

——**自动化专业综合实践** **——基于赛教/科教/产教的三融合方向**

阶段性报告 1

姓名： 李傲琦 学号： 1023009006

题目： 基于 ROS2 系统的自瞄移动机器人

赛教-依托的学科竞赛： 全国大学生电子设计竞赛

目 录

第 1 章 项目背景与需求分析	1
1.1 项目背景	1
1.2 需求分析	1
1.2.1 功能需求	1
1.2.2 性能需求	2
1.3 复杂工程问题分析	2
第 2 章 文献调研与技术现状	3
2.1 移动机器人控制技术	3
2.2 视觉定位技术	3
2.3 云台控制技术	4
2.4 ROS2 机器人操作系统	4
2.5 技术现状总结	4
第 3 章 系统总体设计方案	5
3.1 系统架构设计	5
3.2 硬件选型方案	6
3.2.1 底盘硬件	6
3.2.2 云台硬件	6
3.2.3 视觉硬件	6
3.3 软件架构设计	7
3.3.1 ROS2 节点设计	7
3.3.2 通信协议设计	7
第 4 章 关键技术路线设计	9
4.1 底盘巡线控制方案	9
4.1.1 电机速度环控制	9
4.1.2 底盘角度闭环	9
4.1.3 平均值加权巡线算法	9
4.2 云台三环 PID 控制方案	10
4.2.1 内环：电机速度与电流闭环	10
4.2.2 中环：角度闭环	10
4.2.3 外环：位置闭环	10
4.3 视觉 PnP 与激光位置解算方案	11
4.3.1 离线标定阶段	11

4.3.2 在线解算阶段.....	11
4.4 方案对比与选择依据.....	12
4.4.1 巡线算法对比.....	12
4.4.2 视觉定位算法对比.....	13
第 5 章 方案可行性论证.....	14
5.1 技术可行性.....	14
5.1.1 控制技术成熟度.....	14
5.1.2 视觉算法可行性.....	14
5.1.3 通信架构可行性.....	14
5.2 硬件可行性.....	14
5.2.1 硬件可获得性.....	14
5.2.2 硬件性能匹配.....	14
5.3 开发可行性.....	15
5.3.1 模块化开发.....	15
5.3.2 调试与测试.....	15
5.3.3 技术储备.....	15
5.4 时间可行性.....	15
5.5 风险分析与应对措施.....	15
5.5.1 技术风险.....	15
5.5.2 硬件风险.....	16
5.5.3 进度风险.....	16
第 6 章 下一阶段工作计划.....	17
6.1 硬件选型与采购.....	17
6.2 机械结构搭建.....	17
6.3 底层驱动开发.....	17
6.4 基础通信系统建立.....	17
6.5 阶段二预期成果.....	17
6.6 时间安排.....	18
参考文献.....	19

第 1 章 项目背景与需求分析

1.1 项目背景

随着智能制造与机器人技术的快速发展，具备自主移动与视觉瞄准能力的机器人系统在工业检测、自动化设备、智能物流等领域展现出广泛的应用价值。本项目依托全国大学生电子设计竞赛（赛教融合方向），旨在设计并实现一套基于 ROS2 系统的自瞄移动机器人，该系统能够实现自动巡线移动、激光精准瞄准以及手动/自动模式灵活切换等功能。

如图1.1所示，自瞄移动机器人可应用于工业自动化场景中，通过底盘巡线实现自主移动，利用云台和视觉系统实现对目标的精确瞄准，在智能制造、自动化检测等领域具有重要应用价值。

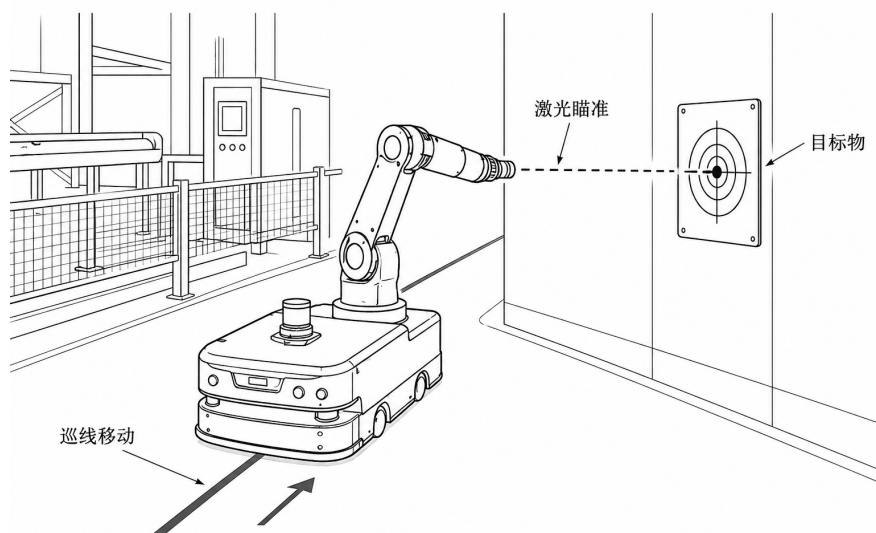


图 1.1 自瞄移动机器人应用场景示意图

1.2 需求分析

1.2.1 功能需求

系统需实现以下核心功能：

1. **自动巡线移动**：机器人能够沿预设黑色轨迹自主行驶，保持稳定跟踪；
2. **激光瞄准目标**：通过视觉定位与云台控制，实现激光点对目标的精确瞄准；
3. **手动/自动模式切换**：支持通过上位机界面实时切换控制模式，满足不同应用场景需求；
4. **实时图像传输**：将摄像头画面实时传输至 Web 端，便于远程监控与操作。

1.2.2 性能需求

系统需满足以下性能指标：

- **定位精度**：激光瞄准精度达到厘米级，满足实际应用要求；
- **控制响应**：系统具备实时性，控制延迟小于 100ms；
- **系统稳定性**：能够连续稳定运行，抗干扰能力强。

1.3 复杂工程问题分析

本项目涉及多个子系统的协同工作，面临以下复杂工程问题：

1. **多系统协同**：底盘控制、云台控制、视觉处理三个子系统需要高效协同，要求合理的架构设计与通信机制；
2. **实时性与计算资源矛盾**：视觉算法计算量大，需在有限计算资源下保证系统实时性；
3. **标定精度影响**：相机内参标定、激光器与相机联合标定的精度直接影响整体系统性能；
4. **控制稳定性**：底盘巡线与云台跟踪均需设计合理的控制策略，保证系统稳定运行。

第 2 章 文献调研与技术现状

2.1 移动机器人控制技术

移动机器人控制技术的实现是自主导航的核心。目前主流的控制方法包括 PID 控制、模糊控制、滑模控制等。其中，PID 控制因其结构简单、鲁棒性强、易于实现等优点，在移动机器人领域得到广泛应用^[1]。对于巡线机器人，常采用串级 PID 结构，外环为路径跟踪环，内环为速度控制环，能够有效提高系统的动态性能和稳定性^[2]。

表2.1对比了不同移动机器人控制技术的特点。从表中可以看出，PID 控制在实现复杂度、计算开销和实时性方面具有明显优势，而模糊控制和滑模控制虽然在鲁棒性方面表现更好，但实现难度较高。综合考虑本项目的实际需求，选择 PID 控制作为主要控制策略是合理的。

表 2.1 控制方法对比

控制方法	优点	缺点	适用场景
PID 控制	结构简单，易于实现； 理论成熟，工程应用广泛； 对线性系统效果好	参数整定依赖经验； 对非线性和强耦合系统适应性差； 抗干扰能力有限	线性系统、低复杂度控制场景，如基础巡线控制
模糊控制	不依赖精确数学模型； 对非线性系统适应能力强； 鲁棒性较好	规则设计依赖经验； 难以保证全局最优； 计算复杂度较高	不确定性较强的系统，如复杂环境下的路径跟踪
滑模控制	鲁棒性强，抗干扰能力强； 对参数变化不敏感； 收敛速度快	存在抖振问题； 实现较复杂； 对执行器有一定要求	高精度、高鲁棒性需求场景，如激光瞄准控制

2.2 视觉定位技术

视觉定位是实现精确瞄准的关键技术。PnP（Perspective-n-Point）算法是计算机视觉领域的经典位姿估计方法，通过已知的 3D 点与其对应的 2D 图像投影点，求解相机的位姿矩阵^[3]。该算法在机器人视觉伺服、增强现实、无人机定位等领域应用广泛。OpenCV 库提供了多种 PnP 求解方法^[4]，包括 ITERATIVE、P3P、EPNP 等，能够满足不同精度和速度要求。

2.3 云台控制技术

云台控制系统通常采用多环串级 PID 结构^[1]。电机内部实现电流环和速度环，主控芯片实现角度环和位置环，形成四环控制结构。这种结构能够有效抑制外部扰动，提高系统的快速性和稳定性。在实际应用中，常结合卡尔曼滤波、低通滤波等方法对传感器数据进行处理^[5]，进一步提升控制性能。

2.4 ROS2 机器人操作系统

ROS2 (Robot Operating System 2) 是新一代机器人操作系统，相比 ROS1 具有更好的实时性、安全性和跨平台能力^[6]。ROS2 采用 DDS (Data Distribution Service) 作为底层通信中间件，支持多种 QoS (Quality of Service) 策略，能够满足不同应用场景的通信需求。ROS2 的节点化架构便于系统模块化开发与维护，已成为机器人领域的主流开发平台。

2.5 技术现状总结

通过文献调研可以发现，视觉伺服与闭环控制相结合是当前移动机器人领域的主流技术方向^[7]。PID 控制因其简单可靠被广泛采用^[1]，PnP 算法在视觉定位中应用成熟^[3]，串级 PID 结构在云台控制中效果良好。基于 ROS2 的分布式架构能够有效支撑多模块协同工作^[6]。这些成熟技术为本项目的实施提供了坚实的理论基础和技术支撑。

第3章 系统总体设计方案

3.1 系统架构设计

本系统采用多主控分布式架构，基于 ROS2 框架实现节点化通信。系统划分为四个主要模块：

1. **底盘控制模块**：负责电机驱动、巡线控制、底盘运动控制；
2. **云台控制模块**：负责云台姿态控制、激光瞄准控制；
3. **视觉处理模块**：负责图像采集、目标识别、位姿解算、激光落点预测；
4. **上位机交互模块**：负责图像传输、远程控制、模式切换。

如图3.1所示，系统采用分层分布式架构，各模块通过 ROS2 的话题（Topic）和服务（Service）机制进行通信，实现数据共享与功能调用。底盘主控与云台主控通过 UART 与 Jetson 通信，上位机通过 Wi-Fi 与 Jetson 连接。这种架构设计具有模块化程度高、易于开发调试、系统扩展性强等优点。

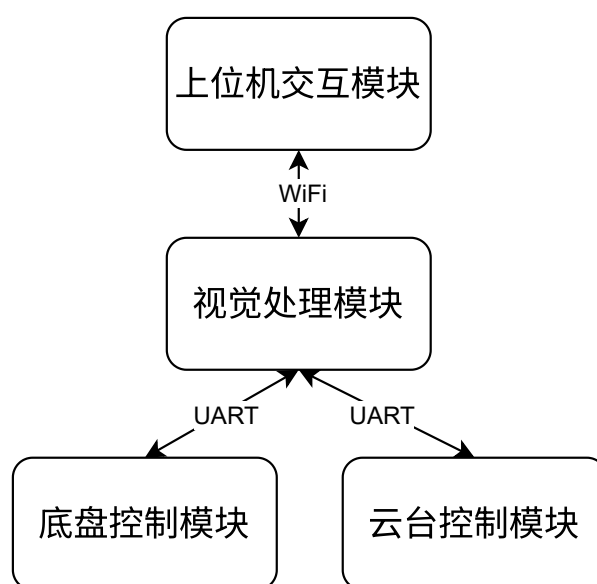


图 3.1 系统总体架构图

3.2 硬件选型方案

3.2.1 底盘硬件

- **主控芯片**：TI MSPM0G3507，32 位 ARM Cortex-M0+ 内核，主频 80MHz；
- **驱动电机**：MG513X 直流电机（带霍尔编码器），额定电压 12V；
- **电机驱动**：DRV8701E 双路电驱模块，支持 PWM 调速；
- **巡线传感器**：感为 8 路模拟量寻迹模块，灰度检测；
- **姿态传感器**：ICM-42688 六轴 IMU，用于底盘角度闭环；
- **机械结构**：亚克力底板，前轮万向轮，后轮驱动轮。

3.2.2 云台硬件

- **主控芯片**：STM32H723VGT6，32 位 ARM Cortex-M7 内核，主频 550MHz；
- **Yaw 轴电机**：DM-G6220 中空电机，驱控一体，内置速度闭环；
- **Pitch 轴电机**：DM-J4310-2EC 关节电机，驱控一体；
- **姿态传感器**：BMI088 六轴 IMU，用于云台角度闭环；
- **通信接口**：CAN 总线连接电机，UART 连接 Jetson；
- **机械结构**：3D 打印轻量化框架。

3.2.3 视觉硬件

- **计算平台**：Jetson Orin Nano Super，8 核 ARM Cortex-A78AE，1024 核 CUDA；
- **相机**：海康全局快门工业相机，分辨率 1920×1200；
- **镜头**：工业无畸变镜头，焦距 8mm；
- **激光器**：红色激光指示器，波长 650nm。

图3.2展示了系统的硬件组成及各模块之间的连接关系。从图中可以看出，Jetson 作为核心计算平台，通过 UART 与底盘主控、云台主控通信，通过 USB 连接工业相机，通过 Wi-Fi 与上位机通信。各硬件模块分工明确，接口清晰，便于系统集成与调试。

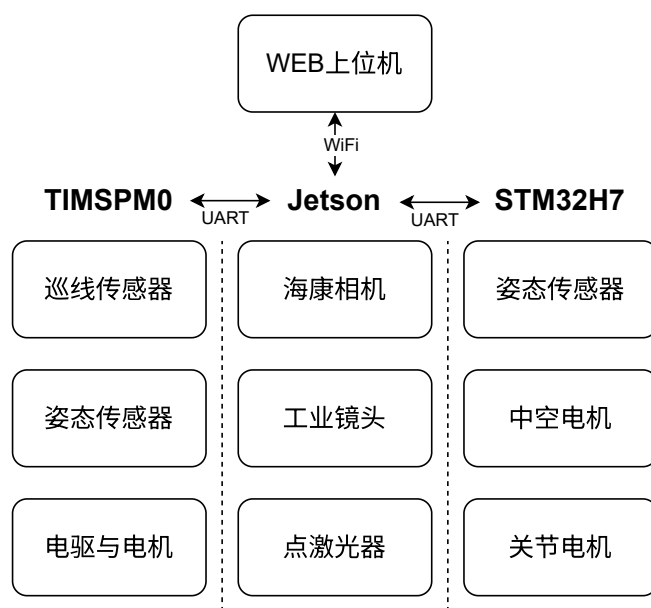


图 3.2 系统硬件组成示意图

3.3 软件架构设计

3.3.1 ROS2 节点设计

系统设计以下 ROS2 节点：

- **camera_driver 节点**：负责海康相机驱动与图像采集；
- **vision_node 节点**：进行目标识别、PnP 解算、激光落点预测；
- **web_server 节点**：基于 Vue 框架的 Web 服务器，提供图传与控制界面；
- **gimbal_control 节点**：云台控制节点，接收视觉误差或手动指令；
- **chassis_control 节点**：底盘控制节点，执行巡线或手动控制。

3.3.2 通信协议设计

- **ROS2 话题通信**：节点间通过话题发布/订阅机制传递数据；
- **UART 串口通信**：Jetson 与底盘主控、云台主控通过 UART 传输控制指令与反馈数据；
- **CAN 总线通信**：云台主控通过 CAN 总线控制电机；
- **Wi-Fi 网络通信**：上位机通过 Wi-Fi 与 Jetson 建立 WebSocket 连接。

如图3.3所示，软件系统采用 ROS2 节点化架构，各节点通过话题机制进行数据交互。camera_driver 节点发布图像数据,vision_node 节点订阅图像并发布视觉误差,gimbal_control

节点和 chassis_control 节点分别订阅控制指令并执行相应动作。这种松耦合的架构设计使得各模块可以独立开发和测试，提高了系统的可维护性和可扩展性。

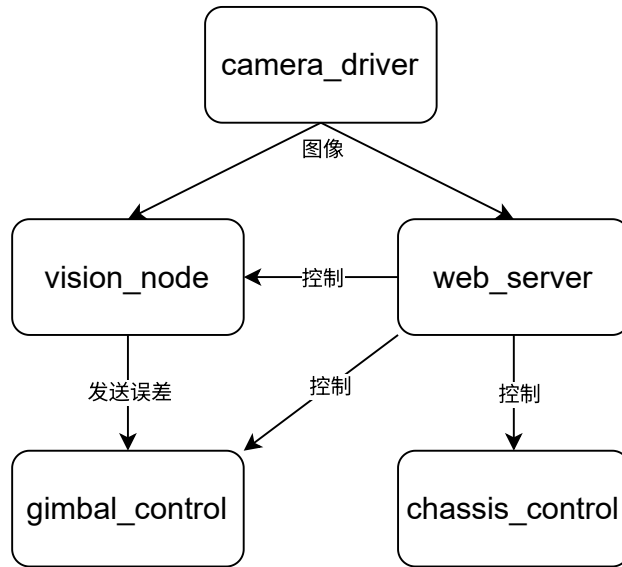


图 3.3 软件架构示意图

第4章 关键技术路线设计

4.1 底盘巡线控制方案

底盘采用三环串级控制结构，由内到外依次为速度环、角度环、巡线环。

4.1.1 电机速度环控制

通过 MSPM0 主控读取 MG513X 电机的霍尔编码器脉冲，计算实时转速。采用增量式 PID 算法^[1]，将编码器反馈的实时转速与目标转速进行比对，输出 PWM 波控制 DRV8701E 驱动模块。速度环 PID 参数通过实验整定，保证快速响应与稳定性。

4.1.2 底盘角度闭环

利用 ICM-42688 陀螺仪获取底盘的实时偏航角 (Yaw)。角度环作为速度环的外环，用于补偿机械差异或地面摩擦导致的偏移，确保底盘具备稳定的直线行驶与精确转向能力。角度环输出作为左右轮速度差的目标值。

4.1.3 平均值加权巡线算法

实时读取 8 路寻迹模块的模拟量数值，设置位置索引为 $\{-4, -3, -2, -1, 1, 2, 3, 4\}$ 。利用公式^[2]

$$Error = \frac{\sum_{i=1}^8 (V_i \times Index_i)}{\sum_{i=1}^8 V_i} \quad (4.1)$$

计算出底盘中心线与黑色轨迹的实时偏差，其中 V_i 为第 i 路传感器的模拟量值， $Index_i$ 为对应的位置索引。将计算出的偏差 $Error$ 经由 PID 控制器计算，输出作为角度环的目标增量，从而实现平滑的自动巡线效果。

图4.1展示了底盘三环控制的完整结构。从图中可以看出，巡线环作为最外环，根据 8 路传感器数据计算偏差；角度环作为中环，利用 IMU 反馈实现航向控制；速度环作为内环，通过编码器反馈实现精确的速度控制。三环串级结构能够有效提高系统的动态性能和抗干扰能力。

4.2 云台三环 PID 控制方案

4.2.1 内环：电机速度与电流闭环

4.2.2 中环：角度闭环

4.2.3 外环：位置闭环

如图4.2所示，云台控制系统采用三环串级 PID 架构。位置环接收视觉误差作为输入，输出目标角度；角度环读取 IMU 反馈，输出目标速度；速度环由电机内部实现。这种多环串级结构能够充分发挥各环节的优势，内环保证快速响应，外环保证精确跟踪，整体系统具有良好的动态性能和稳态精度。

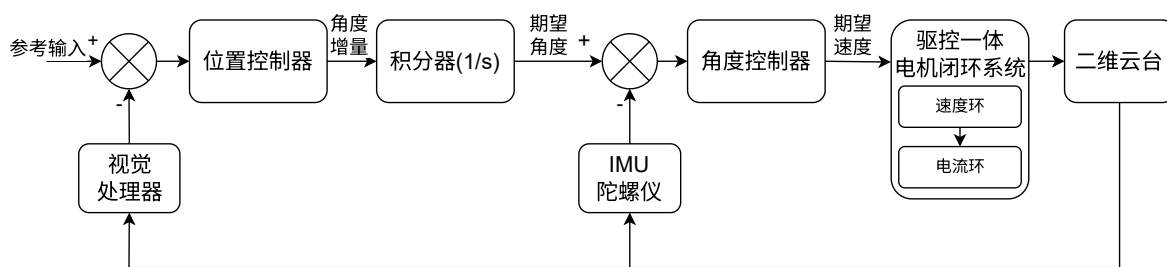


图 4.2 云台三环 PID 控制结构框图

4.3 视觉 PnP 与激光位置解算方案

本方案不依赖画面中激光点的色块识别，而是基于预先标定的”相机-激光器”刚体关系进行模型解算。

4.3.1 离线标定阶段

1. **相机内参标定：**使用棋盘格标定板，通过张正友标定法^[4]获取相机内参矩阵 K 和畸变系数；
2. **激光器与相机联合标定：**
 - 开启激光器，将激光点投射在标定板上，保持相机与激光器相对静止，改变整体姿态，拍摄多组图像；
 - 对每张图像使用 PnP 算法^[3]计算相机外参，提取激光点像素坐标，利用反投影求交计算激光点在相机坐标系下的三维坐标；
 - 使用最小二乘法或 PCA 拟合激光直线方程，得到相机坐标系下激光束的方向向量 $\vec{d}_{cam}$ 和起点 O_{cam} 。

4.3.2 在线解算阶段

1. **获取当前相机位姿：**拍摄包含目标矩形的图像，根据矩形特征点进行 PnP 解算^[3]，获得当前相机位姿矩阵（旋转矩阵 R 和平移向量 t ）；
2. **坐标变换至世界坐标系：**

$$\vec{d}_{world} = R^T \cdot \vec{d}_{cam} \quad (4.2)$$

$$O_{world} = R^T \cdot (O_{cam} - t) \quad (4.3)$$

3. **求解激光点世界坐标：**构建世界坐标系下的激光射线方程 $L(s) = O_{world} + s \cdot \vec{d}_{world}$ ，已知目标平面满足 $Z_{world} = 0$ ，联立求解得到激光点在靶面上的精确坐标 $(x, y, 0)$ 。

图4.3展示了视觉 PnP 与激光位置解算的完整流程。整个过程分为离线标定和在线解算两个阶段：离线标定阶段通过多组标定图像拟合出激光束在相机坐标系下的空间直线方程；

在线解算阶段通过 PnP 算法获取相机位姿，将激光直线变换到世界坐标系，最后与目标平面求交得到激光落点。这种方法不依赖激光点的图像识别，避免了光照变化、背景干扰等因素的影响，具有更高的鲁棒性和精度。

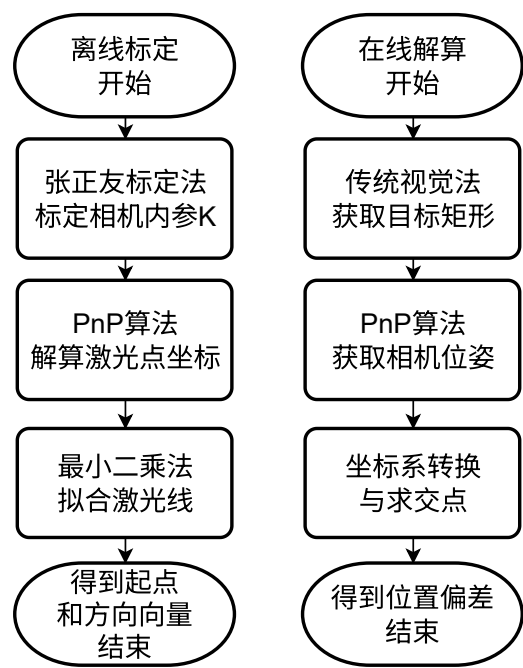


图 4.3 视觉 PnP 与激光位置解算流程图

4.4 方案对比与选择依据

4.4.1 巡线算法对比

表 4.1 巡线算法对比

算法	计算复杂度	抗干扰能力	适用场景
重心法	低	弱	简单轨迹
加权平均法	中	中	一般轨迹
最小二乘拟合	高	强	复杂轨迹

常用巡线算法如表4.1所示，本项目选择加权平均法，原因是：计算量适中，能够满足实时性要求；抗干扰能力较好，适合比赛场地环境；实现简单，便于参数调试。

4.4.2 视觉定位算法对比

表 4.2 视觉定位算法对比

算法	精度	实时性	标定复杂度
色块识别	低	高	低
PnP 位姿估计	高	中	中
SLAM 建图定位	高	低	高

常用视觉定位算法如表4.2所示，本项目选择 PnP 位姿估计结合激光模型解算，原因是：定位精度高，能够达到厘米级；实时性满足要求，单帧处理时间小于 50ms；标定流程清晰，可操作性强。

第 5 章 方案可行性论证

5.1 技术可行性

5.1.1 控制技术成熟度

PID 控制算法是经典的控制方法，理论成熟，工程应用广泛。串级 PID 结构在移动机器人和云台控制中已有大量成功案例。MSPM0 和 STM32 系列微控制器性能强大，完全能够满足实时控制需求。因此，底盘巡线控制和云台三环控制在技术上完全可行。

5.1.2 视觉算法可行性

PnP 算法是计算机视觉领域的经典算法，OpenCV 库提供了成熟的实现。激光器与相机联合标定方法在工业测量领域已有应用。Jetson Orin Nano Super 具备强大的计算能力，能够实时运行视觉算法。因此，视觉定位与激光落点解算在技术上可行。

5.1.3 通信架构可行性

ROS2 是成熟的机器人操作系统，支持多种通信机制和 QoS 策略。UART、CAN、Wi-Fi 等通信协议均为标准协议，硬件支持完善。分布式架构便于模块化开发与调试。因此，系统通信架构在技术上可行。

5.2 硬件可行性

5.2.1 硬件可获得性

所选硬件均为市场上常见的工业级或消费级产品，供货稳定，价格合理。MSPM0、STM32、Jetson 等主控芯片均有完善的开发工具链和技术支持。电机、传感器、相机等外设均可通过正规渠道采购。因此，硬件采购不存在障碍。

5.2.2 硬件性能匹配

MSPM0 主频 80MHz，能够满足底盘控制的实时性要求；STM32H7 主频 550MHz，能够处理云台控制的高频数据；Jetson Orin Nano Super 具备 1024 个 CUDA 核心，能够实时运行视觉算法。各硬件性能与任务需求匹配良好。

5.3 开发可行性

5.3.1 模块化开发

系统采用分布式架构，各模块功能独立，接口清晰。可以并行开发底盘控制、云台控制、视觉处理等模块，最后进行系统集成。这种开发方式能够有效缩短开发周期，降低开发风险。

5.3.2 调试与测试

各模块可以独立调试，降低了系统调试的复杂度。底盘可以先实现基本运动控制，再加入巡线功能；云台可以先实现角度控制，再加入视觉伺服；视觉可以先实现目标检测，再加入位姿解算。分步实施，逐步完善。

5.3.3 技术储备

团队成员具备嵌入式开发、ROS2 编程、计算机视觉等相关技术背景，能够胜任各模块的开发工作。同时，相关技术资料丰富，开源社区活跃，遇到问题可以快速获得技术支持。

5.4 时间可行性

根据项目计划，整个开发周期分为三个阶段：

1. **阶段一（当前阶段）**：完成需求分析、技术调研、方案设计，预计用时 2 周；
2. **阶段二**：完成硬件搭建、驱动开发、基础功能实现，预计用时 3 周；
3. **阶段三**：完成系统集成、功能测试、性能优化，预计用时 2 周。

总计 7 周的开发周期是合理的，能够保证各阶段任务的顺利完成。

5.5 风险分析与应对措施

5.5.1 技术风险

- **风险**：视觉算法精度不足，影响瞄准效果；
- **应对**：提高标定精度，优化算法参数，必要时引入滤波算法。
- **风险**：控制系统不稳定，出现震荡或失控；
- **应对**：仔细整定 PID 参数，加入限幅和滤波环节，进行充分测试。

5.5.2 硬件风险

- **风险：**硬件损坏或性能不达标；
- **应对：**采购时选择可靠供应商，提前预算电流电压等参数，预留备件，做好硬件保护措施。

5.5.3 进度风险

- **风险：**开发进度延误，影响项目交付；
- **应对：**制定详细的开发计划，定期检查进度，及时调整资源分配。

综上所述，本项目在技术、硬件、开发、时间等方面均具备可行性，风险可控，能够按计划完成项目目标。

第 6 章 下一阶段工作计划

根据项目整体规划，第一阶段已完成需求分析、技术调研和方案设计工作。下一阶段（阶段二）将进入系统实现阶段，主要工作内容如下：

6.1 硬件选型与采购

1. 根据设计方案，确定各硬件模块的具体型号和规格；
2. 编制硬件采购清单，包括主控芯片、电机、传感器、相机等；
3. 联系供应商，完成硬件采购，预计用时 1 周。

6.2 机械结构搭建

1. 设计底盘机械结构，确定电机、传感器、电池的安装位置；
2. 设计云台机械结构，完成 3D 建模，进行 3D 打印；
3. 组装底盘与云台，完成整体机械结构搭建，与硬件选型与采购同时期进行，预计用时 1 周。

6.3 底层驱动开发

1. 开发 MSPM0 底盘主控程序，实现电机驱动、编码器读取、IMU 数据采集；
2. 开发 STM32 云台主控程序，实现 CAN 通信、IMU 数据采集、电机控制；
3. 开发 Jetson 相机驱动程序，实现图像采集与 ROS2 话题发布；
4. 完成各模块的基础功能测试，预计用时 1 周。

6.4 基础通信系统建立

1. 搭建 ROS2 通信框架，创建各功能节点；
2. 实现 Jetson 与底盘主控、云台主控的 UART 通信；
3. 实现上位机与 Jetson 的 Wi-Fi 通信；
4. 完成通信链路测试，验证数据传输的正确性和实时性，预计用时 1 周。

6.5 阶段二预期成果

完成阶段二工作后，预期达到以下成果：

- 完成系统硬件搭建，各硬件模块能够正常工作；
- 完成底层驱动开发，能够读取传感器数据，控制电机运动；
- 完成基础通信系统，各模块能够进行数据交互；

- 底盘能够实现基本的前进、后退、转向等运动；
- 云台能够响应控制指令，实现基本的姿态调整；
- 视觉系统能够采集图像并通过 ROS2 话题发布。

6.6 时间安排

阶段二预计用时 3 周，具体时间安排如下：

- 第 1 周：硬件采购与机械结构设计；
- 第 2 周：机械结构搭建与底层驱动开发；
- 第 3 周：通信系统建立与联调测试。

通过阶段二的工作，将为阶段三的功能完善和性能优化奠定坚实基础。

参考文献

- [1] ANG K H, CHONG G, LI Y. PID Control System Analysis, Design, and Technology[J]. IEEE Transactions on Control Systems Technology, 2005, 13(4): 559-576.
- [2] ISMAIL A H, RAMLI H R, AHMAD M H, et al. Vision-based System for Line Following Mobile Robot[J]., 2009: 642-646. DOI: 10.1109/ISIEA.2009.5356366.
- [3] LEPETIT V, MORENO-NOGUER F, FUA P. EPnP: An Accurate $O(n)$ Solution to the PnP Problem[J]. International Journal of Computer Vision, 2009, 81(2): 155-166. DOI: 10.1007/s11263-008-0152-6.
- [4] BRADSKI G, KAEHLER A. Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library[M]. O'Reilly Media, 2017.
- [5] THRUN S, BURGARD W, FOX D. Probabilistic Robotics[M]. MIT Press, 2005.
- [6] MARUYAMA Y, KATO S, AZUMI T. Design and Implementation of ROS2 for Real-Time Systems[J]. IEEE Robotics and Automation Letters, 2016, 1(2): 1070-1077.
- [7] SARIPALLI S, MONTGOMERY J F, SUKHATME G S. Vision-Based Autonomous Landing of an Unmanned Aerial Vehicle[C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). 2002: 2799-2804.