

分类号_____

密 级_____

UDC_____

编 号_____

成都理工大学

全日制专业硕士学位论文

题名和副题名 _____ 基于 ROS 的无人机自主降落技术研究

作 者 姓 名 _____ 丁中涛

指导教师姓名及职称 _____ 方方 教授 黄平 副研究员

申请学位级别 _____ 专业硕士 _____ 类别（领域） _____ 电子与通信工程

论文提交日期 _____ 2020 年 4 月 _____ 论文答辩日期 _____ 2020 年 5 月

学位授予单位和日期 _____ 成 都 理 工 大 学（ 2020 年 6 月）

答辩委员会主席 _____ 杨勇

评阅人 _____

2020 年 06 月

分类号_____

学校代码: 10616

UDC _____

密级_____ 学号: 2017050498

全日制专业硕士学位论文

基于 ROS 的无人机自主降落技术研究

丁中涛

指导教师姓名及职称 _____ 方方 教授 黄平 副研究员

申请学位级别 _____ 专业硕士 _____ 类别(领域) _____ 电子与通信工程

论文提交日期 _____ 2020 年 4 月 _____ 论文答辩日期 _____ 2020 年 5 月

学位授予单位和日期 _____ 成 都 理 工 大 学 (_____ 2020 年 6 月)

答辩委员会主席 _____ 杨勇

评阅人 _____

2020 年 06 月

基于 ROS 的无人机自主降落技术研究

摘要

二十一世纪以来随着传感技术以及材料技术的发展,使得民用领域无人机相关传感器逐渐小型化,这大大促进了民用无人机行业的发展。以消费级无人机领域的中国企业大疆科技为例,其 2018 年全年销售额达 175 亿人民币。与此同时伴随着国内外各开源飞控的不断突破,无人机研发的技术门槛相较于数年前已经大大降低,这促使国内外众多企业与机构开始大量制造或采购无人机并将其投入到各领域中,如商用物流无人机、警用特种无人机、地质勘探测量无人机等早已被大量应用到其相关领域中。随着无人机应用技术的不断发展,当前无人机的发展方向更加倾向于无人机自主控制即无人机不再依赖于人为控制而是依赖于机载计算机,其中无人机自主精准着陆则是无人机自主控制飞行的关键技术。实现多旋翼无人机自主精准降落过程其对未来物流无人机意义重大。本文将引入 ROS (Robot Operating System) 机器人操作系统在四旋翼无人机上实现无人机视觉引导降落过程,并实现 Gazebo 三维仿真软件内仿真飞行以降低开发测试成本。

本设计主要针对物流无人机自主降落问题,旨在使用 opencv 图像处理结合机器人操作系统以及视觉引导算法实现高精度飞行着陆。本文所涉及的工作内容如下:

(1) 通过大量查阅国内外无人机视觉辅助降落相关文献,并比较各类目标识别与定位算法,最终确定了一种改进后的无人机自主降落视觉引导方案。该方案将结合高低两种目标识别方案,有效解决无人机在飞行高度过高(本文设计指标为 20 米)时无法有效识别出靶标问题以及飞行高度过低时摄像机镜头无法完全覆盖靶标的问题。

(2) 重点研究了无人机视觉引导策略,详细介绍了无人机自主降落过程中位姿估算算法以及无人机控制算法,并将本设计所使用的算法与该领域经典算法进行比较。

(3) 研究了在 ROS 系统上进行功能设计的思路,完成在该系统下的程序软件设计,将无人机各项功能节点化并使用相关 ROS 功能包简化开发过程,降低整个系统的复杂度。

(4) 使用 ROS 提供的机器人三维仿真软件 Gazebo,凭借其强大的物理引擎及高质量的图像渲染功能实现整个降落过程的全仿真,降低了调试开发过程中的各项成本。在完成 Gazebo 平台的仿真实验后对无人机进行室外飞行实验,最

终实验测得本设计能够在 20 米高度正确匹配出靶标位置，系统能够稳定可靠的引导无人机实现与靶标平均误差为 30 厘米的精准着陆，本设计能够为物流无人机技术提供一定参考价值。

关键词：自主降落；视觉引导；四旋翼无人机；ROS；目标识别

Research on Autonomous Landing of UAV Based on ROS

Abstract

In recent years, benefited from the development of sensing technology and related material technology, drone-related sensors have been gradually miniaturized, which has greatly promoted the development of the civil drone industry. Take DJ-Innovations, a Chinese company in the field of consumer drones, as an example. Its annual sales in 2018 reached 17.5 billion, accompanied by the continuous development of international and domestic open source flight control. At the same time, along with the continuous development of international and domestic open source flight control, the threshold for the development of drones has been greatly reduced compared to a few years ago. This has led many companies and institutions at home and abroad to start mass manufacturing or purchasing and invest them in various fields, such as commercial logistics drones, police special drones, geological exploration and measurement drones, etc. have also been widely applied to their related fields. With the continuous development of drone technology, the current development direction of drones is more inclined to the autonomous control of drones, that is, drones no longer rely on human control but rely on onboard computers, while drones land on their own, It is the key stage for the drone to control the flight autonomously. The realization of the multi-rotor UAV's independent precision landing process is of great significance to the logistics drone. This paper will introduce the ROS robot operating system to realize the UAV visual guidance landing process on the quadrotor UAV, and realize the Gazebo 3D simulation software simulation flight. Reduce development test costs.

This design is mainly aimed at the problem of autonomous high-precision landing of logistics drones, and aims to use opencv image processing combined with robot operating system and visual guidance algorithm to achieve high-precision flight landing. The work involved in this article is as follows:

(1) Through a large number of domestic and foreign UAV visual aid landing data, and comparing various target recognition and localization algorithms, an improved UAV autonomous landing visual guidance scheme was finally determined. The program will combine the high and low target recognition schemes to effectively solve the problem that the UAV cannot effectively identify the target at a certain height and the camera lens cannot completely cover the target.

(2) Focusing on the visual guidance strategy of UAV, the pose estimation algorithm and UAV control algorithm in the process of UAV autonomous landing are introduced in detail, and the algorithm used in this design is compared with the classical algorithm in this field.

(3) Researched the idea of functional design on ROS system and completed the program software design under this system, node the various functions of the drone and use the relevant ROS function package to simplify the development process and reduce the complexity of the whole system.

(4) Using the robot 3D simulation software Gazebo provided by ROS, with its powerful physics engine and high-quality image rendering function, the whole simulation of the entire landing process is realized, and the various costs in the debugging development process are reduced. After completing the simulation experiment of the Gazebo platform, an outdoor flight experiment was carried out on the drone, and finally the experiment measured that the design can correctly match the target position at a height of 20 meters. The system can guide the UAV stably and reliably to achieve accurate landing with a target error of 30cm. This design can provide a certain reference value for the logistics UAV technology.

Keywords: Autonomous landing; Visual navigation; Quad-rotor Unmanned Helicopter; ROS; Target recognition;

目 录

摘 要.....	I
Abstract.....	III
第 1 章 引言.....	1
1.1 论文研究背景及意义.....	1
1.2 国内外研究现状.....	2
1.2.1 多旋翼无人机应用技术研究现状.....	2
1.2.2 视觉引导降落技术研究现状.....	4
1.2.3 机器人操作系统研究现状.....	6
1.3 研究内容.....	7
1.4 论文结构.....	7
第 2 章 无人机自主着陆系统总体方案设计.....	9
2.1 无人机自主着陆系统方案.....	9
2.1.1 无人机着陆方案设计.....	9
2.1.2 无人机着陆靶标设计.....	10
2.2 无人机自主着陆系统硬件平台搭建.....	11
2.2.1 无人机机载计算机介绍.....	11
2.2.2 无人机自驾仪介绍.....	12
2.2.3 无人机摄像头.....	14
2.2.4 无人机平台及动力系统选型.....	15
2.3 系统软件设计与协议解析.....	16
2.3.1 ROS 节点设计.....	16
2.3.2 MAVlink 协议.....	18
2.4 本章小结.....	19
第 3 章 基于 OpenCV 的合作标识识别.....	20
3.1 相机模型与标定.....	20
3.1.1 相机成像模型与坐标系变换.....	20
3.1.2 下视摄像头标定.....	23
3.2 下视摄像头图像处理.....	25
3.2.1 图像获取与滤波处理.....	25
3.2.2 RGB 色彩空间与 HSI 色彩空间.....	28
3.2.3 HSI 颜色空间下颜色提取算法.....	30
3.2.4 形态学处理.....	32
3.3 对一级靶标轮廓识别与筛选.....	33

3.3.1 Canny 算子边缘提取	33
3.3.2 轮廓筛选与匹配	34
3.4 对二维码的识别	36
3.4.1 功能包的安装与二维码的创建	37
3.4.2 对二级靶标的识别与定位	37
3.5 本章小结	38
第 4 章 基于视觉的无人机位姿估计与着陆引导策略	40
4.1 无人机视觉引导信号的解算	40
4.1.1 视线角 α 为零时横滚与俯仰控制量的解算	41
4.1.2 视线角 α 非零时横滚与俯仰控制量的解算	42
4.2 无人机着陆控制器及其设计	44
4.2.1 位置控制环与速度控制环控制器设计	44
4.2.2 无人机姿态控制器设计	45
4.3 无人机视觉引导着陆软件设计	45
4.4 本章小结	47
第 5 章 无人机视觉引导着陆飞行验证与数据分析	48
5.1 无人机视觉引导着陆过程 Gazebo 软件在环仿真实验	48
5.1.1 SITL 仿真环境搭建	48
5.1.2 靶标误识别率仿真实验	50
5.1.3 着陆误差仿真验证	51
5.2 着陆飞行实验及数据分析	52
5.2.1 着陆飞行实验	52
5.2.2 着陆飞行实验位置控制数据	54
5.2.3 着陆飞行试验速度控制数据	54
5.3 实验飞行着陆精度	56
5.4 本章小结	56
结论与展望	57
结论	57
后续工作的展望	58
致 谢	59
参考文献	60
攻读学位期间取得学术成果	63

第 1 章 引言

1.1 论文研究背景及意义

自航空器问世以来,科研人员便开始尝试对无人驾驶航空器的研究,以便实现航空器在高风险、长航时、低成本等条件下的应用。上世纪限于控制技术、航空材料等技术的限制无人机主要被投入到国防领域中,其中覆盖侦察、通信、靶机等军事用途(Dordrecht, 2015)。直到 21 世纪初,用于无人机控制的各类传感器及通信设备逐渐小型化成本也大幅降低,民用无人机才逐渐得到普及。

近些年来随着各类自驾仪(飞控)的逐渐成熟与开源、动力电池技术的突破以及碳纤维材料的广泛应用,为民用无人机的发展奠定了良好的技术基础。无人机根据其气动布局可分为(John Wiley & Sons, 2013)固定翼无人机、旋翼无人机(MAV)、扑翼无人机。在民用无人机领域考虑到结构复杂度、飞行速度、机动性能等方面通常使用前两种气动布局。旋翼气动布局因其相较于固定翼气动布局具有垂直起降、可悬停、对起降场地条件要求低等特点成为短航程(100 公里以内)应用场景下民/商用无人机应用项目的首选气动布局。现如今除大疆为代表的消费级无人机外,民用无人机也已经被广泛应用到荒野搜救、环境探测、农林植保、电力能源巡检、消防救援、国土测绘、物流派送(侯永宏, 2019)等领域当中。依托旋翼气动布局的特有性能,旋翼无人机应用技术已成为当今世界研究热点领域。

随着民/商用无人机市场的不断发展,传统的低级别无人机自动驾驶已经逐渐不能满足市场的需要,无人机市场对具有高级别自动驾驶能力能够实现起降与飞行阶段完全无人操作的无人机需求在不断提高。现有基于全球定位导航系统(GPS)的无人机全过程自主导航已经不能满足部分应用场景的需求,因为民用 GPS 导航误差大且易受外部因素干扰,使得该方式不能用于引导无人机实现高精度定点降落(张咪, 2018),但是其在无人机自主航线飞行阶段则是首选的引导方案。为实现无人机能够在自主着陆阶段实现精准着陆,在无人机自主降落阶段则必须采用其它导航方式,随着机器视觉技术的不断发展与成熟,机器视觉越来越多的被应用到无人机上。以大疆为例其生产的无人机已广泛配备基于机器视觉的避障功能。在无人机视觉导航方面世界范围内以亚马逊、空客、顺丰等为代表的相关物流领域公司都已推出各自降落阶段采用视觉导航的全自主物流无人机。

目前,具有精确自主着陆能力的无人机已经成为国际上无人机研究的热点方向(索文凯, 2019),其中基于合作标志的无人机视觉引导降落技术以其成本低、精度高、功能负载轻等优点成为民/商用无人机自主着陆技术的热门研究技

术。近些年来亚马逊、顺丰、苏黎世联邦理工、斯坦福大学等各类商业公司及科研机构均开始尝试使用合作标志的方式实现无人机精确自主着陆。研究一款具有精确自主着陆能力的无人机是未来无人机物流、无人机自主动力补给、无人机自主返航等技术的必要前提，其对拓展未来无人机民/商用应用领域具有重要的促进作用。

1.2 国内外研究现状

近些年来，无人机技术的研究已经在飞控算法、航空动力、任务分配等方面取得众多成果（Lee H, 2016）。目前这些技术已经相当成熟，这也使得无人机技术准备水平（TRL）越来越高，因此国内外无人机的实际应用研究项目也越来越多。本节就国内外无人机技术研究现状以及 ROS 机器人系统在无人机自主控制方面应用现状进行阐述。

1.2.1 多旋翼无人机应用技术研究现状

早在第一次世界大战期间人类即以开始尝试将无人机投入到实际应用中，早期其主要被用于军事领域。进入到 21 世纪后无人机应用到民用领域的技术障碍逐渐被突破，民/商无人机应用市场规模逐渐扩大，据研究到 2022 年商用无人机市场预计将达到 221.5 亿美元，从 2015 年至 2022 年的复合增长率为 20.7%（E. Ebeid, 2017）。与此同时多旋翼无人机的研发与应用吸引着越来越多的科研人员投入进来，未来无人机将被大量应用到各行各业。

军事方面以美国军方为代表，其装备的各型无人机数量已经超过 8000 架。虽然美军有“全球鹰”为代表的众多侦打一体无人机用于担负侦察及火力打击等任务，但是受限于起降条件及出动成本，其一线部队依旧选择配备了大量小型无人机用于提升前线部队对战场环境的持续态势感知能力。在美军一再限制采购中国大疆无人机后，其近期就大量采购了美国本土 InstantEye 公司的 MK 系列旋翼无人机用于装备班一级战斗队列，该系列无人机即采用典型的四旋翼气动布局，图 1-1 所示为近期美军使用 InstantEy MK-2 旋翼无人机进行训练。根据 InstantEye 官网显示该型号四旋翼无人机仅重 544 克，但其续航时间却长达 30 分钟。该型号无人机采用了混合通信系统，该系统具有 C2 的加密数字链路和模拟视频链路，飞机不会在机上存储任何数据。因此，如果飞机丢失不会造成任何的数据丢失。



图 1-1 美国海军陆战队准备发射 InstantEye MK-2

在无人机集群飞行方面,最初是由英特尔公司在 2016 年初首次成功进行 100 架规模的四旋翼无人机飞行表演,自 2016 年起无人机集群飞行技术即得到国内外无人机研发团队的重视并纷纷推出各自的无人机集群飞行技术(金伟,2019)。2018 年英特尔公司又以 1218 的数量再次进行无人机集群飞行,成功刷新了“同时放飞数量最多的无人机”的吉尼斯纪录其表演画面如图 1-2 所示。英特尔在自己的集群飞行技术中使用的无人机为一款名为 Shooting Star 的四旋翼无人机,是该公司自主研发的四旋翼无人机,其外形如图 1-3 所示。其使用的集群控制技术可使得无人机群被单台设备操控,其算法可以控制无人机集群编排并规划出最优化飞行路线。国内方面,众多的无人机公司也同时在发展自己的无人机集群控制技术,其中广州亿航白鹭科技有限公司于 2018 年 4 月一举刷新英特尔的纪录完成 1300 架无人机集群表演。



图 1-2 英特尔无人机集群飞行表演



图 1-3 英特尔 Shooting Star 无人机

在移动通信领域,得益于中国 5G 通信技术在国际上的领先地位以及中国无人机产业完善的体系与市场,5G 技术通过各大 5G 产业联盟迅速应用到民/商用无人机上。这使得中国率先开启了 5G 互联网无人机的创新行业应用(李昶,2019),在 2018 年 4 月,中国移动通信运营商中国电信顺利完成了国内 5G 无人机的初次飞行,依托 5G 通信技术优异的技术性能,其成功实现 4K 高清全景视频的低延时回传,能在控制端为操作人员提供延时小于 5ms 的 VR 视角。该验证中使用的无人机即为典型的多旋翼无人机其实验用无人机如图 1-4 所示。



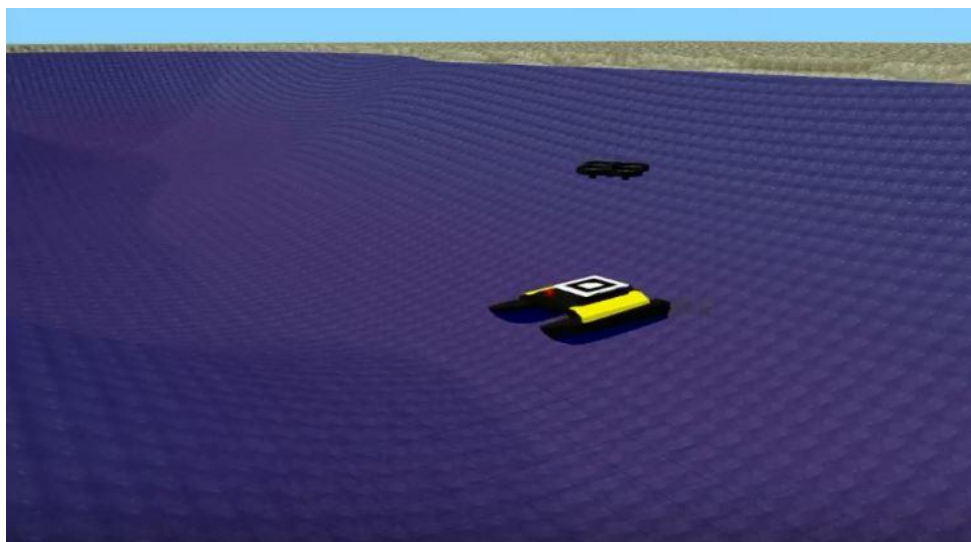
图 1-4 中国电信 5G 无人机

1.2.2 视觉引导降落技术研究现状

对于任何飞行器，其着陆过程都是整个飞行过程中危险性最高的时段，所以飞行器的引导降落一直是无人机技术中最危险的挑战之一（Polvara R, 2018）。传统的导航方式即惯性导航（INS）和 GPS 导航因为其存在误差和定位信号易受干扰等原因的限制不能用作无人机降落阶段的主导航方式（曹美会，2015），因此在无人机自主着陆阶段就需要一种精度高稳定性好的导航方式。近些年来，得益于机器视觉技术的不断发展成熟，国内外众多企业与研究机构都开始了基于视觉的无人机引导降落的尝试与应用。

早在 2003 年 Srikanth Saripalli、Student Member 等人即在 IEEE 机器人与自动化上发表了关于在旋翼机上通过机器视觉引导飞机自主降落的文章，其设计方案中使用合作标识的形式，通过下置单目摄像头采集的图像识别 H 型靶标，其将整个自主降落模式分为两个阶段即搜索模式、跟踪模式。最终，其通过测试得出整个系统的降落误差为 42 厘米。

2018 年 2 月英国普利茅斯大学工程学院自主海洋系统研究小组实现了基于视觉的四旋翼飞机在无人水面航行器甲板上的自主着陆并实现软件仿真。该方案通过下视 RGB 相机识别降落平台并使用扩展卡尔曼滤波器（EKF）估算着陆平台的位姿，其采用的靶标以及仿真环境如图 1-5（a）所示。相同的技术国内华中科技大学则在 2019 年 1 月顺利完成国内首次全自主机艇协同起降的实验飞行，其进一步推动了该技术的实际应用能力，其飞行演示如图 1-5（b）所示。



(a) 普利茅斯大学机艇协同起降仿真实验



(b) 华中科技大学机艇协同起降飞行实验

图 1-5 国内外机艇协同起降飞行实验

不同于其他领域，若将多旋翼无人机成功应用到物流领域自主精准降落则是必须突破的问题。所以国内外各大物流公司在数年前均开始尝试将视觉引导自主着陆技术应用到物流无人机上。以全球最大互联网线上零售公司亚马逊为例，早在 2013 年其就着手开展了 Prime Air 项目，该项目成立的目的是为了打造其未来新一代交付系统，实现使用无人机在 30 分钟以内将包裹安全送达到客户手中。该项目中的无人机均采用图像识别技术用于避障及引导着陆，以其 2019 年 6 月发布的最新第三代 Prime Air 为例，其依托自主研发的立体视觉搭配 AI 算法已经摒弃上代 Prime Air 无人机合作标识的无人机视觉引导着陆技术，能够自动找出无障碍可用着陆区域，图 1-6 为第二代 Prime Air 无人机及其靶标。但是出于技术难度及可靠性等方面考虑，依靠合作标识的着陆技术在国内外物流无人机项目中依旧是主要方式，国内方面京东、亿航、菜鸟、顺丰等公司正在进行的多旋翼

物流无人机方案均采用合作标识的方式,亿航物流无人机及其靶标如图 1-7 所示,可见合作标识的视觉引导降落技术已成为物流无人机自主着陆方案的首选技术。



图 1-6 第二代 Prime Air 及其靶标



图 1-7 亿航物流无人机及其靶标

1.2.3 机器人操作系统研究现状

ROS (机器人操作系统, Robot Operating System)是一套为机器人软件开发而设计的开源操作系统,ROS 可以简单理解为一套开发工具、库以及框架的集合。ROS 系统可以被搭建在 windows、linux 等操作系统上从而为机器人软件开发设计过程提供众多的库以及开发工具。该操作系统起源于斯坦福大学实验室后发展为一款开源操作系统,自 2010 开源以来其已发展成为国际机器人实际软件标准 (Yu X, 2017),包括 NASA、波士顿动力、大疆、百度等国内外知名企业研发的机器人均有采用 ROS 机器人框架,以上使用 ROS 系统的项目中部分项目展示如图 1-8 所示。



(a) 国际空间站上运行 ROS 系统的 R2 机器人



(b) 波士顿动力 ALTAS 机器人

图 1-8 国际上著名 ROS 系统应用项目

由于近些年来无人机自主飞行控制的兴起尤其是引入图像识别、SLAM 等导航避障技术后,传统的无人机应用软件逐渐复杂化耦合度也逐渐增大,这为自主控制无人机的软件开发带来了巨大挑战。机器人操作系统 (ROS) 以其节点对节点的模块化设计、对多种开发语言的支持、架构精简集成度高等优点,近些年来基于 ROS 的无人机控制系统逐渐成为自主飞行无人机的主流控制系统。2015 年无人机巨头大疆发布并在 Github 开源了 Onboard SDK 同时推出了 Onboard SDK

ROS 版本，以帮助开发者研发自己的自主控制无人机。可见 ROS 正逐渐成为自主控制无人机的主流系统。

1.3 研究内容

本文主要研究物流无人机着陆阶段视觉导航问题，实现无人机视觉引导着陆过程。为解决传统导航措施误差大无法实现精准降落的问题，设计中通过机载计算机运行的机器视觉程序实时处理无人机下视摄像头采集的画面并生成无人机导航信息。当无人机被引导程序或手动飞行至标识上空后进行悬停，机载计算机开始运行在搜索模式中并使用目标识别算法查找着陆靶标位置。待图像识别算法确认靶标位置后机载计算机开启降落模式。结合无人机位姿控制与目标识别技术将无人机逐渐引导至靶标上方并最终降落在靶标上。主要研究内容如下：

(1) 研究设计了无人机着陆方案并根据应用需求设计分级式合作标识，最终确定使用颜色、尺寸信息以及形状做为一级靶标的识别依据，使用机器视觉定位中常采用的二维码做为二级靶标为无人机在 2 米以下高度提供视觉引导。

(2) 通过大量阅读国内外相关文献，学习了解了图像处理技术中各类目标识别技术，并比较各类技术的优缺点，最终确定并设计了基于 OpenCV3.0 轮廓提取与匹配技术的目标识别算法用于靶标识别与无人机着陆第一阶段引导识别。当无人机向靶标渐进的过程中识别到靶标中的二维码并满足开启二级引导功能条件后 ROS 系统将开启着陆第二阶段的二维码引导。从而解决无人机渐进至靶标位置上空约 2 米高度摄像头视角无法覆盖整个靶标导致无法识别的问题，以此实现无人机在 20 米高度能够顺利实现靶标识别并能够成功降落至靶标上。

(3) 本设计中使用了机器人操作系统 (ROS) 作为无人机控制系统并运行在机载计算机 (运行 ubuntu16.04 的树莓派 4B) 上，通过 ROS 操作系统提供的库以及开发工具简化开发流程。在设计过程中还研究了 ROS 控制节点的分布与设计。

(4) 设计中为完成无人机在着陆过程中的飞行位姿控制，研究了基于位置控制的无人机着陆控制器的设计，并将控制器应用到着陆过程控制中以实现无人机着陆过程的控制。

(5) 设计中为简化无人机开发测试流程，采用了软件仿真与实际飞行结合的验证方式，在搭建仿真平台过程中研究了 PX4 的 Gazebo 软件在环仿真实现。

1.4 论文结构

本设计从软硬件方面详细阐述了设计方案及算法原理，并最终给出 Gazebo

仿真验证及实际飞行数据，以此给出本设计中无人机的最终着陆效果。本文章节具体内容如下：

第一章介绍了国内外无人机相关技术的发展现状并阐述了视觉引导降落技术在自主控制无人机技术中的重要性以及该技术在物流无人机应用实例，之后给出了无人机自主着陆技术研究的关键内容以及文章结构和内容安排。

第二章主要论述了本设计的软硬件设计方案，硬件方面给出了设计使用的机载计算机、摄像头、无人机平台等硬件的具体选型与参数，软件方面主要介绍了本设计中无人机控制系统（ROS）的节点设计以及通信协议的解析。

第三章详细阐述了本设计中所涉及的图像处理技术与靶标识别算法，其可分为使用 Opencv 对一级靶标的识别与匹配以及使用 ROS 功能包 ar_track_alvar 实现对二维码的识别与定位。

第四章完成了无人机不同姿态下视觉引导信号的解析与无人机着陆器的设计。

第五章完成无人机图像识别算法与着陆过程的 Gazebo 软件在环仿真，并进行实际飞行测试，通过仿真与实际飞行数据分析，确定本设计成果的最终性能。

第2章 无人机自主着陆系统总体方案设计

本章将就本设计总体设计方案以及软硬件搭建进行论述与说明。首先详细说明了设计中无人机的着陆设计方案，其次搭建了系统硬件平台包括电机、电调、机架、PX4 自驾仪、单目摄像头以及机载计算机，最后通过给出 ROS 的节点关系图介绍无人机着陆系统中的软件设计并介绍了无人机通信协议。

2.1 无人机自主着陆系统方案

2.1.1 无人机着陆方案设计

由于本文是针对全自主物流无人机自主着陆问题而设计，其整个飞行过程是在无人工干预下完成，因此该系统的图像处理与无人机位姿控制都是在机载计算机上完成。无人机在航线飞行阶段通过机载计算机通过 MAVROS 连接 ROS 与 Mavlink 协议（Carvalho J P, 2017）发送航线信息引导无人机按一定航线飞行至目标区域上空，由于航线飞行阶段依赖机载 GPS 模块，所以无人机抵达预定空域后与目标的水平相对位置将取决于此时的 GPS 误差。若实现精确自主着陆则需要视觉引导，本设计中着陆过程可分为以下阶段：

第一阶段：无人机由 GPS 导航引导至着陆区域上空悬停在 20 米处调整机头方向指向正北并进入第二阶段（靶标搜索阶段）；

第二阶段：机载计算机通过无人机下视摄像头获取着陆区影像，当识别算法识别出位于着陆区域的着陆靶标后，无人机保持高度不变水平向靶标移动以消除水平误差后，将进入第三阶段（视觉引导阶段）；

第三阶段：机载计算机通过视觉引导算法计算出无人机与靶标的相对位置参数经着陆控制器解析后做为引导数据发送给自驾仪，用于引导无人机接近靶标。当无人机通过靶标引导至 2 米高度后，无人机开始检测靶标中的二维码当成功匹配靶标中的二维码后，无人机开始进行最后阶段的二维码视觉引导着陆。

第四阶段：当机载计算机判断无人机高度低于 30 厘米后无人机将结束视觉引导进行垂直着陆，最终降落至靶标上。

以上无人机着陆流程主体逻辑框图如图 2-1 所示。

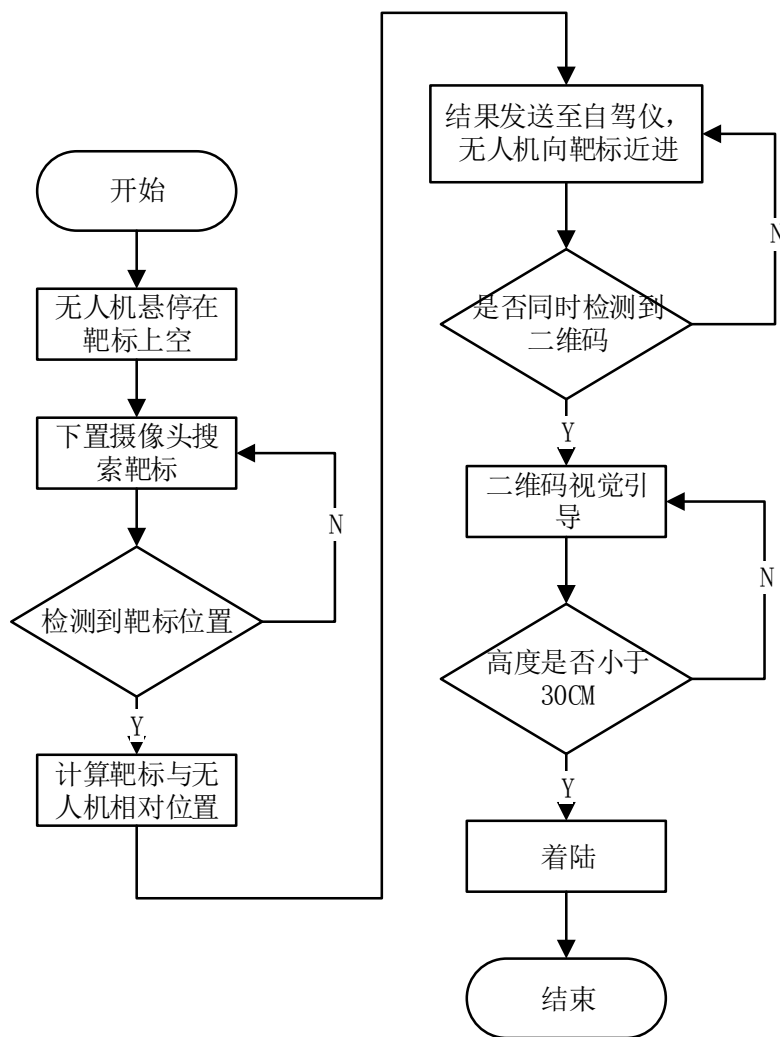


图 2-1 无人机自主着陆流程主体逻辑框图

2.1.2 无人机着陆靶标设计

合作标识即无人机着陆靶标（以下统称为靶标），其被用于为无人机提供着陆视觉引导，合作标识用于机器人视觉引导技术在国内现已经被大量应用到机器人视觉引导中。但在无人机视觉引导着陆技术中，早期研究多是采用单二维码或其他单级靶标方式，该方式配合高效的识别算法能够为无人机提供精度小于 10 厘米的视觉引导方案，但是受摄像头分辨率、摄像头视角以及无人机飞行高度的影响该方案中无人机飞行高度多被限制在 10 米以内，针对该问题本设计提出了一种二级靶标方案其设计如图 2-2(a)所示，其设计尺寸信息如图 2-2（b）所示。

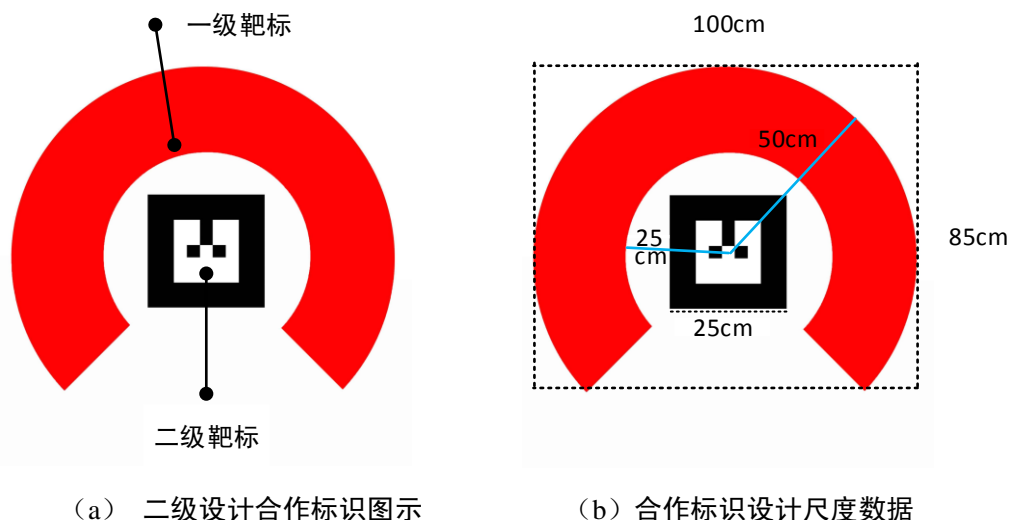


图 2-2 二级设计无人机靶标设计图示

如图 2-2 中所示, 该设计方案中一级靶标为一个四分之三的圆环, 该圆环外环半径为 50 厘米内环半径为 25 厘米, 一级靶标被用于引导无人机在高于 2 米的飞行高度实现对靶标的识别与定位, 其颜色使用红色是因为红色便于将靶标与周围环境区分开来, 同时也便于后续图像处理算法对其进行颜色识别以顺利提取一级靶标轮廓。设计中一级靶标最小外接矩形是一个长为 100 厘米宽为 85 厘米的矩形, 根据尺寸设计可知一级靶标与最小外接矩形的面积之比为 $(9\pi)/64:(2+\sqrt{2})/4$, 其与最小外接圆的面积比为 9:16, 该图像信息将被用于图像处理算法中对目标轮廓的过滤, 以此提高无人机对着陆靶标的匹配正确率。

图中 2 级靶标采用的是一块边长为 25 厘米的正方形二维码, 其作用是当无人机高度降落到 2 米时无人机进入二维码引导着陆阶段配合超声波模块能够为无人机提供高精度的视觉引导, 其二维码中所包含信息可根据需求更改图中所示二维码内信息即 ID 为数字 0。

2.2 无人机自主着陆系统硬件平台搭建

本节就本设计中所涉及的硬件型号及参数进行介绍。跟据设计要求, 本设计中主要涉及嵌入式计算机、无人机自驾仪、下视高清摄像头以及一套无人机动力系统。出于设计性能需求以及成本的考虑最终确定以上设备选型及参数。

2.2.1 无人机机载计算机介绍

本设计中所使用的机载计算机是一块拥有 4G 运行内存的树莓派 4B 单板计算机其外形如图 2-3 所示。树梅派 4B 采用了一枚 ARM V8 架构的 1.5GHz 四核 64 位 Cortex-A72 处理器, 相较于上一代树莓派 3B+在 CPU 处理速度上提高了三

倍。该单片电脑 GPU 采用了 500MHz 的 VideoCor VI 配合 4GB 的 DDR4 内存其在使用 Opencv 处理图像方面完全符合性能要求。其拥有丰富的引脚以及 USB, 40 针引脚中提供了 UART、I2C、SPI 等接口, 同时提供了 2 个 USB2.0 与 2 个 USB3.0 共计 4 个 USB 口。在运行系统方面该单板电脑可以运行 Debian、Ubuntu 以及 Raspbian 等 Linux 发行版本, 在本设计中使用的是 Ubuntu 19.10 系统并关闭了系统的图形界面以提高其运行效率。



图 2-3 树莓派 4B 单板计算机

2.2.2 无人机自驾仪介绍

自动驾驶仪(Autopilot), 其是使用传统控制理论与基于比例积分微分(PID)技术的控制器件(Liu M, 2015), 其可以通过自身搭载的 GPS、气压计、多轴传感器等传感设备采集到的数据经控制算法处理实现多路输出用于控制无人机、无人车、无人艇等自动驾驶设备。本设计中无人机所使用的自动驾驶仪为权盛出品的运行 PX4 2.4.8 开源固件的 Pixhawk, 其是一款成本相对较低且性能优异的飞行控制器(Feng L, 2016), 设计中所采用的自驾仪端口解析如图 2-4 所示。Pixhawk 是由苏黎世联邦理工学院的一个学生项目发展而来, 现如今其已成为无人机飞行控制硬件和自动驾驶软件的常用标准。本设计使用的权盛 Pixhawk 2.4.8 是国内厂商在著名开源控制硬件供应商 3DR 推出的新一代开源飞行控制系统 Pixhawk 2.4.6 的基础上对其供电方案等进行修改后生产的版本并降低了生产成本, 其与原版 2.4.6 在功能上并无差别可运行相同固件。

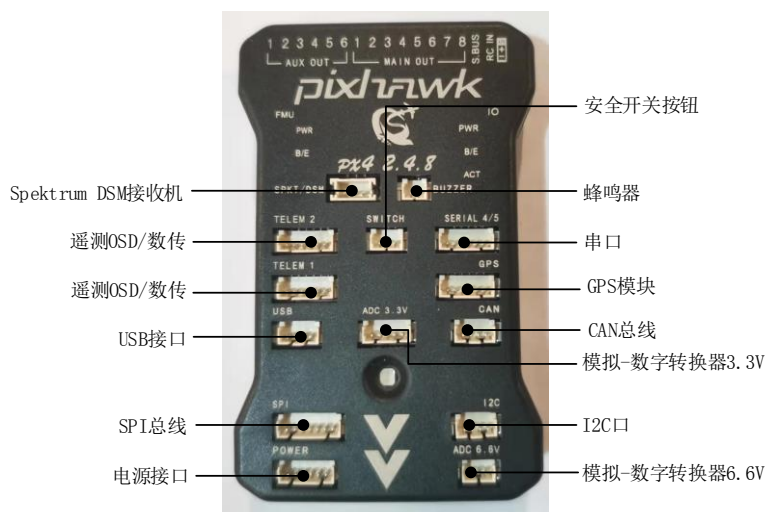


图 2-4 Pixhawk 自动驾驶仪及其端口解析

本设计中所使用的 Pixhawk 自驾仪采用了一个 32 位 STM32F427 芯片作为主处理器并运行 Nuttx RTOS 实时操作系统用于处理各类姿态解析与控制算法，同时自驾仪上还搭载了一枚 STM32F103 协处理器用于控制输出以及故障安全保障。在传感器上，该自驾仪搭载了 ST Micro L3GD20 3 轴 16 位陀螺仪、ST Micro LSM303D 2 轴 14 位加速度计/磁力计、MPU 6000 3 轴加速度计/陀螺仪以及 MEAS MSS5611 气压计。得益于其优秀的软硬件设计该自驾仪在数据采集与处理方面表现出了优异的高效性，其通过多种速率的嵌入式系统可以保证各传感器数据读取中断延时小于 $4\mu\text{s}$ ，而系统内任务间消息传递延时小于 $25\mu\text{s}$ （杨小川，2018）。在接口方面 Pixhawk 继承了上代 PX4IO 的硬件接口，拥有 5 个 UART 其中有一个支持高功率输出可以用于机载计算机供电，其还支持 SPI、I2C、CAN 三种通信方式，设计中各类外设与自驾仪连接方式如图 2-5 所示。

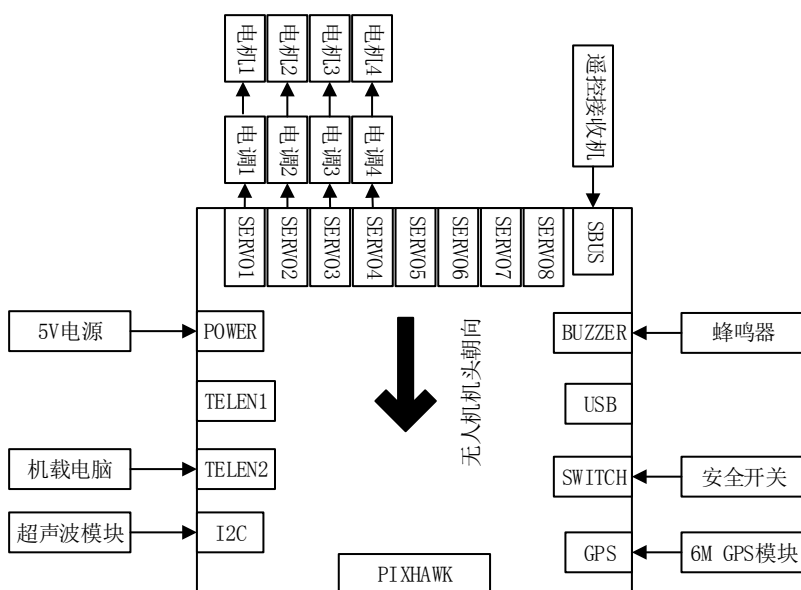


图 2-5 硬件外设与飞控连接示意图

软件设计方面，PX4 软件原生版本可以被划分为四层（Meier L，2015），如图 2-6 所示该软件架构由下至上分别为驱动层、NuttX 系统层、通信层、应用层。第一层驱动层是包含各类硬件设备驱动代码的驱动程序，这里包括陀螺仪、气压计、GPS、超声波传感器等传感设备的驱动，该层将设备抽象为一个设备节点并提供了一个系统接口；第二层 NuttX 系统层即主处理器中运行的 NuttX 实时操作系统，其负责处理各类飞行控制算法；第三层是为对象请求代理层，它能够有效地处理进程间通信，为程序间的通信提供保障；架构中最后一层是应用程序层，该层是由众多与功能相关的应用软件构成，如飞行控制功能、姿态估计功能等以及实现对多旋翼、固定翼、无人驾驶车辆等功能的支持。

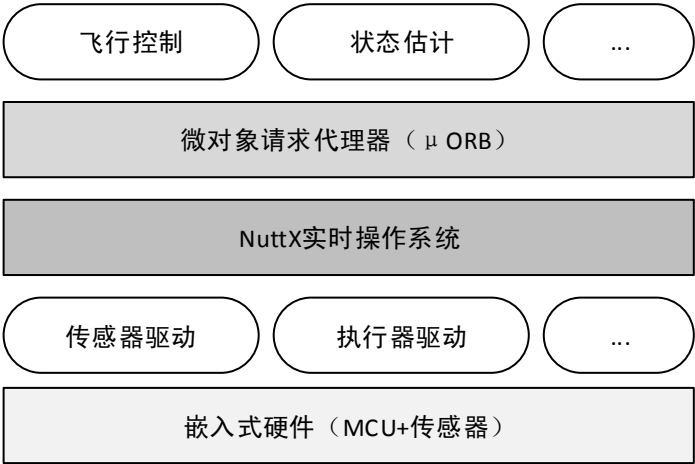


图 2-6 PX4 软件体系结构

2.2.3 无人机摄像头

设计中使用的是锐尔威视生产的 500 万像素的无畸变高清摄像头，该摄像头可在 Windows、Linux、Android 等操作系统上进行免安装驱动使用，即可在本设计中的机载计算机上直接 USB 免驱运行。其使用一块 1/2.5 英寸镁光 MI5100 CMOS 感光片，虽然早期 COMS 传感器存在噪声大的缺点，但随着 CMOS 技术的不断成熟与发展，其图像质量已得到显著提升（何梓溢，2014）。近些年，CMOS 传感器凭着高性价比与开发便捷的优势逐渐受到了青睐。本节所选用的高清相机的详细性能指标如表 2-1 所示。

表 2-1 下视摄像头具体性能参数

参数项	性能参数
输出图片格式	MJPEG /YUV2(YUYV)
感光片	镁光 MI5100(1/2.5 英寸)
感光区域尺寸	5.7mm*4.28mm

续上表		
支持的分辨率与帧率	2048×1536:15 fps 1280×720:30 fps	1280×1024:15 fps 640×480:30 fps
接口类型	USB2.0 High Speed	
镜头规格	尺寸 1/2.5 焦距 3.6mm 光圈 F2.4 视角 90 度	
电压	DC5V	
系统支持	Win xp/Vista/Win7/Win8/Linux with UVC(above linux-2.6.26)/ MAC-OS X 10.4.8 or later/Android4.0 or above with UVC	

其实物图如图 2-7 所示。



图 2-7 锐尔威视 500 万像素 USB 高清摄像头

根据表 2-1 中所示，在 640×480 的显示分辨率下该摄像头可以以每秒 30 帧的速率采集图片，该速率可以有效确保设计中无人机在跟踪靶标过程中图像的实时性，有效避免由图像采集速率过慢引起的时延问题。

2.2.4 无人机平台及动力系统选型

本设计中所使用的无人机平台机架为大疆推出的 F450 型机架，其是目前市场上最为常用的小型四旋翼无人机机架，其机架净重为 282g。其动力系统采用了四个新西达 30A 无刷电机电调以及四个大疆生产的 920kv 无刷自锁电机，无人机实物如图 2-8 所示。



图 2-8 设计用无人机实物图

无人机硬件选型性能参数见表 2-2。

表 2-2 设计用无人机相关配置参数

信息	参数说明
电调型号	新西达 30A 电调
无刷电机型号	2212/920kv 无刷电机
无人机机架型	大疆 F450
无人机总重量	1.3kg
电机轴距	450mm
电池型号	无界 2200mah 3s 25c 电池

2.3 系统软件设计与协议解析

本设计在软件方面使用了运行在 linux 上的 ROS kinetic 为控制系统，使用 C/C++语言进行图像处理、姿态控制以及 ROS 节点的编写。并使用小型无人机通信协议 MAVLink 用于机载计算机与自驾仪进行信息交互。以下为系统软件平台的详细介绍。

2.3.1 ROS 节点设计

ROS 机器人操作系统，其可为基于 linux 系统的机器人开发提供一个模块化软件平台，该操作系统是在分布式处理的基础上结合 TCP/IP 协议，其能够很好的降低软件模块间的耦合关系即实现模块间点对点的松耦合连接，其同时还提供

了硬件抽象、设备控制、消息管理等标准的操作系统服务(刘乃军, 2018)。ROS 在为机器人开发提供模块化软件平台时, 主要引入了以下四项概念(Fernandez E, 2015):

(1) 节点(Node): 在 ROS 系统中一个节点即表示一个任务程序任一节点都是 linux 系统下的任务进程, 一套完整的 ROS 机器人项目中都是由多个不同任务节点组合而成。当 ROS 系统正常运行时, 用户可以使用 `rqt_graph` 指令快速将当前系统中端对端的通信绘制生成节点关系图, 其中节点关系图中椭圆表示节点, 箭头连线表示消息。

(2) 消息(Message): 消息在 ROS 系统中是被用于不同节点间的点对点通信, 其是以一种发布/订阅(Publish/Subscribe)的方式来完成并管理节点间的通信任务。在 ROS 系统中节点间交互的消息有着严格的数据结构, 可被用于传输标准数据类型(整型、浮点型、布尔型等)的数据以及嵌套结构和数组型数据。

(3) 话题(Topic): 话题在 ROS 系统中是被用于管理节点间的消息, 在 ROS 系统中任意节点均可将内部的数据选择性的发送到现有任意话题(Topic)上(此节点称为发布者 Talker)同时节点可从现有任意话题获取该话题上的一项或数项消息(称为订阅者 Listener)。未向话题订阅消息的节点对该话题上的消息不可见。

(4) 服务(Service): ROS 中服务概念的提出是为解决话题的发布/订阅机制不适用于双向同步传输情况的问题, 为此 ROS 以客户端/服务器(Client/Server)模型建立了系统的同步传输机制即服务, 与 Web 服务器相类似服务中也包函了两个部分的通信数据类型: 一个用于请求, 另一个用于应答。

本设计中节点设计如图 2-9 所示, 其为 gazebo 仿真情况下系统 ROS 节点关系, 由图可见设计中主要建立了四个节点分别是 `mavros`、`gazebo`、`commander_node_3` 与 `offboard_node`。其中 `gazebo` 节点用来将下置摄像头采集到的数据通过话题发送至节点 `commander_node_3`, `commander_node_3` 节点用于靶标搜索与跟踪并通过图像算法生成无人机控制信息, `offboard_node` 节点通过订阅 `commander_node_3` 发布的位置控制信息通过 `mavling` 协议发送给自驾仪, 从而让无人机产生机动。

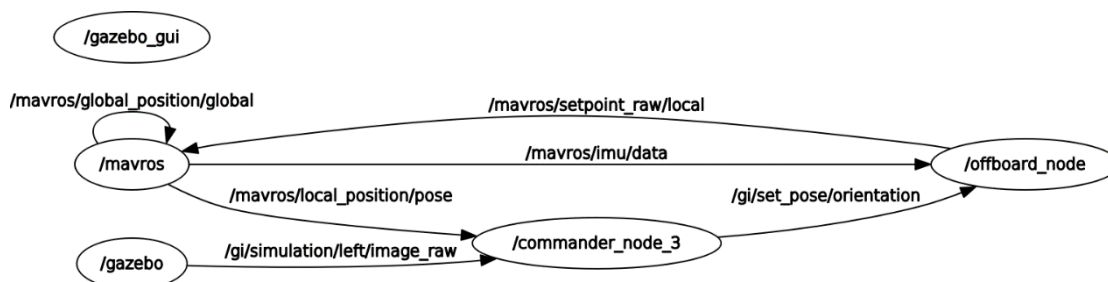


图 2-9 本设计中 ROS 节点关系图

本设计中具体发布订阅关系如图 2-10 所示，图中矩形即为该 ROS 系统内存在的话题，连线即为发布/订阅中信息的传输关系。

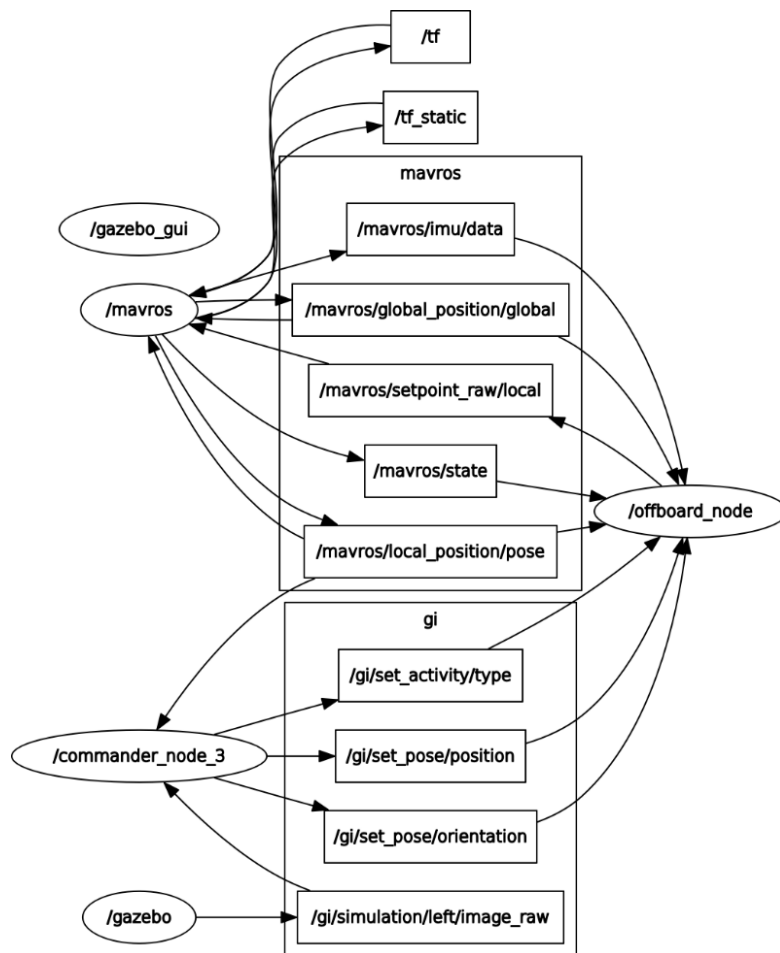


图 2-10 本设计中所有节点与话题关系图

2.3.2 MAVlink 协议

微型飞行器协议（MAVLink）是一种开源的点对点网络协议，常用于进行遥测以及指挥和控制许多小型无人机，该协议现已被广泛应用于如 APM、PIXhawk、Parrot AR.Drone 等自动驾驶仪上。2009 年 MAVLink 协议被洛伦兹·迈耶（Lorenz Meier）等人首次应用到实际应用中并遵循 LGPL 开源协议。该协议支持通过无线通道进行双向通信，可实现无人机与无人机地面控制站 GCS（Ground Control Station）间的无线双向通信，以此实现 GCS 对无人机的控制以及无人机数据回传（Atoev S, 2017）。在无人机自主飞行模式下，用户也需要使用 MAVLink 协议将控制程序输出的控制指令发送到自驾仪上。通过 MAVLink 协议发送的数据帧格式如表 2-3 所示，由表可知其是一帧以 FE 为帧头以 16 位 CRC 校验码为帧尾的数据帧（Domin K, 2017），每一帧带有发送包序从 0 到 255 重复计数用于监测传输过程中的丢包情况。

表 2-3 MAVLink 协议数据帧格式

序号	名称	描述
0	Package start sign	数据包开始标志, 为 FE
1	Payload length	表示有效载荷的长度
2	Package sequence	发送序列, 用于检测丢包
3	System ID	发送系统的 ID, 区分不同的飞行器
4	Component ID	发送组件的 ID, 区分不同的组件
5	Message ID	定义 Data 的类型, 决定解码方式
6~(n + 6)	Data	数据内容, 最大允许 255 字节
(n + 7) (n + 8)	Checksum	16 位 CRC 校验码

2.4 本章小结

本章主要对设计中无人机自主着陆系统使用的软硬件平台以及整体着陆方案进行了详细的描述。在着陆方案方面详细介绍了本设计所采用的地面合作标识以及本设计的整体着陆逻辑。在硬件方面则分别对机载计算机树莓派 4B、自动驾驶仪 Pixhawk、下置摄像头、以及无人机机型与动力系统进行了详细的性能参数分析, 并给出相关硬件连接关系。在软件方面, 本章在给出了软件开发设计逻辑的同时还给出了设计中 ROS 节点的设计与消息传递关系, 最后介绍了机载计算机与自驾仪通信协议 MAVLink 协议。本章对设计中的软硬件设计给出了详细的介绍与说明很好的反映出了本设计的软硬件平台设计思路。

第3章 基于 OpenCV 的合作标识识别

本章将详细阐述本设计中图像处理与识别匹配算法。主要的工作内容有：完成下视单目摄像机内参标定、基于轮廓识别的无人机着陆靶标识别算法设计与实现以及基于轮廓的目标识别算法实现。其中对一级着陆靶标的识别涉及到了 RGB 到 HSI 色彩空间转换、颜色识别、阈值分割、轮廓检测与匹配等图像处理算法，对二级靶标即二维码的识别则是使用 ROS 提供的 `ar_track_alvar` 功能包实现摄像头在 2 米高度下对二维码的识别。本章以下章节将对各图像识别算法设计进行详细阐述。

3.1 相机模型与标定

本节着重对本设计中所使用的单目摄像头成像原理以及摄像头内参标定进行阐述。由于本设计中对无人机着陆靶标位置识别的图像处理算法中位置解析部分是建立在该单目摄像头成像模型原理之上，所以对本设计中所涉及的相机成像模型的分析是本章及后一章节的前提性理论。而本节中的相机标定则是对该相机成像模型进行模型参数确定的操作（卢晓冬，2020），因此本节相机模型的构建与相机内外参数的标定是本设计方案中必不可少的操作。

3.1.1 相机成像模型与坐标系变换

本设计中无人机所搭载的 CMOS 摄像头，其工作原理可以被简化为一个针孔模型。通过该简化后的模型可以很好的解释光线摄入到视网膜或图像传感器成像过程的几何模型。

在针孔模型中，现实物体上的任一点发射出的光线只有一条特定光线能够通过针孔进入到摄像头内部并成功照射到摄像头的感应芯片上，最终经算法处理形成数字图像上的一个像素。在此过程中数字图像所在的平面与小孔所在面的两个面之间的距离由该摄像头固定属性即焦距决定（以 f 表示），此时该距离即为 f 。如图 3-1 所示，图中 Z 为物体至摄像头的距离， X 为该像物体的真实长度， x 为该物体在数字图像中成像大小。由三角相似可以得出各参数之间的关系为：

$$-x = f \cdot \frac{X}{Z} \quad (3-1)$$

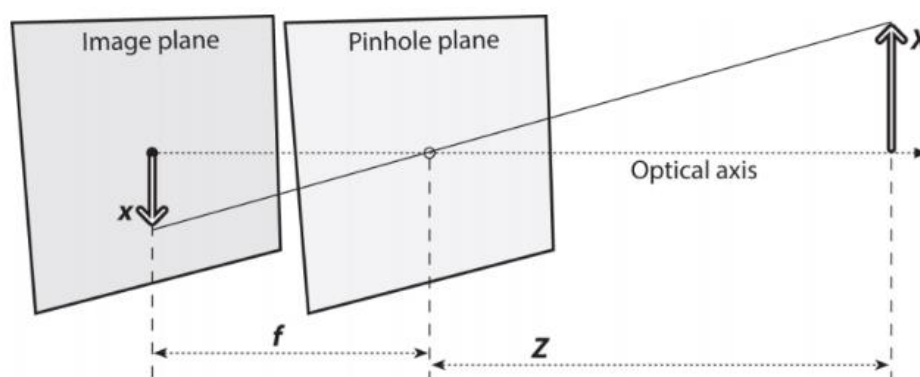


图 3-1 针孔相机成像模型

图 3-1 中为理想状态下针孔相机的成像模型，在其基础上将成像平面置于被摄物体和相机之间并分别建立各类坐标系（江士雄，2016）如图 3-2 所示，图中以摄像机光学中心 O_c 为原点建立了一个三维相机坐标系 $O_c X_c Y_c Z_c$ 其单位为 m ，以成像区域左上角 O 为原点建立坐标系 ouv 该坐标系即为图像像素坐标系其单位为像素，同理以成像区域中心点 O_i 为原点建立了图像物理坐标系 $O_i X_i Y_i$ ，点 O_c 到点 O_i 的距离即为焦距 f 。以 O_w 为原点建立了一个三维世界坐标系 $O_w X_w Y_w Z_w$ 其单位为 m 。根据以上坐标系建立过程可知两个二维坐标系即 ouv 与 $O_i X_i Y_i$ 位于同一个平面且两坐标轴分别平行，而 $O_c X_c Y_c Z_c$ 与 $O_w X_w Y_w Z_w$ 可经刚性变换（即旋转平移）得到。该图中点 M 为现实世界中一个真实点其在世界坐标系中坐标为 (X_w, Y_w, Z_w) 在相机坐标系中的坐标为 (X_c, Y_c, Z_c) ，该点在数字图像中的投影点 m 其 $O_i X_i Y_i$ 坐标表示为 (x_m, y_m) 。

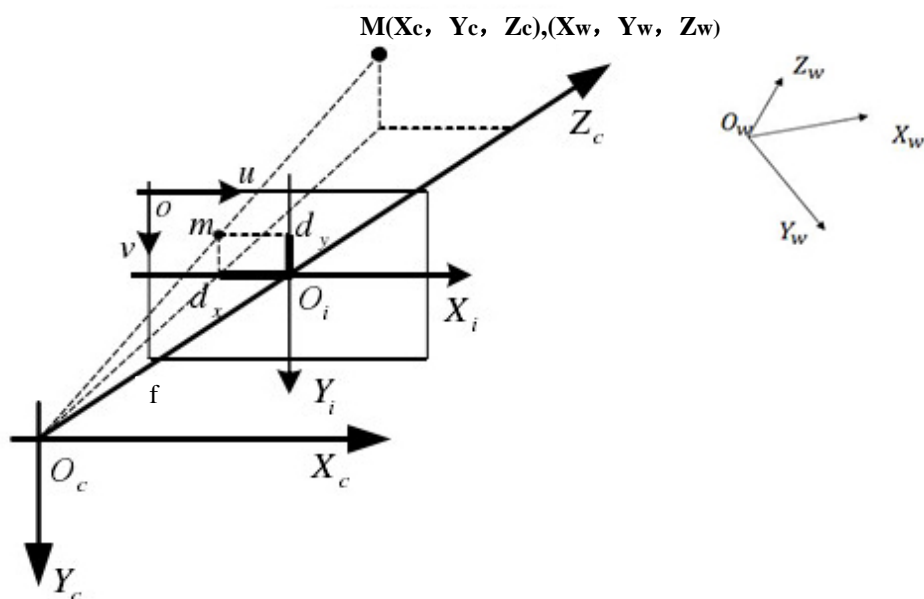


图 3-2 针孔相机成像模型中各坐标系之间关系示意图

通过图 3-2 中在摄像机模型中以不同点建立的各类相关坐标系，结合相机内

外参矩阵（通过相机标定获得）可以建立以上四个坐标系间的转换关系，以便推导出依据被成像物体在图像中的位置信息求解其与摄像头位置误差的问题，以上四类坐标系转换关系如下：

(1)世界坐标系到相机坐标系变换：根据各坐标系的建立方式可知世界坐标系与相机坐标系存在着刚性变换关系，令相机坐标系分别绕世界坐标系的 X、Y、Z 分别轴旋转 φ 度、 ω 度、 θ 度后进行平移可以得到世界坐标系，则当绕 Z 轴旋转未进行平移时满足以下关系。

$$\begin{cases} x = x' \cos \theta - y' \sin \theta \\ y = x' \sin \theta + y' \cos \theta \\ z = z' \end{cases} \quad (3-2)$$

将式 3-2 进行矩阵表示后可得到式 3-3。

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R_1 \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \quad (3-3)$$

公式中 R_1 表示绕 Z 轴旋转的旋转矩阵，同理可以得到绕 X、Y 轴的旋转矩阵 R_2 、 R_3 ，则在该变换过程中的旋转矩阵 $R=R_1R_2R_3$ ，当刚性变换过程中的平移矩阵为 T（其为一个 3×1 矩阵）时则该刚性变换可以表示为：

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + T \quad (3-4)$$

将式 3-4 转化为四阶矩阵可得：

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (3-5)$$

式中 $\begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{bmatrix}$ 即表示摄像头的外参矩阵。

(2)相机坐标系到图像物理坐标系变换：如图 3-2 所示相机坐标系到图像物理坐标系是一个从三维到二维的变换过程，图中根据三角形相似原理可以得出以下结果。

$$\begin{cases} x_m = f \frac{x_c}{z_c} \\ y_m = f \frac{y_c}{z_c} \end{cases} \quad (3-6)$$

将式 3-6 表示为齐次坐标表达式为：

$$Z_c \begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (3-7)$$

式 3-6、3-7 中 f 即相机焦距。

(3)图像物理坐标系到图像像素坐标系变换：由于该变换过程中是在同一二维平面内进行且两个坐标系 X、Y 轴分别平行，所以该操作中仅需要进行坐标系的

简单位移便可完成, 则该转换过程如以下公式:

$$\begin{cases} u = \frac{x_m}{h_x} + u_0 \\ v = \frac{y_m}{h_y} + v_0 \end{cases} \quad (3-8)$$

公式中 (u_0, v_0) 表示点 O_i 在 ouv 坐标系中的位置, (u, v) 表示点 m 的在 ouv 坐标系下的位置, h_x 、 h_y 分别表示成像芯片上单像素的长宽尺寸。将 3-8 表示为矩阵形式则有:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{h_x} & 0 & u_0 \\ 0 & \frac{1}{h_y} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} \quad (3-9)$$

经以上分析点 M 的世界坐标系到图像像素坐标系的转换关系如式 3-10 所示。

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{h_x} & 0 & u_0 \\ 0 & \frac{1}{h_y} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (3-10)$$

令 $\frac{f}{h_x}$ 、 $\frac{f}{h_y}$ 分别用 f_x 、 f_y 表示则式 3-10 可表示为:

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (3-11)$$

式 3-11 中 $\begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ 即为相机内参矩阵, $\begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{bmatrix}$ 为相机外参矩阵。

3.1.2 下视摄像头标定

由于摄像头是精密的光学仪器, 但受摄像头内外部的因素影响, 所有的摄像头在成像过程中都存在着畸变问题, 为了解决摄像头的畸变失真问题减小成像误差, 因此在该摄像头投入使用前还需对其进行标定 (Tianqu Zhao and Hong Jiang, 2016)。在解决摄像头畸变问题的同时相机标定也是实现三维场景重建的重要步骤 (刘艳, 2014), 在摄像头使用前借助摄像头标定程序对其进行标定以获取该摄像头的各类固定性能参数, 是完成单目视觉定位项目前期的必要步骤。

设计中为标定无人机摄像头采用了 ROS 中标定功能包 `camera_calibration`, 该功能包其标定原理是使用张正友标定法对相机进行标定。该标定法是通过特定标定靶即类似于国际象棋棋盘的标定靶其实物如图 3-3 所示, 该标定方法是通过标定靶上的特征点间的已知位置关系与图像中对应点的像素坐标系下的位置来推导出摄像机的内部参数 (王谭, 2019)。

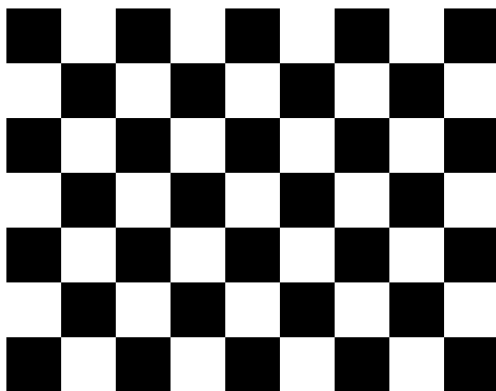


图 3-3 棋盘格图案的标定靶

该功能包在使用期间为提高参数的测量精度要求尽可能多的涵盖各成像角度，因此操作中要不停移动标定靶让标定靶尽量出现在镜头的各个区域以涵盖不同情况下的摄像头失真特性，从而提高程序对摄像头内参的计算精度，标定过程中程序对标定靶上特征点的识别以及操作界面如图 3-4 所示。

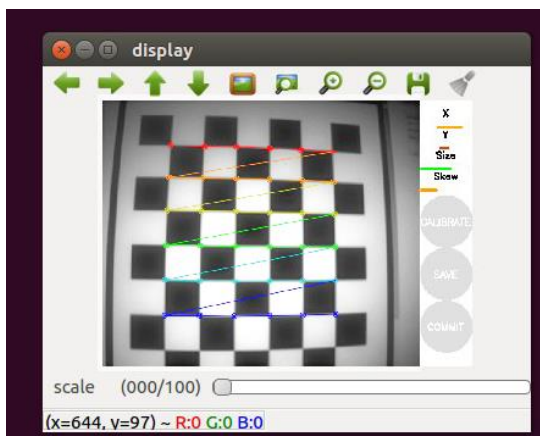


图 3-4 标定过程中程序对特征点的识别

在标定过程中自动保存的不同位置及角度下标定靶的图像以及生成的摄像头参数文件如图 3-5 所示，其中 ost.yaml 文件为生成的标定结果文件。

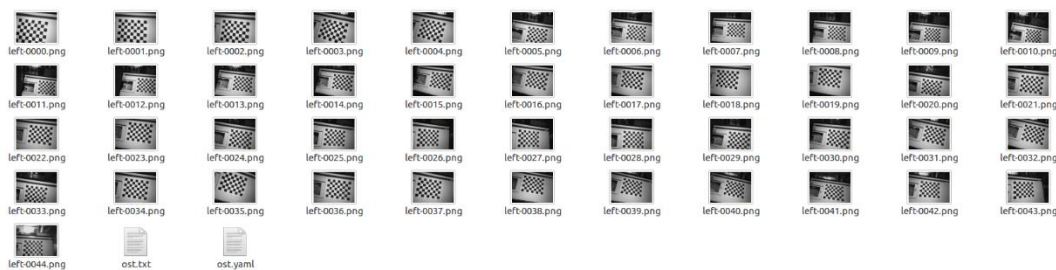


图 3-5 标定过程中生成的文件

根据标定数据最终确定相机内参矩阵为：

$$K = \begin{bmatrix} 791.2377 & 0 & 321.4457 \\ 0 & 794.1536 & 322.9765 \\ 0 & 0 & 1 \end{bmatrix}$$

3.2 下视摄像头图像处理

本设计中为解决低空（低于 2 米）情况下无人机下视摄像头视野不能完全覆盖靶标的问题，对靶标采取了分级设计即一级靶标用于高于 2 米时对靶标的定位与引导，当高度低于 2 米时启动对二级靶标即靶标中二维码的识别。本节及 3.3 节主要介绍了设计中所使用的对一级靶标的图像识别算法，图像处理算法使用 C/C++ 进行编写，其处理流程如图 3-6 所示。

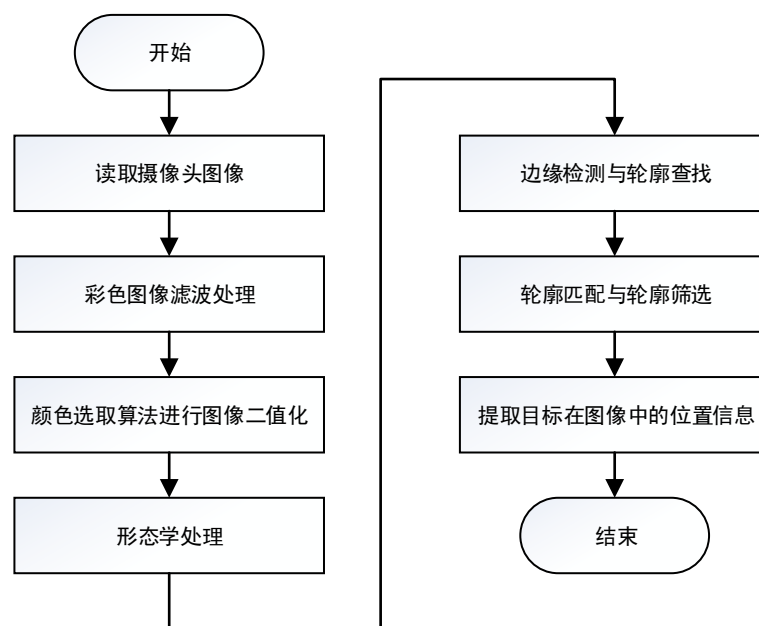


图 3-6 对一级靶标图像识别与处理流程

3.2.1 图像获取与滤波处理

本方案中使用 OpenCV 中的 VideoCapture 类对 USB 摄像头进行调用并读取视频流，图像处理程序首先调用 VideoCapture::open 打开 USB 摄像头后再通过 VideoCapture::read 捕获并解码 USB 摄像头视频帧。当捕获失败时其会返回 false，当捕获成功时将返回一个 cv::Mat 类型的原图像数组。程序中对 USB 摄像头采集到的视频流的帧处理将都在 cv::Mat 类型的数组上进行。此时获取的图像内由于摄像头外部或内部的各种原因都会使得图像内包含大量噪声（韦星，2017），为了提取到感兴趣的视觉特征并过滤掉背景干扰以及图片噪声，在进行目标识别前还需要对图像进行滤波处理以更好的进行后续的图像处理。

在数字图像处理中噪声是决定图像质量的关键，在实际应用中较差的光照条件、高温、电子电路噪声都会导致摄像头在拍摄过程引入噪声，因此降噪是当前图像处理技术中的关键步骤（Jain A, 2015）。OpenCV 中提供了简单模糊和方框滤波器、中值滤波器、高斯滤波器、双边滤波器等 5 种不同的平滑滤波操作，其

中值滤波、高斯滤波和双边滤波是 OpenCV 图像滤波中最常用的平滑滤波方式。本设计对下视摄像头采集到的大小为 640*480 的 RGB 彩色图像进行了高斯滤波与双边滤波,从而降低图片噪声并凸显出目标边缘以更好完成后续轮廓提取工作。

(1) 高斯滤波器

高斯滤波器是机器视觉中被使用最为广泛的滤波器之一,该滤波器是一个低通滤波器。通过对摄像头采集的原图片进行高斯滤波能够有效过滤掉图像内的高频噪声。高斯滤波器的滤波原理是对输入图像中的所有像素点依次进行与邻近像素的加权平均。在使用高斯滤波器的过程中算法采用设定的模板(或称卷积核、掩膜)依次对像素点进行处理,即采用该像素点灰度值及模板大小范围内的相邻像素点灰度值的加权平均值替换该像素点的灰度值。其中应用的高斯模板即高斯核中,当令 x 、 y 表示像素在高斯模板中的坐标,原点为高斯模板中心时,当高斯参数 σ_x 、 σ_y 满足条件 $\sigma_x=\sigma_y=\sigma>0$ 时可得到一维、二维高斯函数如下:

一维高斯函数公式:

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (3-12)$$

二维高斯函数公式:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} = G(x)*G(y) \quad (3-13)$$

由公式 3-12 与 3-13 可知二维高斯函数可由两个一维高斯函数进行乘积得到,在实际数字图像滤波中通常使用二维高斯函数,本设计中采用 x 、 y 方向半宽为 2 的 5*5 的高斯模板对图像进行高斯滤波其模板坐标与高斯核(权重矩阵)如图 3-7 所示。

(-2,2)	(-1,2)	(0,2)	(1,2)	(2,2)		1	4	7	4	1
(-2,1)	(-1,1)	(0,1)	(1,1)	(2,1)		4	16	26	16	4
(-2,0)	(-1,0)	(0,0)	(1,0)	(2,0)		7	26	41	26	7
(-2,-1)	(-1,-1)	(0,-1)	(1,-1)	(2,-1)		4	16	26	16	4
(-2,-2)	(-1,-2)	(0,-2)	(1,-2)	(2,-2)		1	4	7	4	1

(a)

1/273

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

(b)

图 3-7 设计中使用 5*5 高斯模板坐标 (a) 与 5*5 高斯核 (b) 展示

由图 3-7 (b) 所示在设计中高斯滤波器对下视摄像头采集到的图像中一点及其半宽为 2 的周边像素值依据高斯核中的权重进行模糊处理,对采集的图像实际滤波效果如图 3-8 所示。

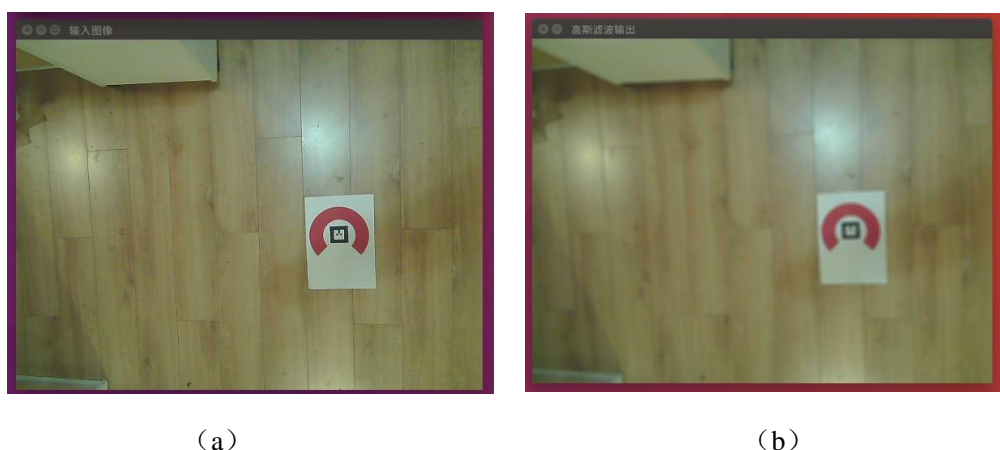


图 3-8 对原图像高斯滤波处理效果 (a) 为输入原图像 (b) 为滤波效果

由图 3-8 可见高斯滤波器对图像进行了明显的模糊处理, 此过程虽然能有效滤除图像中高频噪声但是其对目标的边缘也进行了模糊处理, 若不进一步处理则会影响到对轮廓的提取与对目标的有效识别。

(2) 双边滤波器

高斯滤波器虽然能够很好的降低像素在空间上的变化, 并将领域内像素进行了有效关联, 但是随机噪声却与空间不相关即随机噪声在图像中变化非常大。高斯滤波虽然很好的降低了图像中的噪声同时将小信号值保留, 但在使用该滤波器在优化噪声的过程中却造成目标细节的丢失尤其是对目标的边缘产生影响, 这对本设计中的轮廓提取造成了影响。因此, 在完成高斯平滑后还需对图像进一步处理以达到凸显边缘信息的目的。

双边滤波是一种非线性滤波器, 其与高斯滤波器(高斯平滑)类似都是对图像中一个像素点及其相邻像素的像素值进行加权平均, 其权重可分为高斯平滑、高斯权重两部分, 但是其不再是基于空间距离而是根据色彩强度差进行计算, 在多通道(色彩)图像上强度差由各分量的加权累加代替。双边滤波器的处理方式是在高斯滤波器处理过程中提升相似度高的像素的权值, 从而达到凸显边缘并提高对比度的目的。

但是由于双边滤波相较于其他滤波方式其处理速度较慢, 双边滤波器的大小决定了处理速度, 考虑到本设计中对图像的实时性要求在使用 OpenCV 中的 `cv::bilateralFilter(src, dst, d, sigmaColor, sigmaSpace, borderType)` 函数进行双边滤波时滤波器大小 `d` 设置为对视频处理中常用的 5, 用以在不影响处理速度的情况下实现对高斯滤波器输出的图像进行轮廓改善以及提升对比度。经该滤波操作前后对比如下图所示。



图 3-9 高斯滤波后进行 $d=5$ 的双边滤波效果 (a) 为输入图像 (b) 为双边滤波输出效果

由图 3-9 可见双边滤波器对高斯滤波器输出图像进行了进一步的滤波处理同时对边缘进行了一定优化,并一定程度提升图片对比度。由于考虑双边滤波对处理效率的影响此处使用 $d=5$ 的双边滤波器,其处理效果较不明显,但对后续轮廓提取有一定优化作用。

3.2.2 RGB 色彩空间与 HSI 色彩空间

色彩空间是指使用三种或者更多的特有特征来指定一种颜色的方法或模型,Opencv 中提供了数十种类型的颜色模型,但实际应用中通常只涉及到 RGB、灰度图、CIE XYZ、HSV、HLS、HSI 等多种模型 (García G B, 2015),在本设计中仅涉及到 RGB 及 HSI 两种颜色模型。

RGB 颜色模型是一种加性模型,在该模型中一幅数字图像是由红、绿、蓝三种图像平面进行叠加构成,即在该模型下任何一种颜色都可以通过改变这三个分量值进行叠加得到。虽然在现有图像中多使用该类模型进行成像,但是可以通过 RGB 的线性或非线性变换得到其他颜色空间 (Kolkur S, 2017),其模型可以简单理解为一个如图 3-10 的三维坐标系,该坐标系空间内任意一点都表示一个颜色,坐标系中点 $(0,0,0)$ 代表黑色,点 $(1,1,1)$ 代表白色。在实际数字图像中通常将此坐标系中 0 到 1 用 0-255 进行量化表示,即将 0 到 1 区间等分为 256 份其中 0 表示低亮度 255 表示最高亮度即在数字图像中 $(0,0,0)$ 表示黑色, $(255,255,255)$ 表示白色。其中最大值 255 ($2^8 - 1$) 表示单个色彩通道是由 8 位二进制数字进行表示。所以在实际数字图像中 RGB 色彩空间可以表示出 $2^8 \cdot 2^8 \cdot 2^8 = 16777216$ 种颜色。但在图像处理中采用该模型的颜色表示很容易受到光照强度、阴影等外部因素影响。

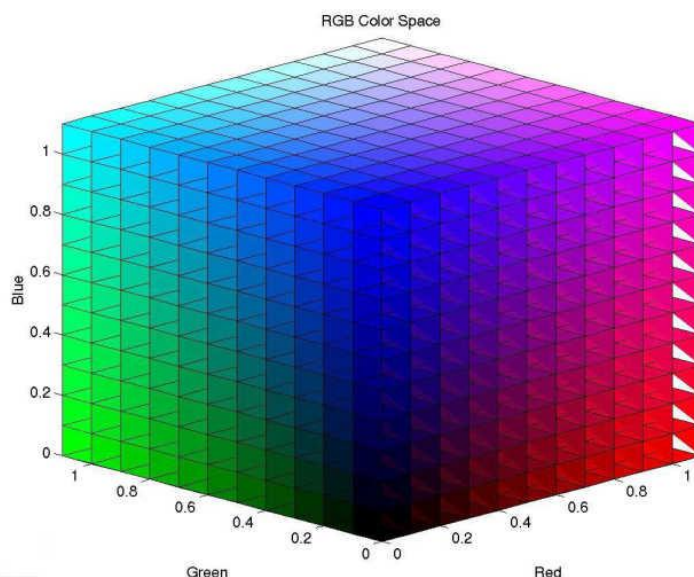


图 3-10 RGB 颜色空间示意图

HSI 模型则是根据色调 (H, Hue)、饱和度 (S, Saturation)、亮度 (I, Intensity) 三个分量来定义颜色的色彩模型, 其颜色空间示意图如图 3-11 所示, 在 HSI 颜色空间中 H 表示对应的颜色其取值范围为 $[0, 360]$, 0 度表示红色, 60 度是黄色, 120 表示绿色, 240 表示蓝色, 300 为洋红色。S 表示饱和度即颜色深浅其取值范围为 $[0, 1]$, I 表示颜色亮度其取值范围为 $[0, 1]$ 其中 0 表示黑色, 1 表示白色。它是一种近似均匀的色彩空间, HSI 颜色模型非常接近于人类色彩感知的仿真模型 (Fatma M, 2014)。对比 RGB 色彩模型中使用三种色彩叠加表示颜色, HSI 颜色模型将亮度分量与彩色图像中颜色承载信息(色调与饱和度)分离, 因为色调、饱和度和强度彼此分离, 所以可以单独处理颜色空间。该颜色空间下能很好的避免不同光照强度对色彩的影响。

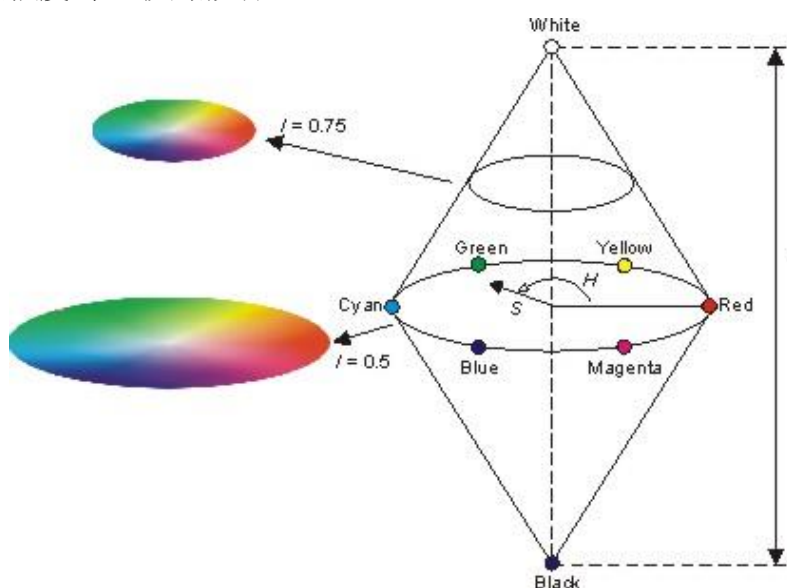


图 3-11 HSI 颜色空间示意图

设计中图像处理程序从下视摄像头视频流中读取的是分辨率为 640*480 的 RGB 格式的数字图像,依据 RGB 与 HSI 两种颜色模型的色彩表示原理可知,对颜色空间内特定颜色的选取在 RGB、HSI 两个色彩空间下均可完成,但是使用 R、G、B 三分量范围选取进行叠加确定颜色范围有着选取易受环境光照强度干扰、颜色选取不直观且三个分量相互关联(汪成龙, 2018)的缺点,HSI 仅需要对 H(色调)进行选取即可确定颜色范围,再通过 S、I 的调节确定颜色深浅、明暗范围,且 HSI 颜色空间相较于 RGB 颜色空间其对光照变化适应性更强(侯宾, 2014)。

3.2.3 HSI 颜色空间下颜色提取算法

根据 3.2.2 节所述,在图像处理过程中若对图像中目标颜色进行提取还需进行图像 RGB 颜色空间到图像 HSI 颜色空间的转换,并在 HSI 空间下针对目标颜色范围使用算法对相应颜色区域处理得到二值化的图像,以便后续进行轮廓匹配,这样能够很好的避免图像中其它不相关物体对目标识别结果产生的干扰。以下将详细说明此部分转换与提取过程。

设计中通过 OpenCV 的 VideoCapture::read 读取的图像是一幅 RGB 图像,但图像中 R、G、B 三个分量的值是将 RGB 颜色空间中的[0,1]用[0,255]进行了量化表示,因此在转换前需要将 R、G、B 三个分量转换到实际 RGB 颜色空间表示下,以 R' 、 G' 、 B' 分别表示图像中一点的 RGB 值对应的[0,1]区间的取值(Saravanan G, 2016),其归一化转换如下:

$$R' = R/255 \quad (3-14)$$

$$G' = G/255 \quad (3-15)$$

$$B' = B/255 \quad (3-16)$$

H(色调)计算公式:

$$H = \begin{cases} \theta & B' \leq G' \\ 360 - \theta & B' > G' \end{cases} \quad (3-17)$$

其中 θ 表示如公式 3-18 所示。

$$\theta = \cos^{-1} \left[\frac{(R' - G')(R' - B')}{\sqrt{(R' - G')^2 + (R' - B')(G' - B')}} \right] \quad (3-18)$$

S(饱和度)计算公式:

$$S = 1 - \frac{3}{(R' + G' + B')} [\min(R', G', B')] \quad (3-19)$$

I(亮度)计算公式:

$$I = \frac{R' + G' + B'}{3} \quad (3-20)$$

根据以上公式实现将 RGB 颜色空间下的一点转换到 HSI 颜色空间下的功能

函数接口设置如下:

```
void RGB_to_HSI(double red, double green, double blue, double& hue, double& saturation, double& intensity)
```

以上部分算法仅完成 RGB 图像中一点到 HSI 颜色空间的转换, 为完成下视摄像头采集的 RGB 彩色图像完全转换成 HSI 图像并对目标颜色进行筛选, 还需要根据图像宽度与高度逐一对图像像素进行 RGB 到 HSI 的转换, 并在 HSI 颜色空间下对红色范围进行筛选。其中在红色范围内的像素将其像素值置为 255 即为白色, 对非红色区域进行置 0 即为白色, 设计算法具体流程图如图 3-12 所示。

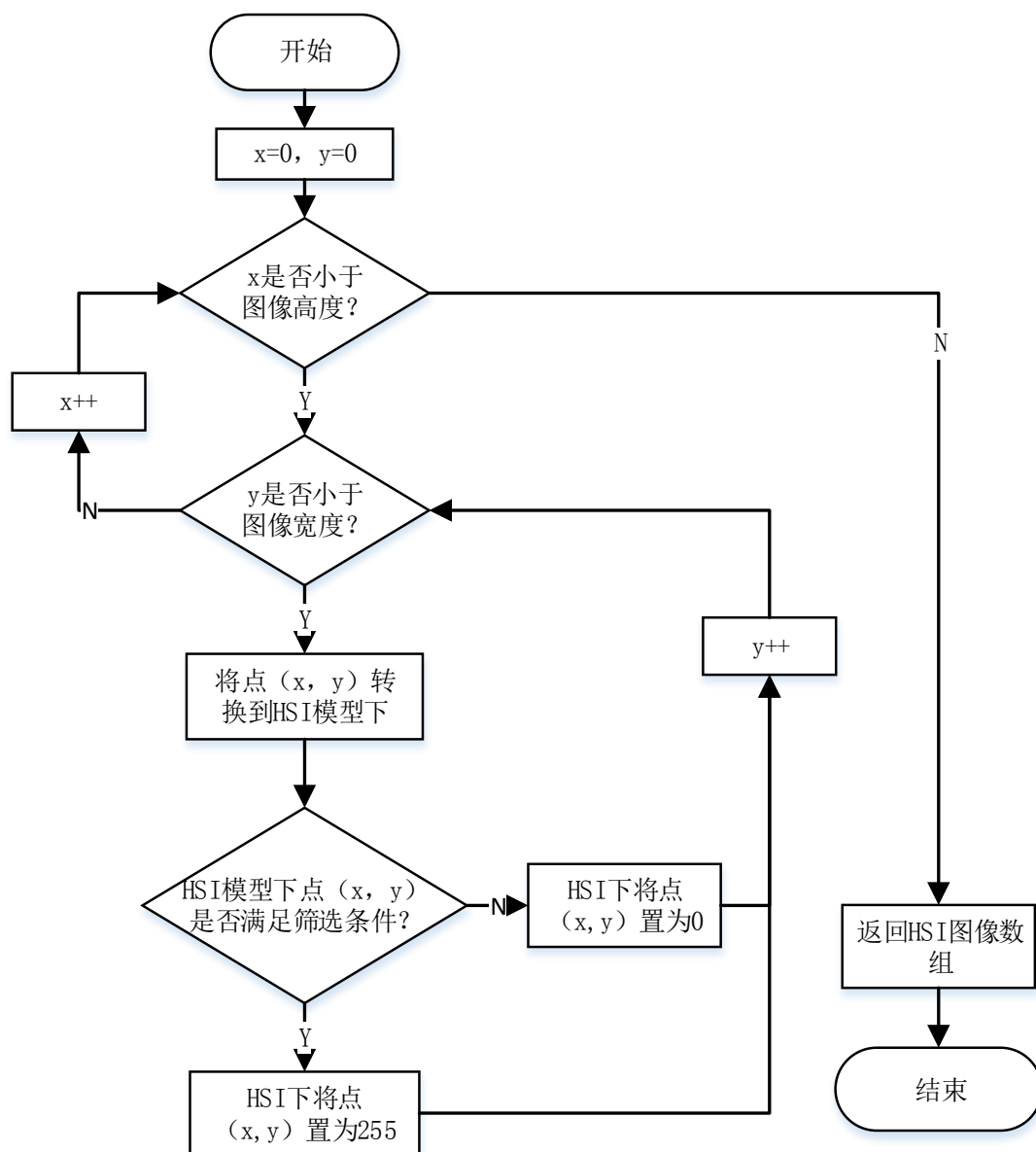
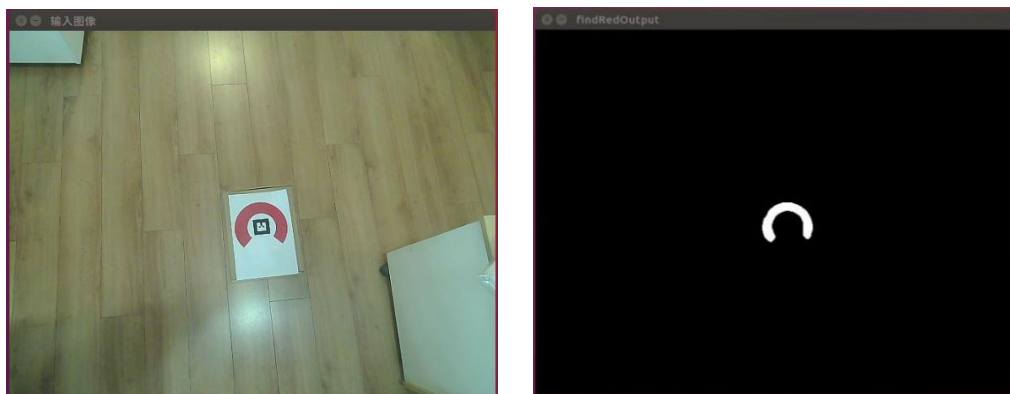
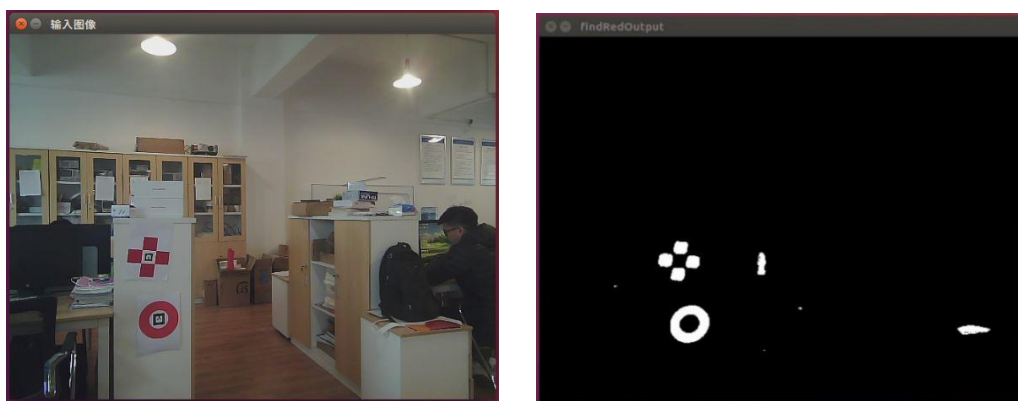


图 3-12 图像 RGB 到 HIS 空间转换并二值化算法流程

使用图 3-12 中算法流程对图像进行二值化处理其中 HSI 颜色空间下对图像红色区域识别的筛选条件为该点像素 HSI 颜色空间下 H 取值范围为 $[312, 360] \cup [0, 8]$, S 取值范围为 $[0.17, 1]$, I 取值范围为 $[0.18, 1]$ 。其识别效果如图 3-13 所示。



(a) 低背景干扰情况下算法对红色区域的提取效果



(b) 复杂背景干扰情况下算法对红色区域的提取效果

图 3-13 不同背景下算法对彩色图像红色区域的处理

3.2.4 形态学处理

由图 3-13 所示颜色提取算法对经滤波处理的彩色图像进行二值化过程中，其处理结果中轮廓边缘有毛刺存在且图像存在孤立白点，为更好的提取红色区域的轮廓还需要对经颜色提取算法处理后的二值化图像进行形态学处理以及去噪。对图像进行形态学处理是图像处理中常用的处理方式，其主要应用于二值图像，图像的形态学处理能够很好的消除图像中部分噪声的影响，起到增强图像的作用（孟建军，2019）。

形态学运算中两个基本的操作是膨胀与腐蚀。其中膨胀运算是将背景到边界添加像素，而腐蚀运算则是清除像素。通过膨胀运算与腐蚀运算相组合，可以组合成其他形态学运算如开运算与闭运算。开运算即对图像先进行腐蚀运算后进行膨胀运算，其常用于消除图像中小对象保留更大的对象，而闭操作则是对图像先进行膨胀运算再进行腐蚀运算，其常用于清除孔洞（Raid A M, 2014）。因此为改善轮廓边缘平滑度以及去除图像中较小的白色点，本设计在继 3.2.3 小节后对图像进行开操作，其效果如图 3-14 所示。



图 3-14 对图像进行形态学开操作效果图

图 3-14 是在图 3-13 (b) 背景下进行形态学处理得到的效果图。根据对比可见开操作有效滤除了轮廓周边的毛刺并去除了孤立的白色点。

3.3 对一级靶标轮廓识别与筛选

在上一节中图像处理算法已对下视摄像头采集到的彩色图像进行了一系列的滤波处理、颜色空间变换、图像指定颜色区域提取、形态学处理等操作，处理后得到保留图像中红色区域的二值化图像，其中白色部分表示图像中红色物体的轮廓。本节就对轮廓的筛选与识别介绍设计中无人机对下视摄像头中合作标识的识别。

3.3.1 Canny 算子边缘提取

在图像处理过程中图像的边缘信息是图像中的重要信息，其可以用于描述目标轮廓或者目标区域内的目标位置 (Rong W, 2014)。为了准确且有效地对无人机合作标识进行识别，设计中图像处理程序需要对图 3-14 中的边缘进行识别。OpenCV 中常用的边缘检测算法有 Canny 算子、Prewitt 算子、laplacian 算子、Sobel 算子等，但是相较于 Canny 算子，其他算法虽然较为简单但是它们去噪能力较差对边缘的提取不够完整 (李宝芸, 2019)，本设计中即采用 Canny 算法对二值图像中白色区域进行边缘检测用于提取白色区域的轮廓，该算法能够在提供较好的检测精度的情况下同时提供较强的降噪能力。

Canny 算子的基本思想是在任意方向上使用二维高斯函数的一阶导数作为噪声滤波器，滤波器通过卷积运算作用在图像上，然后可以找到滤波图像的局部梯度最大值，从而可以确定图像边缘 (Deng C X, 2013)，算法的具体步骤如下。

(1) 高斯滤波

在进行 Canny 边缘检测中，首先会对图像进行高斯平滑处理，结合公式 3-3

可知若输入图像用 $f(x,y)$ 表示，其输出图像 $H(x,y)$ 是输入图像与二维高斯函数的卷积，其输出图像可表示为：

$$H(x,y) = f(x,y) * G(x,y) \quad (3-21)$$

(2) 计算图像梯度的幅值和方向

令 $P_x(i,j)$ 与 $P_y(i,j)$ 分别为 x 、 y 方向的一阶偏导数则图像梯度幅值可表示为：

$$M(i,j) = \sqrt{P_x(i,j)^2 + P_y(i,j)^2} \quad (3-22)$$

图像梯度方向可以表示为：

$$\theta(i,j) = \arctan [P_y(i,j)/P_x(i,j)] \quad (3-23)$$

(3) 梯度幅值行非极大值抑制

对梯度幅值应用非极大值抑制，在梯度方向上具有最大值的像素将保留为边缘像素，并删除其他像素。梯度最大值通常出现在边缘的中心，随着梯度方向上距离的增加，梯度值将减小。

(4) 对梯度选取两个阈值 TH 和 TL ，两者的关系为：

$$TL = 0.5 \times TH \quad (3-24)$$

取出非极大值抑制后图像中的最大梯度幅值，将小于低阈值的点置 0，将大于高阈值的点（即边缘点）置 1，即可提取出完整的边缘。

经 Canny 算子处理后的下视摄像头采集的图片效果如图 3-15 所示，可见其可以有效提取出二值化图像中所有白色区域的轮廓。经 Canny 算子处理后的图像将作为 OpenCV 库中的 `findContours()` 函数的输入图像，该函数将找出图中所有轮廓并保存轮廓信息到一个类型为 `vector<Vec4i>` 的容器 `hierarchy` 中，其是一个包含 4 个 `Int` 类型变量的向量，`hierarchy` 处理过程中存储轮廓内容如图 3-16 所示。

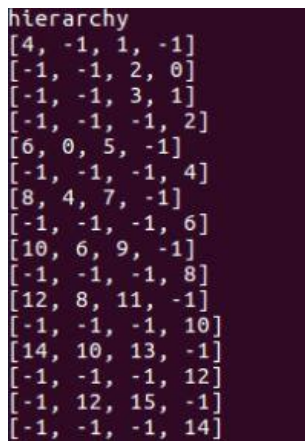


图 3-15 对二值化图像进行 Canny 边缘检测效果 图 3-16 `vector<Vec4i>` `hierarchy` 内容

3.3.2 轮廓筛选与匹配

经 3.3.1 节中的边缘提取算法处理后，如图 3-15 所示机载计算机实时提取到

下视摄像头采集到的图像中所有红色区域部分的轮廓。考虑到实际飞行环境中存在的视觉干扰问题,为了进一步提高识别算法对合作标识的识别准确率,在进行轮廓匹配前还需要结合一级靶标的面积信息对提取到的轮廓进行进一步筛选以过滤掉图像中的干扰轮廓。

首先对于图像中面积过小的轮廓以及轮廓最小外接圆形面积过小的轮廓进行过滤,为确定对轮廓进行面积过滤的阈值,需要计算出无人机在 20 米高度(即无人机开始识别靶标的极限高度)下靶标轮廓的面积。根据 2.1.2 节中靶标约的设计数据可知其一级靶标识是一个四分之三圆环,其最小外接圆形直径为长为 1 米,圆环面积与最小外接圆形面积之比为 9:16。结合公式 3-1 以及摄像头感光片尺寸与有效像素可知,在 20 米飞行高度一级靶标成像面积约为 1800 个像素,其最小外接圆形成像面积约为 3200 个像素。考虑到轮廓处理过程中存在的失真,故对轮廓面积最小阈值设定为 1400 最小外接圆形的面积最小阈值设定为 2800,并设置圆环面积与最小外接圆形面积比例阈值范围为[0.45, 0.65],以此可以简单过滤出干扰轮廓提高程序对靶标的识别正确率。在程序中轮廓面积通过函数 `contourArea()` 获取,其函数原型为 `double contourArea(InputArray contour, bool oriented=false)`,参数 `contour` 为输入轮廓点集,最小外接圆形的半径以及圆心坐标则是通过 OpenCV `minEnclosingCircle()` 函数获取。

经以上一系列阈值过滤后,仍然会有大量干扰轮廓的存在,为准确查找到目标轮廓,还需要对轮廓进行识别。本设计方案中使用 Hu 不变矩对轮廓进行匹配,Hu 矩作为几何矩的一种延伸,具有一种旋转、平移、缩放不变性(徐晓苏,2015),相较于其它轮廓识别技术其更适用于本应用环境,并且其具有运算速度快的优点。

在二值图像 $f(x,y)$ 中,其在区域 ζ 上的 $p+q$ 阶几何矩(标准矩)定义为:

$$m_{pq} = \iint_{\zeta} x^p y^q f(x,y) dx dy \quad (p, q = 0, 1, 2 \dots) \quad (3-25)$$

$p+q$ 阶中心矩定义为:

$$\mu_{pq} = \iint_{\zeta} (x - \bar{x})^p (y - \bar{y})^q f(x,y) dx dy \quad (p, q = 0, 1, 2 \dots) \quad (3-26)$$

其中式 3-26 中的 \bar{x} 与 \bar{y} 表示为图像的重心。

$$\bar{x} = m_{10}/m_{00} \quad (3-27)$$

$$\bar{y} = m_{01}/m_{00} \quad (3-28)$$

对于离散的数字图像则使用求和代替积分,则其几何矩与中心矩表示如下:

$$m_{pq} = \sum_{y=1}^N \sum_{x=1}^M x^p y^q f(x,y) \quad (p, q = 0, 1, 2 \dots) \quad (3-29)$$

$$\mu_{pq} = \sum_{y=1}^N \sum_{x=1}^M (x - \bar{x})^p (y - \bar{y})^q f(x,y) \quad (p, q = 0, 1, 2 \dots) \quad (3-30)$$

其中 N 和 M 分别为图像的高度和宽度,归一化后的中心矩表示为:

$$\eta_{pq} = \mu_{pq}/\mu_{00}^{\rho} \quad (3-31)$$

$$\rho = (p+q)/2+1 \quad (3-32)$$

利用二阶和三阶归一化中心矩可以构造 7 个不变矩 $M_1 \sim M_7$ 如下：

$$\begin{aligned}
 M_1 &= \eta_{20} + \eta_{02}; \\
 M_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2; \\
 M_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2; \\
 M_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2; \\
 M_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})((\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2) + \\
 &\quad (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \cdot (3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2); \\
 M_6 &= (\eta_{20} - \eta_{02})((\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2) + \\
 &\quad 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}); \\
 M_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})((\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2) - \\
 &\quad (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03}) \cdot (3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2);
 \end{aligned} \tag{3-33}$$

通过这 7 个不变矩可以构成图像的一组特征量，Hu.M.K 在 1962 年证明了他们具有旋转、缩放和平移不变性。在本设计中程序中使用了 OpenCV 中的 matchShapes 函数来实现轮廓 Hu 矩的计算、轮廓与模板轮廓的相识度，其模板如图 3-17 所示，该函数通过比较目标轮廓与模板的匹配度最终返回一个相识度值，其值越接近于 0 即表示轮廓与模板越匹配，该算法最终识别效果如图 3-18 所示。

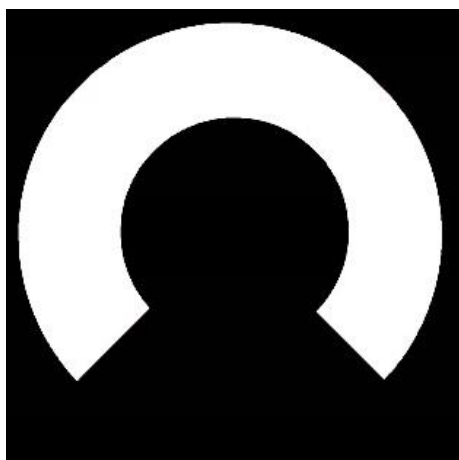


图 3-17 进行轮廓匹配的模板图像

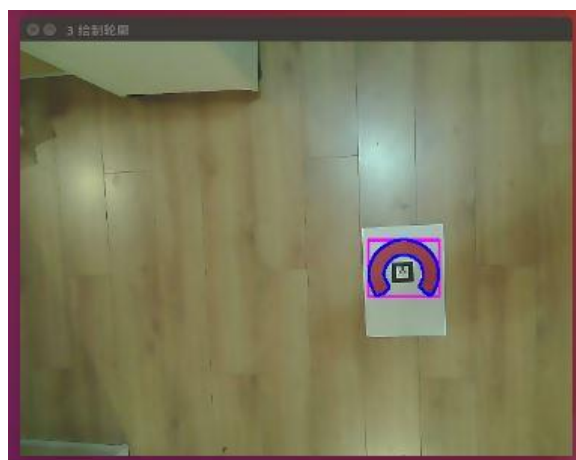


图 3-18 识别算法识别并绘制轮廓效果

3.4 对二维码的识别

本章的以上部分对一级靶标的识别匹配算法进行了详细的介绍与说明，设计中二级靶标的识别则使用了 ROS 机器人操作系统所提供的 ar_track_alvar 功能包以实现无人机对二维码识别与定位。

3.4.1 功能包的安装与二维码的创建

本节所使用的 `ar_track_alvar` 功能包的安装是在已安装 ROS-Kinetic 版本基础上使用 `apt` 指令安装，安装指令如下：

```
sudo apt-get install ros-kinetic-ar-track-alvar
```

`ar_track_alvar` 功能包提供了 QR 二维码标签的生成功能，根据不同的 `AR_ID` 设计需求根据以下指令可以生成 `AR_ID` 从 0-65535 的 65536 种二维码标签。

```
roslaunch ar_track_alvar createMarker AR_ID
```

其中 `AR_ID` 可以是 0-65535 之间的任意数，即当需要生成 `AR_ID` 为 0 的二维码标签时生成指令如下：

```
roslaunch ar_track_alvar createMarker 0
```

其生成的 `AR_ID` 为 0、1、2 的二维码标签如图 3-18 所示。

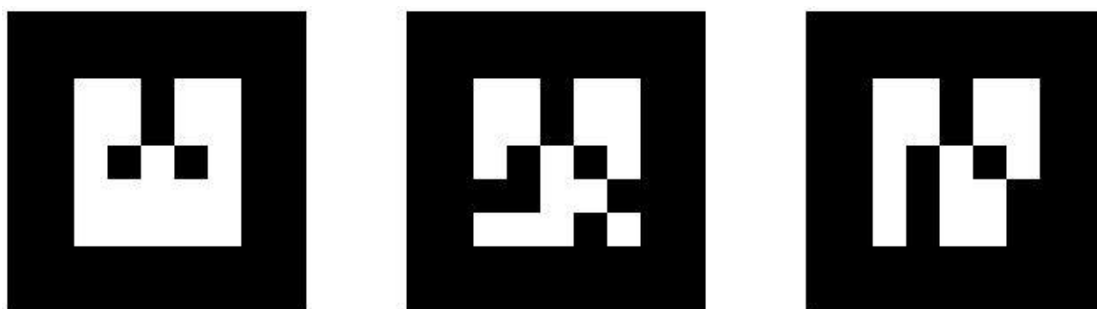


图 3-18 `AR_ID` 分别为 0、1、2 的二维码标签展示

3.4.2 对二级靶标的识别与定位

`ar_track_alvar` 功能包支持 USB 摄像头作为图像采集设备，程序通过订阅 USB 摄像头的图像信息可以完成对图像的二维码的定位并以二维码中心建立坐标系其对二级靶标的识别效果如图 3-19 所示。



图 3-19 `ar_track_alvar` 功能识别并定位二维码

ar_track_alvar 功能不仅能完成对二维码的定位, 通过在对应 launch 文件中设置二维码的尺寸信息, 该功能包还可以确定二维码的空间姿态信息, 设计中对 launch 文件中的二维码尺寸设置如图 3-20 所示。

```
1 <launch>
2   <arg name="marker_size" default="25" />
```

图 3-20 设置二维码实际尺寸

根据图 3-20 所示, 程序运行过程中通过调用参数 marker_size 获取二维码实际尺寸即 25 厘米, 当程序成功定位二维码后节点生成二维码空间位置信息如图 3-21 所示。

```
header:
  seq: 0
  stamp:
    secs: 1444430928
    nsecs: 28760322
  frame_id: /usb_camera
id: 0
confidence: 0
pose:
  header:
    seq: 0
    stamp:
      secs: 0
      nsecs: 0
    frame_id: ''
  pose:
    position:
      x: 0.196624979223
      y: -0.238047436646
      z: 1.16247606451
    orientation:
      x: 0.970435431848
      y: 0.00196992162831
      z: -0.126455066154
      w: -0.205573121457
```

图 3-21 程序对二维码定位识别结果

如图 3-21 所示, 程序确定了二维码的 ID 信息以及与摄像头的实际位置误差信息即 pose::position 中所含信息。

3.5 本章小结

本章主要完成了系统中机器视觉的图像识别算法, 包括对一级靶标轮廓的识别以及对二级靶标即二维码的识别。通过滤波算法完成对图像的降噪优化了目标的边缘信息, 通过颜色空间的变换以及颜色提取算法有效的保留了下视摄像头采集到的图片中红色部分便于完成对一级靶标的轮廓提取, 最终以 Canny 算子提

取的目标轮廓信息为输出使用轮廓 Hu 矩以及轮廓筛选程序完成相似度匹配, 经仿真验证其对靶标的有效识别率大于 90%, 其在机载计算机上的处理速度约为一帧 73ms 其处理速度可以达到无人机控制需要的 10Hz 要求。在二级靶标识别方面设计使用了 ROS 提供的 ar_track_alvar 功能包, 通过该模块完成了无人机对二级靶标的识别定位。

第4章 基于视觉的无人机位姿估计与着陆引导策略

本章将在上一章机器视觉成功获取靶标在图像中的位置信息后的基础上,对图像处理算法处理的结果即靶标位置信息结合各传感器数据,对无人机视觉引导信号进行解析处理从而获得无人机控制信息。本章将主要分析无人机视觉引导信号的解析、无人机整体着陆软件设计以及无人机着陆控制器设计。

4.1 无人机视觉引导信号的解算

在无人机位姿估计中常使用的位姿估计算法为 PnP 算法,但考虑到 PnP 算法是基于点特征的估计算法(邱鹏瑞,2019),且本设计中使用的靶标中特征角点较少,因此在视觉引导信号解算过程中采用求取靶标成像中心点与图像物理坐标系中心点 O_i 在 X、Y 方向的差值 ΔX_i 、 ΔY_i ,结合无人机飞控的各类传感器数据求取无人机与靶标在 X、Y 方向的实际距离差值 ΔX 、 ΔY ,其反映到无人机的控制信号上则表示为横滚与俯仰的控制量。当图像处理算法提取的靶标(包含一二级靶标)中心点在图像像素坐标系中表示为 (x,y) 时,且 O_i 在图像像素坐标系下的坐标为 (x_i, y_i) ,靶标中心点成像在坐标系中如图 4-1 所示。

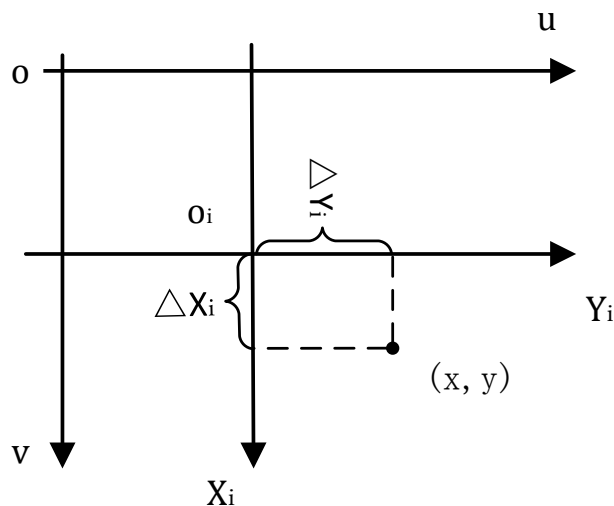


图 4-1 图像坐标系中靶标中心点成像示意图

根据 3.1.1 小节针孔相机成像模型,可知 ΔX_i 、 ΔY_i 的求解过程如下。

$$\begin{cases} \Delta X_i = (x - x_i)h_x \\ \Delta Y_i = (y - y_i)h_y \end{cases} \quad (4-1)$$

其中 (x,y) 是通过第三章中图像识别程序获取, h_x 、 h_y 在 3.1.1 节中分别表示传感器单个像素 X、Y 方向上的长度。

当无人机通过图像处理程序成功识别到着陆靶标并提取出靶标中心点图像像素坐标 (x,y) , 并计算出 ΔX_i 、 ΔY_i 的值后即可通过相机成像模型中的相似关系计算出无人机当前位置与靶标之间的水平偏移量。水平偏移量即指无人机中心点与靶标中心点的相对位置, 如图 4-2 所示, 其中靶标中心点为 P_1 , 无人机中心在该平面投影位置表示为 P_0 , 向量 $\overrightarrow{P_0P_1}$ 即表示为水平方向无人机与靶标位置存在的误差量。向量 $\overrightarrow{P_0P_1}$ 在 X、Y 方向上的分量最终反映到无人机控制上为横滚控制量 ΔX 与俯仰控制量 ΔY , 对横滚控制量与俯仰控制量的解算根据无人机摄像头光心与竖直方向有无夹角令视线角为 α (即无人机有无俯仰、横滚动作), 解算过程可分为 α 为零与 α 非零两种情况, 具体解算过程如下文所述。

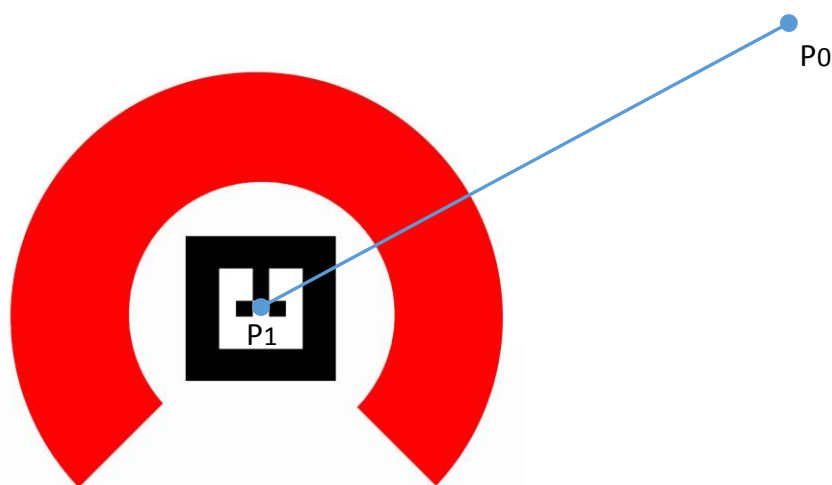


图 4-2 无人机中心与靶标中心水平误差量图示

4.1.1 视线角 α 为零时横滚与俯仰控制量的解算

本设计中使用的 Pixhawk 自驾仪能够以 921600 的波特率为机载计算机提供其传感器测量的无人机姿态数据包括俯仰角度、横滚角度、偏航角度, 由于在着陆过程中偏航角已设定为指向正北, 所以在解算无人机控制量时仅考虑当前状态下无人机俯仰角度、横滚角度, 令横滚角度、俯仰角度分别表示为 β 、 γ 。当无人机当前 $\beta=0$ 、 $\gamma=0$ 时即当前成像平面与靶标平面平行时, 其成像模型如图 4-3 所示, 图中 PP_0 表示摄像机光心离地高度即无人机飞行高度其值由 Pixhawk 自驾仪提供当高度低于 4 米后则使用超声波模块数据, 令其值为 h 。根据三角形相似可知当 $\beta=0$ 、 $\gamma=0$ 时的横滚与俯仰控制量计算公式如 4-2 所示:

$$\begin{cases} \Delta X = \frac{h \cdot \Delta X_i}{f} \\ \Delta Y = \frac{h \cdot \Delta Y_i}{f} \end{cases} \quad (4-2)$$

该式中控制量的正负由靶标成像中心点与图像物理坐标系中心点 O_i 在 X 、 Y 方向的差值 ΔX_i 、 ΔY_i 的正负决定，程序中规定当 $\Delta X > 0$ 时无人机左侧两电机转速提高飞机横滚角度沿逆时针旋转一定角度向右飞行，当 $\Delta X < 0$ 时无人机右侧两电机转速提高横滚角度沿顺时针旋转一定角度向左飞行，当 $\Delta Y > 0$ 时无人机前方两电机转速提高机头抬升一定角度使无人机向后飞行，当 $\Delta Y < 0$ 时无人机后方两电机转速提高机头下沉一定角度使无人机向前飞行。

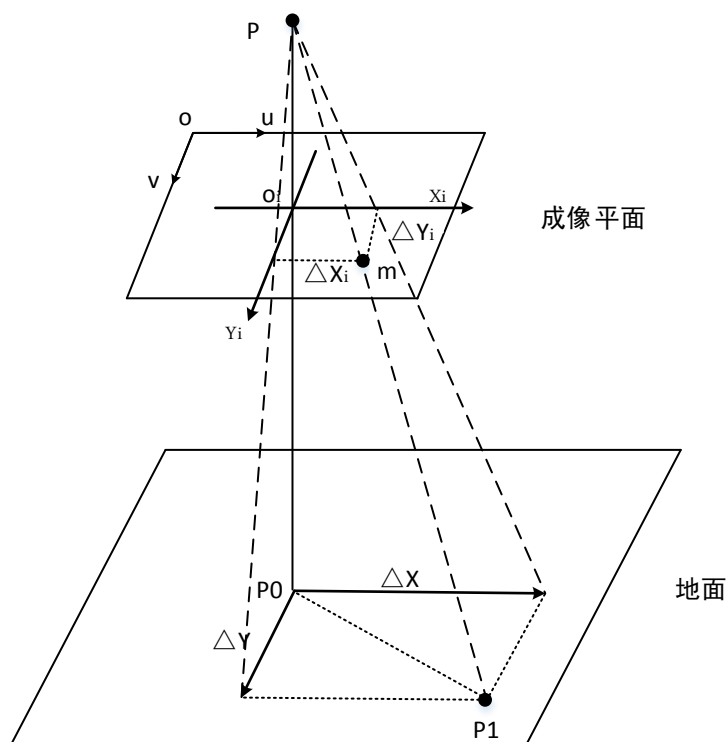


图 4-3 视线角为 0 的状态下控制量解算模型

4.1.2 视线角 α 非零时横滚与俯仰控制量的解算

当无人机当前处于机动状态此时横滚角 β 与俯仰角 γ 为非零状态，该情形下由于摄像头光心线不再垂直于地面，即无人机摄像头光心与靶标平面交点 O 与无人机在该平面的垂直投影点 P_0 不重合，此时图像坐标系下的 ΔX_i 、 ΔY_i 将与实际环境中的 ΔX 、 ΔY 不再存在相似关系。但在 β 、 γ 与无人机高度 h 已知的情况依旧可以根据光心与地面交点同靶标中心点位置关系分为图 4-4 中的两种情况对 ΔX 、 ΔY 进行推导。其中图 4-4 (a) 中表示靶标中心点位于无人机摄像头光心与靶标平面交点 O 和 P_0 点之间，图 4-4 (b) 表示靶标中心点位于两点之外。

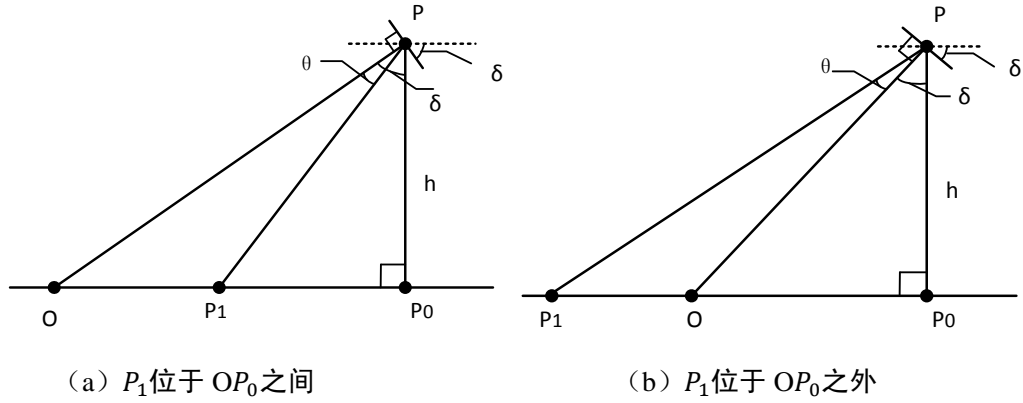


图 4-4 视线角非零情况中靶标不同位置下控制量的解析模型

图中 δ 角表示无人机的横滚角 β 与俯仰角 γ 的和角度满足 $\cos \delta = \cos \beta \cos \gamma$, 即图 4-4 表示复合角度情况下的情况, 对于 X 与 Y 方向上的偏移量计算可根据图中模型进行计算, 其中 β 与 γ 取值范围为 $[0, 90)$ 在降落过程中由飞控正常控制下一般小于 45° , 图中 θ 角表示摄像头光心线与摄像头光心到靶标中心点间连线的夹角, 根据相机成像模型可知 θ 角表示如式 4-3, 式中 ΔK_i 表示图像坐标系中的 ΔX_i 、 ΔY_i 。

$$\theta = \arctan \frac{\Delta K_i}{f} \quad (4-3)$$

当无人机与靶标位置处于图 4-4(a) 情况下时向量 $\overrightarrow{P_0P_1}$ 的计算公式如下。

$$P_0P_1 = h * \tan(\delta - \theta) \quad (4-4)$$

结合公式 4-3 与 4-4 可得到向量 $\overrightarrow{P_0P_1}$ 的计算公式为:

$$P_0P_1 = h * \tan(\delta - \arctan \frac{\Delta K_i}{f}) \quad (4-5)$$

式 4-5 即解算出该情形下无人机与靶标水平方向的距离实际误差量向量 $\overrightarrow{P_0P_1}$, 依据此公式计算得到 ΔX 、 ΔY 的表示公式如下。

$$\begin{cases} \Delta X = h * \tan(\beta - \arctan \frac{\Delta X_i}{f}) \\ \Delta Y = h * \tan(\gamma - \arctan \frac{\Delta Y_i}{f}) \end{cases} \quad (4-6)$$

同理当无人机与靶标位置处于图 4-4(b) 情况下时向量 $\overrightarrow{P_0P_1}$ 的计算公式如下。

$$P_0P_1 = h * \tan(\delta + \arctan \frac{\Delta K_i}{f}) \quad (4-7)$$

即该情况下 ΔX 、 ΔY 的表示公式为:

$$\begin{cases} \Delta X = h * \tan(\beta + \arctan \frac{\Delta X_i}{f}) \\ \Delta Y = h * \tan(\gamma + \arctan \frac{\Delta Y_i}{f}) \end{cases} \quad (4-8)$$

由以上推导过程可知无人机在 X 、 Y 方向上的位移偏差 ΔX 、 ΔY 的正负分别由横滚角 β 与 ΔX_i 和俯仰角 γ 与 ΔY_i 的正负决定。

4.2 无人机着陆控制器及其设计

通过 4.1 节中对无人机与靶标水平方向位置误差的解算，无人机能够顺利计算出不同情况下无人机下一时刻的期望位置，但该结果不能直接作为无人机的控制输入，还需对其进行处理最终产生无人机机动所需要的速度控制与角度控制量。所以无人机在接近靶标的过程仍需要设计一个着陆控制器。在四旋翼无人机的着陆控制系统中主要由位置环控制和姿态环控制（曾聪，2018）。本设计引入 PX4 的位置控制器使用串联 PID 控制算法，设计了无人机着陆控制器用于引导无人机向靶标接近，其总体设计框图如图 4-5 所示。

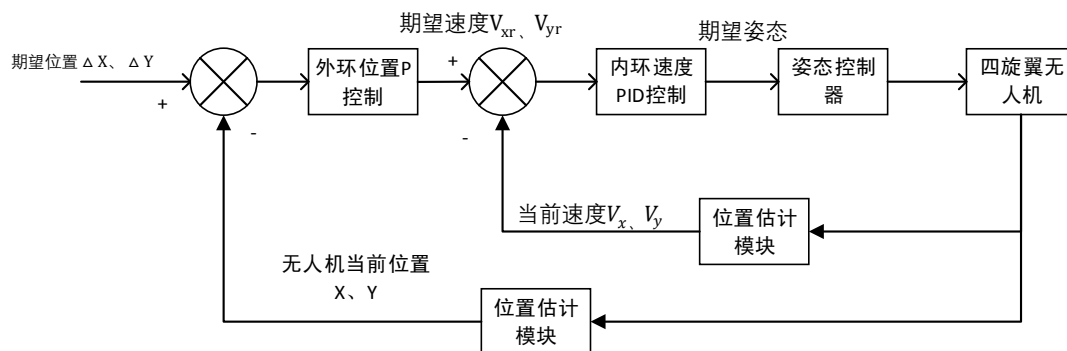


图 4-5 无人机控制系统总体框图

4.2.1 位置控制环与速度控制环控制器设计

根据图 4-5 所示，外环位置控制器其通过无人机 GPS 反馈信号获取当前无人机的坐标信息以及图像处理算法获取的无人机与靶标水平位置误差产生期望位置，通过比例 P 控制产生无人机在 X、Y 方向上的期望速度 V_{xr} 、 V_{yr} 并做为内环速度环的 PID 控制输入。内环速度控制器通过 PID 控制输入分别为位置控制器输入的期望速度以及飞控通过陀螺仪、加速度计等传感器实时测量出的无人机当前状态下 X、Y 方向的实际速度 V_x 、 V_y ，速度控制器通过输入的期望速度与实际速度应用 PID 控制方法解算出无人机下一时刻的期望姿态，包括期望横滚角度 β_r 与期望俯仰角度 γ_r 和期望偏航角度 θ_r ，其中偏航角 θ 固定为正北方向不变。位置控制器与速度控制器结构如图 4-6 所示。

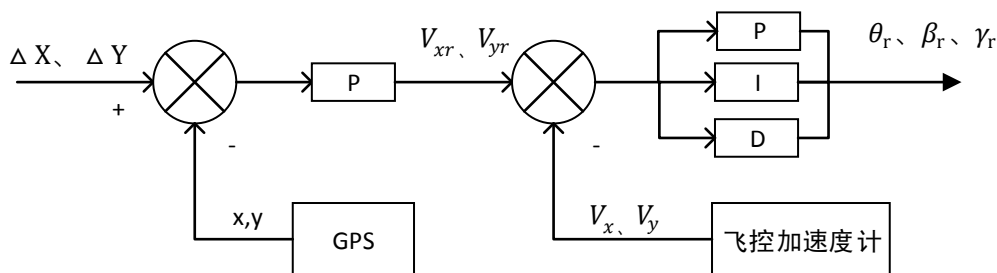


图 4-6 位置控制器与速度控制控制器结构

4.2.2 无人机姿态控制器设计

无人机姿态控制器是用于无人机姿态控制，其输入为 4.2.1 中位置控制器与速度控制器输出的期望姿态角以及飞控陀螺仪采集的实时姿态角数据，产生最终的俯仰、横滚、偏航控制量。姿态控制器是由内环角速度控制器与外环角度控制器组成（张舸，2018）其结构如图 4-7 所示。

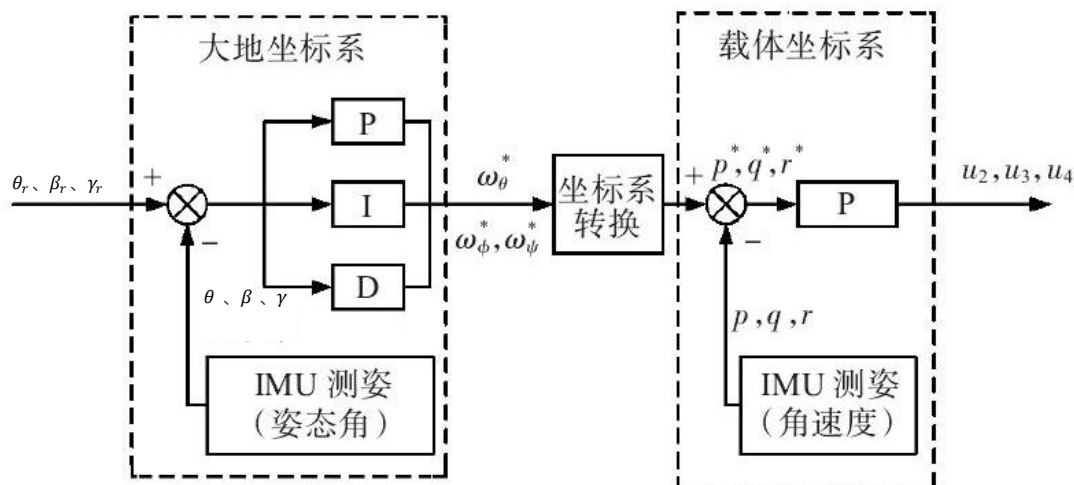


图 4-7 无人机姿态控制器结构

由姿态控制器结构图可知，外环姿态角控制环输入为期望姿态以及飞控姿态传感器测量的实时姿态角（以欧拉角表示），经比例积分微分（PID）控制计算输出期望角速度，其运算是在大地坐标系即世界坐标系下进行。内环角速度控制器采用比例（P）控制其是在载体坐标系即相机坐标系下进行输入为经坐标系转换后的期望角速度以及经飞控姿态传感器测量出的实时角速度，输出为在俯仰、横滚、偏航方向上的运动控制量 u_2 、 u_3 、 u_4 。

4.3 无人机视觉引导着陆软件设计

本设计中无人机的视觉引导着陆过程中软件程序流程可以分为四个阶段，四

阶段介绍如 2.1.1 节所述。无人机视觉引导着陆软件逻辑流程图如图 4-8 所示。

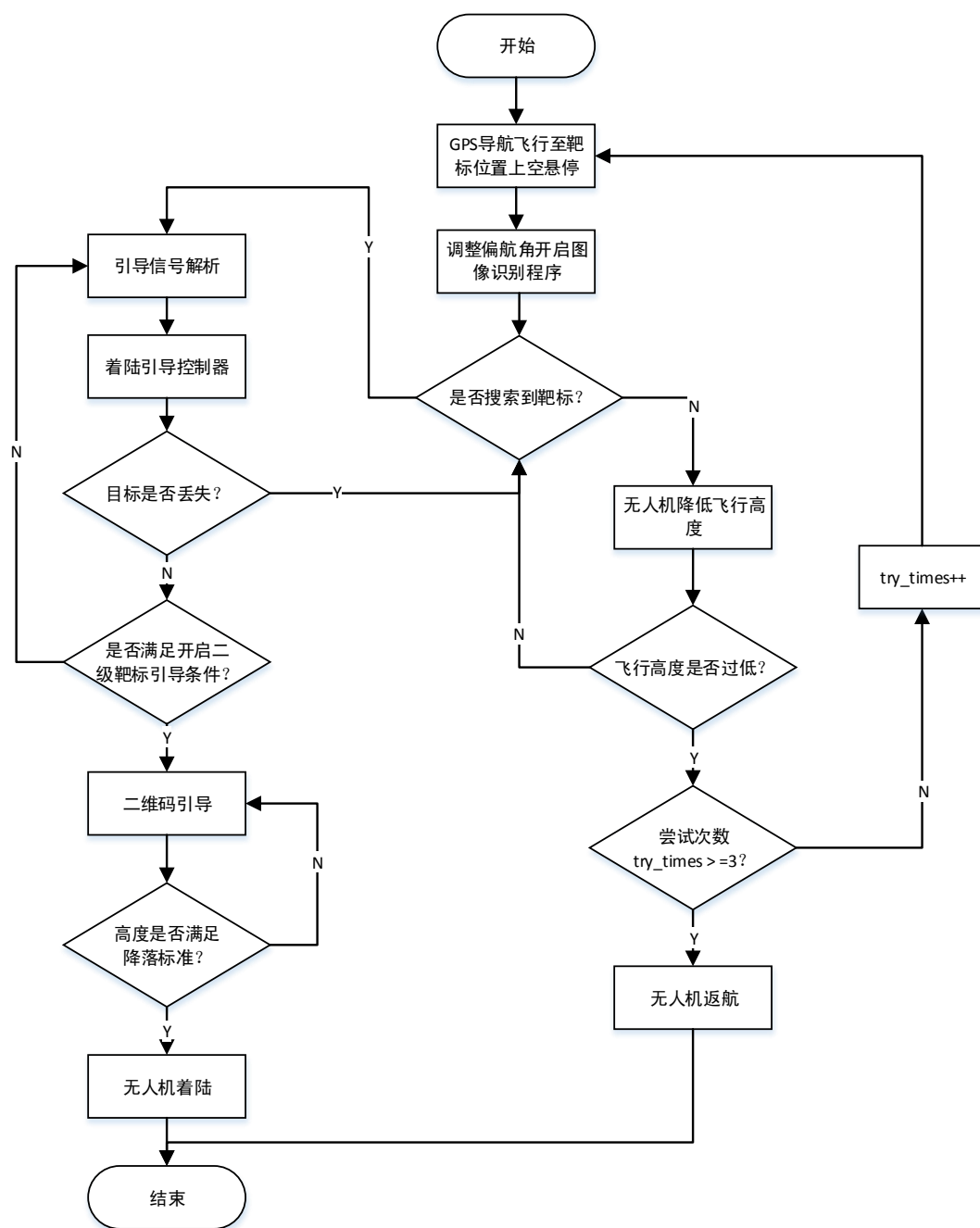


图 4-8 无人机视觉引导着陆软件逻辑流程图

在上图的软件逻辑流程图中当无人机悬停在靶标上空约 20 米处无法识别到靶标位置时，无人机尝试降低高度，当高度达到 15 米仍无法正常识别时无人机将重新执行着陆程序并记录尝试次数，当尝试 3 次仍未寻找到靶标时无人机将执行 PX4 飞控的返航程序，以此避免无人机因无法识别靶标而滞空的情况。图中当无人机识别到一级靶标并执行视觉引导程序后，当二维码检测程序成功检测到二级靶标并成功匹配信息后无人机二维码识别程序将会对二维码正确识别标志布尔型变量 `qr_identify` 置 1，当高度低于 2 米同时 `qr_identify` 为真时视觉引导信

号将转变为二级靶标中心位置即启动二维码导航,其导航过程容错设计参照一级靶标。最终当超声波雷达监测在正确引导过程中高度低于 30 厘米,无人机将执行着陆命令进行着陆,自此完成整个的无人机视觉引导着陆过程。

4.4 本章小结

本章中对无人机视觉引导信号的解算过程进行了详细论述,阐述了无人机视觉算法识别到靶标后解算无人机与靶标位置误差的过程,并介绍了无人机控制系统的设计以及整体着陆软件逻辑设计流程。

第 5 章 无人机视觉引导着陆飞行验证与数据分析

本章在完成无人机软硬件设计的基础上，将在 Gazebo 仿真软件上对图像识别算法的准确率以及无人机着陆过程进行软件在环仿真，后对无人机视觉引导着陆进行实际飞行测试，并根据飞控在飞行过程中保存在 SD 卡内的日志数据进行数据分析，以此完成本设计的实验论证与分析。

5.1 无人机视觉引导着陆过程 Gazebo 软件在环仿真实验

由于无人机的特殊性，其在开发实验过程中存在着较多风险与问题，例如需要承担在实际飞行中由软硬件问题带来的高风险的财产损失、由无人机故障或人为操作失误造成的人员意外受伤风险、频繁试飞增加研发时间成本与物力成本等问题，所以对无人机进行软件在环仿真 Software-in-the-Loop-Simulation (SITL) 或硬件在环仿真 Hardware-in-the-loop simulation (HILS) 是无人飞行器 (UAV) 系统设计中的两种有效论证方法 (Nguyen K D, 2018)，以上两种方法都可以让工程师在软件上使用无人机模型对无人机图像识别算法以及控制算法进行仿真测试。考虑到 HILS 是半硬件仿真即仿真过程需要连接飞控到电脑其仿真建立过程较为耗时，且 SITL 是对算法和控制仿真的最优仿真方案，所以本设计选用 SITL 的仿真方案。

近年来，SITL 仿真中经常使用 Matlab / Simulink 或 PX4 源码开发飞行控制算法，还有 X-Plane、FlightGear、JMAVSim 或 Gazebo 用于建造无人机模型。其中，PX4 源码和 Gazebo 仿真是最优的 SITL 仿真方案其原因有下：首先，PX4 是具有许多库的开源库，可用于开发无人驾驶飞机和地面无驾驶车辆。它包括飞行控制的基本控制器代码。其次，构建 PX4 并将其上传到开放硬件平台 Pixhawk 上其是一种非常流行的硬件无人机应用。Gazebo 是一款专业化三维仿真平台，可以提供仿真物体动态模型、传感器模型和 3D 可视化 (K. Dang Nguyen and T. Nguyen, 2019)。因此本文采用了 PX4 源码与 Gazebo 仿真的 SITL 仿真模式。

5.1.1 SITL 仿真环境搭建

本设计中所涉及的仿真环境，是搭建在运行有 Ubuntu16.04 LTS 的台式计算机上，其仿真搭建过程主要包括 linux 下 ROS-Kinetic 的安装、MAVROS 功能包的安装、PX4 源码编译，机载计算机环境安装同于此处环境搭建但机载计算机环境搭建过程中不涉及 PX4 源码编译。其仿真环境具体搭建过程如下。

(1) ROS-Kinetic 的安装: 本设计所使用的 ROS 系统为 Kinetic 版本, 其安装过程中主要安装指令如下:

```
sudo apt-get install ros-kinetic-desktop-full
```

通过以上指令可以完成对 ROS 操作系统以及 ROS 仿真软件等一系列工具的安装其中包括用于仿真实验的 Gazebo7，其详细过程可参考 ROS 官网 ROS-Kinetic 的安装。

(2) **PX4 源码编译**: 在仿真实验中本设计使用开源飞控 **PX4** 源码 1.8 版本进行仿真飞行控制, 其源代码通过 **Github PX4** 开源项目获取, **PX4** 源码下载至本地通过编译后可在仿真环境下实现对仿真无人机的飞行控制, 其通过 **cmake** 指令进行编译效果如图 5-1 所示。

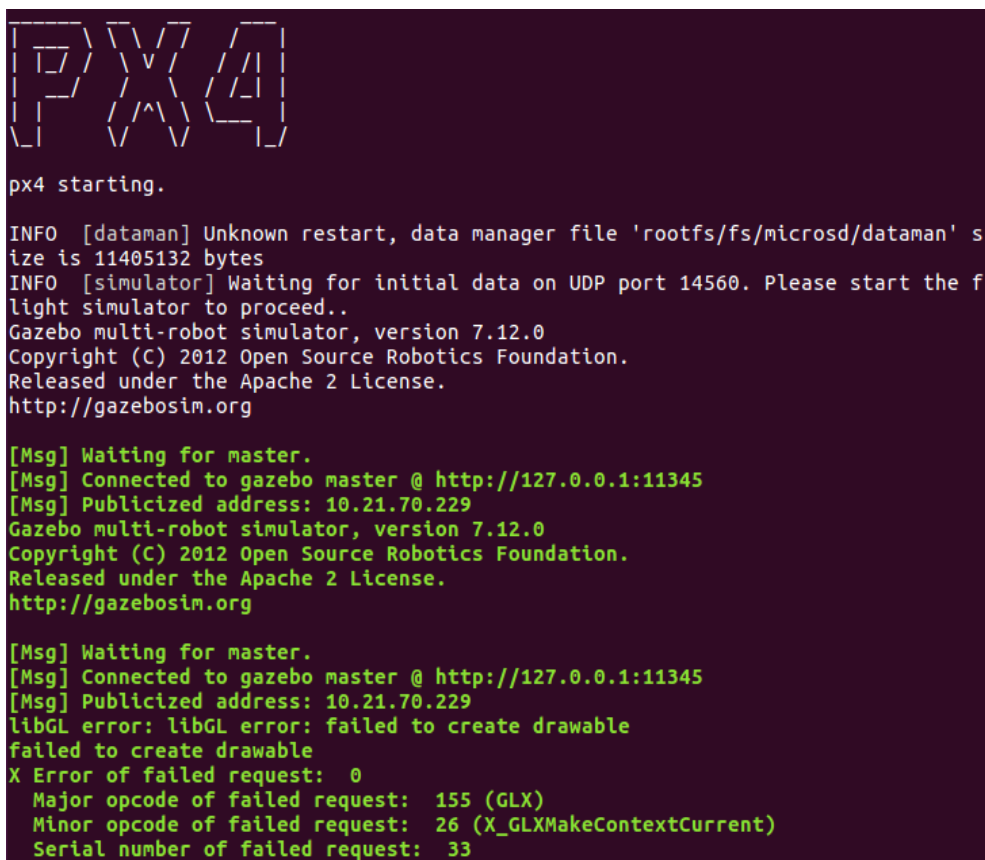


图 5-1 PX4 源码本地编译

通过以上编译完成后启动 PX4 Gazebo 仿真效果如图 5-2 所示。

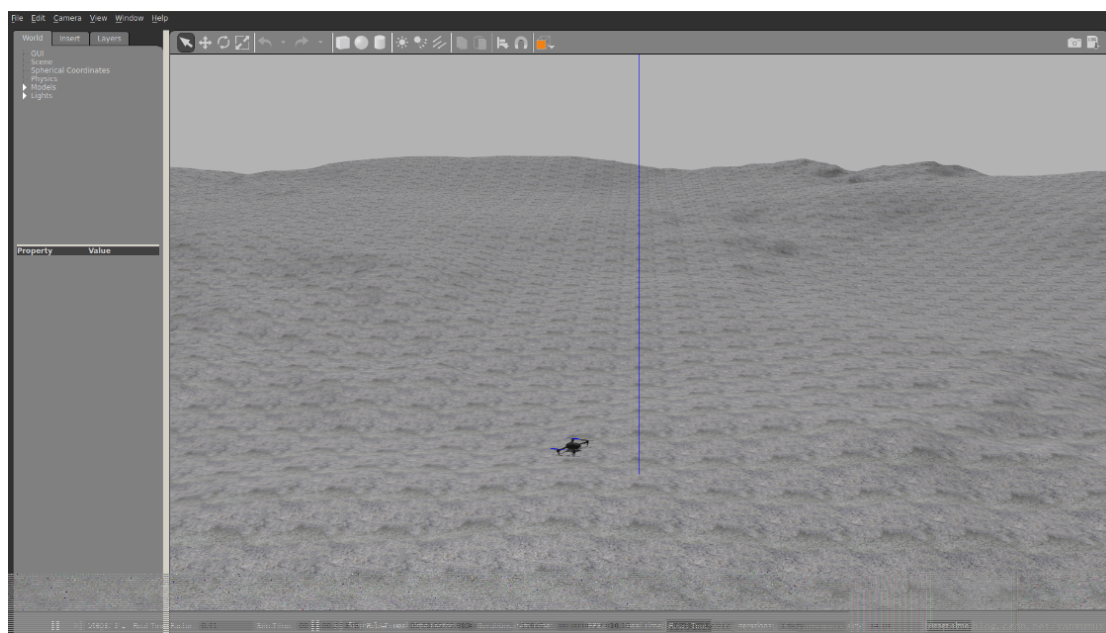


图 5-2 PX4 源码 Gazebo 仿真效果

(3) MAVROS 功能包安装：无人机是使用 MAVLink 小型无人机通信协议进行交互控制的，为了与设备进行信息交互 ROS 使用了 MAVROS 功能包，该功能包实现了使用标准 ROS 数据类型通过 MAVLink 协议发送机器控制指令的功能 (Taganov A A, 2019)，通过该功能包 ROS 可以连接到飞行自驾仪或 Gazebo，以实现与飞控的信息交互。设计中该功能包安装在 ROS 系统的工作区路径为 `~/catkin_ws/src` 其安装指令如下：

a. MAVLink 的安装

```
rosinstall_generator --rostdistro kinetic mavlink | tee /tmp/mavros.rosinstall
```

b. MAVROS 的安装

```
rosinstall_generator --upstream mavros | tee -a /tmp/mavros.rosinstall
```

5.1.2 靶标误识别率仿真实验

在仿真实验环境中采用的无人机模型使用的是 PX4 源码仿真部分提供的 `iris_fpv_cam.sdf` 文件中搭建好的无人机模型，其是一款带有下视单目的四旋翼无人机其中摄像头位置及姿态可根据需求修改，根据本文需求将下视摄像头设置为无人机正下方且摄像头正向下，其配置文件摄像头位姿部分设置如图 5-3 所示，其中参数 `pose` 中 6 个数据中前 3 位参数分别表示以无人机中心为原点摄像头在 X、Y、Z 方向上的距离，后 3 位数据分别代表摄像头绕 X、Y、Z 轴旋转角度，即姿态的欧拉角表示。

```
10 <include>
11   <uri>model://fpv_cam</uri>
12   <pose>0 0 0 0 0 0</pose>
13 </include>
```

图 5-3 仿真无人机摄像头位姿配置参数

为验证设计中图像识别算法对靶标的误识别率，本实验在仿真环境下对无人机进行复杂环境干扰情况下的仿真飞行实验。其中干扰源设置为不同形状大小的红色目标，其仿真测试场景如图 5-4 所示。

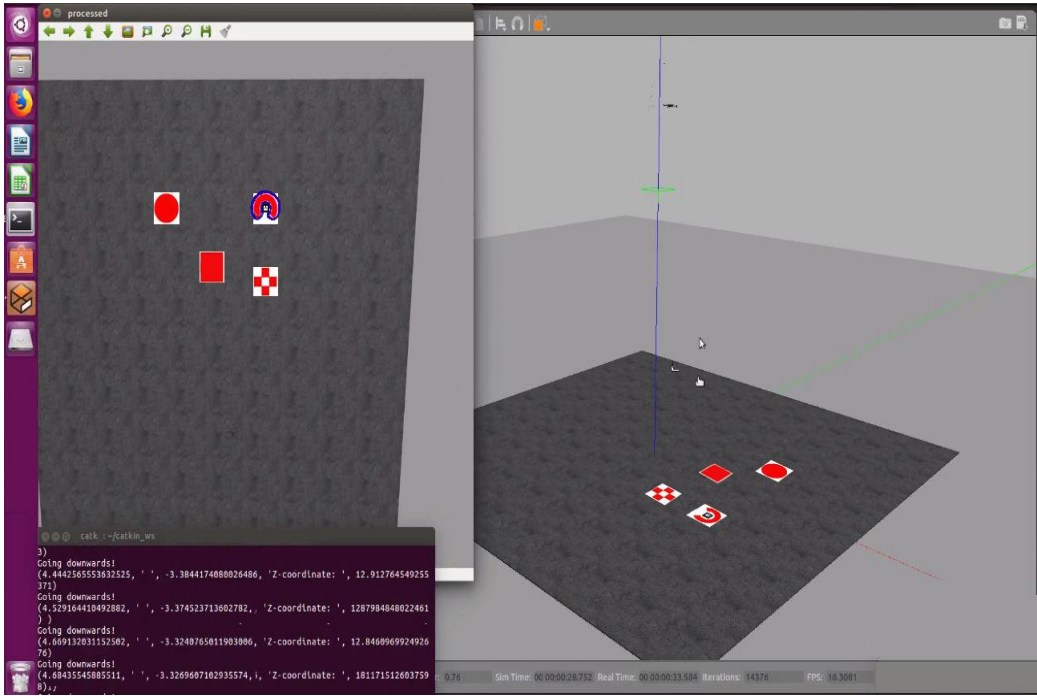


图 5-4 图像识别算法 Gazebo 仿真测试场景

在图 5-4 中的左侧为仿真环境下无人机下视摄像头所拍摄的画面，图中图像识别算法正确识别到一级靶标后通过蓝色线条在图像中描绘出满足条件且匹配度最高的目标轮廓，并且位置解析算法成功解析出无人机与靶标的 X、Y 方向上的位置误差。通过该仿真环境下的 100 架次自动重复飞行验证，该仿真条件下图像算法的首次正确识别率达到 92%，通过容错逻辑修正后无人机正确引导率达到 96%。

5.1.3 着陆误差仿真验证

为了验证无人机视觉引导着陆精度，该部分实验在上一节仿真环境下将无人机着陆靶标移动至仿真世界的原点位置即靶标中心在仿真世界坐标系下表示为 (0,0,0)，并通过无人机着陆后的世界坐标系位置确定无人机着陆误差。通过统计 15 次实验结果得到平均着陆误差为 0.25 米，其详细数据如表 5-1 所示。

表 5-1 着陆误差仿真实验数据

实验	着陆 X 坐标	着陆 Y 坐标	着陆误差
1	0.132	-0.267	0.298
2	0.213	0.258	0.335
3	-0.176	-0.159	0.237
4	0.183	0.127	0.223
5	-0.275	0.196	0.338
6	0.039	-0.217	0.220
7	0.167	-0.138	0.217
8	-0.089	-0.057	0.108
9	-0.073	0.283	0.292
10	0.097	0.142	0.172
11	-0.332	-0.245	0.413
12	0.151	0.097	0.179
13	-0.225	0.183	0.290
14	0.039	0.171	0.175
15	0.168	-0.135	0.216

5.2 着陆飞行实验及数据分析

通过仿真实验对图像识别算法以及无人机位姿控制软件进行验证分析后,该部分实验对无人机进行实物着陆飞行实验,以验证本设计实际效果并获取无人机视觉引导着陆过程中的相关数据信息。

5.2.1 着陆飞行实验

本实验出于安全以及实验效果考虑,实验在室外开放环境下进行。在实验进行前需要对硬件连接进行复检,检测完毕后还需对无人机自驾仪的传感器进行校准,该校准采用无人机开源地面站 QGroundControl (QGC) 进行,校准过程无人机自驾仪通过 USB 口连接笔记本电脑,其中无人机传感器校准过程如图 5-5 所示。

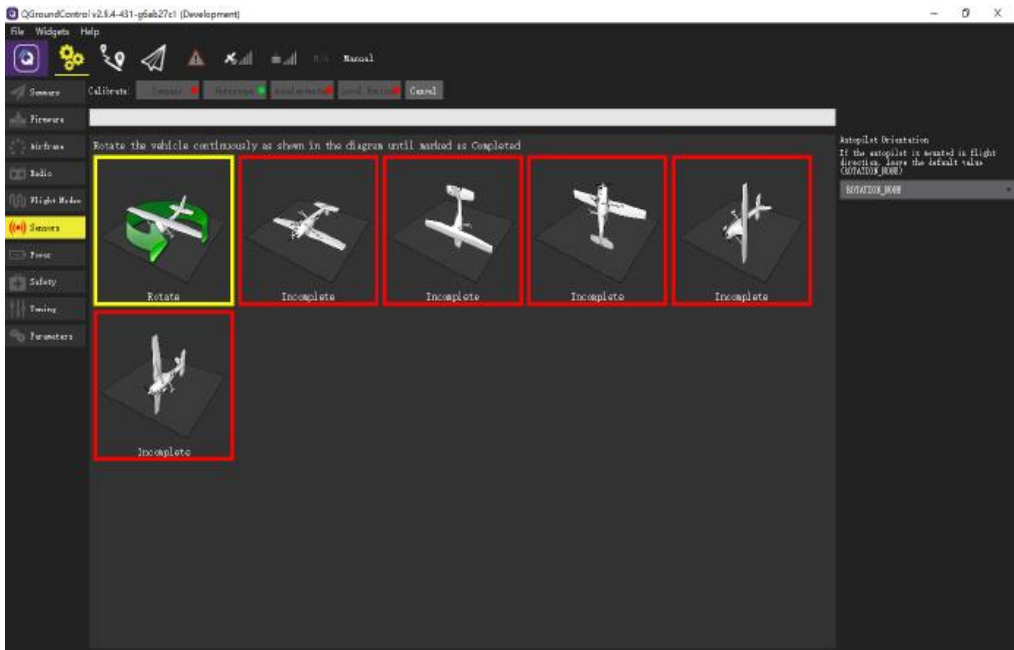


图 5-5 QGroundControl 软件无人机传感器校准界面

由于本设计中并未涉及航迹引导规划问题，所以实验过程中首先通过遥控器将无人机引导至约 20 米高度，并通过遥控器的通道 1 开关控制无人机飞行模式，在抵达预定高度后拨动开关启动飞行自驾仪的 offboard 模式，机载计算机通过 MAVROS 监控自驾仪模式状态，当确认无人机进入 offboard 模式后启动应用程序即开启自主着陆引导流程。实验飞行过程中首先将无人机升至靶标上空，然后开启无人机 offboard 模式无人机在切换控制模式后悬停等待机载计算机的控制指令，其飞行过程中高度信息如图 5-6 所示，图中垂直轴为高度单位为米，水平轴为数据编号其数据采集频率为 10Hz。



图 5-6 实验飞行过程中无人机高度数据

图 5-6 中无人机最大飞行高度约 23 米，悬停阶段约 100 秒其中模式切换在约 40 秒处，着陆发生在约 170 秒处，图中起飞降落阶段受地效影响由于无人机自驾仪高度数据来源于气压计所以发生高度为负情况，无人机着陆最后阶段高度数据来源于超声波模块故不受影响。无人机着陆过程实验场景如图 5-7 所示。



图 5-7 无人机着陆实验实验场景及过程

5.2.2 着陆飞行实验位置控制数据

着陆飞行实验过程中通过位置解析程序识别到无人机着陆靶标后将无人机与靶标水平位置误差以飞控日志保存速率即 10Hz 将数据以日志形式保存至机载计算机中,水平位置误差数据如图 5-8 所示,其中垂直轴为误差距离单位为米,水平轴为数据编号其数据采集频率为 10Hz。

水平位置误差

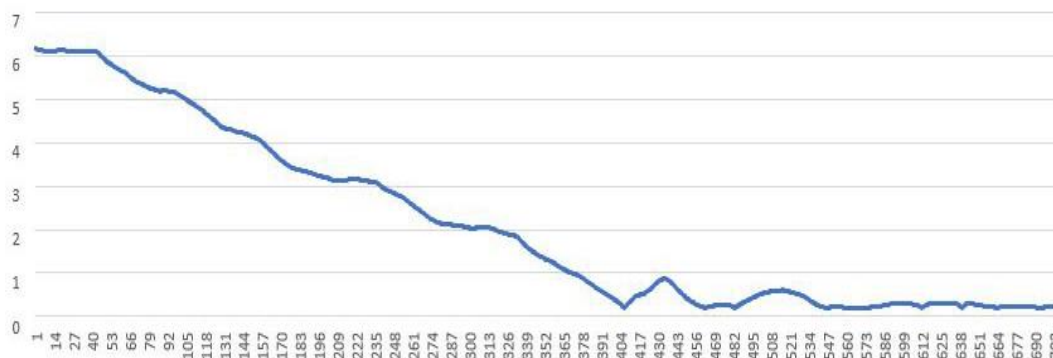


图 5-8 无人机与靶标水平位置误差

根据图 5-6 与 5-8 可知,无人机在启动后约 100 秒后成功识别到一级靶标,在约 104 秒无人机开始视觉引导着陆,该过程持续约 70 秒。在该阶段中飞行器在逐渐靠近着陆标识,无人机与靶标水平位置误差逐渐减小,受位置控制程序以及外部因素如试验过程中风力影响水平位置误差在视觉引导开启约 40 秒至 53 秒过程中略有增加后最终稳定。

5.2.3 着陆飞行试验速度控制数据

Pixhawk 飞行自驾仪在飞行过程中会以 10Hz 的速率实时记录无人机状态数

据到 SD 卡中其中包括无人机 X、Y、Z 方向上的速度及加速度，本节中通过提取飞行日志数据以反映本设计中速度控制器控制效果。其中着陆过程中各方向上的速度信息如图 5-9 所示。

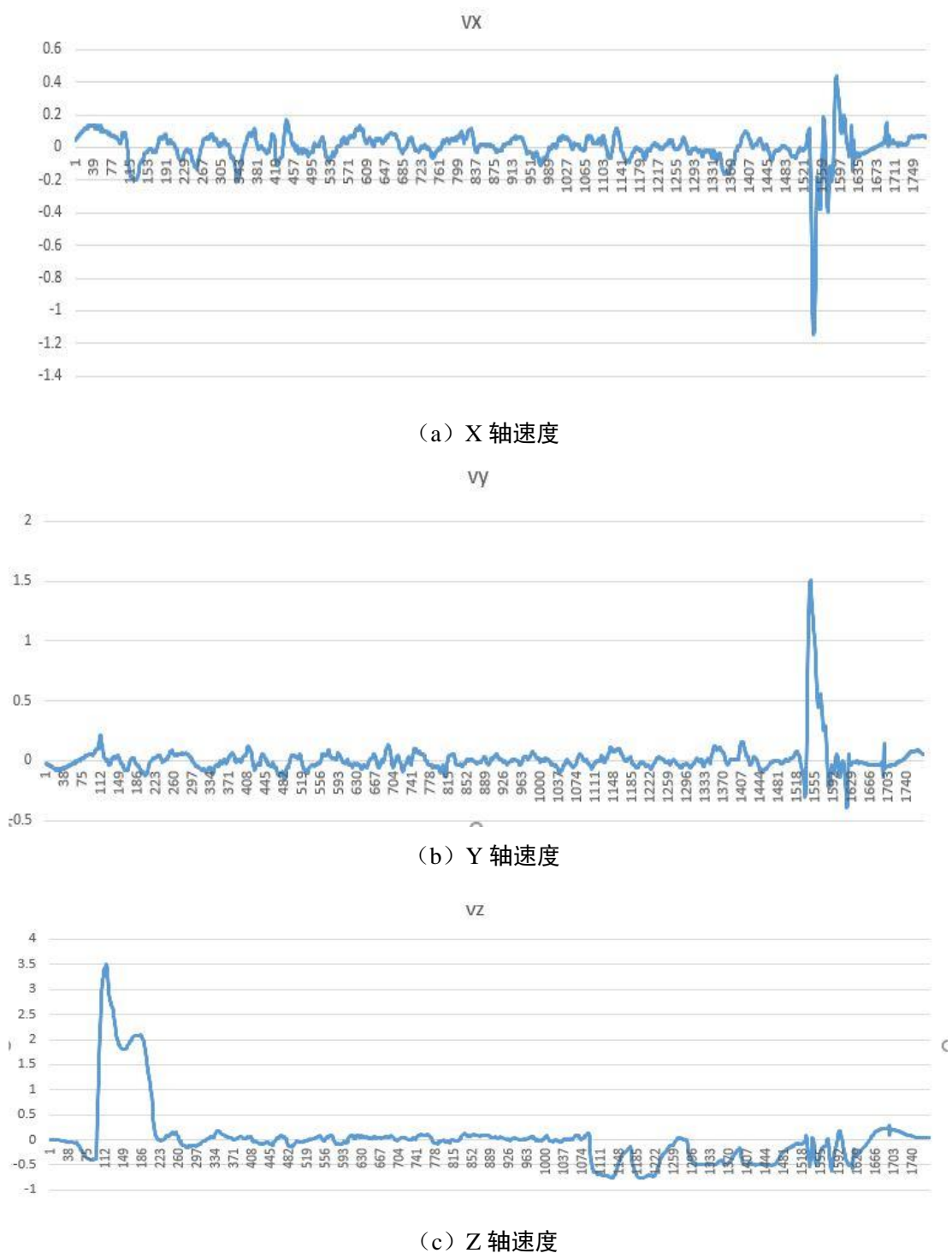


图 5-9 实验过程中无人机速度信息

由上图可知着陆过程中无人机在 X、Y 方向上的速度整体较为平稳，仅在一二级靶标切换过程中产生轻微抖动。Z 轴方向速度在起飞过程中波动较大，在一级靶标引导过程（约为 1074-1530 区间数据）中变化较小，在二级靶标引导过程

中（约为 1530-1610 数据）由于高度较低所以采用逐渐靠近的方式所以 Z 轴方向上速度反复波动呈现出多个倒峰状。

5.3 实验飞行着陆精度

为确定室外实际测试环境下无人机视觉引导着陆实验精度，本部分实验进 10 架次反复着陆测试，最终通过测量无人机中心位置至靶标中心位置距离确定了无人机实验环境下的着陆精度，其数据如表 5-2 所示。

表 5-2 实验飞行着陆精度数据

实验	着陆误差（m）
1	0.23
2	0.36
3	0.31
4	0.29
5	0.39
6	0.17
7	0.28
8	0.42
9	0.26
10	0.33
平均误差	0.30

根据数据显示实际测量着陆误差略大于仿真环境下的误差，考虑到无人机靶标尺寸为 1m×1m，以及在物流无人机领域中无人机着陆场地较大，该着陆精度完全满足高精度视觉引导着陆的需求。

5.4 本章小结

本章主要完成了本设计的实验验证部分其中包括软件在环仿真验证以及实际飞行数据分析。该部分的软件仿真对本设计中的图像识别误识别率以及仿真环境下的着陆误差精度进行了验证，实际飞行验证部分通过分析无人机飞行日志对设计中的位置控制以及速度控制进行了数据分析并对着陆精度进行了进一步的实验验证。根据以上部分的实验很好的验证了本设计的最终飞行效果。

结论与展望

结论

随着近些年来多旋翼无人机应用的兴起,多旋翼无人机的商业应用逐渐走向大型化与智能化,依靠人工遥控操作或后台操作的商业无人机在众多领域已不能满足需求,尤其是在物流领域物流无人机必将成为未来无人化物流中的重要一环。但在自主着陆技术验证方面现有研究多应用在 10 米以下飞行高度,其设计的靶标较小不能满足较高飞行高度条件下的应用。针对物流无人机在自主飞行过程中识别高度高、着陆误差小的需求,本文设计使用基于 Opencv 的图像识别算法在 Pixhawk 2.4.8 版本自驾仪的板外控制(offboard)模式下完成飞控自驾仪的视觉导航二次开发,最终实现符合该应用条件下的视觉引导着陆方案。最后对本设计的主要研究成果总结如下:

(1) 提出并设计了一款尺寸为 1m×0.85m 的分级式无人机着陆引导标识,即一级靶标为无人机在 20 米高度下提供视觉导航二级靶标为最终降落提供误差为 30 厘米的视觉引导。实验过程中无人机对一级靶标的识别高度略高于 20 米在约 23 米处即可实现对一级靶标的识别,最终在该靶标的引导下可达到平均误差为 30 厘米的着陆精度,达到了实际应用水平。着陆精度达到国内外现有研究的同等水平,并优于部分现有研究。

(2) 设计使用 Opencv 对图像进行处理,图像处理算法实现了对一级靶标的识别与匹配,使用一级靶标的面积信息、颜色信息以及轮廓对目标进行匹配筛选提高了图像处理算法对目标的识别准确率,经 Gazebo 软件仿真验证其正确识别率可达到 96%,其能够很好的在运行环境下过滤环境干扰并正确识别出靶标位置。

(3) 设计引入了 ROS 操作系统作为无人机的软件开发平台,简化了开发流程降低了程序间的耦合度,并实现整体软件及着陆流程的 Gazebo 仿真,在无人机控制软件开发完成后在 Gazebo 仿真软件上实现了软件的全仿真测试,为开发节约了大量的测试飞行时间与成本,并完成全仿真下图像识别准确率等的数据采集,其为现有无人机开发流程提供了新的验证思路。

(4) 通过完成整个无人机视觉引导着陆的设计后进行的仿真与实际飞行数据采集,对数据进行分析证实了本设计方案的可行性与实用性,结果说明本设计初步完成了设计要求。

后续工作的展望

无人机全自主飞行是当前民/商用无人机的发展趋势，由其是在物流无人机领域，由于其着陆区域的限制以及其频繁的起降过程，其对无人机自主着陆精度、飞行稳定性、抗风能力等都有着非常高的要求，由于本设计仅限于技术论证所以其还存在众多技术不足，具体如下：

（1）本设计的图形处理平台是 Opencv，图像处理算法虽然在仿真环境及测试环境下能够达到非常高的正确识别率，但是在实际应用环境下如夏日强光照、夜晚低光照强度、实际复杂干扰环境、光污染环境等应用场景下对靶标的识别情况还需要进行大量测试与优化，限于个人精力、投入时间、能力等限制未能完成以上各项测试与优化。

（2）随着机器学习的发展与成熟，将机器学习技术应用到该设计的图像识别中能够大大提升无人机图像识别的效率与准确率。

（3）本设计着陆过程中控制器仅采用简单的位姿控制，并未对着陆过程进行着陆航迹规划，实际着陆过程中无人机飞行效果抖动较大。后期可以通过深入研究无人机的位姿控制算法，结合更加合理的着陆逻辑可以进一步提升飞行稳定性以及着陆精度。

（4）本设计中着重开发并实现多旋翼无人机自主控制着陆功能，其对无人机整体性能如抗电磁干扰能力、功耗管控、GPS 导入地图定位等功能与性能均未进行测试或融合。

致 谢

时光荏苒，日月如梭。转眼间我即将结束三年的硕士研究生的学习生活，通过这三年的学习与锻炼我在学术能力以及学术水平上得到了大幅提升。这三年对我来说，是宝贵的三年，是收获的三年，在这里我由衷地感谢在成都理工大学遇见的各位老师和同学！

这三年里我特别感谢我的导师方方教授以及工作室指导老师曾维副教授！从本科升至硕士研究生以来是方老师与曾老师的不断指导才让我逐渐走向学术研究的道路上来。在学术上，我紧跟方老师与曾老师的步伐，是方老师与曾老师带我领略了电子与通信工程专业的风采、带领我潜心钻研，让我真正的喜欢上了我所学的专业；方老师与曾老师渊博的学识、严谨的治学态度和正直的为人风格令我敬佩，为我树立了良好的学术榜样，在教会我如何为人处事的同时，也让我重新认识了自己的人生轨迹。在论文的完成过程中，老师们不厌其烦地一次次地帮我修改，给出了很多论文写作的不足之处，在此我向方老师与曾老师致以崇高的敬意，十分感谢两位恩师！同时我还要感谢我的校外指导老师黄平老师，感谢黄老师在工作之余还认真负责的不断对我的论文提出宝贵的指导意见，黄老师这种认真负责的态度不管是在科研上还是工作上都值得我去认真学习。

感谢成都理工大学信息科学与技术学院的领导和老师们！他们通过授课、讲座的形式为我们传授专业知识，丰富了我的理论体系。

在此，我要感谢我的师兄们，感谢师兄们对我硕士研究生期间的指导以及帮助，是他们帮助我快速的走上自己方向领域的道路；同时也感谢工作室的同学们以及与我同导师的四位同窗，感谢他们在我需要帮助的时候慷慨帮助，也是他们让我三年的学习生活不那么枯燥。还要特别感谢罗梅同学在我迷茫之际频繁给予的思想上的开导以及时刻的监督，能让我时刻注意提高自己的科研水平及自身能力。

在此，还要感谢杭州迅蚁科技有限公司嵌入式部门的同事们，是他们在我的实习期间的指导才让我不断提高自己逐渐掌握毕业论文相关领域的知识；最后我想感谢我的家人，感谢他们在我读研过程中给予我莫大的鼓励与支持。

2020 年是特殊的一年，一场不可抗拒的全球性新冠疫情首先爆发于我国，受此次疫情影响我们未能正常返校，这在一定程度上影响了毕业论文的实验数据获取和实验验证，研究中部分工作仍待后续完成。因此我还要感谢各位评审老师，为了我们这些毕业生能够顺利毕业，在这一特殊时期你们仍然肩负起了这份责任，感谢你们！

参考文献

- 曹美会, 鲜斌, 张旭, 等. 2015. 基于视觉的四旋翼无人机自主定位与控制系统[J]. 信息与控制, 44(2): 190-196, 202.
- 何梓溢. 2014. 基于 FPGA 的高动态摄像机的设计[D]. 吉林大学.
- 侯永宏, 刘艳, 吕华龙, 吴琦, 赵健, 陈艳芳. 2019. 一种基于双目视觉的无人机自主导航系统[J]. 天津大学学报(自然科学与工程技术版), 52(12): 1262-1269.
- 侯宾, 张文志, 戴源成, 田洪强. 2014. 基于 OpenCV 的目标物体颜色及轮廓的识别方法[J]. 现代电子技术, 37(24): 76-79+83.
- 金伟, 周震博. 2019. 2018 年无人机研发热点回眸[J]. 科技导报, 37(01): 163-172.
- 江士雄. 2016. 面向位姿估计的相机系统标定方法研究[D]. 华中科技大学.
- 李昶, 程锦霞, 杨光, 于江. 5G+无人机的低空数字化发展与应用[J]. 移动通信, 2019, 43(09): 47-52.
- 李宝芸, 范玉刚, 高阳. 2019. 基于 OTSU 和 Canny 算子的红外图像特征提取[J]. 陕西理工大学学报(自然科学版), 35(06): 33-40.
- 刘乃军, 鲁涛, 席宝, 等. 2018. 基于 ROS 的 UR 机器人遥操作系统设计[J]. 兵工自动化, 37(3): 88-90.
- 刘艳, 李腾飞. 2014. 对张正友相机标定法的改进研究[J]. 光学技术, 40(06): 565-570.
- 卢晓冬, 薛俊鹏, 张启灿. 2020. 基于圆心真实图像坐标计算的高精度相机标定方法[J]. 中国激光, 47(3): 0304008.
- 孟建军, 程思柳, 李德仓. 2019. 基于形态学处理的轨道扣件定位算法研究[J]. 计算机仿真, 36(11): 105-109+170.
- 邱鹏瑞, 刘筠, 刘聪. 2019. 四旋翼无人机单目视觉自主着陆系统研究[J]. 云南民族大学学报(自然科学版), 28(03): 289-292.
- 索文凯, 胡文刚, 伍锡山, 张彪. 2019. 基于光学视觉辅助无人机自主降落研究综述[J]. 激光杂志, 40(04): 9-13.
- 王谭, 王磊磊, 张卫国, 段晓韬, 王婉丽. 2019. 基于张正友标定法的红外靶标系统[J]. 光学精密工程, 27(08): 1828-1835.
- 韦星, 马智愚. 2018. 一种基于 OpenCV 的彩色图像滤波算法设计[J]. 电子科学技术, 2017, 04(04): 91-95.
- 汪成龙, 陈国壮, 马秋立, 方浩群, 何宣, 陈泽涛, 孙培宜. 2018. 基于 OpenCV 的电线颜色识别系统软件[J]. 电子测量技术, 41(16): 91-97.
- 徐晓苏, 岳增阳, 张涛, 汤郡郡. 2015. 基于 Hu 矩的水下地形二维特征匹配辅助导航方法[J]. 中国惯性技术学报, 23(03): 363-368.
- 杨小川, 刘刚, 王运涛, 孙岩, 孟德虹, 李伟. 2018. Pixhawk 开源飞控项目概述及其航空应用展望[J]. 飞航导弹, (04): 25-32+76.

- 曾聪. 2018. 基于视觉反馈的微型四旋翼飞行器自主着陆系统研究及应用[D]. 武汉大学.
- 张咪, 赵勇, 布树辉, 张臻炜, 杨君. 2018. 基于阶层标识的无人机自主精准降落系统[J]. 航空学报, 39(10):213-221.
- 张舸, 伊国兴, 高翔. 2018. 基于视觉导航的旋翼无人机自主降落系统[J]. 传感器与微系统, 37(09):77-80+83.
- Atoev S, Kwon K R, Lee S H, et al. 2017. Data analysis of the MAVLink communication protocol[C]//2017 International Conference on Information Science and Communications Technologies (ICISCT). IEEE, 1-3.
- Carvalho J P, Jucá M A, Menezes A, et al. 2017. Autonomous UAV outdoor flight controlled by an embedded system using Odroid and ROS[M]//CONTROLO 2016. Springer, Cham, 423-437.
- Domin K, Symeonidis I, Marin E. 2016. Security analysis of the drone communication protocol: Fuzzing the MAVLink protocol[J].
- Deng C X, Wang G B, Yang X R. 2013. Image edge detection algorithm based on improved canny operator[C]//2013 International Conference on Wavelet Analysis and Pattern Recognition. IEEE, 168-172.
- Dordrecht. 2015. Handbook of unmanned aerial vehicles[M]. Springer Netherlands.
- E. Ebeid, M. Skriver and J. Jin, 2017. "A Survey on Open-Source Flight Control Platforms of Unmanned Aerial Vehicle," 2017 Euromicro Conference on Digital System Design (DSD), Vienna, pp. 396-402.
- Feng L, Fangchao Q. 2016. Research on the Hardware Structure Characteristics and EKF Filtering Algorithm of the Autopilot PIXHAWK[C]//2016 Sixth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC). IEEE, 228-231.
- Fernandez E, Crespo L S, Mahtani A, et al. 2015. Learning ROS for robotics programming[M]. Packt Publishing Ltd.
- Fatma M, Sharma J. 2014. Leukemia image segmentation using K-means clustering and HSI color image segmentation[J]. Int. J. Comput. Appli, 94: 6-9.
- García G B, Suarez O D, Aranda J L E, et al. 2015. Learning image processing with opencv[M]. Packt Publishing Ltd.
- John Wiley & Sons, 2013. Unmanned aerial vehicles: Embedded control[M].
- Jain A, Gupta R. Gaussian filter threshold modulation for filtering flat and texture area of an image[C]//2015 International Conference on Advances in Computer Engineering and Applications. IEEE, 2015: 760-763.
- Kolkur S, Kalbande D, Shimpi P, et al. 2017. Human skin detection using RGB, HSV and YCbCr color models[J]. arXiv preprint arXiv:1708.02694.

- K. Dang Nguyen and T. Nguyen, 2019. "Vision-Based Software-in-the-Loop-Simulation for Unmanned Aerial Vehicles Using Gazebo and PX4 Open Source," *2019 International Conference on System Science and Engineering (ICSSE)*, Dong Hoi, Vietnam, pp. 429-432.
- Lee H, Jung S, Shim D H. 2016. Vision-based UAV landing on the moving vehicle[C]//2016 International conference on unmanned aircraft systems (ICUAS). IEEE, 1-7.
- Liu M, Egan G K, Santoso F. 2015. Modeling, autopilot design, and field tuning of a UAV with minimum control surfaces[J]. *IEEE Transactions on Control Systems Technology*, 23(6): 2353-2360.
- Meier L, Honegger D, Pollefeys M. 2015. PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms[C]//2015 IEEE international conference on robotics and automation (ICRA). IEEE, 6235-6240.
- Nguyen K D, Ha C. 2018. Development of hardware-in-the-loop simulation based on Gazebo and Pixhawk for unmanned aerial vehicles[J]. *International Journal of Aeronautical and Space Sciences*, 19(1): 238-249.
- Polvara R, Sharma S, Wan J, et al. 2018. Vision-based autonomous landing of a quadrotor on the perturbed deck of an unmanned surface vehicle[J]. *Drones*, 2(2): 15.
- Raid A M, Khedr W M, El-Dosuky M A, et al. 2014. Image restoration based on morphological operations[J]. *International Journal of Computer Science, Engineering and Information Technology (IJCSEIT)*, 4(3): 9-21.
- Rong W, Li Z, Zhang W, et al. 2014. An improved CANNY edge detection algorithm[C]//2014 IEEE International Conference on Mechatronics and Automation. IEEE, 577-582.
- Saravanan G, Yamuna G, Nandhini S. 2016. Real time implementation of RGB to HSV/HSI/HSL and its reverse color space models[C]//2016 International Conference on Communication and Signal Processing (ICCSP). IEEE, 0462-0466.
- Tianqu Zhao and Hong Jiang, 2016. "Landing system for AR.Drone 2.0 using onboard camera and ROS," 2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC), Nanjing, pp. 1098-1102.
- Taganov A A, Shashev D V. 2019. Simulation of the behavior of an unmanned aerial vehicle in virtual 3D scenes[C]//IOP Conference Series: Materials Science and Engineering. IOP Publishing, 516(1): 012046.
- Yu X, Gao L, He B, et al. 2017. Design and Implementation of Platform of Educational Mobile Robots Based on ROS[C]//International Conference on Mechatronics and Intelligent Robotics. Springer, Cham, 476-482.

攻读学位期间取得学术成果

已发表论文:

丁中涛,方方,冉述,杨剑锋,杨雪,夏凡,郭浩攀.基于无线传感网络的可移动式环境监测系统设计[J].无线电通信技术,2018,44(05):479-482.

夏凡,方方,丁中涛,刘宇为,杨剑锋,杨雪.基于机器学习的 IPFIX 的僵尸通信检测[J].信息技术与网络安全,2019,38(09):50-54.

冉述,曾维,杨剑锋,杨雪,丁中涛,谭浩.基于物联网技术的无线多路温度采集系统设计[J].无线电通信技术,2018,44(05):458-462.

其他研究成果:

曾维,杨昌杰,杨明翰,丁中涛,夏凡,刘宇为,杨雪,冯坤,陈川,任际周.2018-3-23.基于 Leap Motion 的手势操控无人机系统设计:中国,ZL201820399532.0[P].

