算法依旧表现了较好的检测性能，普适性较强。由表 3-6 可知改进后的算法在 P、R、mAP@0.5 与 mAP@0.5:0.95 上较原始模型相比，分别提高了 2.9%、0.3%、1.9%和 4.9%，改进后的模型性能有较大提升。

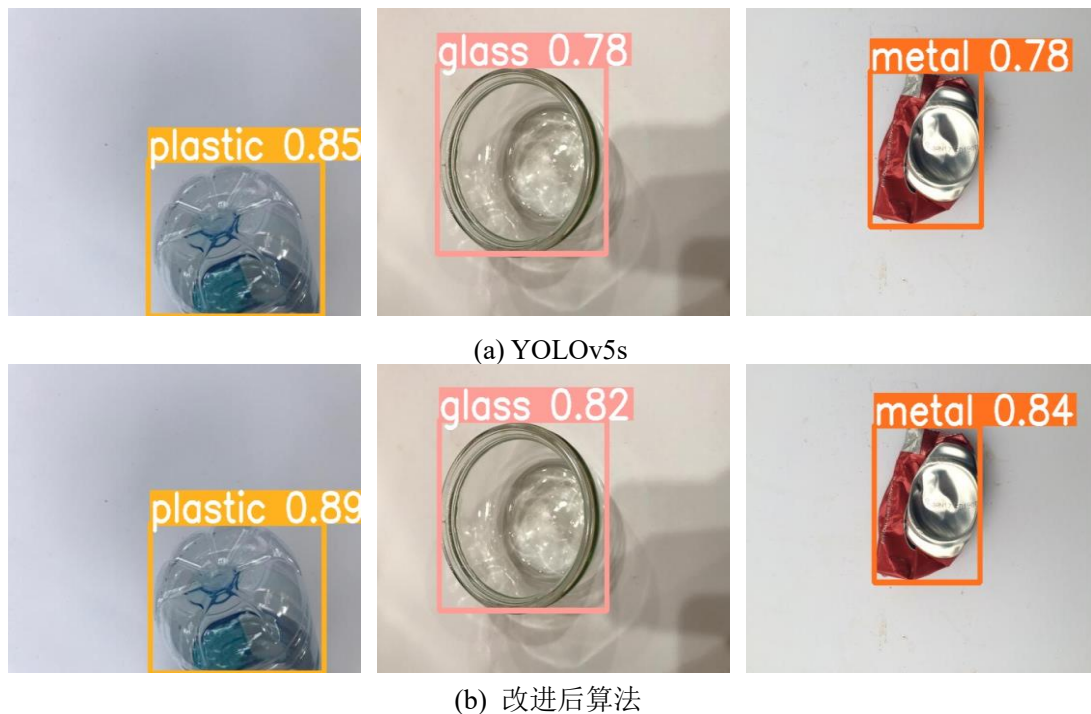


图 3-22 Trashnet 数据集上 YOLOv5s 与改进后模型检测效果对比

表 3-6 不同数据集对比试验

算法	P	R	mAP@0.5/%	mAP@0.5:0.95/%
YOLOv5s	88.5	84.1	89.8	81.4
Proposed model	91.4	84.4	91.7	86.3

3.4.8 不同算法对比试验

将改进后算法与 YOLOv4-tiny、YOLOv7^[64]、YOLOv7-tiny 和 YOLOX-s^[65] 算法在相同的实验环境下进行实验。

表 3-7 不同算法对比试验

算法	P/%	R/%	mAP@0.5/%	size/MB
YOLOv4-tiny	75.6	69.5	76.1	22.3
YOLOX-s	82.9	73.1	82.6	35.1
YOLOv7	87.4	74.8	85.4	71.4
YOLOv7-tiny	80.7	71.5	80.6	24.9
YOLOv5s	78.4	73	80.9	13.6
Proposed model	83.3	75.8	83.8	13.3

由表 3-7 所示改进后的 YOLOv5s 模型在精度方面仅低于网络复杂度最高的 YOLOv7, 明显高于其余目标检测模型。mAP@0.5 较参数量相近的 YOLOv4-tiny, YOLOv7-tiny, YOLOX-s 分别提高了 8.7%、3.2%、1.2%。改进后的算法和其他算法相比具有一定的优势。

3.5 本章小结

本文基于 YOLOv5s 设计了一种垃圾检测算法。主干使用基于密集连接思想构建的 C3_DG 模块, 提高背景复杂情况下的检测能力; 颈部使用 C3Ghost, 减少了模型参数量, 减轻对计算资源的消耗。同时使用 I-BiC 模块优化多尺度融合模块, 改善原始模型面对目标尺寸多样情况下检测能力弱的问题, 最后引入 SimAM 注意力机制, 提高复杂背景中待检测目标区域显著度。针对现有数据集的问题, 通过融合 2020 年“华为云人工智能大赛-垃圾分类挑战杯”生活垃圾像数据集与 TACO 数据集, 获得了 14 类常见垃圾数据集。通过实验验证, 改进后的模型精度、普适性与鲁棒性较原始算法都有较大提升。

第 4 章 机械臂路径规划算法

4.1 引言

第 3 章提出了基于改进 YOLOv5s 的垃圾检测算法，该算法能够准确的确定目标的位置与类别信息。而面对工作空间有障碍物的情况，机械臂完成垃圾分拣还需高效的路径规划算法。

RRT-Connect 算法在 RRT 算法的基础上，提高了搜索效率，但仍存在扩展树的生长方向随机，搜索路径冗长等问题。本节在 RRT-Connect 算法的基础上进行了优化，提出了 DAPF-RRT(RRT algorithm with dynamic step size and artificial potential field)算法。

4.2 RRT 与 RRT-Connect 算法基本原理

RRT 算法是一种基于采样的路径规划算法，因其具有规划时间短、探索能力强等优点，广泛应用于三维空间的路径规划问题^[66-67]。RRT 基本思想是从起始点开始以树的形式进行探索，在可行空间内随机采样，延伸枝叶，直到探索树覆盖目标区域。RRT 算法的路径规划过程如图 4-1，图中 N_{source} 为初始点， N_{rand} 为随机采样点， N_{new} 为扩展的新节点， N_{near} 为到 N_{rand} 距离最近的节点， N_{goal} 为目标点，obstacle 为空间中的障碍物。

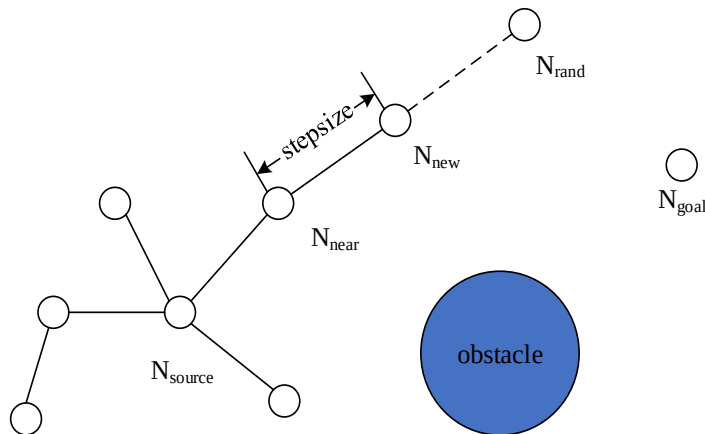


图 4-1 RRT 算法扩展示意图

RRT 算法的具体流程如下:

(1)初始化树 $T=(V,E)$, V 为扩展的所有节点集合, E 为 N_{new} 和 N_{near} 的路径集合, 初始化时为空集。 ρ 为设定的阈值, 当 N_{new} 与 N_{goal} 间的距离小于 ρ 时则表示找到目标点。

(2)使用 RANDOM 函数在空间中生成随机采样点 N_{rand} 后, Extend 函数如表 4-1 所示。 Extend 函数中的 NEAREST_NEIBOR 函数遍历搜索树中所有节点, 找到最近的父节点 N_{near} 。

(3)使用 Steer 函数连接 N_{rand} 和 N_{near} , 沿连线方向以设定的 stepsize 生成新节点 N_{new} , COLLISION_FREE 函数用于判断 N_{near} 与 N_{new} 之间是否与障碍物发生碰撞, 如果有则丢弃 N_{new} , 返回步骤(2), 重新生成采样点, 如果没有碰撞则将新 N_{new} 添加集合 V 中, 并返回 Advanced, 代表新节点扩展成功。

(4)若 N_{new} 与 N_{goal} 间的欧式距离小于设定值 ρ 且两点间无碰撞表示规划成功, 并返回 Reached, 代表到达目标点; 若以上情况都未出现则返回 Trapped, 表示扩展失败。 重复以上步骤至 Extend 函数返回 Reached, 或达到设定的最大迭代次数, 结束随机采样。 最后, 通过从终点回溯随机树的所有节点, 得到从起点到终点的路径。 RRT 算法伪代码如表 4-2 所示:

表 4-1 Extend 函数伪代码

1:Extend($G(V,E),N_{rand}$)
2: $N_{near} \leftarrow \text{NEAREST_NEIBOR}(G(V,E),N_{rand})$
3: $N_{new} \leftarrow \text{Steer}(N_{near},N_{rand})$
4:if COLLISION_FREE (N_{near}, N_{new}) then
5: $V \leftarrow V \cup \{N_{new}\}; E \leftarrow E \cup \{(N_{near},N_{new})\};$
6:Return Advanced;
7: if $\ N_{new}-N_{goal}\ < \rho$ then return Reached
8:else rerurn Trapped

表 4-2 RRT 算法伪代码

Algorithm1:RRT
1: $V \leftarrow \{N_{source}\}; E \leftarrow \emptyset; N_{goal}; \rho \in R$
2:For $i=1,..N$ do
3: $N_{rand} \leftarrow \text{RANDOM}();$
6: if Extend($G(V,E),N_{rand}$)=Reached then break;
7:return $G(V,E);$

Lavalle 在 RRT 算法的基础上提出了 RRT-Connect 算法^[68], 扩展过程如图 4-2 所示。RRT-Connect 算法通过以下两个关键的改进, 改善了搜索效率, 缩短搜索时间:

(1) 不同于 RRT 只从起点生长一棵树, RRT-Connect 同时从起点和终点构造 T1 和 T2 两棵树, 两棵随机树相互靠近生长, 直到这两棵树相连。

(2) 表 4-3 为 CONNECT 函数伪代码。CONNECT 连接函数采用节点贪婪扩展策略, 按照 RRT 算法扩展第一棵树 T1 的新节点之后, 会尝试将树 T2 朝 T1 的新节点方向进行扩展, 使两棵树快速连接。同时, 当 T2 的节点在该方向扩展时没有遇到障碍物时, 则继续向该方向扩展节点。表 4-4 为 RRT-Connect 算法伪代码, 其中的 Swap 函数用于使两棵树交替生长。

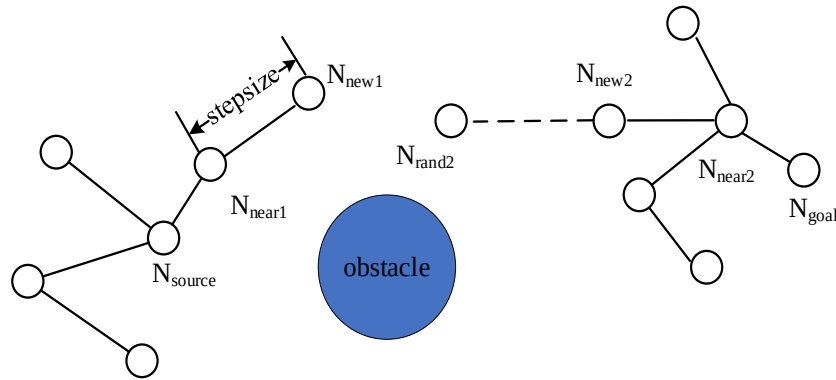


图 4-2 RRT-Connect 算法扩展示意图

表 4-3 CONNECT 函数伪代码

CONNECT(T1, N_rand)
1: repeat
2: S ← Extend(T1, N_rand)
3: Until not S = Advanced
4: Return S

表 4-4 RRT-Connect 算法伪代码

Algorithm1: RRT-Connect PLANNER(N_source, N_goal)
1: T1.init(N_source); T2.init(N_goal);
2: For k=1 to K do
3: N_rand ← RANDOM();
4: if not(Extend(T1, N_rand)) then
5: if (Connect(T2, N_new) = Reached) then
7: Return PATH(T1, T2);
8: SWAP(T1, T2);
9: Return Failure

4.3 基于 DAPF-RRT 算法的路径规划

RRT-Connect 算法使用两个树进行双向扩张，速度较 RRT 算法有较大提升。但算法采样点过于分散，扩展随机，使用固定步长造成搜索效率低，路径转折多等问题还有待解决。

4.3.1 目标偏置

在 RRT-Connect 算法中添加目标偏置，使节点以更高的概率向目标点方向扩展，改善采样点过于分散的问题，并防止节点扩展落入死区。目标偏置思想是在随机采样过程中设置一个采样概率 $P \in [0,1]$ ，同时在 0 到 1 间随机生成一个概率 P_{rand} 。由于改进算法是两个树双向搜索，若概率 P_{rand} 小于采样概率则以 N_{goal} 或 N_{source} 为采样点，否则使用 $Random()$ 函数在规划空间中随机生成采样点。添加目标偏置的采样公式如式(4-1)所示。阈值的设定与所要规划的环境有关，当环境中的障碍物较少时，将阈值适度增大，加速向目标扩展。当环境中障碍物较多时，阈值设定较小。

$$N_{rand} = \begin{cases} N_{goal} \text{ or } N_{source}, & (P_{rand} < P) \\ Random(), & (P_{rand} \geq P) \end{cases} \quad (4-1)$$

4.3.2 动态扩展步长

RRT-Connect 算法对于不同的环境场景在扩展新节点时都使用固定的步长进行扩展。对于环境简单的场景，固定步长对于规划效率并无太大影响。但规划场景杂时，当随树扩展到障碍物较为密集的区域，使用固定步长扩展会增加与障碍物碰撞的概率，造成新节点扩展失败，进入下一次搜索，使算法花费更多时间才能绕开障碍。本文结合文献[69]中提出的步长计算方法，提出一种动态步长策略，步长计算公式如式(4-2)所示：

$$\lambda_d = (\zeta e^{A \cdot D_{obs}} - 0.8) \cdot \lambda \quad (4-2)$$

式中， λ 为初始步长， λ_d 为调整后的动态步长， ζ 为调节系数， A 为衰减系数， D_{obs} 是 N_{near} 与障碍物之间的距离。当 N_{near} 距离障碍物距离较远的情况下使用较大的步长进行扩展，加快收敛速度。当 N_{near} 靠近障碍物时，步长随 D_{obs} 的减小而

变化，加快算法绕开障碍物。通过动态调整步长，既帮助扩展树跳出密集障碍物区域，又能增加路径的平滑性。在实验过程中，根据扩展随机树与障碍物之间的距离确定的动态步长有一定几率会出现过小的情况，造成密集障碍物区域扩展缓慢的情况。本文对目前的动态扩展策略进行了修改，动态步长策略生成的步长表达式如式(4-3)所示：

$$stepsize = \begin{cases} \lambda, & D_{obs} > D_s; \\ \lambda_d, & D_{obs} \leq D_s \\ \lambda_{min}, & \lambda_d < \lambda_{min} \end{cases} \quad (4-3)$$

如图 4-3 所示，首先设置了步长调整范围 D_s ，当 N_{near} 与障碍物圆心距离 D_{obs} 大于 D_s 时表示距离障碍物较远，使用原始步长进行扩展。当 D_{obs} 小于 D_s 时开始动态调整步长。随着 D_{obs} 减小，代表着进入障碍物密集区域，使用较小步长扩展，当步长小于最小步长 λ_{min} ，则使用最小步长 λ_{min} 加速探索。

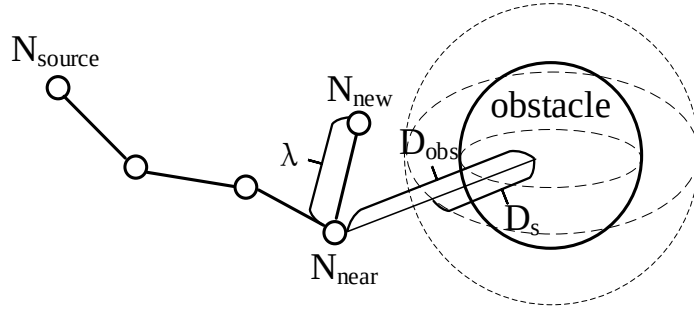


图 4-3 动态步长扩展示意图

4.3.3 节点扩展策略优化

为了提高 RRT-Connect 算法在路径规划时的效率，减少扩展的盲目性，加快收敛速度，将人工势场法中的目标引力函数和障碍物斥力函数融入到两棵树生长过程中，对节点生长过程进行优化。在克服传统人工势场算法容易落入局部最优的缺点的同时，更好地利用地图中的障碍信息，提高路径搜索目的性，两颗树在势场作用下的能够迅速向目标点扩展。

人工势场法因其算法简单，计算量小，收敛速度快在路径规划领域得到广泛应用。人工势场的基本原理为：在目标点设置引力势场，障碍物构建斥力势场，并设置斥力作用范围。机器人在引力势场和斥力势场共同作用下，沿着合力方向朝目标运动，并完成避障^[70]。机器人在人工势场中的受力如图 4-4 所示，其中虚

线圆形为障碍物斥力的作用范围， F_{att} 表示目标点对机器人的吸引力， F_{rep} 表示障碍物对机器人的斥力， F_{total} 为二者合力。

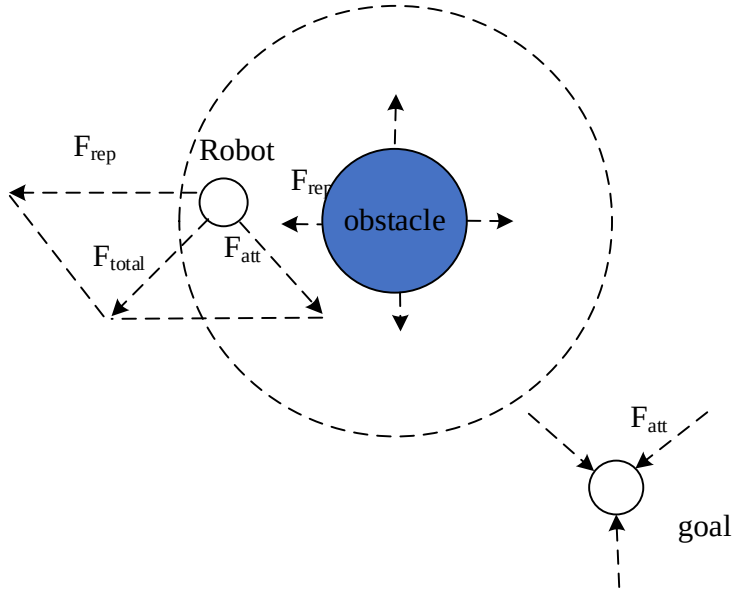


图 4-4 机器人在人工势场中受力示意图

引力势场表示为式(4-4):

$$U_{att}(N) = \frac{1}{2} \xi d^2(N, N_{goal}) = \frac{1}{2} k_{att} \|N - N_{goal}\|^2 \quad (4-4)$$

式中， ξ 是引力场增益， $d^2(N, N_{goal})$ 表示物体当前点到目标点的距离。对引力势场表达式求负梯度即可得到引力，引力方向为当前点指向目标点，引力如式(4-5)所示。

$$\begin{aligned} F_{att}(N) &= -\nabla U_{att}(N) = -\nabla \left(\frac{1}{2} \xi d^2(N, N_{goal}) \right) \\ &= -\xi d(N - N_{goal}) = k_{att} \|N - N_{goal}\| \end{aligned} \quad (4-5)$$

斥力势场可以表示为式(4-6):

$$U_{rep}(N) = \begin{cases} \frac{1}{2} n \left(\frac{1}{d(N, N_{obs})} - \frac{1}{d_0} \right)^2, & d(N, N_{obs}) < d_0; \\ 0, & d(N, N_{obs}) \geq d_0 \end{cases} \quad (4-6)$$

式中， n 表示斥力场增益， $d(N, N_{obs})$ 表示采样点到障碍物的欧几里得距离， d_0 为采样点被障碍物排斥的最大距离也就是斥力场作用范围。斥力可由斥力势场的负梯度求出，斥力方向为障碍物指向当前点，斥力的数学表达式如式(4-7)，

其中, N_{obs} 为障碍物中心位置, 从式(4-7)可以看出斥力随着与障碍物的距离减小而增大。

$$F_{rep}(N) = \begin{cases} n \left(\frac{1}{d(N, N_{obs})} - \frac{1}{d_0} \right) \frac{1}{d(N, N_{obs})^2}, & d(N, N_{obs}) < d_0; \\ 0, & d(N, N_{obs}) \geq d_0 \end{cases} \quad (4-7)$$

机器人在受到合力可以表示为式(4-8):

$$F_{total} = F_{att} + \sum_i^n F_{rep} \quad (4-8)$$

式中, n 为障碍物总数, i 为对当前点产生斥力的障碍物数量。

借鉴人工势场算法的思想, 将障碍物斥力分量与目标引力分量融入到 RRT-Connect 算法的新节点扩展中。其主要方法为: 在目标点设置作用全局的引力场, 每次扩展的新节点 N_{new} 都会受到目标引力分量引导作用。同时为防止目标引力作用过强, 增加在障碍物密集区域扩展失败的概率, 引入了障碍物斥力分量。若进入斥力势场作用范围后, N_{new} 的生长受到采样函数分量、目标引力与障碍物斥力三者的合力方向的影响。

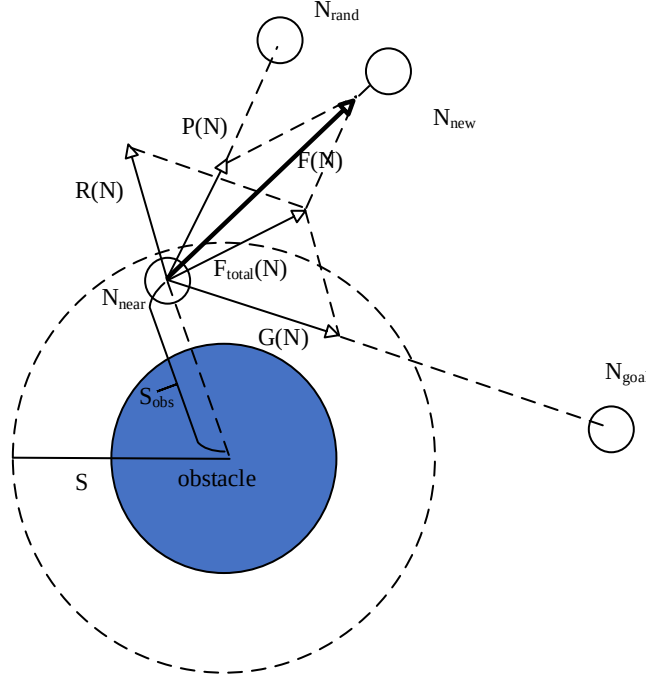


图 4-5 加入目标引力分量与障碍物斥力分量

图 4-5 加入目标引力分量与障碍物斥力分量后新节点的扩展方式示意图。图中为 $R(N)$ 障碍物斥力分量, $G(N)$ 为目标引力分量, $P(N)$ 为随机采样方向分量, $F_{total}(N)$ 为 $G(N)$ 和 $R(N)$ 分量之和。 $F(N)$ 为 N_{new} 生长方向, S_{obs} 为到 N_{near} 最近障

障碍物的距离, S 为障碍物斥力作用范围。在目标引力分量以及采样函数分量作用下, 使新的节点偏向目标点扩展, 同时利用周围的障碍物信息, 对扩展方向进行修正。每一次扩展 N_{new} 都更靠近目标点, 提高了路径搜索的目的性, 显著减少路径规划时间。

目标引力分量的单位向量 $\vec{g}$ 如式(4-9), 其中 $\|N_{\text{goal}} - N_{\text{near}}\|$ 为 N_{goal} 和 N_{near} 两点的欧式距离。

$$\vec{g} = \frac{N_{\text{goal}} - N_{\text{near}}}{\|N_{\text{goal}} - N_{\text{near}}\|} \quad (4-9)$$

根据式(4-9)可构造出目标引力分量 $G(n)$ 如式(4-10)所示:

$$G(n) = \alpha \cdot \frac{N_{\text{goal}} - N_{\text{near}}}{\|N_{\text{goal}} - N_{\text{near}}\|} \quad (4-10)$$

式中, α 为目标引力系数, 其取值与空间的障碍物有关。空间中障碍物较少时, 适当增大 α , 加快向目标扩展, 障碍物较多时适当减小 α , 加快向无障碍区域探索。

同理可得 N_{near} 处节点的障碍物斥力分量可以表示为式(4-11):

$$R(n) = \beta \cdot \frac{N_{\text{near}} - OBS}{\|N_{\text{near}} - OBS\|} \quad (4-11)$$

式中, OBS 为障碍物的中心坐标, β 为障碍物斥力系数, 其表达式如(4-12)所示:

$$\beta = \begin{cases} \frac{1}{2}K \left(\frac{1}{S_{\text{obs}}} - \frac{1}{S} \right)^2, & S_{\text{obs}} < S; \\ 0, & S_{\text{obs}} \geq S, S_{\text{obs}} < S_{\text{min}} \end{cases} \quad (4-12)$$

式中, S_{obs} 为 N_{near} 到最近障碍物的距离, S 为障碍物斥力最大作用范围, S_{min} 为斥力最小作用范围, K 为增益系数。 β 随 S_{obs} 的减小而增大, 障碍物斥力分量的作用越大, 使采样点向远离障碍物的区域扩展。同时为防止斥力分量过大, 设置最小作用范围 S_{min} , 当 S_{obs} 小于 S_{min} 时斥力分量为 0。RRT-Connect 算法计算公式为式(4-13), 其中 λ 为步长式。加入目标引力函数与障碍物斥力分量, DAPF-RRT 算法的节点计算公式为式(4-14)所示, 式中 λ_d 为前一节中提出的动态步长。

$$N_{new} = N_{near} + \lambda \cdot \frac{N_{rand} - N_{near}}{\|N_{rand} - x_{near}\|} \quad (4-13)$$

$$N_{new} = N_{near} + \lambda_d \cdot \left(\frac{N_{rand} - N_{near}}{\|N_{rand} - N_{near}\|} + R(n) + G(n) \right) \quad (4-14)$$

4.3.4 路径修剪

RRT-Connect 算法经过上述优化后，生成的路径中仍包含较多冗余节点，特别是在障碍较多的复杂环境中。为了减少不必要的冗余节点，同时改善路径中的转折情况，使路径接近最优，使用贪婪策略对生成的路径进行修剪，修剪过程示意图如 4-6 所示，图中橙色实线为修剪后的路线。

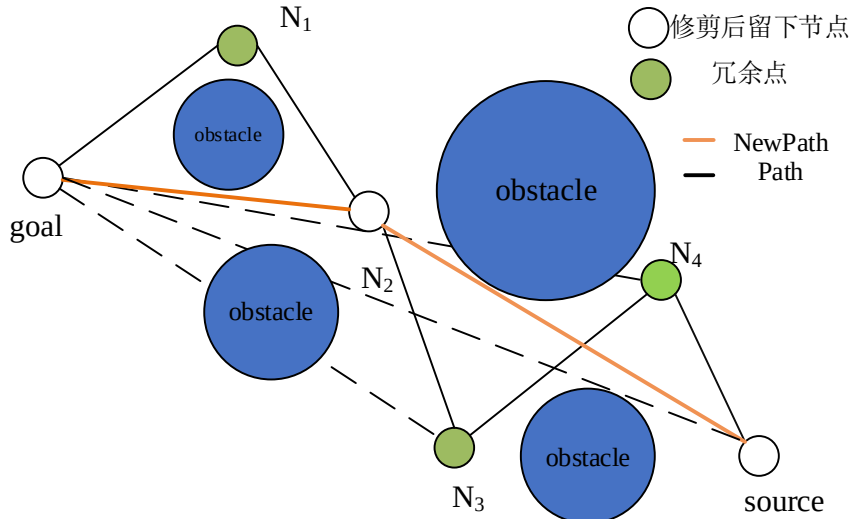


图 4-6 路径修剪过程示意图

具体修剪步骤如下：

(1) 创建两个变量 **Path** 和 **NewPath** 用于接收修剪前后的路径点。假设通过 DAPF-RRT 算法得到最终路径 $\text{Path}=[N_{\text{source}}, N_1, N_2, N_3, \dots, N_{q-2}, N_{\text{goal}}]$ ；

(2) 以目标点 N_{goal} 为起始点，依次尝试连接路径中的所有节点，每连接一次都要进行碰撞检测，若 N_{goal} 与 N_q 连线间发生碰撞时，则 N_{q-1} 与 N_{goal} 间的节点皆为需要修剪的节点。 N_{q-1} 为经过修剪后留下的节点，将 N_{q-1} 加入到 **NewPath** 中；

(3) 下一次修剪以 N_{q-1} 作为起始点，重复上述步骤，直至遍历完 **Path**；

(4) 最后，根据保存的修剪后节点，通过回溯得到修剪后的路径 $\text{NewPath}=[N_{\text{source}}, N_1', N_2', N_3', \dots, N_{q-2}', N_{\text{goal}}]$ 。

4.3.5 轨迹平滑

DAPF-RRT 算法得到的路径为两棵随机树在空间中扩展节点依次连接的连续线段，并不是一条平滑的曲线。当机械臂沿不平滑的路径运动时会存在抖动，影响机械手的稳定性，加剧磨损^[71]。为了改善这个问题有必要对规划路径进行路径平滑处理。

B 样条曲线具有连续性和局部性好的优点，在运动规划中得到了广泛的应用。因此，利用 B 样条曲线来拟合事先修剪过的路径点，以生成具有连续曲率的平滑轨迹，k 阶 B 样条曲线的表达式如式(4-15)所示：

$$C(u) = \sum_{i=0}^n d_i N_{i,k}(u) \quad (4-15)$$

式中， $d_i (i=0, \dots, n)$ 为控制顶点坐标， $u=[u_0, u_1, \dots, u_{n+k+1}]$ 是 B 样条函数的节点向量， $N_{i,k}(0, \dots, n)$ 为 k 次 B 样条基函数，B 样条基函数如式(4-16)：

$$\begin{cases} N_{i,0}(u) = \begin{cases} 1 & u_i \leq u \leq u_{i+1} \\ 0 & other \end{cases} \\ N_{i,k}(u) = \frac{u-u_i}{u_{i+k}-u_i} N_{i,k-1}(u) + \frac{u_{i+k+1}-u}{u_{i+k+1}-u_{i+1}} N_{i+1,k-1}(u) \end{cases} \quad (4-16)$$

式中， i 为节点序号， k 为阶数。阶数越高平滑度越高，但计算的复杂度也会增加。考虑到计算的复杂程度和路径的平滑要求，本文选取了三阶 B 样条曲线，拟合的曲线可以满足本文的路径平滑要求。三次 B 样条基函数如式(4-17)所示：

$$\begin{cases} N_0(u) = \frac{1}{6}(-u^3 + 3u^2 - 3u + 1) \\ N_1(u) = \frac{1}{6}(3u^3 - 6u^2 + 4) \\ N_2(u) = \frac{1}{6}(-3u^3 + 3u^2 - 3u + 1) \\ N_3(u) = \frac{1}{6}u^3 \end{cases} \quad u \in [0, 1] \quad (4-17)$$

根据上式三次 B 样条曲线如式(4-18)所示：

$$c_i(u) = d_{i-1} \times N_0(u) + d_i \times N_1(u) + d_{i+1} \times N_2(u) + d_3 \times N_{i+2}(u), u \in [0,1] \quad (4-18)$$

4.3.6 DAPF-RRT 算法流程

DAPF-RRT 算法流程总体如图 4-7 所示：

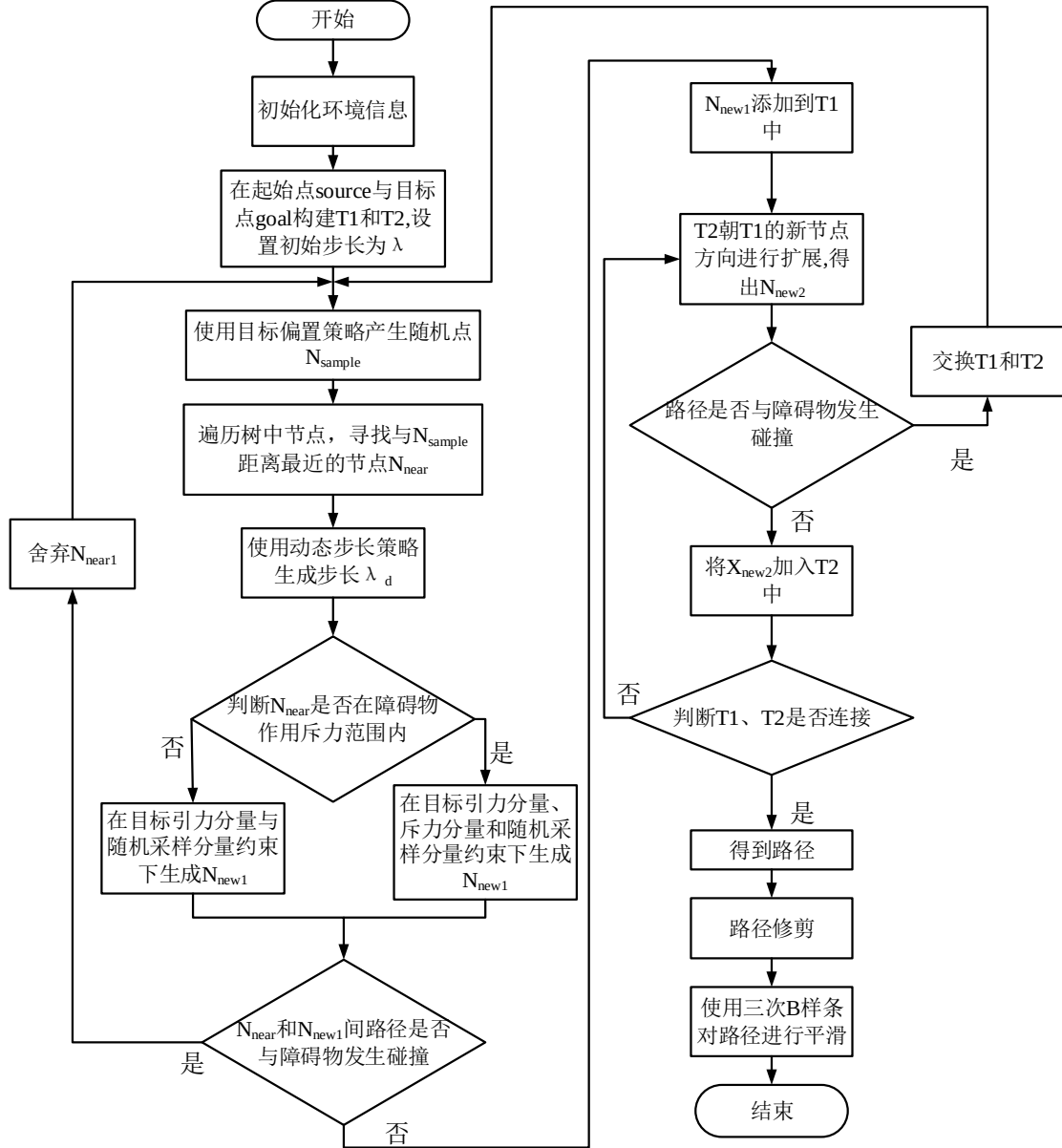


图 4-7 DAPF-RRT 算法流程图

DAPF-RRT 算法具体步骤如下：

(1)初始化规划空间，添加障碍物。在起始点与目标点构建 T1 和 T2，设置初始步长为 λ ；

(2)通过使用目标偏置策略产生随机点 N_{sample} ，遍历树中节点，寻找与 N_{sample} 距离最近的节点 N_{near} ，使用动态步长策略生成步长 λ_d ；

(3)判断 N_{near} 是否在障碍物作用斥力范围内,若在范围内,在目标引力分量、斥力分量和随机采样分量约束下生成 N_{new1} 。否则,在目标引力分量和随机采样分量约束下生成 N_{new1} ;

(4)判断 N_{near} 和 N_{new1} 间路径是否与障碍物发生碰撞,若发生碰撞则舍弃 N_{new1} 并返回步骤(2), 否则将 N_{new1} 添加到 T1 中;

(5)T2 朝 T1 的新节点 N_{new1} 方向进行扩展,获得 N_{new2} 并判断生成的路径是否发生碰撞,若未发生碰撞则将 N_{new2} 添加到 T2 中,若发生碰撞则交换随机树,返回步骤(2);

(6)判断 T1 和 T2 是否相连,若相连则返回规的路径点。获得路径后,对路径进行修剪并使用三次 B 样条对路径进行平滑处理。

4.4 三维空间仿真实验及分析

为了验证 DAPF-RRT 算法在复杂的三维空间进行路径规划中的可行性及性能,设计实验将 RRT 算法、RRT-Connect 算法与 DAPF-RRT 算法在相同环境下进行路径规划对比仿真,实验中的 RRT 和 RRT-Connect 算法均在原有基础上添加了目标偏置。仿真实验环境为:操作系统 Windows10,处理器(CPU) Intel (R) Core (TM) i5-1135G7 CPU@2.40 GHz,内存为 16G,所有仿真实验都通过 MATLAB 2021a 软件来实现。

仿真实验设置的搜索空间为[250 250 250],起始坐标为[20 20 20],终点坐标为[200 200 180]。为方便计算,实验中的障碍物均为球体,空间中设置了 7 个球体。球体中心坐标分别为: [125 125 125; 100 90 60; 150 100 80; 90 150 80; 200 200 120; 200 160 100; 40 180 100],半径分别为: [30; 30; 20; 20; 20; 15; 20],初始步长设置为 10,最大迭代次数为 10000。

3 种 RRT 算法在设定的三维空间中生长过程如图 4-8 所示。图 4-8(a)与 4-8(b)为 RRT 算法与 RRT-Connect 算法扩展过程。由于这两种算法生长的随机性强,扩展导向性差,扩展过程中在空间中生长出了大量的无效树枝。图 4-8(c)为 DAPF-RRT 算法扩展过程,可以看出扩展的导向性较强,从两棵树分别向目标点快速生长。如图 4-8(c)中蓝色箭头所示,扩展过程中在距离障碍物较远时,步长较大,加快收敛速度;扩展至障碍物密集区域时,步长较小,如图中红色箭头所

示，减小与障碍物碰撞的几率，更快的找出一条无碰撞路径。

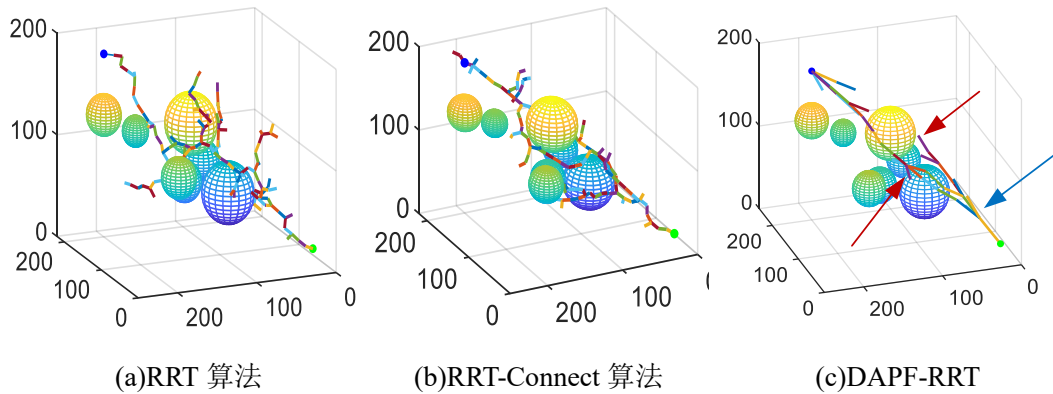


图 4-8 扩展过程

图 4-9 中图(a)到(c)中的红线为最终规划的路径,改进后的算法规划的路径更好,路径转折较改进前有很大程度减少。

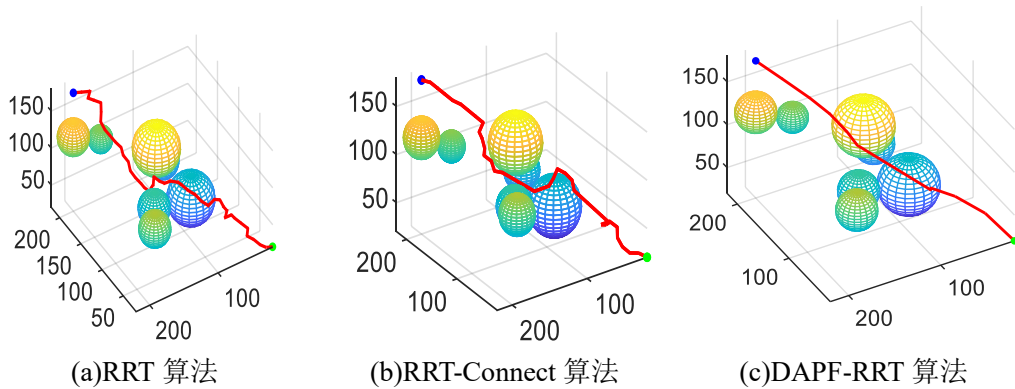


图 4-9 最终规划路径

图 4-10 蓝色线为经过路径修剪后的路径对比。图 4-10(a)中红线为修剪前路径,蓝线为修剪后路径,图 4-10(b)中蓝线上黑色圆点为修剪后留下的较优点。可以明显看出,通过修剪后的路径更短,进一步减少了路径上的转折与冗余节点。

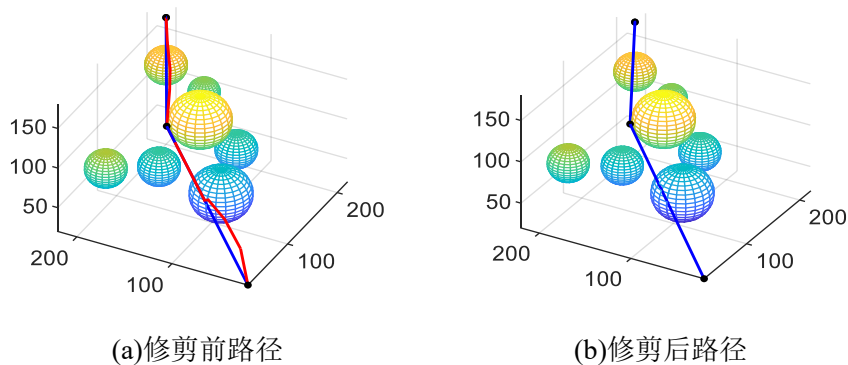


图 4-10 路径修剪对比图

使用三次 B 样条对修剪后的路径进行平滑处理,结果如图 4-11 所示。图 4-

11(a)蓝色实线为平滑前的路径，图 4-11(b)红色实线为平滑后的路径。由图可以看出修剪后的路径经过三次 B 样条插值后，路径中的转折处更为平滑。

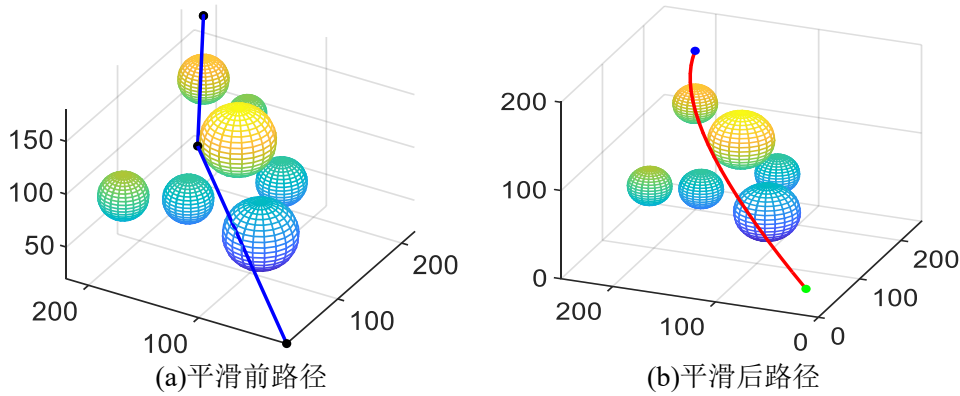


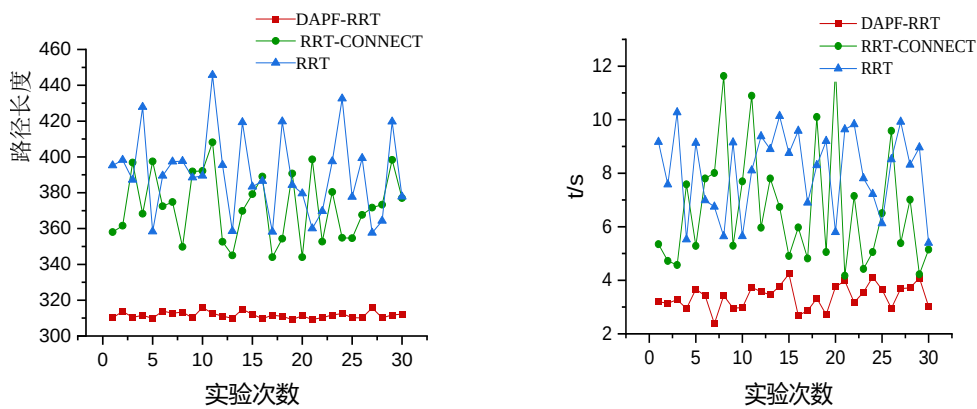
图 4-11 平滑前后路径对比图

由于 RRT 系列算法的随机性，通过一次比较难以验证改进后的算法优越性，分别将 DAPF-RRT 算法与另外两种算法在图 4-8 环境中进行 30 次仿真，30 次路径规划平均结果如表 4-5。

表 4-5 不同算法性能对比

路径规划算法	时间/s	路径长度	节点/个
RRT	8.1	390.5	40
RRT-Connect	6.7	372.2	38
DAPF-RRT	3.4	314.6	6

改进后的算法相较于 RRT 和 RRT-Connect 算法路径长度分别下降了 19.4% 和 15.4%；在空间中规划一条可行路径的时间分别相较于其他两种算法下降了 58% 和 49.2%；经过修剪后的节点数平均后近似 6 个，较 RRT-Connect 算法下降了 84.2%。



如图 4-12 所示，改进后的算法性能较原始算法有较大提升，规划时间与路

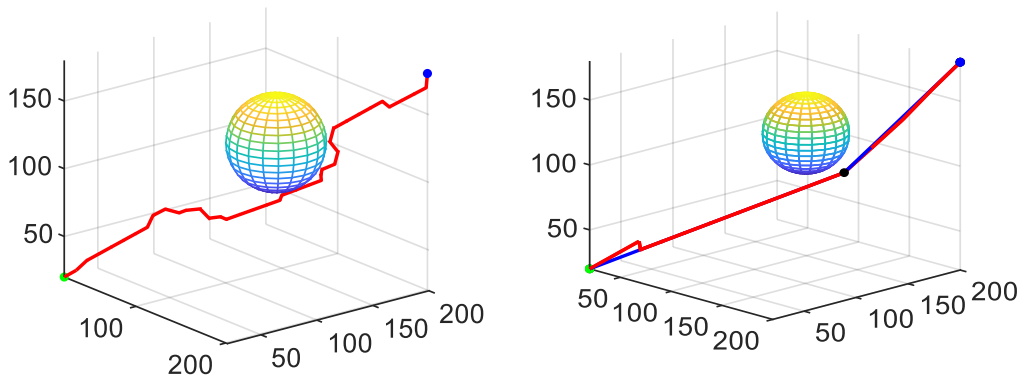
径长度有明显改善。

为进一步验证 DAPF-RRT 算法的有效性, 分别在 4 种障碍物数量不同的空间进行了 30 次仿真, 结果如表 4-6 所示。在 Map1 中, 本文提出的 DAPF-RRT 算法规划的路径长度与使用的时间相比于改进前减少了 18.3%与 48.8%; 在 Map2 中, DAPF-RRT 算法规划的路径长度与使用的时间比 RRT-Connect 算法分别减少了 16.9%与 43%; 在 Map3 中, 本文提出的算法规划的路径长度与使用的时间分别比 RRT-Connect 算法减少了 17.9%与 39.7%; 在 Map4 中本文提出的算法规划的路径长度与使用的时间分别比 RRT-Connect 算法减少了 22.3%与 42%。

表 4-6 不同空间规划路径长度与规划时间

算法	地图	路径长度	规划时间/s
DAPF-RRT	Map1	307.7	2.2
	Map2	317.4	3.7
	Map3	320.4	4.1
	Map4	324.8	5.1
RRT-Connect	Map1	376.5	4.3
	Map2	381.8	6.5
	Map3	390.1	6.8
	Map4	417.9	8.8

图 4-13 为改进前后算法规划出的路径。图 4-13(a)中红线为 RRT-Connect 算法规划的路径, 图 4-13(b)为 DAPF-RRT 算法规划的路径, 红线表示规划出的路径, 蓝线表示修剪后的路径。在不同的复杂程度的三维空间中, DAPF-RRT 算法都表现出了较原算法更优秀的避障效果和更好的规划路径。



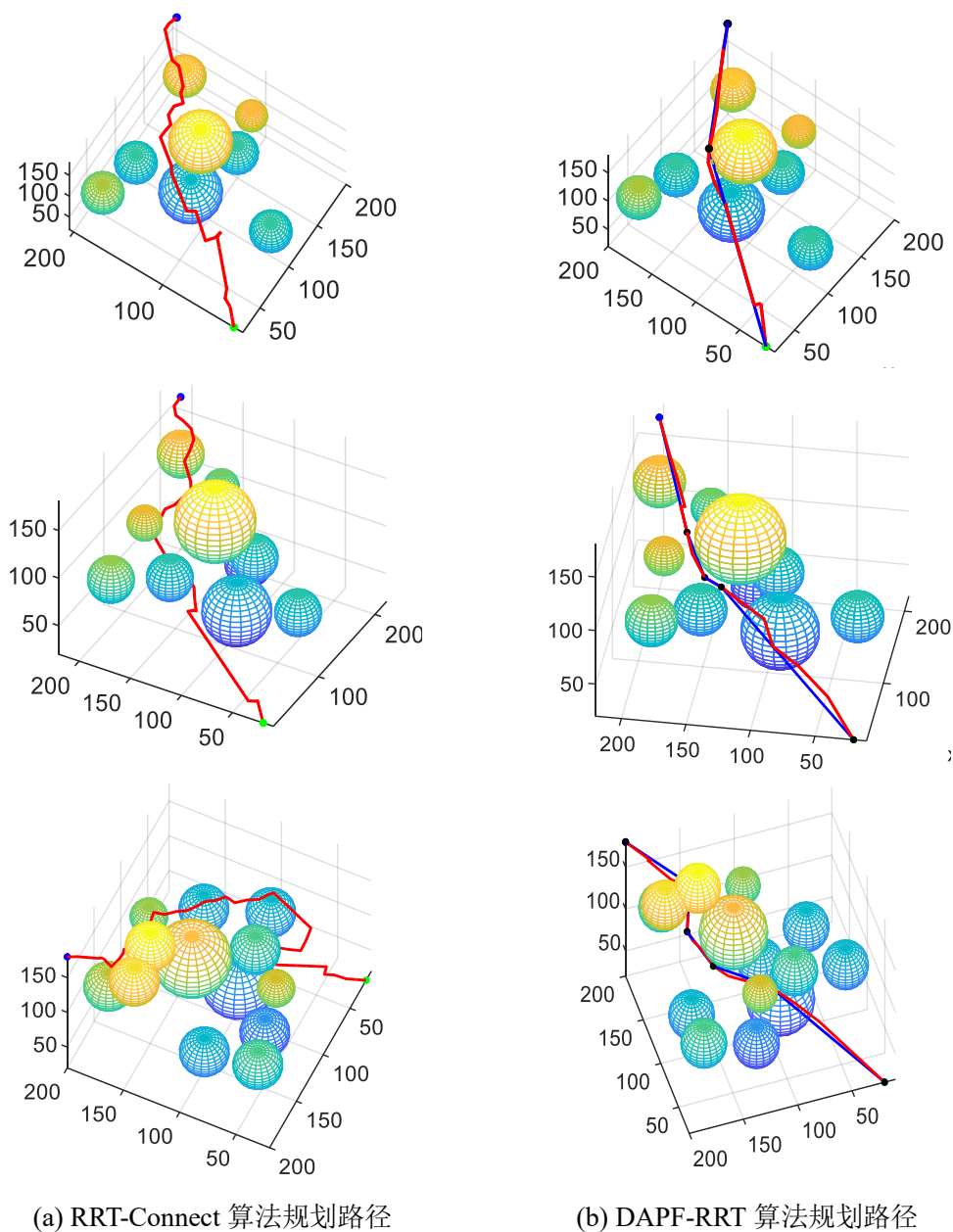


图 4-13 不同空间仿真对比图

4.5 机械臂路径规划仿真

在 4.4 节中已经验证了 DAPF-RRT 算法三维的路径规划性能, 本节将进一步验证其应用于机械臂避障路径规划的实际效果。

机械臂碰撞检测是机械臂路径规划中的重要环节, 也是能够规划出一条无碰撞路径的前提。同时由于机械臂与障碍物都不是规则的物体, 机械臂与障碍物两者之间位置关系很难精准描述, 也很难通过仿真实现。因此为了提高机械臂在工作空间中碰撞检测效率和准确性, 使用包络法对机械臂与障碍物间的碰撞检测进

行简化。如图 4-14(a)所示，机械臂连杆使用半径为 R_a 的圆柱体进行包络，障碍物使用半径为 R_b 的球体进行包络。如图 4-14(b)所示，通过包络法的简化将原本机械臂与障碍物间的碰撞检测转化为线段与球体间的碰撞检测，通过比较 $D_{\min}$ 和 R_c 的大小与分析 AB 线段与包络球间的位置关系就能判断出机械臂是否与障碍物发生碰撞。

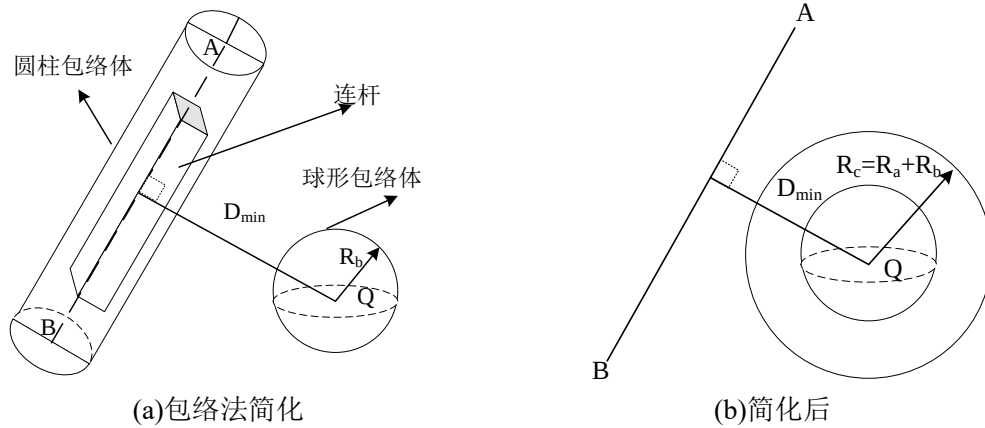


图 4-14 包络法简化模型示意图

为了进一步验证本文提出的 DAPF-RRT 算法在空间中规划的路径经过修剪与三次 B 样条平滑后的路径是否满足机械臂的运动学约束。在 MATLAB 中使用 Robotics Toolbox 导入六轴机械臂模型，在空间中创建球体障碍物，使用 DAPF-RRT 算法进行了路径规划仿真。

导入 D-H 参数并利用 Robotics Toolbox 中的 Link 函数和 Seriallink 函数建立机械臂模型。本文使用改进的 DH 参数法，可以选择“modified”建立相应杆件，通过“offset”设置关节初始位置，建立的机械臂模型如图 4-15 所示：

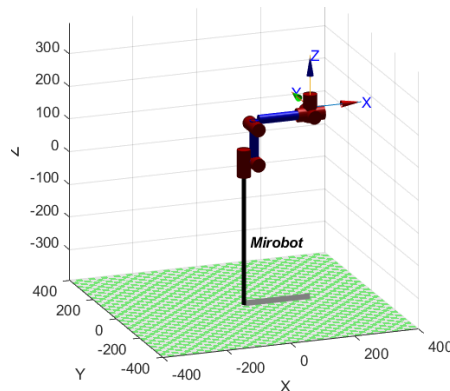


图 4-15 Mirobot 机械臂模型

机械臂路径规划仿真过程图 4-16 所示：

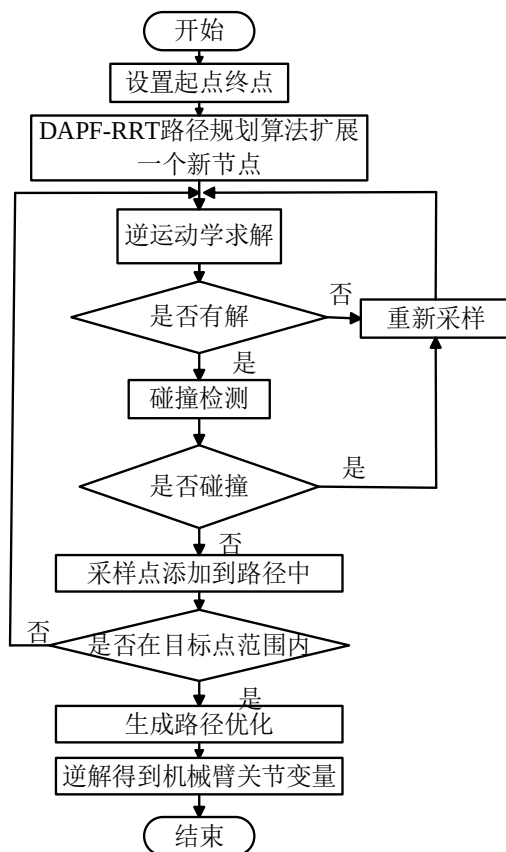


图 4-16 机械臂路径规划仿真过程

仿真过程具体步骤如下：

- (1)设置起点与终点，初始化三维空间；
- (2)利用 DAPF-RRT 算法产生一个新的候选节点；
- (3)检查该节点对应的末端执行器位姿是否存在解析解,若无解则放弃该节点,重复步骤 2；
- (4)采用包络法判断各连杆在该节点时是否会与障碍物发生碰撞，发生碰撞则舍弃该节点，否则视为一个潜在路径点；
- (5)判断新节点是否到达目标区域附近，未到达则循环步骤 2，到达时生成初步路径并进行后续优化；
- (6)求解每个路径点处的逆运动学解，从而得到机械臂各关节的角度值序列。

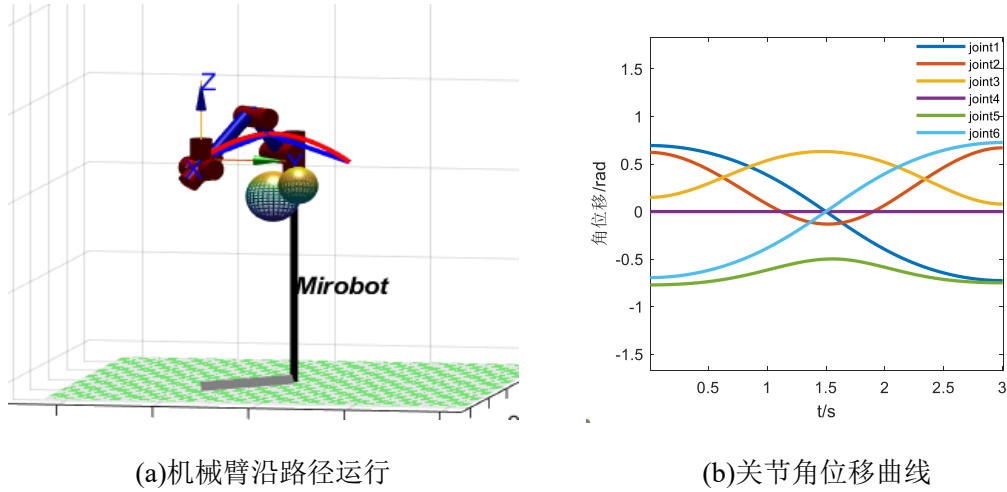


图 4-17 机械臂沿轨迹运动效果图

机械臂沿路径运行如图 4-17(a)所示。图中 4-17(a)蓝色实线为修剪后但未经平滑的路径，红色实线代表经过三次 B 样条平滑处理后路径。图 4-17(b)为机械臂运动结束后输出的各关节角位移曲线，机械臂经过沿平滑后的路径运动时，关节角位移曲线变化连续、无跳变，保证了机臂在规划路径上的平稳运行。

4.6 本章小结

本章首先介绍了 RRT 和 RRT-Connect 算法的基本原理。针对 RRT-Connect 算法的不足做出以下改进：引入目标偏置策略，使新节点以更大概率向目标扩展。为改善原算法使用固定步长带来的扩展缓慢问题，引入了动态步长。通过融合目标引力函数、障碍物斥力函数及采样函数对新节点扩展进行共同约束，优化节点生长策略，解决 RRT-Connect 算法中节点扩展随机，导向性弱的问题。通过使用贪婪策略对规划后的路径进行修剪使路径更短，转折更少。利用三次 B 样条曲线法使修剪后路径更平滑，以减少机械臂沿路径运动时因路径转折较多产生的抖动。通过 MATLAB 对 DAPF-RRT 算法的有效性进行了验证，使用 MATLAB 中的 Robotics Toolbox 对机械臂路径规划进行仿真，通过仿真可以看出机械臂关节角位移曲线较平滑。综上所述，本章为机械臂用于垃圾分拣任务中的路径规划方法提供了技术支撑。

第 5 章 机械臂垃圾分拣方法应用与实现

5.1 引言

第 3 章对 YOLOv5s 算法进行了改进,提升了算法在垃圾检测任务上的性能。第 4 章提出了 DAPF-RRT 机械臂路径规划算法,缩短规划时,获得更优路径。本章搭建了机械臂垃圾分拣实验平台,完成相机标定实验与手眼标定实验,获得了相机内参与相机坐标系与机械臂坐标系间的转换关系。将改进后的目标检测算法、DAPF-RRT 路径规划算法在实际场景下进行测试。最后进行了垃圾分拣实验,验证了基于深度学习的机械臂垃圾分拣方法的可行性。

5.2 实验平台介绍

实验平台采用一台搭载 Intel (R) Core (TM) i5-1135G7 CPU@2.40 G 芯片的 PC 机作为上位机,操作系统为 Ubuntu20.04。上位机使用 Modbus-TCP 通讯连接的运动控制器,由运动控制器驱动的六轴微型工业机械臂机 Mirobot。机械臂与深度相机采用“眼在手外”的方式,如图 5-1 所示。软件部分使用 Ubuntu 20.04 操作系统,ROS(Robot Operating System)系统,Moveit!功能包,RViz。



图 5-1 垃圾分拣实验平台

在垃圾分拣任务中,需要根据目标的位置信息来完成分拣,目标位置信息包括目标的 X-Y-Z 三维坐标。普通 RGB 相机只能获取目标的 X-Y 轴信息,无法获取目标的深度信息。为完成分拣任务,选择 Intel RealSense D435i 深度相机(后简称 D435i 深度相机)来获取图像信息与深度信息。D435i 视觉深度相机前端包括了一对立体红外传感器、红外激光发射器和 RGB 相机,实验实物图如图 5-2 所示。



图 5-2 Intel RealSense D435i 深度相机

5.3 相机标定及手眼标定

5.3.1 D435i 深度相机标定

为了建立图像像素坐标系与摄像机坐标系之间的关系,需要对摄像机进行标定获得相机内参。常见的标定方法包括传统相机标定法、相机自标定法和主动视觉标定法。考虑到成本、精度和鲁棒性等原因,采用通 Matlab 中 `camera_calibration` 功能包对相机标定,该功能包使用张正友标定法^[72]。在标定时仅需要使用相机拍摄不少于三张的不同位姿的标定板即可标定出相机内参。该方法相对简单且鲁棒性好、精度高。

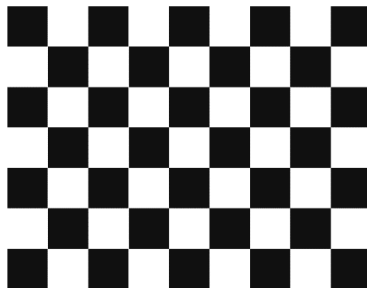


图 5-3 标定棋盘格

如图 5-3 所示，实验设定的标定棋盘的规格为 10×7 ，标定板中白格和黑格的大小设定为 $25\text{mm} \times 25\text{mm}$ 。

将标定棋盘打印贴在刚性板上，使用 D435i 相机从不同位置和角度拍摄出 20 组照片。在 Camera Calibrator 中导入图片，对角点进行提取，角点为两黑格连接处，图 5-4(a)为角点提取后的结果，图(b)为标定板与相机间的位置。

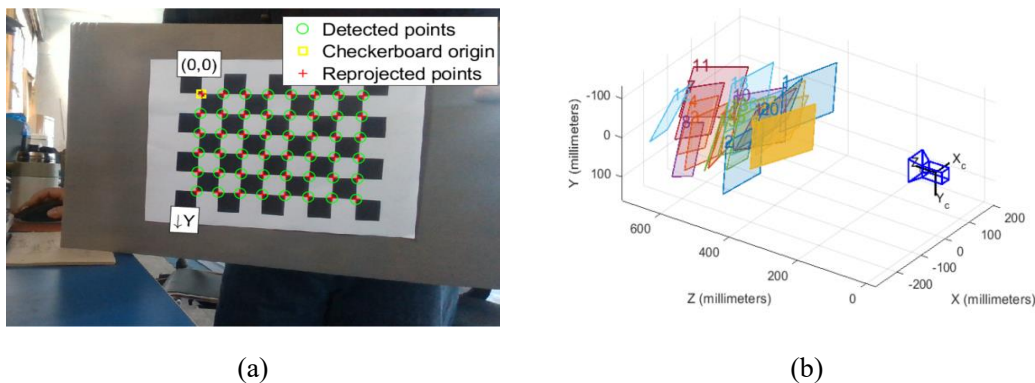


图 5-4 角点标定结果与标定板相对相机之间的位置关系

标定后的相机内参数如表 5-1 所示：

表 5-1 相机内参数

	参数	RGB 相机
焦距	f_x	606.948
	f_y	603.736
中心点	ppx	330.244
	ppy	242.714

5.3.2 手眼标定

获得待分拣目标的相机坐标系下的位置后，还需要确定机械臂坐标系与相机坐标系之间的变换关系，以获得目标在机械臂坐标系下的位置，手眼标定正是用来求解这一变换关系的方法。通过标定机械臂能够获取目标在机械臂坐标系下的位置，使机械臂能够到达目标位置完成分拣。

本文采用眼在手外(Eye to hand)的方式，相机固定在工作台上，并面向机械臂工作台，具体布局如图 5-5 所示。

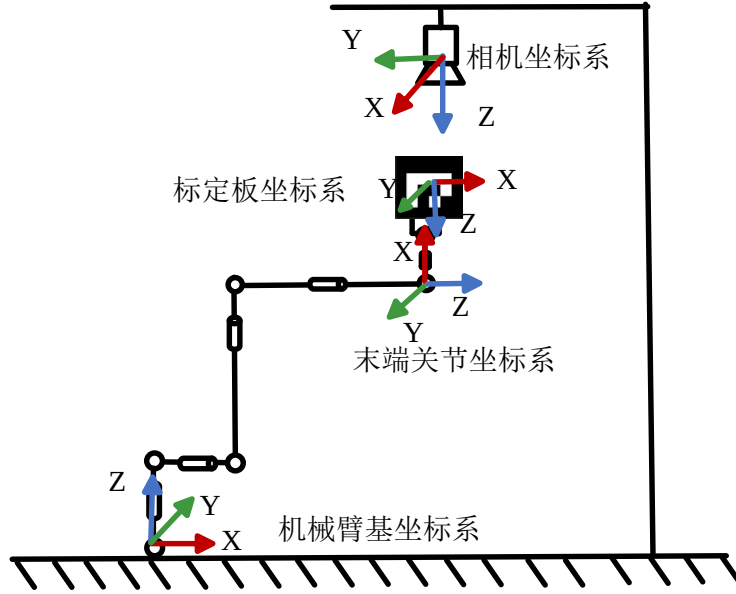


图 5-5 眼在手外示意图

眼在手外标定的标定原理为：设机械臂末端关节坐标系在机械臂基坐标系下位姿为 ${}^{base}_{end}T$ ，末端关节坐标系到标定板坐标系间转换为 ${}^{end}_{board}T$ ，标定板坐标系到相机坐标系之间转换为 ${}^{board}_{camera}T$ 。机械臂基坐标系与相机坐标系间变换为 ${}^{base}_{camera}T$ ，也就是所求解的转换关系，由于采用眼在手外，相机相对机械臂转换关系 T^{base}_{camera} 不变。

通过移动机械臂可以获得前后两次 ${}^{end1}_{board1}T$ 、 ${}^{end2}_{board1}T$ 如式(5-1)与(5-2)所示：

$${}^{end1}_{board1}T = ({}^{base}_{end1}T)^{-1} * {}^{base}_{camera}T * {}^{camera1}_{board1}T \quad (5-1)$$

$${}^{end2}_{board2}T = ({}^{base}_{end2}T)^{-1} * {}^{base}_{camera}T * {}^{camera2}_{board2}T \quad (5-2)$$

由于标定板是置于末端关节上，因此 ${}^{end}_{board}T$ 为定值，也就是 ${}^{end1}_{board1}T = {}^{end2}_{board2}T$ ，因此可得式(5-3)，通过整理可得式(5-4)。

$$({}^{base}_{end1}T)^{-1} * {}^{base}_{camera}T * {}^{camera1}_{board1}T = ({}^{base}_{end2}T)^{-1} * {}^{base}_{camera}T * {}^{camera1}_{board2}T \quad (5-3)$$

$$({}^{base}_{end2}T * ({}^{base}_{end1}T)^{-1}) * {}^{base}_{camera}T = {}^{base}_{camera}T * ({}^{camera1}_{board2}T * ({}^{camera1}_{board1}T)^{-1}) \quad (5-4)$$

令 $A = ({}^{base}_{end2}T * ({}^{base}_{end1}T)^{-1})$ ， $B = ({}^{camera1}_{board2}T * ({}^{camera1}_{board1}T)^{-1})$ ， $X = {}^{base}_{camera}T$ ，上式就转化为 $AX=XB$ 形式，可以利用最小二乘法进行求解 X 。

标定过程中使用 ArUco 标签作为标定板，ArUco 标签能够通过 Opencv 快速被识别，提供其在空间中位置和姿态信息。本次实验使用 ArUco 标签 ID 为 1，

标签大小为 85mm×85mm，使将 ArUco 标签置于机械臂末端，使其中心点位于末端端正中心。

ROS和Moveit!环境下，通过调用MoveGroupInterface::getCurrentPose函数可以获得 ${}^{end1}_{board1}T$ 与 ${}^{base}_{end2}T$ ， ${}^{camera}_{board1}T$ 与 ${}^{camera}_{board2}T$ 则可以通过Opencv中的库函数对机械臂末端ArUco码进行识别获得。图5-6为Moveit!中手眼标定过程，通过控制机械臂使末端的ArUco码出现在相机画面中，并移动机械臂末端，使ArUco码的姿态不断变化。

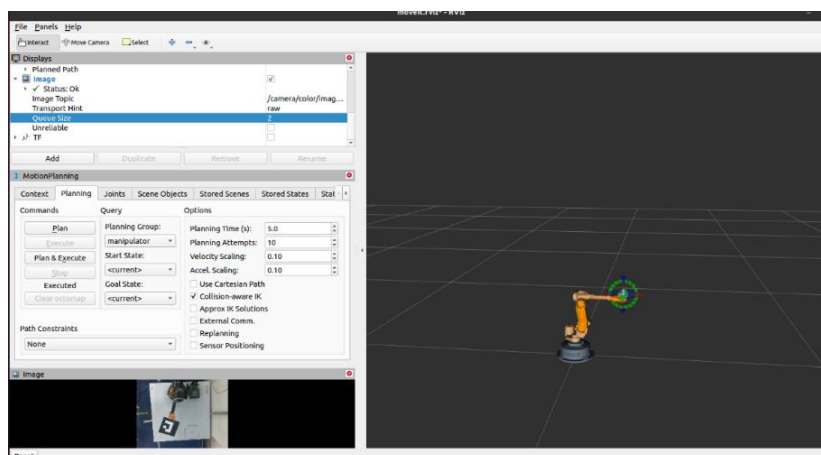


图 5-6 Moveit!中手眼标定过程

一般需要记录三组以上的数据才可以确定位姿数据，实验采集了 10 次不同坐标位置，标定过程部分图片如图 5-7 所示：

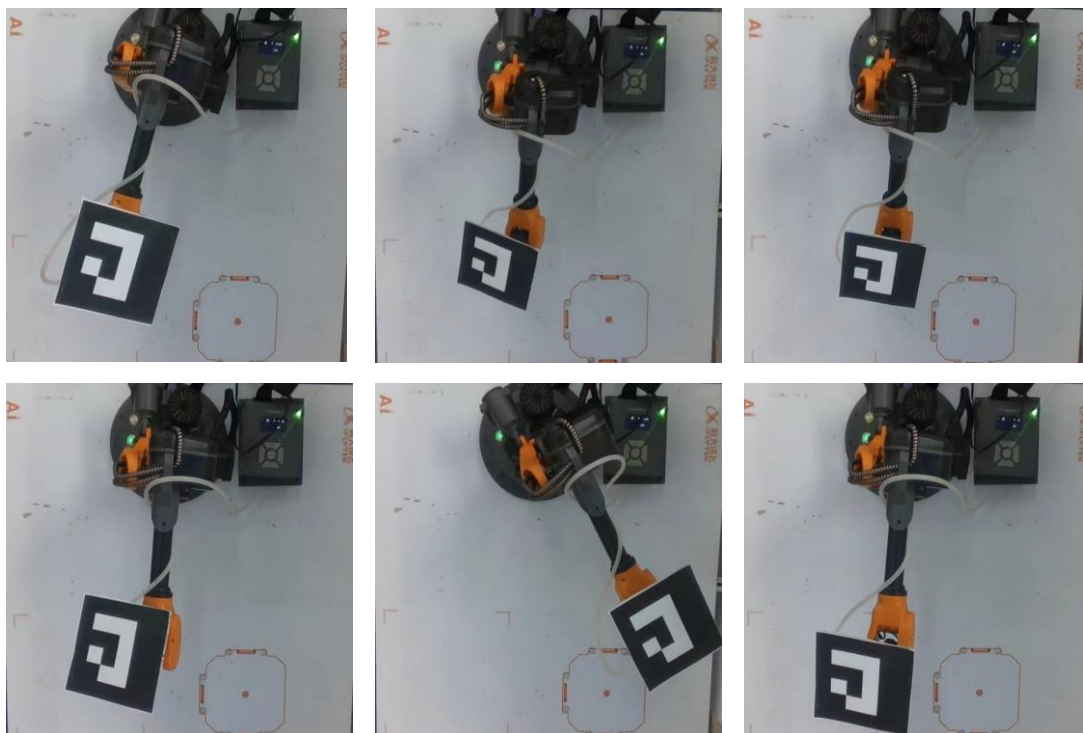


图 5-7 标定过程部分图像

手眼标定的最终获如下：

$${}_{base}^{camera}T = \begin{bmatrix} 0.006788 & -0.003361 & 0.999975 & 0.336348 \\ 0.999977 & 0.000817 & -0.006790 & -0.009539 \\ -0.000477 & 0.999993 & 0.003370 & 0.79449 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5-5)$$

Moveit!中通过标定结果确定的相机与机械臂基座标的位姿关系如图 5-8 所示，与现实中搭建的相机和机械臂位姿关系一致。图中黄色直线为从相机坐标系指机械臂基座标系的箭头。

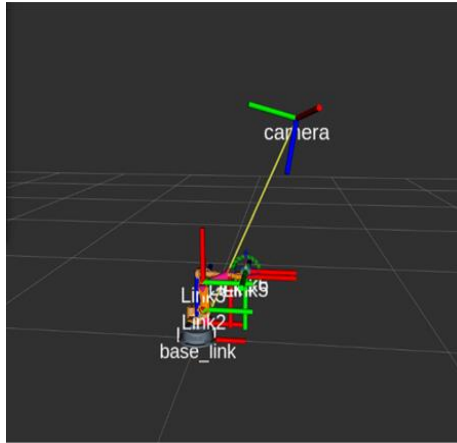


图 5-8 手眼标定结果

5.4 目标检测算法部署与 ROS 环境下机械臂路径规划实验

5.4.1 目标检测算法部署

第 3 章中提出一种基于改进 YOLOv5s 的垃圾检测算法，改进后算法在检测精度与检测能力上有明显提升。在上位机上完成部署，通过 D435i 深度相机实时目标检测结果如图 5-9 所示。图(a)为 YOLOv5s 算法，图(b)为改进后算法，当目标较为密集，同时部分有遮挡时，原始 YOLOv5s 算法面对这种情况出现了漏检，dry cell 和 chrisma 都未检出；检测目标尺寸多样，改进后的算法面对不同尺寸目标，展现了更好的检测能力。在模拟垃圾分拣场景中改进后的算法特征提取能力、鲁棒性更强。

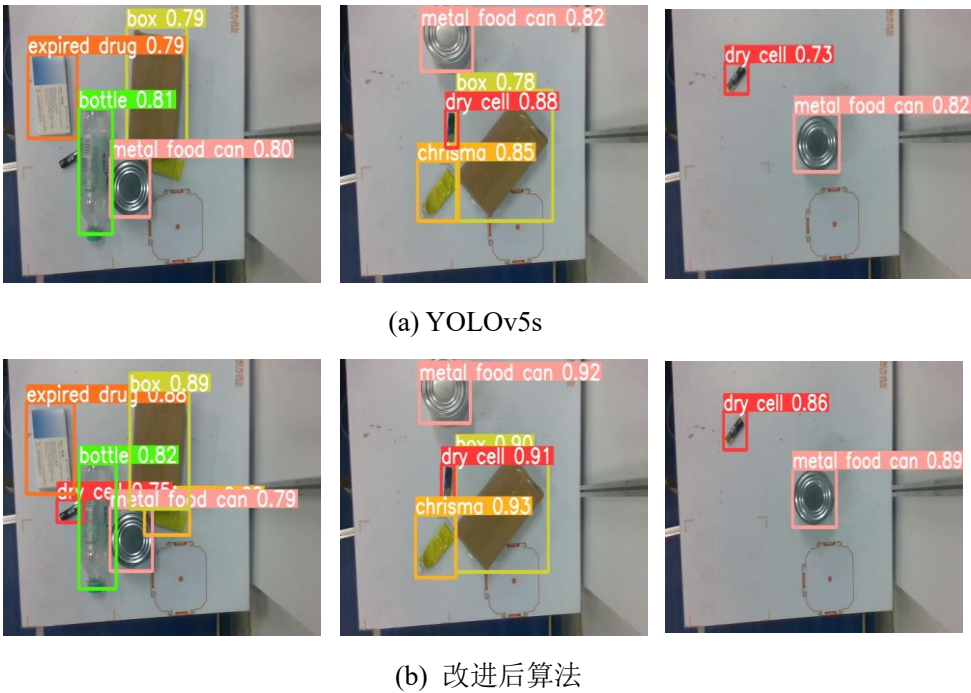


图 5-9 模拟垃圾分拣场景下检测效果

5.4.2 ROS 环境下机械臂路径规划实验

本次实验是在 ROS 环境下通过 Moveit!实现对机械臂的控制。Moveit!功能包涵盖了运动规划、路径规划、三维感知集成、运动学计算和碰撞检测等关键领域^[73]。利用这些功能，用户可以实现复杂的机械臂任务规划和执行，同时确保机械臂的动作符合物理约束和避免碰撞。

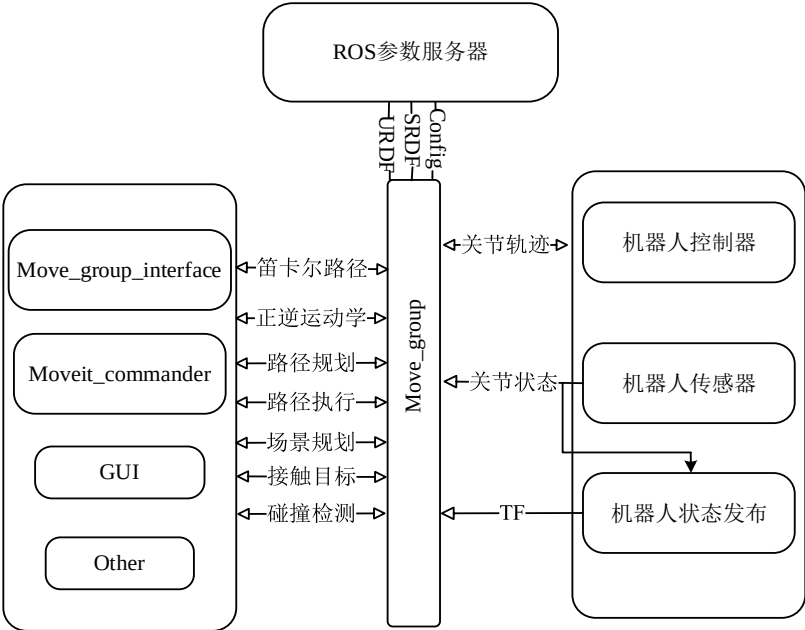


图 5-10 Moveit!基本架构

Moveit!的基本架构如图 5-10 所示。Moveit!核心组件 Move_group 协调机械臂的各个模块，提供了一整套动作指令和服务接口给用户。Move_group 通过订阅 ROS 参数服务器来获取机器人的统一机器人描述格式(URDF)和语义机器人描述格式(SRDF)以及其他 Moveit!配置，如关节限制、运动学参数、运动规划策略和感知等信息，这些配置由 Moveit!Setup Assistant 自动生成，并保存在机器人的 Moveit!配置包中。在 Moveit!框架中，常用的是可作为插件集成的开源运动规划库 OMPL (The Open Motion Planning Library)。该库提供一系列的规划器以实现路径规划，且允许用户根据具体需求对规划算法进行定制和重写。

Rviz(ROS Visualization)是一个 3D 可视化工具^[74]，用于 ROS 环境中。它允许用户以图形化方式查看模拟或实际机器人的传感器数据、状态信息、计划的运动等。Rviz 在本次实验中实时显示机器人模型，如图 5-11 所示，并按照真实环境中在 Rviz 中添加障碍物。

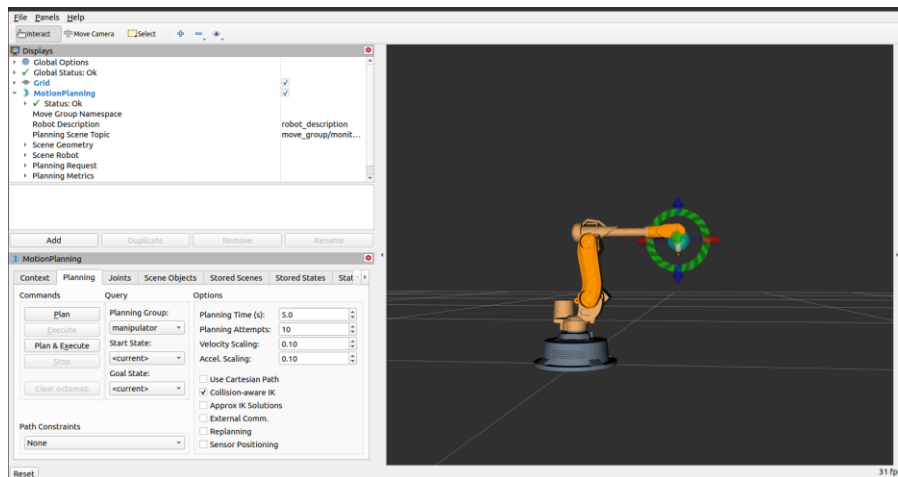


图 5-11 Rviz 中的 Mirobot 机械臂

为了进一步验证 DAPF-RRT 算法的可行性，本章在仿真环境和现实环境下控制机械臂运动，实验步骤如下：

- (1)在 Ubuntu 操作系统中安装 Moveit!和其他必要的功能包，以搭建实验所需的软件环境；
- (2)计算障碍物相对于真实机械臂的位置来获取障碍物的坐标信息；
- (3)使用 Moveit!功能包，结合 Rviz 工具，建立机械臂的仿真模型。同时，根据障碍物的坐标信息在仿真环境中添加相对应的场景物体；
- (4)在开源运动库中 OMPL 中添加 DAPF-RRT 算法；

(5)准备工作完成后，开始进行机械臂的路径规划实验。

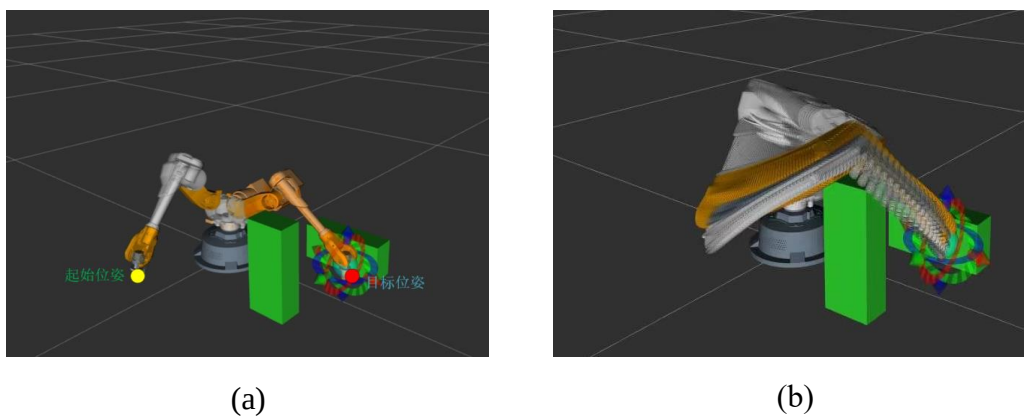


图 5-12 机械臂运动轨迹

图 5-12(a)为设置的起始位姿与目标位姿，图(b)为仿真环境中机械臂沿后 DAPF-RRT 算法规划路径运动的轨迹，绿色长方体为障碍物，根据现实障碍物信息导入至 Rviz 中。

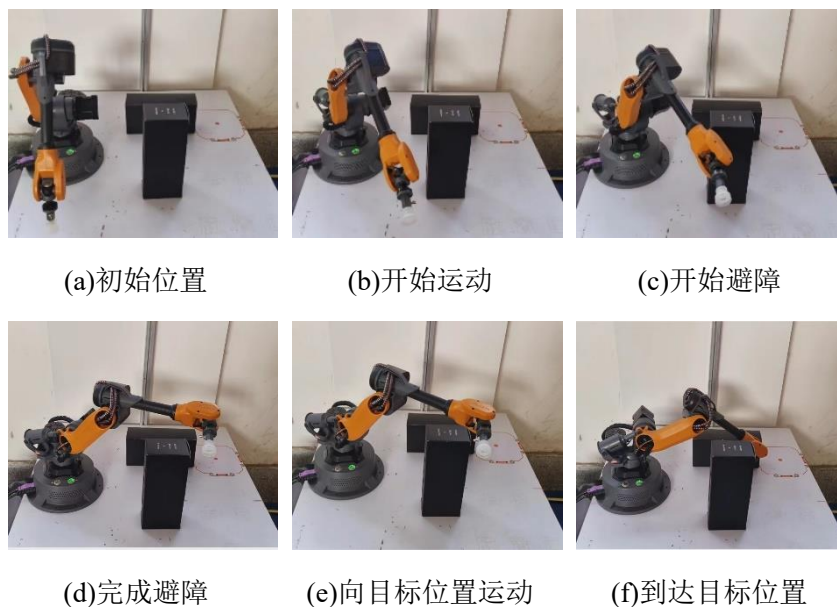


图 5-13 机械臂沿 DAPF-RRT 算法规划路径运动

通过在环境中创建两个节点一个用于 Rviz 中发布的/joint_states 消息，另一个节点订阅此主题，并按照 G 代码协议发布控制指令控制 Mirobot 机械臂运动。通过此种方式实现仿真与真实机械臂同步运动。图 5-13(a)为机械臂起始位置，等待路径规划完成后，开始运动。5-13(b)到 5-13(e)为机械臂沿规划路径运动过程，图 5-13(f)机械臂运动到达目标位置。

5.5 机械臂垃圾分拣实验

将改进后的目标检测算法与 DAPF-RRT 机械臂路径规划算法应用到机械臂分拣实验中，分拣流程大体如图 5-14 所示。

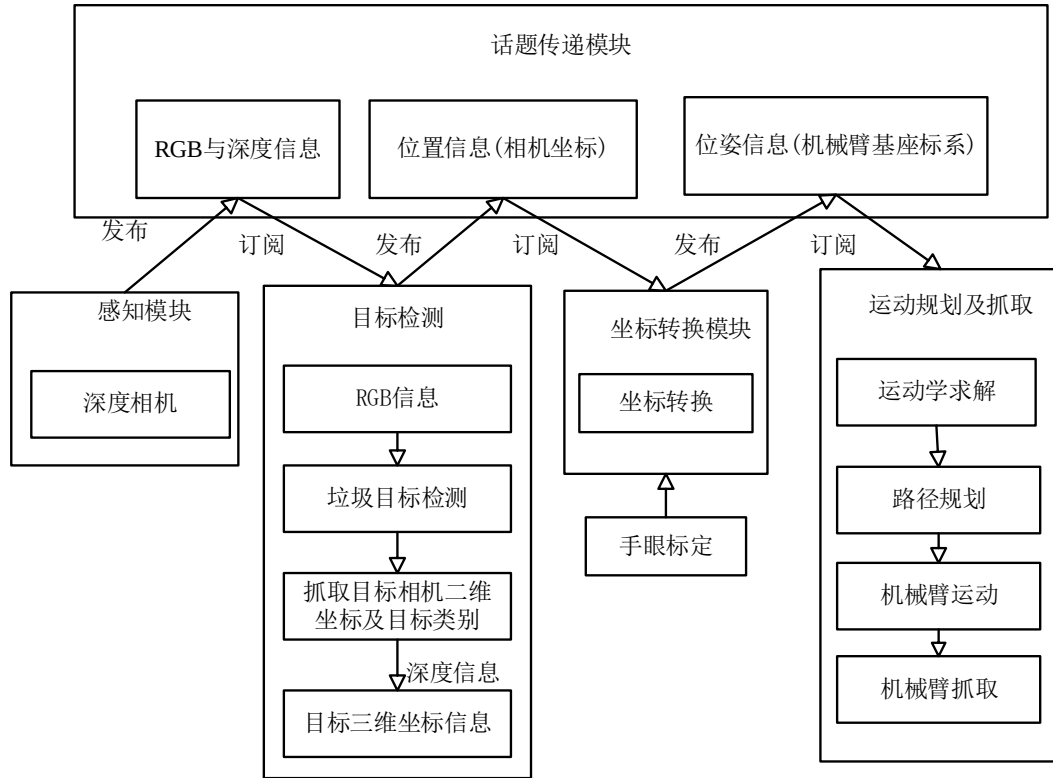


图 5-14 分拣实验流程图

实验具体步骤如下：

- (1)初始化系统，机械臂与上位机建立通讯；
- (2)订阅 D435i 深度相机获取 RGB 图像，使用改进后的目标检测算法对输入 RGB 图像进行检测，输出目标的像素坐标系下的二维坐标；
- (3)经过转换将像素坐标系下坐标转换为相机坐标系下坐标，融合深度信息，获得目标相机坐标系下的三维坐标信息，并发布出去；
- (4)订阅目标位置信息的话题，根据 5.3 节中手眼标定的结果进行变换，将相机坐标系下的目标位置转换为机械臂坐标系；
- (5)获得目标位置，在 Moveit! 平台下利用 DAPF-RRT 算法进行路径规划，执行轨迹并完成分拣。

本次实验选择检测框中心位置作为机械臂分拣位置，如图 5-15 中白色实心点。通过垃圾目标检测算法获得目标像素坐标系下的二维坐标，经过转换和结合

D435i 相机输出的深度信息，就得到了目标相机坐标系下的三维信息。由于本文并未为对目标的位姿估计进行研究，本次实验将目标放置在于机械臂同一平面。

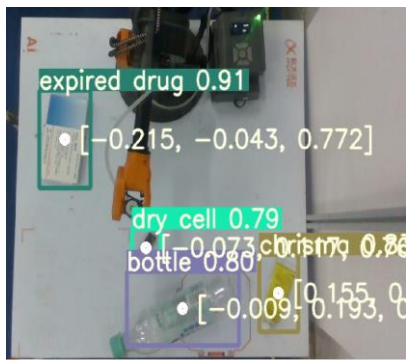


图 5-15 相机画面

5.5.1 机械臂工作空间中无障碍物

本次实验设置了空间中无障碍物与空间中有障碍物两种情况，并分为两种方案，方案 1 使用 YOLOv5s 与 DAPF-RRT 算法，方案 2 使用改进后的目标检测算法算法与 DAPF-RRT 算法。本次实验进行了 20 组实验，分拣目标为 *expire drug*，并将其放置在不同位置。识别出目标对象类别同时完成路径规划，到达分拣目标位置完成分拣操作即为成功。

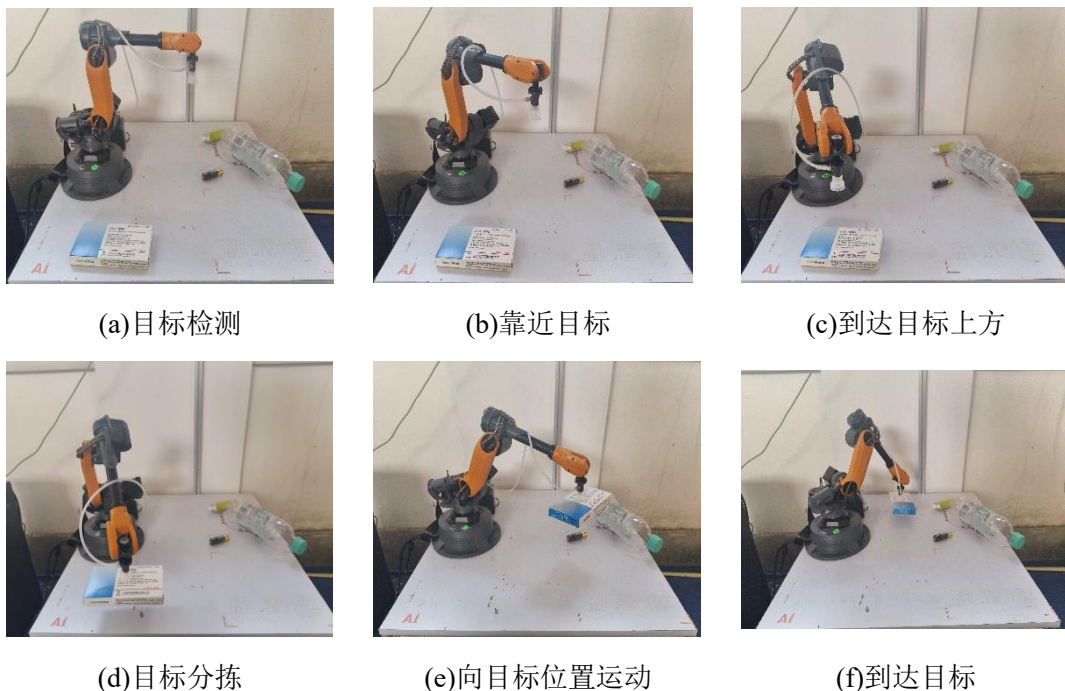


图 5-16 工作空间中无障碍物机械臂分拣过程

机械臂成功进行抓取任务的过程可如图 5-16，六幅图展示了抓取过程的不

同阶段。图 5-16(a)为执行目标检测，机械臂在等待接收抓取指令的初始状态。在这一阶段，机械臂保持静止，等待识别并定位目标物体。图(b)为接收到目标物体位置信息后，通过 Moveit!平台进行运动规划，规划完成后机械臂开始向目标物体移动。图(c)显示了机械臂按照规划的路径到达目标物体的位置，并成功吸起目标物体的瞬间。图(d)到(f)为完成分拣后，返回固定位置。实验结果如表 5-2 所示，20 次实验中使用方案 2 全部识别成功，并成功完成分拣。

表 5-2 工作空间中无障碍物分拣结果

方案	目标识别成功率	完成分拣平均时间/s
1	85%	18.4
2	100%	19.1

5.5.2 机械臂工作空间中有障碍物

图中 5-17 黑色长方体为障碍物，模拟分拣过程中机械臂操作空间中存在障碍物的情况，分拣前将障碍物信息导入到 Rviz 中。分拣过程如图 5-17 所示，过程与上一小节相似，增加了图 5-17(e)中避障过程。

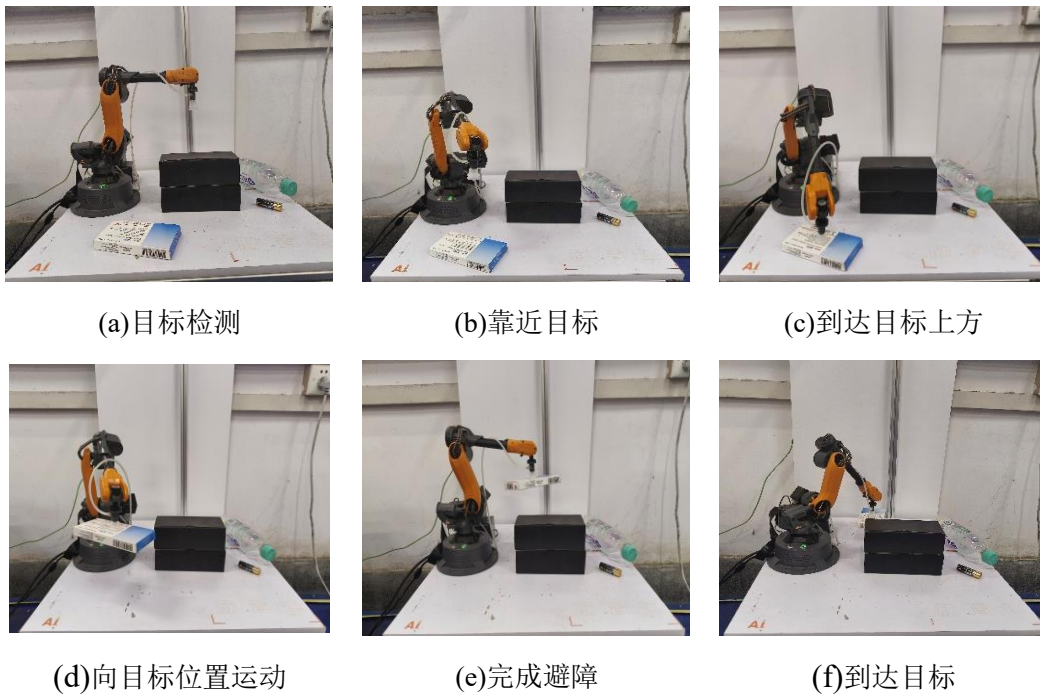


图 5-17 机械臂分拣过程

20 组实验中，方案 2 目标全部识别成功，完成避障垃圾分拣成功次数为 17 次；方案 1 目标识别成功 18 次，分拣成功次数为 15 次。表 5-3 为其中六组实验中的数据，从表中 5-3 可以看出，在识别精度方面方案 2 较方案 1 有一定提升。

方案 2 分拣的失败的原因主要包括以检测框中心作为抓取点的方法并不准确,手眼标定存在误差,坐标转换不精确。

表 5-3 六组实验部分数据

目标三维坐标	置信度	
	方案 1	方案 2
(-0.198, 0.161, 0.773)	80%	89%
(-0.101, 0.163, 0.772)	81%	87%
(-0.186, -0.017, 0.778)	84%	89%
(-0.263, -0.085, 0.769)	80%	87%
(-0.256, -0.155, 0.774)	85%	89%
(-0.277, 0.175, 0.771)	86%	86%

5.6 本章小结

本章使用 Mirobot 与 D435i 深度相机完成垃圾分拣平台的搭建,相机与机械臂采用眼在手外的配置方式。通过完成相机标定与手眼标定,获得相机内参与完成相机坐标系与机械臂坐标系之间的转换。在模拟垃圾分拣场景下,验证了改进 YOLOv5s 垃圾检测算法的性能。在 ROS 环境中通过在 Moveit!导入修改后的 DAPF-RRT 机械臂路径规划算法,按照真实环境导入障碍物,完成对 DAPF-RRT 算法有效性的验证。最后结合改进后的目标检测算法与路径规划算法,验证了所提垃圾分拣方法的可行性。

第 6 章 总结与展望

6.1 总结

目前基于深度学习的机械臂垃圾分拣方法涉及的垃圾目标检测算法与机械臂路径规划尚有提升空间：一方面现有的垃圾检测算法面对场景复杂，各类别间尺寸差异大时，存在漏检、误检，同时还存在模型参数量大的问题。另一方面 RRT-Connect 算法存在算法搜索时间长，扩展盲目，路径不平滑等问题。针对以上问题本文的主要工作内容如下：

(1)提出了基于改进 YOLOv5s 垃圾目标检测算法。首先设计了 C3_GD 与 C3Ghost 模块，分别在主干与颈部使用，用以改善模型在背景复杂情况下的的特征提取能力，和减少模型参数量。构建 I-BIC 模块，改善对不同尺度目标识别精度。最后引入 SimAM 注意力机制，抑制背景噪声影响，提升模型性能。在自制数据集上，与原始模型相比，改进后模型的 $mAP@0.5$ 提高了 2.9%，精度提高 4.3%，召回率提高 2.8%。在公开的 TrashNet 数据集上改进后模型 $mAP@0.5:0.95$ 增加了 4.9%。

(2)提出了 DAPF-RRT 机械臂路径规划算法。使用种动态步长策略，改善固定步长在加速在密集区域的扩展效率。减少生长过程中的盲目性，使新节点的生长受到随机采样函数、目标引力函数和障碍物斥力函数的约束。通过路径剪枝减少冗余点与转折。最后通过三次 B 样条对路径进行平滑处理。在相同的环境下 DAPF-RRT 算法相较于原算法路径长度和规划时间分别下降了 15.4%和 49.2%；路径中的节点下降了 84.2%。在障碍物数量不同的环境中，改进后的算法同样有更好的性能。在最后通过导入机械臂模型验证了 DAPF-RRT 算法规划的路径能够保证了机臂在规划路径上的平稳运行。

(3)机械臂垃圾分拣方法应用与实现。搭建了机械臂与深度相机融合的垃圾拣实物平台。完成了改进的垃圾目标检测算法的部署，并验证了检测性能，在 ROS 环境下通过导入 DAPF-RRT 算法完成了路径规划仿真，并控制真实机械臂完成避障。最后结合改进后的目标检测算法与路径规划算法，验证了基于深度学

习的机械臂垃圾分拣方法的可行性。

6.2 展望

由于实验室环境与设备限制，研究中存在亟待改进与有待研究的之处，主要包括以下几个方面：

(1)文中使用的机械臂末端使用的是吸盘，对于重量较轻，表面平整的分拣目标来说，分拣的成功率较高。但并距离实际场景下的垃圾分拣有一定差距，可以选择末端执行器为夹爪或带有双末端执行器的工业机械臂进行分拣。

(2)本文并未对目标分拣位姿进行研究。例如可以利用 LINEMOD 的算法，通过对目标的位姿的获取，机械臂能够更准确的进行分拣。

(3)本文检测算法并未在嵌入式设备上移植，可以通过对网络进行剪枝或者蒸馏等，在进一步提升检测速度的同时，使算法更适合在嵌入式设备上部署。

参考文献

- [1] 叶瑞克, 卞梦颖, 易珂琪. 城市生活垃圾分类准确率研究: 基于环境关心、制度信任和社区归属感的分析[J]. 环境污染与防治, 2023, 45 (12): 1743-1750.
- [2] 郭明坤, 王菲, 刘宽等. 黄河流域主要城市生活垃圾处理处置特征与对策建议[J]. 环境科学研究, 2024, 37(01): 151-159.
- [3] 张乃仁. 生活垃圾焚烧处理的环境危害[J]. 生态经济, 2023, 39(11): 5-8.
- [4] 高飞. 基于机器视觉的景区清洁机器人垃圾分拣方法研究[D]. 河北科技大学, 2023.
- [5] 广东省人民政府办公厅转发国务院办公厅关于转发国家发展改革委住房城乡建设部生活垃圾分类制度实施方案的通知[J]. 广东省人民政府公报, 2017, (32): 10-16.
- [6] 我国自 2019 年起在全国地级及以上城市全面启动生活垃圾分类[J]. 大社会, 2019, (Z1): 13.
- [7] 靳晓菲. 我国城市生活垃圾分类政策实施现状与改进对策[J]. 清洗世界, 2020, 35(12): 112-114.
- [8] 陶永, 王田苗, 刘辉等. 智能机器人研究现状及发展趋势的思考与建议[J]. 高技术通讯, 2019, 29(02): 149-163.
- [9] 房德职, 李克勋. 国内外生活垃圾焚烧发电技术进展[J]. 发电技术, 2019, 40(4): 367-376.
- [10] 田震. 基于深度学习的垃圾分类算法研究[D]. 哈尔滨工业大学, 2022.
- [11] 李俊杰. 基于深度学习的垃圾分拣技术研究[D]. 天津科技大学, 2024.
- [12] Deng B, Lv H. Survey of target detection based on neural network[C]//Journal of Physics: Conference Series. IOP Publishing, 2021, 1952(2): 022055.
- [13] Lowe D G. Object recognition from local scale-invariant features[C]//Proceedings of the seventh IEEE international conference on computer vision. Ieee, 1999, 2: 1150-1157.
- [14] Cortes C, Vapnik V. Support-vector networks[J]. Machine learning, 1995, 20: 273-

-
- [15] Yang M, Thung G. Classification of trash for recyclability status[J]. CS229 project report, 2016(1): 3-9.
 - [16] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.
 - [17] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.
 - [18] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. Advances in neural information processing systems, 2015, 28.
 - [19] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]//Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing, 2016: 21-37.
 - [20] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.
 - [21] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.
 - [22] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arxiv preprint arxiv:1804.02767, 2018.
 - [23] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arxiv preprint arxiv:2004.10934, 2020.
 - [24] Wang Y, Zhang X. Autonomous garbage detection for intelligent urban management[C]//MATEC Web of Conferences. EDP Sciences, 2018, 232: 01056.
 - [25] Ma W, Wang X, Yu J. A lightweight feature fusion single shot multibox detector for garbage detection[J]. IEEE Access, 2020, 8: 188577-188586.
 - [26] Sun Q, Zhang X, Li Y, et al. YOLOv5-OCDS: An Improved Garbage Detection Model Based on YOLOv5[J]. Electronics, 2023, 12(16): 3403.

-
- [27] Chen L, Zhu J. Water surface garbage detection based on lightweight YOLOv5[J]. Scientific Reports, 2024, 14(1): 6133.
- [28] 罗安能, 万海斌, 司志巍等. 基于改进 YOLOv5s 的可回收垃圾检测算法[J]. 激光与光电子学进展, 2023, 60(10): 130-137.
- [29] 魏铖磊, 南新元, 李成荣等. 一种具有多尺度感受视野注意力机制的生活垃圾单阶段目标检测方法[J]. 环境工程, 2022, 40(01): 175-183.
- [30] Dijkstra E W. A note on two problems in connexion with graphs[M]//Edsger Wybe Dijkstra: His Life, Work, and Legacy. 2022: 287-290.
- [31] Hart P E, Nilsson N J, Raphael B. A formal basis for the heuristic determination of minimum cost paths[J]. IEEE transactions on Systems Science and Cybernetics, 1968, 4(2): 100-107.
- [32] Li Y, Zhao J Y, Chen Z H, et al. A robot path planning method based on improved genetialgorithm and improved dynamic window approach[J]. Sustainability, 2023, 15(5): 4656.
- [33] Ni Y, Zhuo Q H, Li N, et al. Characteristics and optimization strategies of A* algorithm and ant colony optimization in global path planning algorithm[J]. International Journal of Pattern Recognition and Artificial Intelligence, 2023, 37(03): 8345-8352.
- [34] Huang C, Zhou X B, Ran X J, et al. Adaptive cylinder vector particle swarm optimization with differential evolution for UAV path planning[J]. Engineering Applications of Artificial Intelligence, 2023, 121(01): 105942.
- [35] 冯林, 贾菁辉. 基于对比优化的 RRT 路径规划改进算法[J]. 计算机工程与应用, 2011. 47(03): 210-213+228.
- [36] Lavalley S. Rapidly-exploring random trees: a new tool for path planning[J]. Research Report, 1998: 293-308.
- [37] Kavraki L E, Svestka P, Latombe J C, et al. Probabilistic roadmaps for path planning in high-dimensional configuration spaces[J]. IEEE transactions on Robotics and Automation, 1996, 12(4): 566-580.
- [38] Ding J, Zhou Y, Huang X, et al. An improved RRT* algorithm for robot path planning based on path expansion heuristic sampling[J]. Journal of Computational

Science, 2023, 67: 101937.

- [39] Li J, Huang C, Pan M. Path-planning algorithms for self-driving vehicles based on improved RRT-Connect[J]. Transportation safety and environment, 2023, 5(3): tdac061.
- [40] 尹庆文. 基于 GB-APFB-RRT~*算法的机械臂路径规划[J]. 组合机床与自动化加工技术, 2023, (10): 11-15.
- [41] 江韩, 晁永生, 周江林等. 改进 RRT 算法的机械臂路径规划[J]. 机械设计与制造, 2023, (12): 288-292.
- [42] Li Z, Liu F, Yang W, et al. A survey of convolutional neural networks: analysis, applications, and prospects[J]. IEEE transactions on neural networks and learning systems, 2021, 33(12): 6999-7019.
- [43] Lecun Y, Bottou L. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
- [44] Chen L, Li S, Bai Q, et al. Review of image classification algorithms based on convolutional neural networks[J]. Remote Sensing, 2021, 13(22): 4712.
- [45] Apicell A, Donnarumma F, Isgr F, Prevete R. A survey on modern trainable activation functions[J]. Neural Networks, 2021, 32(05): 128 -138.
- [46] Gu J, Wang Z, Kuen J, et al. Recent advances in convolutional neural networks[J]. Pattern recognition, 2018, 77: 354-377.
- [47] 胡子瑞, 王丙旭, 杨景等. 基于轨迹灵巧度的机械臂尺寸优化[J]. 制造业自动化, 2023, 45(11): 96-101.
- [48] Denavit J, Hartenberg R S. A Kinematic Notation for Lower-Pair Mechanisms[J]. Journal of Applied Mechanics, 1955, 22: 215-221.
- [49] Zhou D, Xie M, Xuan P, etc. A teaching method for the theory and application of robot kinematics based on MATLAB and V-REP[J]. Computer Applications in Engineering Education, 2020, 28(2): 239-253.
- [50] 贾瑞清, 周东旭, 谢明佐. 机器人学: 规划、控制及应用[M]. 北京: 清华大学出版社, 2020.
- [51] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]//Proceedings of the IEEE conference on computer vision and pattern

-
- recognition. 2017: 4700-4708.
- [52] Han K, Wang Y, Tian Q, et al. Ghostnet: More features from cheap operations[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 1580-1589.
- [53] Li C, Li L, Geng Y, et al. YOLOv6 v3. 0: A Full-Scale Reloading[J]. arXiv preprint arXiv: 2301.05586, 2023.
- [54] Li D, Hu J, Wang C, et al. Involution: Inverting the inherence of convolution for visual recognition[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 12321-12330.
- [55] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.
- [56] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.
- [57] Yang L, Zhang R Y, Li L, et al. Simam: A simple, parameter-free attention module for convolutional neural networks[C]//International conference on machine learning. PMLR, 2021: 11863-11874.
- [58] 陈健松, 蔡艺军. 面向垃圾分类场景的轻量化目标检测方案[J]. 浙江大学学报(工学版), 2024, 58(01): 71-77.
- [59] 郭洲, 黄诗浩, 谢文明等. 融入注意力机制的轻量化可回收垃圾检测方法[J]. 包装工程, 2023, 44(09): 243-253.
- [60] 陈胜选, 王爱民. 基于改进的 YOLOv5 的户外垃圾检测识别[J]. 激光与光电子学进展, 2023, 60(22): 78-85.
- [61] 邢洁洁, 谢定进, 杨然兵等. 基于 YOLOv5s 的农田垃圾轻量化检测方法[J]. 农业工程学报, 2022, 38(19): 153-161.
- [62] Proenca P F, Simoes P. Taco: Trash annotations in context for litter detection[J]. arXiv preprint arXiv: 2003. 06975, 2020.
- [63] Melinte D O, Travediu A M, Dumitriu D N. Deep convolutional neural networks object detector for real-time waste identification[J]. Applied Sciences, 2020, 10(20): 7301.

-
- [64] Wang C Y, Bochkovskiy A, Liao H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[J]. arXiv preprint arXiv: 2207.02696, 2022.
- [65] Ge Z, Liu S, Wang F, et al. YOLOX: Exceeding yolo series in 2021[J]. arXiv preprint arXiv: 2107.08430, 2021.
- [66] Tahir Z, Qureshi A H, Ayaz Y, et al. Potentially guided bidirectionalized RRT* for fast optimal path planning in cluttered environments[J]. Robotics and Autonomous Systems, 2018, 108: 13-27.
- [67] 陈满意, 张桥, 张弓等. 多障碍环境下机械臂避障路径规划[J]. 计算机集成制造系统, 2021, 27(04): 990-998.
- [68] 游达章, 杨智杰, 张业鹏. 基于改进 RRT-Connect 算法的机械臂运动规划[J]. 电子测量技术, 2023, 46(08): 112-119.
- [69] 成浩浩, 杨森, 齐晓慧. 基于改进 RRT 算法的四旋翼无人机航迹规划[J]. 计算机工程与设计, 2018, 39 (12): 3705-3711.
- [70] 章一鸣, 姚文广, 陈海进. 动态环境下自主机器人的双机制切向避障[J]. 浙江大学学报(工学版), 2024, 58(04): 779-789+846.
- [71] 代伟, 李创业, 杨春雨等. 基于低差异序列与快速扩展随机树融合算法的机械臂路径规划[J]. 控制理论与应用, 2022, 39(1): 130-144.
- [72] ZHANG Z. A flexible new technique for camera calibration[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(11): 1330-1334.
- [73] Kingston Z, Kavraki L E. Robowflex: Robot motion planning with MoveIt made easy[J]. arXiv preprint arXiv, 2021, 2103: 12826.
- [74] Pütz S, Wiemann T, Hertzberg J. Tools for Visualizing, Annotating and Storing Triangle Meshes in ROS and RViz[C]. 2019 European Conference on Mobile Robots (ECMR) IEEE, 2019: 1-6.

攻读硕士学位期间主要研究成果

(1) 发表的学术论文:

- [1] 易方旭,王正刚,何宇豪,欧阳嘉泰. 基于改进 YOLOv5s 的生活垃圾检测算法[J].哈尔滨商业大学学报(自然科学版), (已录用, 第一作者)。

(2) 获奖情况:

第十六届全国大学生智能汽车竞赛华为 AI 组三等奖

第十六届“西门子杯”中国智能制造挑战赛信息化网络化全国初赛特等奖

2022 年安徽省机器人大赛-机器人赛道一等奖

第十七届全国大学生智能汽车竞赛华为 AI 组二等奖

第十七届全国大学生智能汽车竞赛室外 ROS 无人车竞速赛(高教组)三等奖

第十七届“西门子杯”中国智能制造挑战赛信息化网络化全国初赛二等奖

第十八届全国大学生智能汽车竞赛讯飞智慧农业组一等奖

致 谢

光阴似箭，转眼间三年的研究生生涯即将结束。在完成论文之际，我衷心地感谢这些年来一直给予我支持、陪伴和关怀的亲友和老师们的。

首先，我要向王正刚教授和王冠凌教授致以诚挚的感谢。在这三年中，无论我在学术上遇到何种困惑，还是遭遇生活中的困难，他们都耐心地为我的答疑解惑，细致地提供帮助和鼓励。他们用渊博的知识和严谨的治学态度，引导我全面掌握课题，从开题报告、实验研究到最终的论文撰写，每一步都少不了他们宝贵的意见与指导。他们是我学术道路上的导师和引路人，他们的优良品质一直激励着我。衷心祝愿两位老师桃李满天下，事业更上一层楼。

同时，我要感谢我的同门何宇豪、欧阳嘉泰和兰天。这三年里，我们在学术道路上携手共进，互相支持和帮助，共同面对各种挑战。此外，我还要特别感谢我的师弟王坤相、王世康、杨思远和江自豪。他们对我的论文提出了许多宝贵的建议，使我的研究更加完善，也让我对课题有了更深刻的理解。

最后，我要感谢我的家人和朋友。在整个研究生阶段，他们给予了无尽的鼓励和支持，使我在面对生活和学习的挑战时，始终充满勇气和信心。

尚德敏学 唯实惟新

