

武汉理工大学

硕士专业学位论文

基于 STM32 智能小车的设计与实现

论文作者： 方国贤

指导教师： 徐震

专业学位类别： 工程硕士

专业学位领域： 计算机技术

所在院系： 电气与电子工程学院

2018 年 12 月

分类号 _____

密 级 _____

U D C _____

学校代码 10496

武汉理工大学
硕士专业学位论文

基于 STM32 智能小车的设计与实现

Design and implementation of intelligent car
based on STM32

论文作者 方国贤

指导教师 徐震

学位类别 工程硕士

专业领域 计算机技术

论文提交日期 2018.12.10 论文答辩日期 2018.12.11

答辩委员会主席 郑林 评阅人 _____

2018 年 12 月

摘要

近年来，无线网络通信技术、无线视频监控技术、图像采集技术以及嵌入式技术日臻成熟，人们对智能化产品的需求随之不断增加。智能小车采用无线网络通信，相比传统的通信方式，具有布线简单、成本低等优势，能够应用在一些复杂的环境中。

本文设计了一款以 STM32 系列开发板作为核心控制器， μ C/OS-III 操作系统为软件平台的具有自动避障功能的智能小车。以 STM32 主控制器来对直流电机进行驱动，驱动车载系统上的摄像头对图像信息进行采集，控制小车的运行；以 WIFI 信号为媒介，采集到的视频信息实时地传输到 Android 手机或者 PC 上位机显示；而实现手机或者 PC 上位机控制小车的关键在于它们可以将相应的控制信号准确地传递到主控制器。

系统的硬件设计与软件设计共同构成了总体设计。其中，软件设计是 μ C/OS-III 操作系统的移植、硬件驱动程序设计、自动避障算法、PC 上位机和 Android 手机控制端的设计。硬件设计主要是设计系统中的电路，具体包括超声波避障模块设计、视频采集模块设计、电机驱动模块设计、无线路由器模块设计和 STM32 最小系统模块设计等几部分。在自动避障算法中，对目标障碍物的确定需要借助区域生长的方法，并且以图像采集模块采集到的周围环境图像信息为依据。根据障碍物与智能小车之间位置关系提出了基于固定区域分割思想的避障算法。智能小车的软硬件系统测试结果表明，小车的各项功能达到了预期设定的目标，系统的运行状况良好。本系统的实现对于防盗系统以及智能家居今后的发展具有重要意义。

关键词：智能小车；机器视觉； μ C/OS-III 操作系统；避障算法

ABSTRACT

With the rapid development of embedded technology, video image acquisition technology, wireless video surveillance technology and wireless network communication technology, the demand for intelligent products is increasing. The smart car adopts wireless network communication, which has the advantages of simple wiring and low cost compared with the traditional communication method, and can be applied in some complicated environments.

This paper designs a smart car with automatic obstacle avoidance function with STM32 series development board as the core controller and μ C/OS-III operating system as the software platform. The STM32 main controller is used to drive the car DC motor, controlling the actual operation of the car and driving the camera to collect video information; the video information collected by the camera on the car system will be transmitted to the PC host computer or Android mobile phone interface through the WIFI signal for real-time. The display of the PC host computer or mobile phone can transmit the control signal sent to the main controller, thereby achieving the purpose of controlling the car by the PC host computer or the mobile phone.

The hardware design and software design of the system together constitute the overall design. Among them, the software design is the transplantation of μ C/OS-III operating system, hardware driver design, automatic obstacle avoidance algorithm, PC host computer and Android mobile phone control terminal design. The hardware design is mainly to design the circuit in the system. Specifically, it includes ultrasonic obstacle avoidance module design, video capture module design, motor drive module design, wireless router module design and STM32 minimum system module design. In the automatic obstacle avoidance algorithm, the image acquisition module is used to acquire the road environment image information, and the region growth method is used to extract the target obstacle. According to the positional relationship between smart cars and obstacles, an obstacle avoidance algorithm based on fixed region segmentation is proposed. The test results of the software and hardware system of the smart car show that the functions of the car have reached the expected target, and the system can work correctly. The realization of this system is of great significance for the anti-theft system and the future development of smart home.

Keywords: smart car; Machine vision; μ C/OS-III operating system; Obstacle avoidance algorithm

目 录

摘要	II
ABSTRACT	III
1 绪论	1
1.1 智能小车研究背景和意义	1
1.2 智能小车国内外发展及研究现状	2
1.2.1 智能小车国外发展及研究现状	2
1.2.2 智能小车国内发展及研究现状	3
1.3 论文主要工作及章节安排	4
2 智能小车系统的总体设计	6
2.1 系统的总体设计方案	6
2.1.1 智能小车系统的总体设计思路	6
2.1.2 智能小车系统的总体设计框图	6
2.1.3 设计的可行性分析	7
2.2 智能小车系统开发环境	8
2.2.1 硬件开发环境	8
2.2.2 软件开发环境	8
2.3 WIFI 技术	9
2.3.1 WIFI 的基本工作原理	9
2.3.2 WIFI 技术的特点	9
2.4 本章小结	10
3 智能小车系统的硬件设计	11
3.1 系统硬件设计	11
3.2 系统硬件模块选型	11
3.2.1 微处理器模块选型	11
3.2.2 电机驱动模块选型	12
3.2.3 视频采集模块选型	13
3.3 无线路由器的设计	14
3.4 STM32 的最小系统	14
3.4.1 电源电路	14
3.4.2 晶振电路	16
3.4.3 SWD 接口电路	16
3.4.4 串口通信电路	17
3.5 本章小结	18

4 智能小车避障系统的设计与实现	19
4.1 避障流程设计	19
4.2 基于固定区域分割原理的避障算法	20
4.3 机器视觉的图像采集及处理	21
4.3.1 图像的获取和数字化	21
4.3.2 图像预处理技术	21
4.4 障碍自动检测	23
4.4.1 障碍物特征参数获取	23
4.4.2 多个障碍物的处理	25
4.4.3 障碍物区域确定	27
4.5 本章小结	28
5 智能小车系统的软件设计	29
5.1 上位机应用程序设计	29
5.1.1 APP 程序设计概述	29
5.1.2 APP 基本文件解析	30
5.1.3 APP 的 UI 设计	31
5.1.4 APP 的主要程序描述	31
5.2 Android 手机应用程序设计	33
5.3 操作系统的移植	33
5.3.1 μC/OS-III 移植的条件	33
5.3.2 μC/OS-III 移植的主要工作	34
5.3.3 μC/OS-III 的文件移植	34
5.4 硬件驱动程序设计	36
5.4.1 无线数据传输程序设计	36
5.4.2 电机驱动模块程序设计	37
5.4.3 视频采集程序设计	38
5.5 本章小结	40
6 系统测试	41
6.1 系统硬件检测	41
6.1.1 电机驱动的测试	41
6.1.2 无线路由器的测试	42
6.2 系统软件测试	44
6.2.1 PC 上位机的测试	45
6.2.2 Android 手机端的测试	46
6.3 自动避障功能测试	47
6.3.1 避障进程以及躲避障碍物后运行轨迹的复原	48

6.3.2 避障实验测试	48
6.4 本章小结	51
7 总结与展望	52
7.1 总结	52
7.2 展望	52
参考文献	54
致 谢	58
附录 I	59
附录 II	60
附录 III	62
附录 IV	63
附录 V	64

1 绪论

1.1 智能小车研究背景和意义

无线网络通信技术、视频处理技术以及嵌入式技术近年来日臻成熟，智能小车的技术也由此得到了飞速发展，目前已经涌现出多种多样的智能控制系统小车，尤其是基于 WIFI 的嵌入式视频监控系统智能小车得到了广泛应用^[1]。智能小车，通过内部和外部传感器，识别外部环境信息及其自身状态，在有障碍的复杂环境中，实现自主运动，从而完成相应操作的机器人系统。它融合了传感器、信息融合、通信接口、导航控制人工智能以及自动控制等一系列高新技术^[2]。

近年来人工智能技术的快速发展，智能小车技术也已经发展到可以模仿人类行为程度，主要是因为自动化生产设备越来越精密，它不仅给传统产业带来了翻天覆地的变化，而且提高了我们生活生产的质量^[3]。智能小车具有自动循迹、避障和可控行驶等诸多功能，可以在不同环境完成工作，智能小车在各个行业都有不错的应用价值。例如，在日常生活中，智能小车可以像导盲犬那样给盲人指引正确的道路。在军事方面，智能小车可以代替我们在危险区域的排雷和侦察等任务^[4]。在科学的研究中，智能小车可以在外星球上完成探索或返回照片。另外，它可以帮助人们在复杂的环境中执行设备检查和货物处理等任务^[5]。

目前国内许多高校、科研机构加入智能小车研究的行列，并且也取得了不错的成就。对智能小车的研究不仅仅是停留在理论研究层面上，很多研究成果已经转化为实用性产品。传统设计的智能小车主要有自循迹功能，避障功能，视频监控功能，无线通信功能等功能，这些智能小车都存在共同的问题：功能结构单一，扩展性差，可操控性差。但在一些特殊的场合，传统的智能小车由于容错率低，无法完成任务，尤其是在环境很复杂的情况下。

智能小车在工作中为了避免发生碰撞，就必须拥有一个完整的自动避障系统。自动避障系统是一个综合系统，集智能控制、规划运动路径、检测环境于一体。该系统首先对外部环境进行检测，分析处理获得的环境数据将其转化为系统能够识别的语言，核心处理器以此为基础规划小车路径、制定正确的运行指令，最后控制指令由系统统一发出^[6]。所以，自动避障系统的研究与开发对于智能小车而言具有重要的现实意义。

智能小车目前的传感器技术仍有待改善，智能水平不高，因此相关科研人员仍旧将障碍自动识别以及精确定位作为核心工作。对已有的相关研究进行充分地了解后，发现目前多功能移动机器人以及智能小车的设计中仍然普遍存在的问题就是目标定位以及自动识别所遇障碍然后主动地对其规避。现阶段，智能小车仍然是借助超声与红外这两种主要手段来实现障碍识别，但超声与红外存在着两大不足：①识别的准确度不高；②采集的信息量不大，这些缺陷都大大地限制了智能小车的发展与大规模应

用^[7, 8]。科学研究发现：机器视觉技术是解决这些问题的良方，借助科技来对事物进行观察，通过识别特定的特征点或者像素，来对小车的视觉识别能力进行大幅度地提高。更为重要地是，通过科技的“眼睛”获得的视觉信息更加丰富，开展的研究深度能够显著地提高，而且智能小车的障碍自动识别系统得到完善后其能够更加有效地规避障碍。

本文设计的 WIFI 视频自动避障智能小车，在传统小车设计的基础上增加 WIFI 视频监控等諸多功能。该款智能小车通过 WIFI 网络视频实时地监控小车的行驶，能够方便地检测到一些人类不方便进入的环境，因此 WIFI 网络视频监控能够实现实时地监控，有效地解决这一困境。以 WIFI 技术为基础的多功能小车的控制终端通常是手机或 PC 电脑，新型智能小车指令的传输媒介是 WIFI。作为智能小车的控制终端，PC 机易于连接，稳定性好且视频画面清晰。而手机则相对更加便于携带。对于智能小车，在移动终端进行控制能够改善智能小车的操作性差的问题。智能小车在遇到障碍物时，通过设定的算法实现避障功能。该项技术还可用来控制家用电器，实现智能家居，故有很高的应用价值。

1.2 智能小车国内外发展及研究现状

1.2.1 智能小车国外发展及研究现状

对于智能小车系统的研究，最早是美国于上世纪 50 年代开始的。全球首个自动导车系统是美国巴雷特电子公司研发出来的，也是世界上第一个固定路线运行的智能小车系统，它具有智能小车最基本的功能^[9]。这一革命性发明拉开了智能小车时代的序幕，智能小车的研究在发达国家开始呈现蓬勃发展的趋势。国外许多大型的科技公司、研究所纷纷在智能小车领域的研发投入大量财力、精力和人力。其中最具代表有微软、谷歌以及 IBM 这三大巨头，他们连续推出自己公司设计出的新型智能小车。随着科技水平的提高，人们开始对智能小车的行进线路进行研究^[10, 11]。

小车“智能化”的概念是美国科学家于上世纪 60 年代提出来的，这些科学家认为在设计智能小车的过程中避障功能的实现是非常重要的方面。智能小车能够采集到不同的环境信息，以此为依据对避障规划进行分类：①局部规划，在智能小车所处环境未知的情况下，以采集到的实时环境信息为依据来分析数据并进一步规划路径；②全局规划，显然地，这是在小车行车路径中的所有障碍物的信息数据均已获取的情况下制定最为科学合理的路径^[12]。

为了实现智能小车有效地避开障碍，J. Holland 在 70 年代以进化论中的“物竞天择、适者生存”为依据提出了著名的遗传算法^[13]，顾名思义，就是借助变异、交叉以及选择等对程序的运算过程进行控制。之后，Thrun 进一步完善了遗传算法的编码，为了提高运算效率 Thrun 采用了直角坐标法来进行编码^[14]。在深入研究栅格理论后，

Kleeman 以及 Chong 在遗传算法中加入了栅格序列编码，这使得障碍识别更加准确^[15]。

传感器技术一步步发展，研究人员在 90 年代开始将红外与超声技术应用于智能小车来对障碍物进行检测^[16]。不过，超声与红外这两种技术手段存在着较大的缺点。借助超声技术来规避障碍时，环境温度的影响不容忽视；而借助红外避障时，圆形物体很可能检测不到而使得小车撞到障碍。在小车运行过程中，若是采集的环境信息不受温度影响则会提高避障的有效性，信息的独立性有利于智能小车的判定。

机器视觉通常被应用于路径识别以及跟踪，其能够监测较大的信息量、能够遥测，但是庞大的数据处理量大大降低了实时性。所以，提高机器视觉性能的关键就在于制定高效的算法以及研发高性能的硬件。

在智能小车全局避障规划方面，著名的位姿空间法是由 T.Lozano.Perez^[17]等人提出的，也就是借助位姿空间的一点来表征智能小车的方向与所处位置。而位姿空间法通常选用的是可视图法，也就是在无向图中将智能小车的大小忽略，用点的形式来表征行车轨迹中小车的位置。Tangent Graph^[18]等人对可视图法进行了改进，提高其灵活性。

1.2.2 智能小车国内发展及研究现状

相对于国外，我国近年来才开始研究智能小车，不过我国科学界已经关注到了其自动避障技术的发展。在我国，目前主要是研究所、高校等具有丰富资源的机构在研究智能小车，主要以清华大学、北京理工、中科院自动化研究所、国防科技大学等为代表。现阶段，智能小车产业得到了国家政府部门的高度重视并且制定了相关政策来大力扶持这一行业。通过对国外相对成熟的智能小车产品进行研究，国内的技术已经取得了突破性进展^[19, 20]。

北京理工大学付梦印等在 2005 年将 GPS 与数字地图导航这两者相结合，从而实现了智能小车能够自动导航。其以数字地图信息作为依据，制定合理的全局路径，之后借助数学建模来进行导航^[21]。

哈尔滨工业大学在 2007 年与香港中文大学进行联合研究，致力于解决智能机器人避障问题。刘进英与董涛等设计了一款智能小车，其遥控功能的实现是以红外作为依托，能够有效地避开障碍，该小车主控模块的处理器选用的是 STC89C52 单片机，而小车遥控命令的接收是由 IRM-2638 红外线接收头来完成的。障碍物的探测由红外传感器进行，然后控制直流电机来控制智能小车的整个系统实现避障^[22]。

南昌大学卢威深入地研究了模糊算法，他于 2012 年在 Matlab 平台上利用多传感器信息融合技术模仿识别障碍算法，将小车的控制核心选为 STC89C52，借助超声波来实现障碍规避这一功能^[23]。

2014 年，中南大学郑海华在研究智能小车的避障功能时从单目视觉的视角出发，以三坐标测量机实验平台为依托，通过采集 CCD 图像中的各个激光点间的位置关系

来对障碍物所在地进行确认。但是实验过程中的障碍物只能部分避开，这是由于借助激光点识别来避障时目标障碍物的轮廓并不能非常清晰地判断出来^[24]。

X1 清洁机器人是哈工大在 2015 年 7 月份研制出来的，能够以手机 APP 为操作端对其周围环境进行观察并进一步远程控制机器人是这款产品的最大优点。X1 清洁机器人的避障功能是借助红外传感器实现的，它能够较好地完成清理，很少发生碰撞事故，主要由于它在遇到清理路径中的障碍物时能够主动地转换方向进行规避。

就全局直线道路模型研究而言，徐友春^[25]制定了识别道路边界的方法，实现了可能性最大的边界直线的寻找；而就机器视觉提取所遇物体形状，沈明霞^[26]研究出了两种方法：①以形态特征为基础来分割图像；②以纹理谱的方向度参数为基础来分割图像，边缘的检测依据小波模极大。车道边缘具有明显的平行线性特征，陆建业^[27]以此为依据在市内交通环境中提出了道路检测法。对于提取到的特征线集合，该算法对其进行模拟匹配，所有的特征线集合均要通过卡尔曼滤波检测，然后借助退火诊断技术来对道路模型参数进行估计。程洪^[28]等综合了图像的色度与局部统计特征，在道路识别研究中提出了一种算法，其以局部统计特征与主元分析为依据。在该算法中，借助广义算法对神经网进行训练，神经网络的输入则选用局部统计特征与像素的分量，对特征矢量进行聚类时借助了均值分类器。二维重建方法是由周欣^[29]提出的，该法以分道线灰度与几何特征为依据来构建约束方程，对分道线进行提取；车道形态是根据采集的特征点来加以明确的，然后借助双曲线模型重建弯道，借助变换重建直道。

利用二维码的某些特性，刘国栋^[30]等人实现了路径二维码向一维编码的简化，这一尝试对研究小型汽车的智能避障有着一定的参考价值。冯琦^[31]等人则将工作重心放在遗传算法上，引入了极坐标的思想，并且改进了算法中的操作算子。而同样是针对遗传算法，周明^[32]等人另辟蹊径，用来解决连续空间中诸如路径点这样的问题，与此同时，他们还将遗传算法结合到了模拟退火算法中，创造了一种全新的算法——遗传模拟退火算法，通过这一算法人们能够显著提高避障效率。在面临不规则障碍物时，某些传统方法可能起不到效果，针对这点，罗熊^[33]等人提出了初始种群有效快速生成法和随机指导式搜索策略结合的观点，为我们提供了一些策略。周浦城^[34, 36]等人则实现了对遗传算法鲁棒性的提升，他们所采取的方式是将混沌变化引入到前者的变量优化中。

1.3 论文主要工作及章节安排

本课题研究目的是以 STM32 为基础设计多功能智能小车，将它应用我们的实际生活中。智能小车能够实现自主避障、自主行驶、实时采集小车周围的图像信息并在终端实时显示。将采集到的数据通过 WIFI 信号发送到上位机和手机终端，同时用户在终端可通过监控界面对智能小车发送控制指令从而达到灵活的控制智能小车。

论文各章节主要内容如下：

第1章 阐述了论文的研究目的及意义，国内外对于自动避障系统的研究现状及发展现状。

第2章 阐述了智能小车系统的整体设计方案及其设计方案的可行性分析。另外，也介绍了系统的软硬件平台的开发环境，这是系统开发的基础。

第3章 提出了系统硬件总体设计方案，建立了智能小车避障系统的总体结构框架，并对系统的硬件模块选型进行了详细的分析。

第4章 详细阐述了基于机器视觉的障碍识别技术研究，提出了一种以固定区域分割原理为依据的算法。综合了固定区域分割避障原理与障碍物识别，从而提出了适用于本课题智能小车的方法。

第5章 阐述了系统软件平台的构建与实现， μ C/OS-III 操作系统和 Qt 图形库的移植，设计了 PC 上位机和 Android 手机客户端的应用程序。

第6章 系统测试。测试主要包括硬件测试和软件驱动的测试。硬件测试包括电机测试和无线路由器测试测试。软件测试主要体现在应用程序的测试。最后对避障功能可行性的实验论证。

第7章 对本文的工作进行了总结以及后续工作的展望。

2 智能小车系统的总体设计

当今电子技术高速发展的背景下，基于嵌入式系统技术的产品越来越多，相对应的系统体系结构日益完整。作为一种新的设计理念，视频实时监控的智能小车应运而生。这种智能小车具有多种优点，多功能、低成本，并且结构灵巧，能够在复杂的工业场景、军事活动中得到广泛应用。

2.1 系统的总体设计方案

2.1.1 智能小车系统的总体设计思路

嵌入式操作系统与处理器近年来的发展突飞猛进，由此我们的日常生活中逐渐开始大量应用无线传输技术与图像采集技术。本文的主要设计思想是将智能小车的核心控制器选为 STM32，对直流电机进行驱动，进而对小车行驶状况进行控制，利用摄像头来实现图像信息采集工作；并将采集到的图像信息，通过 WIFI 传输到 PC 端或手机端进行显示，应用人员，可通过手机或者 PC 控制界面来发送相应的指令给小车的主控制器，小车根据收到的指令做出相应的动作，从而可实现远程实时监控智能小车的功能。

本文设计的智能小车预期实现的功能有如下几点：

- 1、运动功能：主要包括小车的前进、后退、左转和右转以及变速等必不可少的运动功能；
- 2、避障功能：能够对所遇障碍物进行有效规避，借助算法选用最为合理的行驶路径，从而达到无障碍的行驶；
- 3、无线通信功能：上位机与下位机（智能小车）通过 WIFI 无线信号连接起来，对小车进行实时远程控制与视频信息显示。

2.1.2 智能小车系统的总体设计框图

一般来说，智能小车的设计主要包括 2 个方面：硬件模块和软件模块。硬件模块设计的核心是选用一个性能满足应用需求的处理器—STM32 芯片，并以 STM32 为基础在它的外部添加一些必要的功能模块。硬件驱动模块程序设计与应用程序设计共同组成了软件设计。其中，前者的主要部分有三个：无线数据传输模块、摄像头采集视频模块以及电机驱动模块。应用程序模块由 PC 上位机和 Android 手机控制界面共同组成。其总体设计框架如图 2.1。

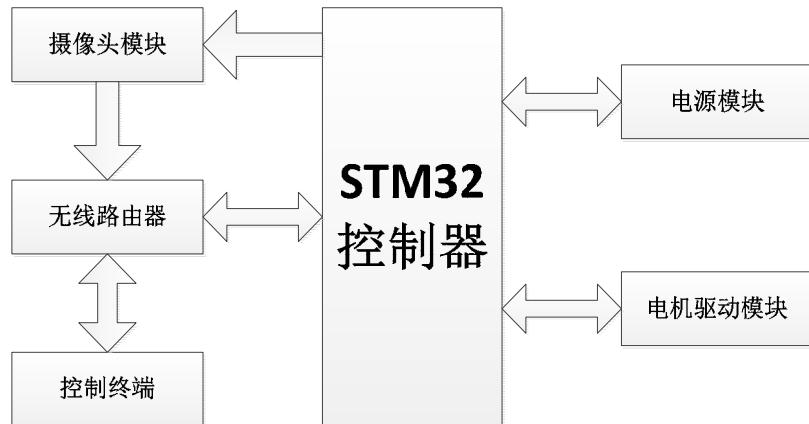


图 2.1 智能小车总体设计框图

在以上的设计图中，小车的每个模块都很重要，发挥着重要作用：

(1) 微控制器模块

在整个系统的设计中，微控制器模块最为核心，它的作用是负责收集信息、处理数据、协调各个模块功能的实现。

(2) 电源模块

电源模块主要是给整个系统提供相应的电压。在本设计中的电源模块可以提供两种电压来满足系统的需求：5V 和 3.3V。该模块的作用是负责小车的运行，并在电动机控制芯片等作用下，控制或调整车轮运转速度。

(3) 电机驱动模块

该模块负责控制小车的运行状态，通过电机驱动芯片和 PWM 脉冲编码输出单元改变车轮的转速。

(4) 视频采集模块

车身前方的 USB 数字摄像头可以收集小车周围环境的视频，然后通过 WIFI 信号将捕获的视频数据传输到控制界面，从而实现实时监控的目的。

(5) 无线通信模块

智能小车与所有控制终端间的通信都需要借助通信模块实现。本系统选用无线通信来传输数据主要是因为小车会发生运动。通信模块将智能小车收集的数据实时传输到控制终端，进而用户可在终端上对小车进行相应的指令操作。短距离无线通信的优点颇多，具有稳定的性能、较强的抗干扰性，并且不受地域影响。

(6) 控制终端模块

本文提供的控制终端有两种：PC 上位机和 Android 手机客户端。用户可以通过任意一种控制端来实时地显示小车运行以及控制小车运动。

2.1.3 设计的可行性分析

本文选择的主控制器是目前主流处理器-STM32 系列，其性价比明显优于同类型的 ARM7 系列处理器。该控制器采用的是高度集成的结构方式，缩减了很多不需要

的外设，提高了芯片的整体性能。电机驱动器芯片选用了 L298N，该芯片的 Enable 控制信号有两个，所用的控制信号是 TTL、CMOS 逻辑电平，以对外部电阻进行检测并反馈控制电路的变化量。汽车的速度可由 PWM 脉冲编码速度进行控制，电动机速度的改变是由平均电压值的改变来实现的。该相机使用 USB 摄像头，使用简单-即插即用，降低了系统结构的复杂性，提高了系统运行的速度^[37]。该应用程序使用的开发工具是 Qt Creator 4.4.1 软件，并采用 UI 界面设计对上位机的操作界面进行设计；Android 应用程序是在 Eclipse 软件平台上开发，其中用到的编程语言是 JAVA 语言。

智能小车的设计主要是包括硬件和软件系统的设计。硬件的设计主要就是分模块的设计，然后将每个模块搭载在小车上面，同时也要保证各个模块的引脚接线图无误；软件设计主要是应用程序设计和驱动程序设计。应用程序的设计是让用户可以通过控制界面远程操控小车和实时显示视频信息；驱动程序设计主要对系统中各个模块设计相应的驱动程序。

2.2 智能小车系统开发环境

2.2.1 硬件开发环境

硬件设计工作是以电路图为原理进行的，并在图纸引导下制定 PCB 电路板。本文通过 Altium Designer 09 等软件的作用，完成电路的设计工作。该软件主要有这样几部分构成：控制信号完整性与可行性分析、电路板印刷与预制、系统电路仿真与实现。

另外，Altium Designer 09 主要包含三部分内容，即原理符号、仿真模型以及集成库。组件集成库则有电器特定的组件构成，从而在相当程度上提升了工程师的开发效率，并减少开发所用时间。

2.2.2 软件开发环境

该系统的软件开发环境为 Keil μ Vision5，其不但可以与 C 语言程序相兼容，实现文件管理、链接等多项功能，同时还提供在线仿真与调试服务。Keil μ Vision5 可以实现上述一系列功能，并且便捷截面十分的简洁、美观，容易操作，其主要包含五方面特性，具体为：

- (1) 代码编辑器功能比较齐全，可以提供多元化服务；
- (2) 内置数据库可以实现同时实现多种配置与操作；
- (3) 具备功能强大、性能良好的编译工具；
- (4) 对话框可以根据需求与爱好设置成多种形式；
- (5) 提供在线调试功能；
- (6) 与之相配套的信息资料十分完善。

借助 Keil μ Vision5 软件进行设计开发的过程中，需要按照下列五个步骤进行操作，具体为：

- (1) 根据设计要求，建设一个科学合理的新工程，并结合实际需求，确定开发控制器，最后进行工程配置与调试。
- (2) 参照功能需求与系统特点，初步编制完善的程序代码；
- (3) 通过科学合理的方式来编译工程文件；
- (4) 对程序进行调试，找出程序中存在的不足，然后进行修改与优化；
- (5) 将 object 文件 download 到开发板并进行调试。

2.3 WIFI 技术

当前，网络对我们的日常生活影响越来越大，不论是在家里、在公司、在商场网络都随处可见。WIFI 技术由于便捷和覆盖面广等特点被广泛应用。WIFI 信号的强度较高，并且不会受制于地理位置的约束，应用范围较广，极大的改善了有线网络应用范围小的缺陷，同时也可以满足高速传输的功能需求。在远程监控系统之中，有线传输存在环境、位置和布线的局限性。而 WIFI 就可以很好的解决这些缺陷，其区域覆盖面广、节省布线成本、易于后期维护、结构也非常简单^[38]。

2.3.1 WIFI 的基本工作原理

WIFI 工作形式和内置无线路由器工作形式较为相似，而无线网卡的主要作用，是接受客户终端发送的信号。具体工作流程为：首先，以主动或被动的形式，获取有关的工作信息与数据，当被动扫描仪运行的过程中，工作站之中的频道列表将会对不同频道的信号进行快速切换，同时对信标帧信息进行全面扫描；当主动扫描仪运行的过程中，工作站将会在不同的通道之中对外发送 Probe Request 帧信号，以便特定网络可以接受处理器信号；当扫描结束之后，便会根据所扫描的信息自动生成 BSS 参数报告^[39]。把全面扫描结果进行汇总处理，在访问点关联前，工作站便会自动加入至 BBS 之中，同时工作站对其具体身份进行验证分析；当身份验证流程全部结束之后，工作站便会根据其份信息分配相对应的访问网路的权限^[40]。

2.3.2 WIFI 技术的特点

- (1) WiFi 辐射范围大、信号强：经过实验证与理论分析得出，WiFi 的信号半径最高可以达到 100m 左右，一般被应用于写字楼层、办公室等场合。
- (2) WiFi 传输速率较快、安全性与可靠性较高：802.11b 无线网络规的带宽最高可以达到 1Mbps 左右，当处于信号弱或者存在外部干扰的环境之中，为了保证信号强度，也可以结合实际情况对带宽进行调整，即调为 5.5Mbps 或者 2Mbps，带宽自动调整功能一方面可以确保网络信号强度，同时也可提高网络的安全性以及稳定性。
- (3) WiFi 技术不需铺设线路：无需布线、不受空间环境的约束是 WiFi 技术的最大优点。当前，WiFi 技术已经不再仅仅局限于管理、库存、医疗等特殊领域，正

在逐步向更为多元化的领域发展，在一些教育机构、家庭教育培训方面已经开始应用 WiFi 技术。

(4) WiFi 的安全性、可靠性比较高：： IEEE802.1 明确规定，其发射功率应当在 100hw 以下，实际发射功率的范围一般为 60-70hw，其中对讲机、手机的发射功率分别为 5w、100hw-1w。另外，无线网络在具有应用过程中，不需要与人体进行直接接触，因而安全性能较好。

2.4 本章小结

本章节主要是结合设计需求，从不同层面、不同视角对智能小车系统进行剖析，并确定了总体设计思路以及具体的实现方案。首先，对硬件单元进行全方位、多层次的介绍与分析，其次，对智能小车的功能实现进行阐述与探析，同时从软件、硬件两个不同的层面来验证设计方案的可操作性。最后，详细介绍智能小车系统的程序调试环境、硬软件集成开发环境，并且从不同层面分析了 WIFI 技术的特点。

3 智能小车系统的硬件设计

3.1 系统硬件设计

智能小车硬件结构设计框图如图 3.1。

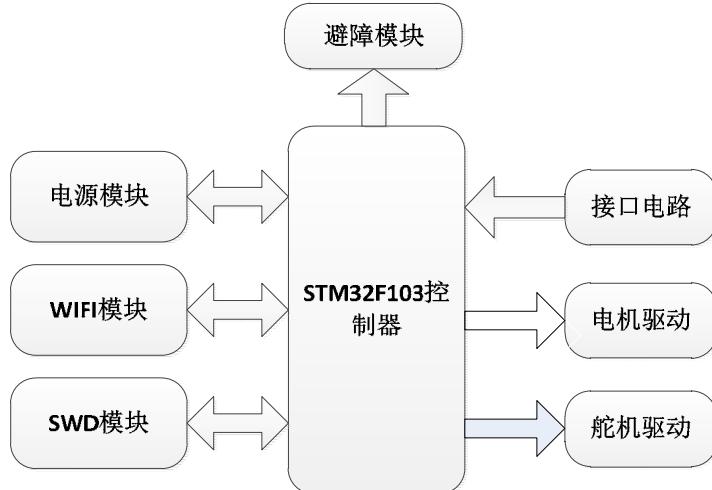


图 3.1 智能小车硬件结构设计框图

硬件系统各模块主要功能如下：

- (1) 系统中各个功能模块使用不同的工作电压，需要电源模块分别提供不同的电压给各个模块正常工作。
- (2) WIFI 模块主要负责智能小车和 PC 机或 Android 手机之间的数据传输。WIFI 模块采用的是无线传输，能够灵活地将智能小车的运动状态以及采集到的各种数据上传到 PC 机或 Android 手机，用户通过 PC 机或 Android 手机将相应的指令发送至小车的控制中心，从而用户通过 PC 机或 Android 手机实现对智能小车的控制。
- (3) SWD 模块是系统程序烧写模块，通过 ST-Link V2 仿真器将已经调试编译好的程序烧写到小车的核心板里，从而让小车执行相应的程序指令。
- (4) 由于小车的核心板上没有并口电路模块，需要使用转并口电路模块来提高系统的兼容性。
- (5) 电机驱动模块的作用是驱动小车车轮的转动，如正转，反转，停止等。
- (6) 舵机驱动模块主要依靠内部的基准电路发挥作用，其能够产生具有一定周期和宽度的基准信号，然后利用信号比较器将基本信号与其他信号进行综合对比，选取符合要求的信号值与方向，进而确定电机的转动信号。
- (7) 避障模块通过基于固定区域分割原理的避障算法，通过设计小车与障碍物之间的位置关系，采用相应的运动策略，完成对不同障碍物的躲避。

3.2 系统硬件模块选型

3.2.1 微处理器模块选型

智能小车的微处理器要具备以下辅助功能，包括与外部设备建立通信联系、控制智能小车的电机转动速度等，综合这些要求，本次选择的控制模块是性能较为优良的 STM32F103 系列芯片。此种芯片的内部框架主要由 Cortex-M3 内核决定，并采用与系统相统一的嵌入式设计，这种结构使得系统具有功耗较低、性能高及成本低等显著优势，并且系统的集成度大大提升，降低了开发难度^[41]。系统采用 STM32 处理器，由增强型 I/O 端口、APB 总线构成，APB 总线有两条与设备进行连接，系统能够实现对外部设备的良好控制；此外，该处理器还可以参照系统特点对其内部功耗进行完善，从而与系统要求相适应。基于 STM32 处理器的性能优势较为明显，在工业及消费行业的应用较为广泛，其优势主要有以下几个方面：

- (1) 速度方面：主运行频率最大时 72MHz；
- (2) 存储器方面：芯片的型号与系列不一致，其内部 Flash 容量也存在较大差异，主要处于 32K 至 512K 两数值之间，而 SRAM 的容量值则处在 6K 至 64K 之间；
- (3) 时钟、复位和电源管理方面：构成芯片的集成电路振荡频率为 8MHz，电源电压是一个范围值，位于 2.0V-3.6V 之间，并具有 RTC 校准功能，它的晶振频率为 32KHz；
- (4) 功耗方面：处理器具备三种功耗模式，分别是：休眠、待机和停止，当系统断电时，可以使用备份寄存器和后备电池为其提供持续电能；
- (5) 外设资源方面：处理器内部芯片的组成结构主要是 USART 控制器、DMA 控制器、A/D 转换器以及多种接口，其中转化器采用 12 位，USART 型控制器有五个，DMA 型控制器有两个。

3.2.2 电机驱动模块选型

智能小车上最常见的电机驱动芯片有 L298N 和 L293D 芯片，性能参数对比如表 3.1：

表 3.1 L293D 和 L298N 参数对比

	L293D	L298N
电路结构	单 H 桥集成电路	双 H 桥大电压驱动器
最高工作电压	36V	46V
最高输出电流	2A	3A
额定电流	1A	2A
去驱动电机个数	两台直流电机	两台直流电机
性能	电机驱动能力弱	电机驱动能力强，可驱动大电流的机器

表 3.1 中的数据表明，L298N 的整体性能优于 L293D 的芯片。本系统的电机驱动芯片选择 L298 型号。

本设计采用的是直流电机，相比于其他电机，具有如下的优点：

- (1) 能够在速度及信号不恒定的情况下稳定工作，其应用范围较广；

(2) 电机可以承受较大载荷，具有明显的负载特性，使系统的稳定性得到提升^[42]；

(3) 系统处于空载状态下时，电机的力矩较大。小车在接收到不同指令的时候，由于具备惯性还会继续保持原有的状态继续运动，直流电机可以给小车提供足够大的力矩来阻止惯性运动^[43]；

(4) 作噪声小，性能稳定；

(5) 直流电机的用途很广泛，在生活中随处可见。

L298N 模块与电机的引脚接线图如图 3.2。



图 3.2 L298N 电机驱动模块

在 L298N 驱动芯片的作用下，驱动模块不仅能够驱动多个步进电机同时工作，还能够驱动多个直流电机同时运行。本次设计的智能小车中共包含四个直流电机，小车的两侧各有两个直流电机，将两个电机相联由同一个控制信号进行控制。信号两端分别为 ENA 和 ENB，能够实现对电机的有效控制，使电机可以实现正转、反转、自由滑行以及停止四种运行模式。

3.2.3 视频采集模块选型

视频采集任务由摄像头负责，其将信号采集后传输到显示单元，因此，摄像头性能的好坏与监控效果有着密切联系。摄像头的选择依据主要有下列两种：

(1)采用 CMOS 数字图像传感器。其内部结构包括 A/D 转换器、视频编码 Camera Interface^[44]、图像传感器。目前，OV9650 图像传感器和 OV7620 图像传感器应用较为广泛。

(2) 利用 USB 数字化摄像头。此种类型的摄像头具备开发简单、灵敏度高、画面清晰等一系列优点，且其结构十分紧凑，所以只需在 USB-Host 驱动上配设 USB 驱动即可，如此一来便可以实现视频信息实时采集功能。

本系统硬件设计具有一定的复杂度，而且各个模块之间相互独立，故系统可选用 USB 数字摄像头来降低硬件设计的复杂性。此种类型的摄像头的内部配设有集成电路，在数据信息传输的过程中，主要是借助集成电路将信息数据输送至电脑，当处理器的响应速度达到一定值后，CCD 所捕捉到的图像信号便可以利用信号传输的形式来实现动态显示的目的^[45]。此款 USB 数字摄像头具备多方面优势，具体为：

- (1) “一键通”远程监控：具有数字录像、网络传输、支持网页浏览、动态检测、云台控制、报警输入输出、文件备份、文件回放和系统管理；
- (2) 智能化的报警功能：报警时会发出较为强烈但对声音，同时可以实现图片随机抓拍、短信报警等一系列功能；
- (3) 智能分析、移动侦测会自动触发录像功能：在录像功能触发之前，可以实现预先录制，时间长达 5 秒钟左右；
- (4) 智能分析功能：丢失物品检测、面部表情识别与检测，物体计数，绊线检测，逆行检测；
- (5) 工作电压可完全依靠无线路由器提供，不用再外接电压，降低了电路设计的复杂度；
- (6) 支持手机观看和远程网络观看，在夜间也可以正常采集视频信息。

3.3 无线路由器的设计

本系统需要无线路由器创建覆盖范围较小的局域网，小车控制信号、系统控制终端可以进行信息数据的实时交互。首先，借助控制终端网络程序，与小车 UDP 建立连接，当连接成功之后，用户便可以根据功能需求，通过控制终端将一系列信息指令发送给小车系统，并得到相关的视频信息。本设计在具体进行的过程中，视频信息、控制信号等诸多信息数据均是通过无线传输实现的。

考虑到系统设计的美观性和实用性，本文选用 TP-LINK 的 mini 型无线路由器，该路由器传输速率大约为 54Mbps。在室外空旷的空间内最大可覆盖 35 m²，在室内封闭的空间内最大覆盖范围可达 10m，该无线路由器完全满足设计的带宽需求以及系统性能的稳定性。mini 型无线路由器如图 3.3。



图 3.3 无线路由器

3.4 STM32 的最小系统

3.4.1 电源电路

电源模块是给系统中硬件模块提供稳定的、有效的电压，以确保整个系统可以正确地运转。该模块对整个系统的最终性能至关重要。由于系统中的各个电路模块需要的额定电压值都不相同，需要把最初电池组提供的最大电压 7.5V 经过降压技术进行电压转换，转换到各个模块需要的额定电压值。电源模块结构图如图 3.4：

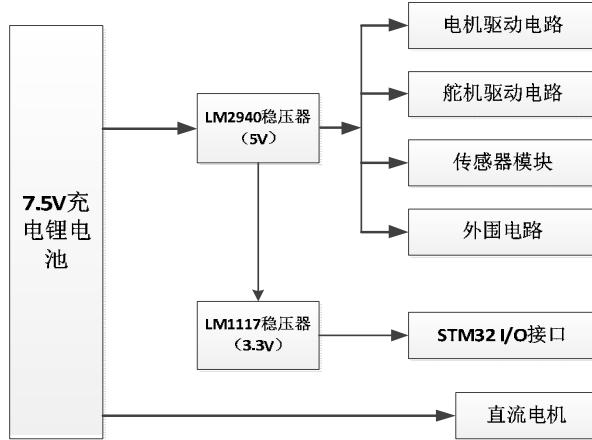


图 3.4 电源模块结构图

STM32 外围硬件模块额定电压是 5V。由于锂电池组只能提供 7.5V 电压，无法满足外围硬件模块正常工作。如果电压高于硬件模块的额定电压，该模块不仅不能正常工作反而还会被烧毁）。因此，需要对电池组提供的 7.5V 电压进行稳压电路处理，将 7.5V 电压降到需要用的 5V 电压。本系统选用 LM2940 型号的稳压芯片，LM2940 电路图如图 3.5。

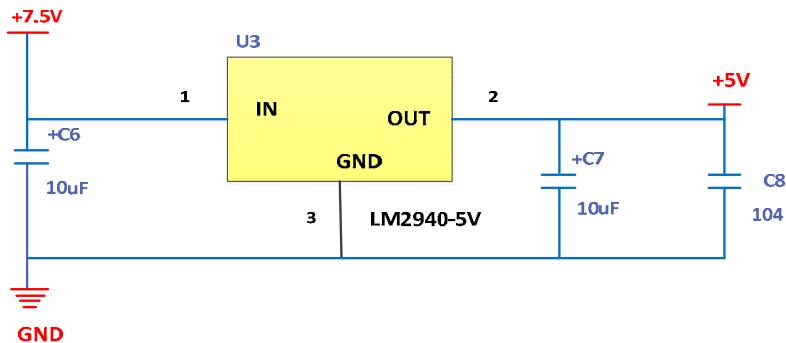


图 3.5 LM2940 电路图

STM32 芯片以及它的 I/O 接口的额定工作电压是 3.3V。虽然锂电池组的 7.5V 电压经过稳压处理后得到了 5V 的工作电压，但是仍高于 STM32 芯片和它的 I/O 接口的正常工作电压，因此还需要对 5V 电压进一步处理。本系统通过采用 LM1117 芯片对 5V 电压进行稳压处理，得到所需的 3.3V 电压。LM1117 芯片稳压电路图如图 3.7。

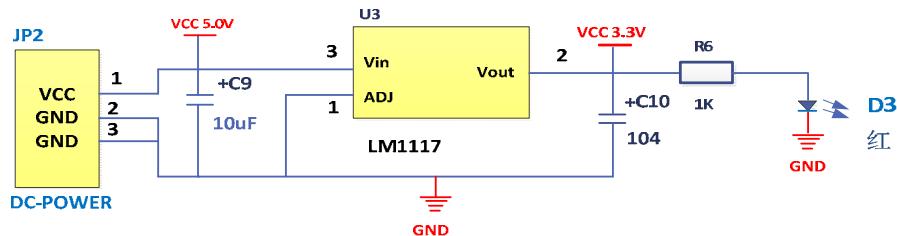


图 3.6 LM1117 稳压电路图

3.4.2 晶振电路

主控制器 STM32 上电复位后，默认使用内部 8MHz 的晶振，如果外部接了 8MHz 的晶振，可以自动切换为使用外接的 8MHz 晶振，PLL 倍频到 72MHz。本电路选用 8MHz 外部晶振，外接的 32KHz 晶振是为 RTC 提供时钟信号。本系统的晶振电路图如图 3.7。

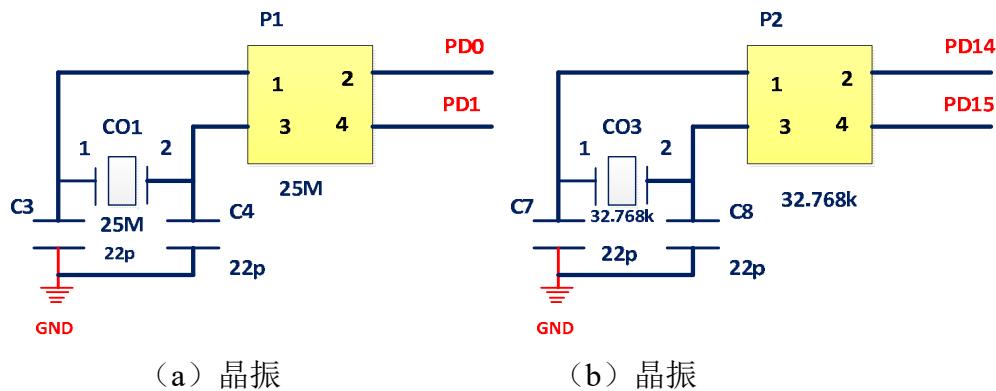


图 3.7 晶振电路

3.4.3 SWD 接口电路

主控制器下载程序可采用 SWD 接口和串口 ISP 下载二种方案，本设计选用 SWD 接口。SWD 接口是由 ARM 公司开发的，可以减少调试接口所使用的引脚数。SWD 接口使用两个信号线：双向的数据线（SWDIO）和数据线的时钟信号（SWDCK）。最新的 SWD 调试协议仅需要两根数据线可以完成 JTAG 接口所有功能，抛弃占用引脚多、占用面积较大的传统 20P 接口底座。ST-Link V2 电路引脚接连图如图 3.8。SWD 和传统的调试方式区别：

- (1) 同 JTAG 相比较而言，在高速模式之下，SWD 的安全性与可靠性更高；
- (2) 当数据量比较大时，JTAG 下载程序会自动停止，即下载失败，但是 SED 发生的概率非常微小；
- (3) 在 GPIO 比较少的时候，可以使用 SWD 仿真，这种模式支持更少的引脚，需要的 PCB 空间较小；
- (4) 在开发板的体积受限时推荐使用 SWD 模式。

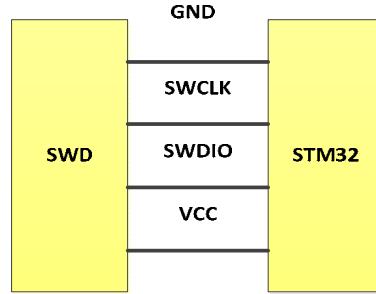


图 3.8 ST-Link V2 引脚图

3.4.4 串口通信电路

本系统选用的 STM32 处理器是简洁型的主控制器，在核心板上只有 USB 接口，而没有通信串口，如 RS232。为了便于通信测试，系统采用芯片 CH340G 实现 USB 转串口。CH340G 芯片转串口电路图如图 3.9。

该款芯片具备很多优点：

- (1) 高速 USB 接口，兼容 USB V2.0；
- (3) 支持多种电压，如 5V、3.3V、3V。
- (3) 支持多种通信接口，如 RS485、RS232 外接接口。
- (4) 支持常用的 MODEM 联络信号 RTS、DTR、DCD、RI、SR、CTS。

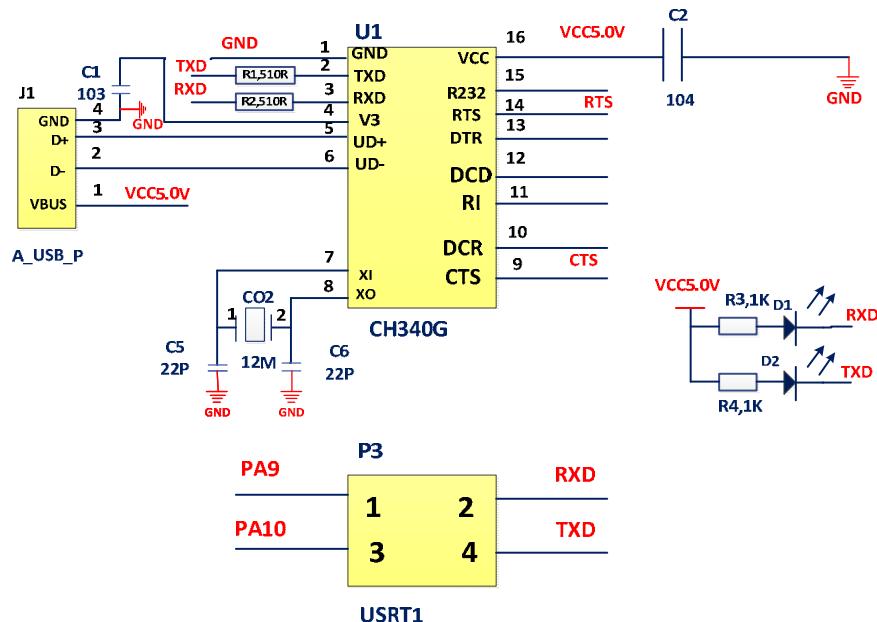


图 3.9 转串口电路

在系统运行的过程中会出现一些操作失误或者程序运行出线 bug，可能导致系统进入死机状态，整个系统停止工作。为避免此类情况发生，在系统中设置复位电路，通过采取一键复位的操作，系统立即就恢复到初始化进程。系统复位电路图如图 3.10。

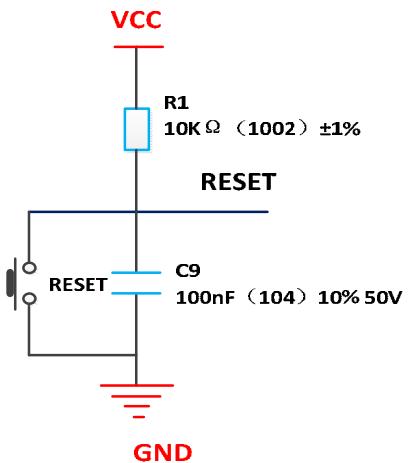


图 3.10 系统复位电路图

3.5 本章小结

本章节主要是对智能小车硬件系统的整体方案进行全方位、多层面的剖析，同时结合整体方案中的相关要求与标准，论述了 STM32 的一系列性能与外设硬件选型等相关问题。最后对电机驱动模块设计、视频采集模块设计、无线路由器模块设计和 STM32 最小系统的设计等内容设计进行了相关的分析。

4 智能小车避障系统的设计与实现

本文的自动障碍系统主要借助机器的视觉采集功能来对机器周边环境进行信息采集，然后进行图像处理与分析，最后根据处理分析结果有计划、有目的的躲避障碍物。自动避障系统主要利用了三方面技术，具体为：第一，障碍检测。其是由两部分构成，即微控制器、传感器，主要实现小车附近信息收集等功能。第二，信息处理。由于视觉系统所收集信息存在一系列不确定因素，再加上图像采集摄像头自身存在诸多不足，进而使小车的具体走向受到外界无用信息的严重影响，基于此，必须要制定一种科学合理、切实可行的方案来实现有用信息的提取。第三，躲避决策。参照已经处理分析过信息数据，确定一条小车可以顺利通过，并且不会发生碰撞的路线。

4.1 避障流程设计

为了确保小车躲避功能的可靠性以及安全性，本文根据设计要求，首先确定了环境坐标系，并借助视觉技术获取障碍物的相关参数，通过区域分割法得出小车、障碍物之间的距离信息。若障碍物对小车产生了影响，那么小车将会自动执行躲障策略。小车躲避障碍物时，会与既定方向以及路线产生偏颇，应当根据实际情况对其行驶路线进行调整，以便使小车按照既定路线继续行驶。小车避障流程如图 4.1。

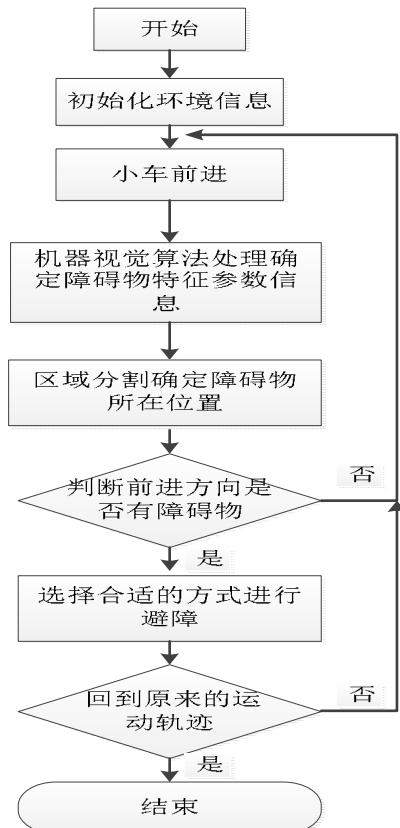


图 4.1 小车避障流程图

4.2 基于固定区域分割原理的避障算法

本文在设计分析的过程中，把摄像头放置于特殊位置，即与路面保持一定夹角，确保摄像头所拍摄的画面为路面信息，并且不会受到外界因素的影响。一般情况之下，路面特征信息并不会发生较大变化，但是当路表产生障碍时，图像便会产生被破坏，即出现破损区域。智能小车、障碍物具备五种不同的位置关系，具体见下图 4.2 所示。以分割聚类思想为重要基础，结合实际需求，设计开发了一种以固定区域分割原理为基础的躲避算法。区域分割实质上就是指把小车前方的一部分区域细化为一个个面积固定的小区域，当小车前方区域出现障碍物时，那么小车便会自动采取相应的策略。对利用同一策略的小区域进行整理汇总，并划分为统一类别，当小车发现障碍物之后，便会造成几种比较固定的躲避方式，如此一来，便可以对小车的躲避形式进行深入、全面的剖析。

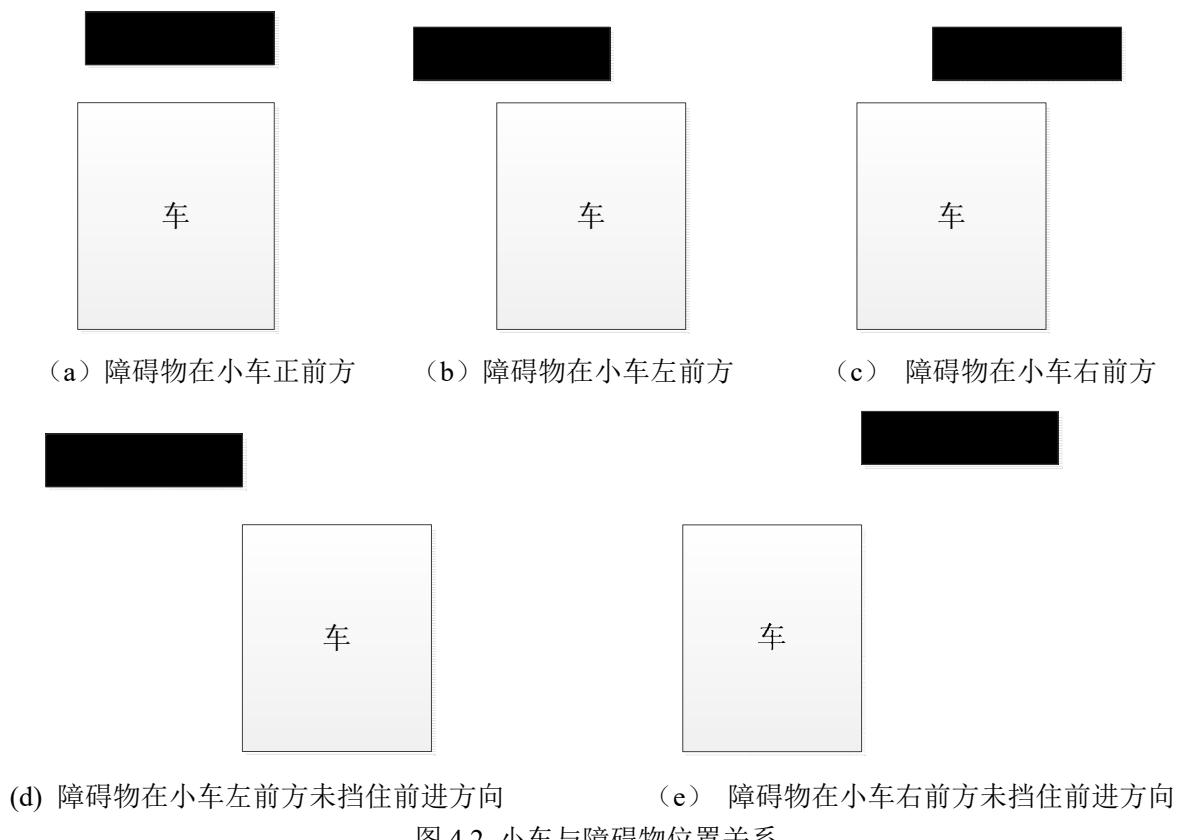


图 4.2 小车与障碍物位置关系

在图 4.3 所示的固定区域分割方法，若区域①②不存在障碍物，小车便按照既定路线行驶。若区域①②存在障碍物，那么将会对区域③④进行分析与判别，并选取无障碍物的一条路线。若区域①存在障碍物，但是区域②不存在，那么便会对区域④进行分析与判别，若④中午障碍物，那么小车便会自动转向右边，然后顺利驶过；若均存在障碍物，那么将会对③进行分析。若小车前方均存在障碍物，那么小车便会优先向右进行转动，然后进行二次判别。同理，若区域①②只存在一个障碍物，那么将会

对区域③④进行分析与判别，并从无障碍物的区域通行。其中区域⑤代表安全区，主要目的是为了应对一些意外事故与突发情况，若区域⑤存在障碍物，不管其它区域处于何种状态，小车均会立即停止运动。

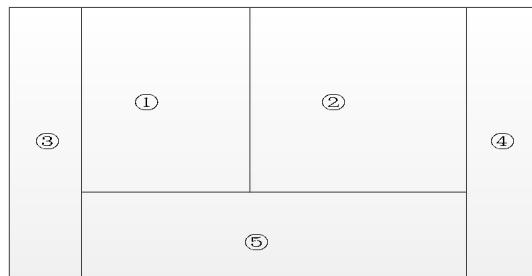


图 4.3 固定区域分割法

4.3 机器视觉的图像采集及处理

4.3.1 图像的获取和数字化

本设计主要是利用机器视觉的方式实现障碍物自动躲避，如同给小车加配一双明亮的“眼睛”，以便小车可以及时、全面的掌握前方道路的信息数据，从而使系统可以在短时间内规划出一条科学、完善的行使道路以及应对方式。图像信息的获取主要是利用若干个样点对图像进行分析与描述，样点越多，则代表图像的清晰度越高^[46]。

4.3.2 图像预处理技术

在进行图像采集的过程中，由于硬件设备质量、外部环境等一系列因素的干扰，图像的清晰度较低，所以必须要采取相应的措施对图像进行预处置，以此来进一步提升图像质量、消除外界干扰源，达到增强图像清晰度以及准确度的目的。当前，对图像进行预处理的过程中，主要包括下列二大部分，具体为：

1、图像的灰度化

图像的灰度化实质上就是把彩色图像转变成为灰度图像。彩色图像中，各像素颜色是由三个分量决定，即蓝、绿、红，分别用 B、G、R 表示。灰度化图像是指三个分量值完全相同时的一种特殊色彩图像，所以灰度图像仅存在 256 种的颜色可能，如此一来，便可以使后续图像处理的运算量大大减少^[47]。

目前，图像灰度化的处理方式主要包括三种，具体为：

(1) 最大值法：假定灰度值 Y_1 取三个分量中的最大值，可以得到亮度较高的灰度图像，公式 4.1 为：

$$Y_1 = \max(R, G, B) \quad (4.1)$$

(2) 平均值法：假定灰度值 Y_2 取三个分量中的平均值，可以得到亮度柔和的灰度图像，公式 4.2 为：

$$Y_2 = (R + G + B) / 3 \quad (4.2)$$

(3) 平均值法：假定灰度值 Y_2 取三个分量中的平均值，可以得到亮度柔和的灰度图像，公式为 4.3：

$$Y_3 = 0.299R + 0.587G + 0.114B \quad (4.3)$$

2、图像的阈值化

图像的阈值化主要是参照不同物体表面对光的反射率存在较大差异，然后结合实际要求，确定一个比较合理、恰当的阈值，将需要检测分析的障碍物从大环境之中进行分离^[48]。图像通过阈值化处理之后，色彩信息将会被再一次简化，通常情况下可以通过具体公式 4.4 表示：

$$p(z) = \begin{cases} a, & f(x, y) \leq T \\ b, & f(x, y) > T \end{cases} \quad (4.4)$$

式中， $f(x, y)$ 的含义为 (x, y) 点处的灰度值大小，当 a 、 b 分别为 255 和 0 时，那么在阀值 T 的作用之下，图像各点仅存在两种状态，即黑白。

图像分割值的选取是图像阈值化处理的重点与核心，其中反二进制阀值法、二进制阀值法、Otsu 等均是比较普遍的阀值确定方式。本文在设计过程中，参照路面状况，通过 Otsu 方式确定了两个阀值，并把图像细化为三大部分，运算分析公式 4.5 如下，具体为：

$$p(z) = \begin{cases} 0, & f(x, y) \leq T_1 \\ 255, & T_1 \leq f(x, y) \leq T_2 \\ 0, & f(x, y) \geq T_2 \end{cases} \quad (4.5)$$

式中， T_1 、 T_2 属于固定阀值，而 $f(x, y)$ 的含义为 (x, y) 点处的灰度值大小。经过实验分析与论证可知，采取 Otsu 方式所得到了图像比较符合本设计要求，具体为下图 4.4、4.5。在图示中，将路面作为背景，并且选取椅子、烧水壶作为障碍物，(a)、(b)、(c) 分别表示原始图像、灰度图像以及采取 Otsu 方式处理后得到的图像。

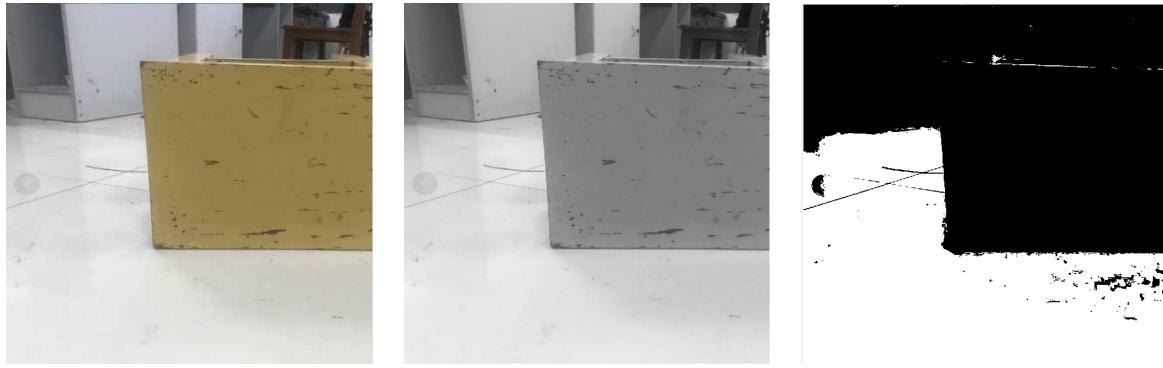


(a) 原始图像

(b) 灰度图

(c) 二值化图像

图 4.4 Otsu 算法图像处理障碍物 A



(a) 原始图像

(b) 灰度图

(c) 二值化图像

图 4.5 Otsu 算法图像处理障碍物 B

从图上可以看出，经过 Otsu 算法得到的图像，外部环境的影响已经被完全消除，同时还可以降低外界噪声的影响，使小车躲避功能可以得到更好的发挥。从整体效果来说，基本实现背景、待测目标完全分离的效果，但是进行障碍物分割时，同路面灰度值比较接近的一小部分无法消除，但是该点并不会对后续障碍判别产生实质性的影晌，一般情况下，障碍物尺寸是最值得关注与分析的问题，在图 4.4、4.5 之中，(c) 图留存了该部分信息数据。

4.4 障碍自动检测

通过以上机器视觉技术处理之后，已经基本实现背景、待测目障碍物分离的目的，但这仅仅是初步分离，与预期效果相差甚远，并且计算机也无法对此种分离效果进行识别，不能获取相应的特征信息数据，仍需进一步细化分离。因为，其中最核心的环节就是准确识别障碍物，让机器可以自动识别障碍物并进行躲避。

4.4.1 障碍物特征参数获取

对于特定的目标来讲，首先需要对其进行拍摄图片并对图像实施二值化处理，可以选择断点坐标法来对障碍物主要的特征参数进行表征。当对障碍物进行定位时，主要是以物体的中心坐标作为定位坐标，忽略障碍物本身的深度信息，只考虑障碍物从左至右的宽度，具体的测量可以通过测试物体最右端中点和物体中点坐标计算得出。

在进行计算过程中，假设物体的底边中点坐标为 (m_1, n_1) ，最左端中点坐标为 (m_2, n_2) ，则计算如公式 4.6、4.7、4.8、4.9：

$$m_1 = \frac{\max(i) + \min(i)}{2} \quad (4.6)$$

$$n_1 = \max(j) \quad (4.7)$$

$$m_2 = \min_{i \in S_{ij}}(i) \quad (4.8)$$

$$n_2 = \frac{\max_{j \in S_{ij}}(j) + \min_{j \in S_{ij}}(j)}{2} \quad (4.9)$$

其中，在上述式中 (i, j) 表示图像中每个像素点的坐标， S_{ij} 表示目标障碍物所占的连通域。

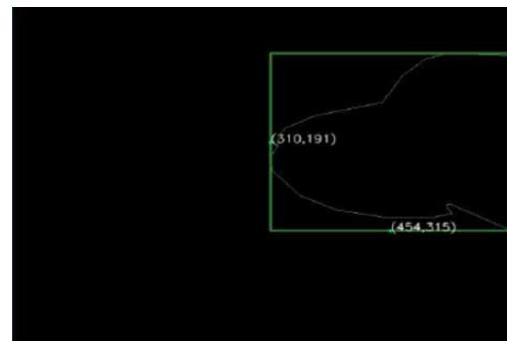
根据上述得到的坐标点参数，便可以绘制出一个包含整个物体的矩形轮廓，该矩形的长度以及宽度可以通过以下公式 4.10 计算得到：

$$\begin{cases} L = 2 \times (m_1 - m_2) \\ W = 2 \times (n_1 - n_2) \end{cases} \quad (4.10)$$

在上一章节的讨论中，列举了具体的案例进行分析，本章节中继续按照该案例进行分析，如下图 4.6，在进行计算过程中，首先得到障碍物图片中底边中点坐标以及最左端中点坐标依次为 $(454, 315)$ 和 $(310, 191)$ 。然后根据这两个参数来得到一个囊括整个障碍物的矩形框架。按照这一测试原理我们也同样可以得出本章节研究中的热水壶效果如图 4.7。



(a) 原始障碍A图

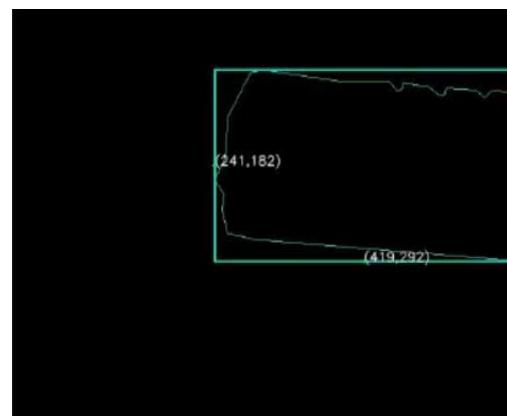


(b) 获取的障碍物A信息

图 4.6 障碍物A特征参数获取



(a) 原始障碍物B图



(b) 获取的障碍物A信息

图 4.7 障碍物B 特征参数获取

4.4.2 多个障碍物的处理

上述研究主要是针对单一的障碍物为主进行研究分析，但是在实际的道路环境中，障碍物的数量远远不止一个，经常会存在多个障碍物。要想保证设备可以准确地对各种障碍物进行识别，就需要准确获取所有障碍物的特征参数，并将这些障碍物进行区分。

当前主要存在下述集中方法来实现多个障碍物的图像分割：

第一，边缘提取法：对于图像来讲，一般明暗变化最显著的区域就是物体的边缘。通过分析图像中某一区域的灰度值变化，就可以发现物体边界处的灰度值存在较大的差异。因为物体多数的图像特征都是体现在图像边缘部位，因此准确寻找到物体的轮廓对于设备判断障碍物具有非常大的影响。边界提取方法能够高效地实现图像的分割，通过边缘增强算子来表征物体的轮廓特性，然后选择合适的阈值获取物体边缘的集中点。但是在实际的图像拍摄过程中往往会有存在图像模糊或者噪音较大的影响，进而会使得通过这一方法获取的物体轮廓出现断裂现象。

第二，图像特征分割法：对于某一障碍物来讲，其自身的特性以及所处的环境都会存在一定的差异。例如物体的形状大小，颜色明亮程度等等。其中图像的色彩属于整体性的特征，主要是体现出物体在特定环境下的外观差别。经常采用的方法是颜色直方图，这一方法不会受到图像的旋转以及移动的影响，但是也经常会检索出一些没有必要的图像。对于物体的形状特性，至今没有有个健全的数据库资源，即便建立了数据库，如果物体的实际形状发生了变化也会对测试结果造成影响。位置关系特征就是指特定空间下不同物体的相互位置关系，在实际使用过程中需要结合物体的自身特征来配合。

第三，贝叶斯函数优化法：这种计算方法将物体的所有属性以点进行表示，通过有向边来保证不同属性的关联，以根结点表征其类别属性，余下的属性均属于其子节点。贝叶斯函数优化法在统计学中的运用非常普遍，尤其是用于系统的判别。这种方法具有判断精准，考虑全面，可以将各种存在的失误全部考虑其中进行综合的计算。

对于图像的分割来讲，主要是将图像划分为多个不同的区域，进而可以将障碍物的特征信息进行全面的分析。因此本研究采用区域的角度来实现图像的分割，并将所有区域的集合 R 表征全部图像，即假设图像包含 R_1, R_2, \dots, R_n 等多个不同区域。

根据这一划分方法，我们可以实现将特定目标从背景中进行分离。在本文研究中选择一种类似区域生长的方法进行图像区域的划分。这一方法从种子点的集合开始，将与种子属性类似的灰度，强度等参数相似的像素进行整合。区域增长属于迭代的过程，每个像素种子都实现迭代增长，直至将所有的像素全部处理完成，最终将整个图像划分为多个区域，对于不同区域的边界主要是以闭合的多边形定义。

根据图像像素的高度 m 以及宽度 n 构造一个标记矩阵 H ，保证矩阵的行数与列数

与 m 和 n 等同，随后建立一个同等大小的空间矩阵 Y ，实现对每次扫描之后标记信息的记录，具体操作过程如下所述：

(1) 对矩阵 H 遵照从左至右，从上到下的顺序进行扫描，如果扫描到像素值为“1”的像素点时，这一像素点就是靠近图像左上角的第一个目标障碍物的起始种子点，并在 Y 中相应位置做出标记。

(2) 按照上一过程继续对矩阵 H 进行扫描，当出现像素值为“1”的像素点时，就认为其与矩阵 Y 中已经标记的像素点存在连通性，并将其在矩阵 Y 中进行标记。

(3) 当对矩阵 H 进行多次扫描之后，从矩阵 Y 中统计出连通域像素个数 N_1 。为了避免在对障碍物标记时出现冲突，需要对图像在进行一次逆向扫描，即从右至左，从下到上的顺序进行扫描，具体的操作与步骤(2)一样，随后从矩阵 Y 中得到对应的连通域像素个数 N_2 。当 N_1 与 N_2 相等时，按照步骤 4 继续进行操作，如果两个数值不等，继续按照上述步骤进行操作。

(4) 按照上述方法依次获取各连通域像素端点坐标 $\max(i)$ 、 $\min(i)$ 、 $\max(j)$ 、 $\min(j)$ ，通过以矩形来讲连通域进行标记，进而获得各个连通域的底边中点坐标和左侧中点坐标参数。

(5) 将两个矩阵 H 与 Y 进行相减计算，也就是将两个矩阵中共同存在的连通域部分的像素删除。然后判断图像是否完全实现分割，分割完成则结束整个操作，如果没有完成，则需要将矩阵 Y 清零，继续按照步骤(1)进行操作。

下图 4.8 所示即为多个障碍物识别效果图，通过这一方案的设计就能够顺利实现对多个目标障碍物的识别处理。

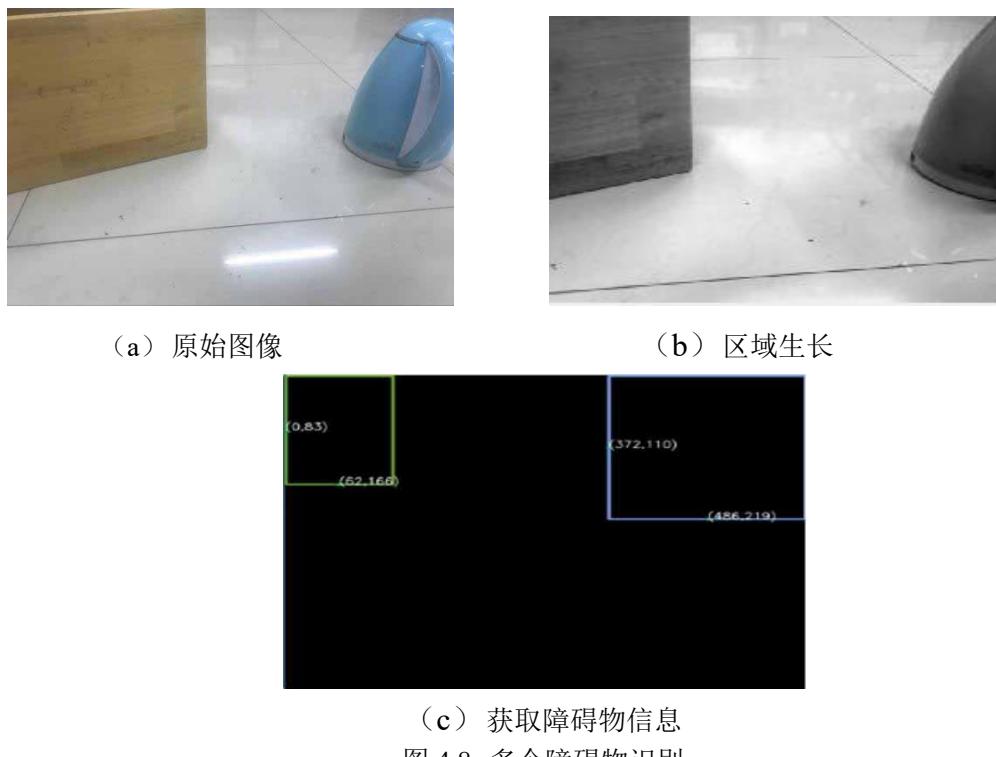


图 4.8 多个障碍物识别

4.4.3 障碍物区域确定

通过上节的分析，实现了对目标障碍物的分割以及矩形表示，这个矩形框也就是障碍物所能够占据的最大范围，换句话说就是将障碍物以矩形的方式进行识别。在前文的分析中提出了基于固定区域分割原理的避障算法，将障碍物的区域划分为五大部分。一旦障碍物目标出现在智能设备视野范围时，就将障碍物的所有区域标记为“1”。此时，就实现了障碍物所占据区域的标记，具体如下图 4.9 中（a）所示，其显示画面为各个区域存在障碍物的画面，随后智能运动车根据障碍物的区域分布自动选择运动方式。下图 4.9 中（b）-（d）表示障碍物在②，④区域时小车运动情况。图 4.9 中（e）-（g）表示障碍物在区域③时小车的运动情况。



(a) 各个区域有障碍物时效果图



(b) 障碍物位置A



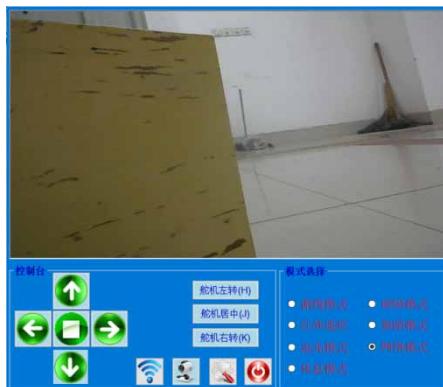
(c) 小车视角采集的画面



(d) 固定区域分割的判定



(e) 障碍物位置 B



(f) 小车视角采集的画面



(g) 固定区域分割的判定

图 4.9 固定区域分割原理测试效果

通过对上图中的 (d) 和 (g) 可知，在图 (d) 表示②，④区域有障碍物的情景，根据小车躲避障碍物的原理，此时区域①不存在障碍物，区域②存在障碍物，而区域③没有障碍物时，小车就会通过左转弯来实现对障碍物②的躲避。进一步观察小车实际运行线路可知，小车直行缺失受阻，需要通过左转弯来正常通行。在图 (g) 中所示为③区域存在障碍物，其他区域没有障碍物，此时障碍物不会影响小车正常的行驶，通过图片也可以明显看出此时的小车运行正常。所以，通过本方法能够实现对障碍物的高效识别和自动规避。

4.5 本章小结

本章节主要是研究小车在自动规避障碍物过程中的几种常见的位置关系为主进行分析，提出了基于固定区域分割原理的避障算法。借助于设备视觉技术来实现对道路中障碍物的识别，或者障碍物的相关特征参数。最后结合障碍物识别技术以及区域分割原理来实现小车自动识别障碍物和规避障碍物的功能。

5 智能小车系统的软件设计

软件设计包括应用程序和硬件驱动程序。硬件程序设计是给系统中的硬件模块设计驱动程序。应用程序设计是用户结合用户的实际需求，结合相关的操作平台以及软件技术等来实现本研究的设计。

5.1 上位机应用程序设计

在整个系统的设计过程中硬件平台的搭建只占了整个系统设计的一部分，还需要加载相关系统程序，否则智能小车只能完成简单的操作。整个系统是基于 STM32 智能小车的视频监控系统，需要用到的硬件模块种类多，系统结构复杂。因此，上位机的程序设计需要操作简单、界面功能强大，能够控制下位机和视频显示。当上位机应用程序启动时需要对整个系统进行初始化处理，并以无线 WIFI 信号实现与下位机的连接。用户通过在上位机上操作不同的按钮来发出控制指令，用于控制小车的运行，并且将小车的运行状态信息动态化地显示在终端设备上。

5.1.1 APP 程序设计概述

本文 APP 程序设计是在 Windows 系统开发环境下进行开发的，开发工具是 Qt Creator 4.4.1 软件，编程语言是 C++。上位机的功能是实现用户与智能小车之间的交互和视频信息显示。通过控制界面将用户的需求按照设定的通信协议发送给处理器，信息经过核心处理器处理后反馈给控制界面，并实时显示视频信息。APP 实现的功能包括控制小车的行驶、视频信息实时显示等。

APP 开发主要分以下几个部分：

- 1、UI 界面设计，利用 Qt 自带的 Qpointer 和 QPixmap 类对象和布局功能完成图片的加载设计。
- 2、网络连接，利用 Qt 自带的 QTcpsocket 类，完成网络通信。
- 3、信号与槽机制，通过按钮单击事件，触发相应槽函数的执行，从而实现面向对象操作。
- 4、事件处理方法，通过 Qt 自带的时间触发 event () 和事件过滤器 event-filter 对不同事件的判断和处理。

APP 使用的是 Qt 自带的数据接口 QSqlDatabase 实现的。中间的控制将接收到的数据进行逻辑解析，View 使用 QWidget 和 QGraphicsView 的框架来搭建 UI 效果。对于网络不稳定和数据丢包的问题，在程序中设置了相应的心跳包检测以及断线自动连接机制等一系列措施^[49, 50]。上位机软件设计流程图如图 5.1。

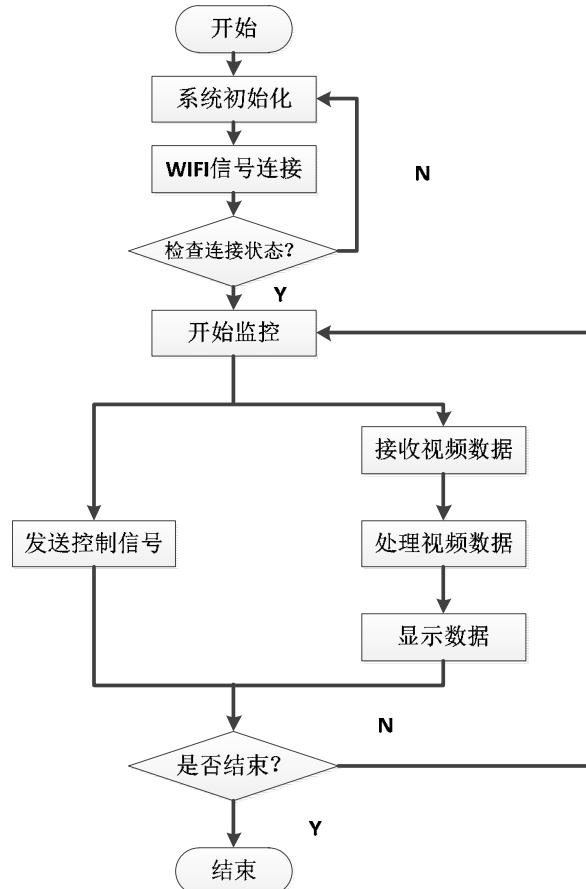


图 5.1 上位机软件设计流程图

5.1.2 APP 基本文件解析

```

QtAuto.pro // 工程统等文件
widget.h // 声明了一个类Widget
main.cpp // 函数入口
widget.cpp // 定义了Widget的类体
widget.ui // 可以编辑的图形界面

namespace Ui {
    class Widget;
}

class Widget : public QWidget
{
    Q_OBJECT

public:
    explicit Widget(QWidget *parent = 0);
    ~Widget();

private:
    Ui::Widget *ui;
};

#endif // WIDGET_H

```

图 5.2 APP 的基本文件

如图 5.2 为 Qt 设计的智能小车控制软件 APP 的基本文件，其中.pro 为工程文件，每次只需点这个文件就可以进入 Qt 软件设计界面。文件夹里包含许多的.h 头文件，头文件里面包括许多与设计 APP 相关的类。main.cpp 是所有函数的入口，而其他的.cpp 文件则是对具体的类进行分模块操作。.ui 文件用于美化 APP 界面。伪代码解释如下：

(1) widget.h 介绍

```

namespace UI{
    class Widget; //UI::Widget 声明
} // UI::Widget 类与::Widget 类不是同一个类
class Widget :public QWidget
{
    //::Widget Q_OBJECT
    .....
}

(2) widget.cpp 介绍
Widget::Widget(QWidget *parent)
    QWidget(parent); //指定父类构造函数执行的版本
    ui(new UI::Widget);
{
    //给成员变量 ui 赋值
    ui->setupUi(this); //给传入指针，所指代的控件，添加内置部件
}

(3) main.cpp 介绍
int main(int argc,char *argv[])
{
    //a 是关于 APP 的一个管理类对象
    QApplication a(argc,argv);
    Widget w; //w 是与 UI 相关， W 是一块白板
    w.show(); //显示 UI
    //int res=a.exec();
    //qDebug("%d",res);
    return a.exec(); //开启一个死循环
}

```

5.1.3 APP 的 UI 设计

在 APP 的开发通过 push button 按钮触发单击信号，执行相应的槽函数。在槽函数里获得用户输入的信息，打包成 JSON 格式的数据，通过 TCP socket 的 write 函数将数据发送给服务器，服务器对数据进行处理并将结果反馈回来，然后使用 TCP socket 的 read 函数接收数据，解析 JSON 格式的数据，并对接收的数据做出相应的判断，完成智能小车实时控制功能。

5.1.4 APP 的主要程序描述

(1) 画主界面，如图 5.3。

```
void SmartH_Widget::paintEvent(QPaintEvent *event)
```

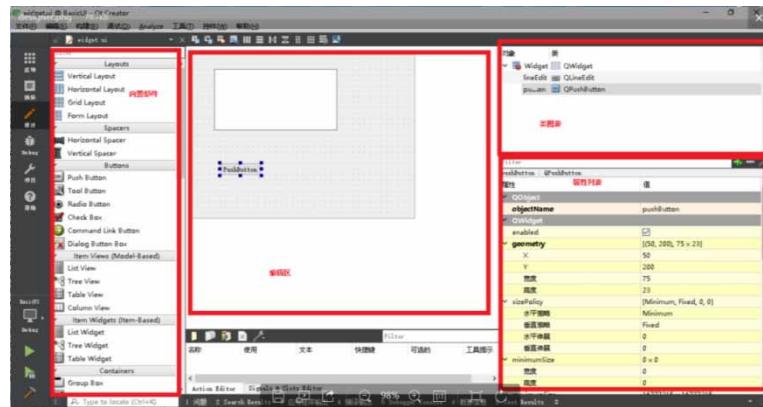


图 5.3 Qt 画图主界面

(2) 定义数组接收用户输入的 UP、DOWN、LEFT、RIGHT 和 STOP。

```
buf[BUFSIZE-2]=ui->lineEdit_UP->text().toStdString().c_str();
qDebug()<<"buf="<

(3) 将用户输入的信息通过 TCP 打包成相应的格式发送给服务器。


```

```
buf[BUFSIZE-2]=ui->lineEdit_UP->text().toStdString().c_str();
qDebug()<

```

(4) 如果服务器返回正确的信息，则跳转到用户界面。

(5) 接收服务器发送过来的数据，在用户界面上显示。通过 Qt 界面设计出来的效果如图 5.4。



图 5.4 上位机界面效果图

智能小车上位机机客户端主接口主要包括以下内容：

- 1、方向指令：前进、后退、左转、右转以及停止按键。
- 2、舵机方向的控制：左转、右转、居中。
- 3、功能模式选择：巡线模式、避障模式、红外模式、舞蹈模式、追光模式、网络模式、休息模式等。
- 4、WIFI 的连接、摄像头的开启按键、系统快捷键设置和系统启动按键。
- 5、菜单栏：快捷按键的设置、模式指令的设置和 WIFI 参数的设置。

5.2 Android 手机应用程序设计

Android 手机应用程序设计包括控制界面和视频界面。前者主要是控制小车不同的运行状态，例如向前行驶、向后行驶、向左转动、向右转动等各种操作。视频界面的设计是让用户可实时观察到小车周围的图像情况，并根据观察到的视频信息做出相应的动作。这两部分设计的内容构成一个完整的客户端。手机通过 WIFI 将控制命令发送给小车，使之执行相应的操作。Android 手机端的开发模块程序见附录 I。

在 Android 系统的应用程序设计中，各个组件之间信息传递的途径是以事件进行处理的。通过嵌套字 Socket 通信机制实现应用程序与系统的相互通信，利用特定的 IP 地址连接端口，获取的相关数据信息以 Send 函数进行传输，设备的连通域断开以 Close 函数进行控制^[51]。应用程序加载在 Android 手机上的视频界面和控制界面如图 5.5 和 5.6。



图 5.5 Android 视频界面图



图 5.6 Android 控制界面图

5.3 操作系统的移植

5.3.1 μC/OS-III 移植的条件

当利用 μC/OS-III 来对处理器的寄存器进行读写操作时，只能以汇编语言或者在 C 语言中嵌入汇编代码来实现。μC/OS-III 操作系统最主要的功能就是保证汇编部分

以及汇编代码与 C 代码进行正常的连接。为了保证 μC/OS-III 能够正常运转，必须要保证处理器的编译器具备以下特征：

- 1、能够产生可重入代码。
- 2、可以在程序中打开或者关闭。
- 3、可以产生定时中断（一般为 10Hz - 100Hz）。
- 4、可容纳特定容量数据的堆栈。
- 5、可以对堆栈指针和 CPU 寄存器下达操作指令。

经过编译之后的 μC/OS-III 内核大小通常为 6-10KB。STM32 内部包含一个 2M 的 RAM 和 2M 的 FLASH 程序存储器，可以进行定时中断。其内部包含 Keil μVision5 工具包，具有大量的可重入代码，还可以在 C 程序中嵌入汇编语句。因此，其可以实现 μC/OS-III 的移植。

5.3.2 μC/OS-III 移植的主要工作

根据 μC/OS-III 移植的相关要求，要想实现在新的体系上移植 μC/OS-III，必须依靠 os_cpu.h、os_cpu.c 和 os_cpu.a.asm 三个文件来实现。此外，还需要根据实际对 IRQ.inc 文件进行修改。IRQ.inc 具有一个可以提供一个 STM32 通用的中断服务程序与 C 函数接口的汇编代码，简称汇编宏。通过这个宏可以实现对各 IRQ 中断部分之前数据的存储，随后调用出对应的中断程序实现系统的中断，当中断程序结束之后，系统就会结合实际情况调用出中断前的数据并将其返回被终止任务中继续进行执行，也可以转移至级别更高的就绪任务上运行。

5.3.3 μC/OS-III 的文件移植

1、配置文件 cpu_cfg.h os_cfg.h lib_cfg.h os_cfg.app.h	2、用户应用程序 app.c app.h
3、μC/OS-III 与 CPU 无关的代码 os_int.c os_prio.c os_core.c os_flag.c os_mbox.c os_mem.c os_mutex.c os_q.c os_time.c os_tmr.c os_cfg.app.c os_type.h ...	4、库文件 lib_ansic.c lib_ansic.h
	5、μC/OS-III 与移植相关的代码 os_cpu.h os_cpu_a.asm os_cpu.c.c
	6、μC/OS-III 与 CPU 相关的代码 cpu_def.h cpu_c.c cpu_a.asm cpu_core.c cpu_core.h
7、其他 CPU 相关文件	

图 5.7 μC/OS-III 操作系统的文件结构

如图 5.7, μ C/OS-III 操作系统文件结构包含配置文件, 用户应用文件, 内核服务文件, 库文件, CPU 移植文件, CPU 配置文件, 其他 CPU 相关文件。每类文件具有不同的功能。

- (1) 配置文件: 主要是通过对文件中的宏进行定义来对 μ C/OS-III 的功能进行裁剪;
- (2) 用户应用文件主要是对各应用任务进行定义和声明;
- (3) 内核服务文件: 具体代码不受CPU影响, 可以移植到任何 CPU;
- (4) 库文件: 字符串的常规操作和常用的数学计算;
- (5) CPU 移植文件: 可以根据需要将 μ C/OS-III 移植到需要的平台上, 其具体代码主要是受到不同的 CPU 影响而存在差异;
- (6) CPU 配置文件: 例如 CPU 的常规工作模式以及服务函数;
- (7) 其它的 CPU 相关文件。

根据 μ C/OS-III 操作系统的文件结构, 操作系统的移植依赖于三个文件:
os_cpu.h、os_cpu.c.c 和 os_cpu_a.asm。

- (1) os_cpu.h 文件包含用户预编译定义的处理器常量、宏和类型, 系统需要根据不同的处理器对数据类型做出定义。
- (2) os_cpu_c.c 文件有 10 个函数与 μ C/OS-III 的移植有关: void OSTaskstInit(), void OSTaskCreatHook() , void OSTaskDelHook() , void OSTaskSwHook() , void OSTaskdleHook() , void OSTaskStatHook() , void OSTimeTickHook() , void OSTCBInitHook() , void OSInitHookBegin() , void OSInitHookEnd()。用户可以根据自己的需要如扩展 μC/OS-III 的功能, 在这些函数里面增加代码, 并在用户任务中可以对其进行系统的功能调用。

(3) os_cpu_a.asm 文件

当采用汇编语言来对 os_cpu_a.asm 文件进行编写时, 其包含的 3 个具体函数主要是受到系统移植的影响:

- ◆ OSStarHighRdy(): 首先运行具有较高级别的就绪任务;
- ◆ OSCtxSw(): 在进行不同任务切换过程中主要是通过一次软中断来实现;
- ◆ OSIntCtxSw(): 可以实现在中断服务程序中进行不同任务的切换。

1、OSStarHighRdy() 函数

在 μ C/OS-III 中需要启动多个任务环境的函数叫做 OSStart()。OSStart() 函数通过调用 OSStarHighRdy() 来保证已经就绪的较高级别任务进行运行。在高级任务运行之前, 需要将所有处理器寄存器按照一定的顺序从任务堆栈中进行恢复, 遇到中断操作时返回上一操作。这一函数具体的运行流程为以下几部分:

- ① 首先将 OSTaskSwHook() 函数调用出来;
- ② OSRunning = TRUE ;

③ 根据需要恢复运行的任务，获取其对应的堆栈指针：

SP = OSTCBHighRdy->OSTCBStkPtr；

④ 通过新任务堆栈来实现所有处理器寄存器的恢复；

⑤ 按照中断指令执行返回操作。

2、OSCtxSw() 函数 OSCTXSw() 的处理流程：

① 将处理器寄存器进行存储

② 将当前任务的堆栈指针存储在对应的任务控制块中：

_OSTCBCur->_OSTCBStkPtr = 堆栈指针；

③ 调用 OSTaskSwHook() 函数；

④ OSTCBCur = OSTCBHighRdy；

⑤ OSPrioCur = OSPrioHighRdy；

⑥ 获取需要恢复任务的堆栈指针：= OSPrioHighRdy->OSTCBStkPtr；

⑦ 将处理器的寄存器从新任务的任务堆栈中进行恢复；

⑧ 根据中断指令执行操作。

3、OSIntCtxSw() 函数

OSIntCtxSw() 主要是实现在中断服务程序中进行任务的切换，通过 OSIntExit() 函数进行调取。OSIntCtxSw() 函数与 OSCTXSw() 函数最大的不同就是前者在中断服务程序中已经将部分的 CPU 寄存器进行存储，无需在 OSIntCtxSw() 中额外开展这一操作。以下所示为函数 OSIntCtxSw() 的具体操作过程：

① 根据当前的堆栈模式，从中获取相应的 R12、PC；

② 将 LR、PC 存储在至当前任务堆栈中；

③ 将 R4 ~ R12 保存在当前任务堆栈；

④ 读取此时任务堆栈中的 R0 ~ R3；

⑤ 根据需要对堆栈指针进行调整；

⑥ 将其存储在此时的任务堆栈中；

⑦ 将 CPSR、OsEnterSum R0 ~ R3 分别存储在运行的堆栈中；

⑧ 按照 OSTCBCur->OSTCBStkPtr = SP 指令执行操作。

5.4 硬件驱动程序设计

5.4.1 无线数据传输程序设计

在对视频模块进行设计过程中，除了要对视频信息进行完整的采集以及压缩处理之外，还需要对视频信号进行实时传输来满足用户的查阅以及在客户端的显示。本研究中主要是通过 TCP/IP 协议来实现对视频信号的控制以及传输。在服务器端，该协议主要是为了满足用户的需求，并与客户建立远程连接进行数据的传输或者断

开，满足用户对数据的需求。对于用户来讲，该协议主要是实现与客户端的连接以及信息的沟通。本研究中的TCP/IP协议主要包含网络接口层、互联网层、传输层和应用层四层结构^[52]。

该程序设计中具体的参数类型可以根据与本地嵌套字连接目标的方式不同划分为三类：即通信对象、地址与端口号以及传输协议等^[53]。根据这些参数就能够对各个网络之间的通信进程有一个清楚的了解。然后经过路由器初始化处理之后，通过设置的IP地址来绑定对应的端口号，进而实现与连接对象的正常通信^[54]。本次设计中，视频接口选用型号为80，控制端口为2001。客户端在系统中通过判断网络是否成功连接，来判断监控数据的传输工作。无线数据传输程序设计如图 5.8。

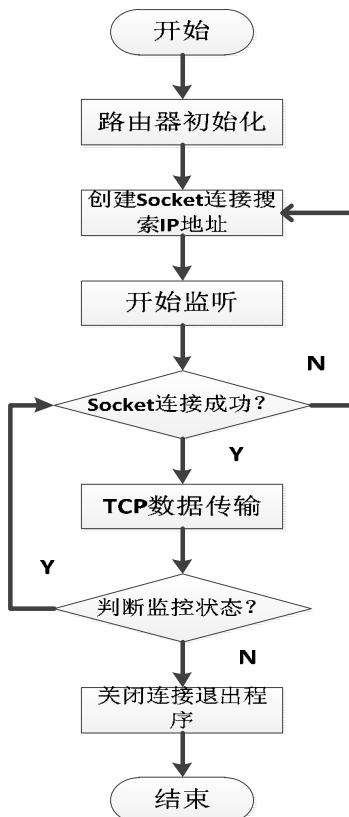


图 5.8 无线数据传输程序设计

5.4.2 电机驱动模块程序设计

本文以L298N驱动芯片来作为本系统的电机驱动模块，通过这一设计能够实现直接提供电压信号来保证电机的正常运转，并可以通过对电源输入信号的调节来改变电机运转的速度。从处理器里读取控制信号，并将对应的高低电平，接入系统接口，从而保证小车的有效运转。接口和小车运动情况如表 5.1。

表 5.1 接口和小车运动情况

电 机	旋 转 方 式	控制端 INT1	控制端 INT2	控制端 INT3	控制端 INT4	调试通过输入 PWM 信号改变脉宽实现	
						调速端A	调速端B
M1	正转	高	低	/	/	高	/
	反转	低	高	/	/	高	/
	停止	低	低	/	/	高	/
M1	正转	/	/	高	低	/	高
	反转	/	/	低	高	/	高
	停止	低	低			/	高

电机驱动小车转动的实质是改变电机的占空比，本设计小车是四轮车，电机需要驱动小车四个车轮的运转。电机驱动程序部分代码见附录 II。

电机驱动设计流程图如图 5.9。

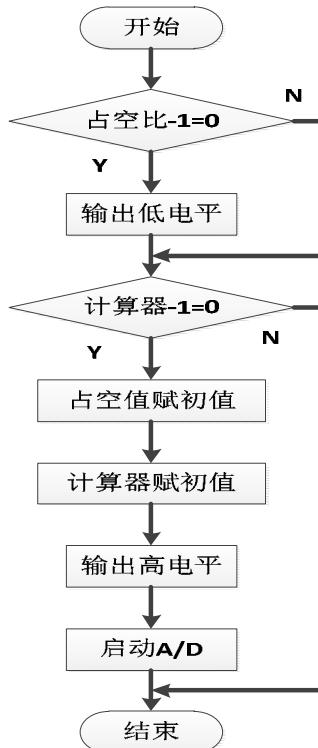


图 5.9 电机驱动设计流程图

5.4.3 视频采集程序设计

视频摄像头的作用机理是把光学讯号转成电讯号的进程，利用图像传感器把物镜采集到的图像转变成电讯号，之后再利用 A/D 转化为数字格式图像讯号，然后把数字格式图像讯号传递到摄像头，由其 DSP 处理芯片实施处置。该系统策划方案将无线路由器当作承载体，把数字类型摄像头具体的驱动程序添加到其中，主控制器下达

视频采集命令时，那么摄像头就会进入工作状态，把收集到的相关讯息利用无线路由 WIFI 讯号传输到显示窗口并呈现在上面^[55]。

该设计方案使用的是 HD1080P 高清模式带有 USB 接口的数字类型摄像头，这类摄像头内置有数字讯号处理芯片，可以把收集到视频讯息实施压缩处理并进行相应编码，把模拟讯号转变成数字讯号。依据视频传导体系收集端的功能需要，视频收集端的工作是对智能小车周围的环境视频讯息进行收集，然后进行压缩处理，最后把压缩数据传递出去。视频收集功能模块任务是利用摄像头内置的图像传感仪器收集具体的视频讯息，同时把收集到的视频讯息保存到压缩线程所具有的随机队列缓冲之中，以保证压缩线程处理视频讯息的及时性。当视频收集轻量进程顺利向压缩轻量进程队列缓冲区域存入一段初始视频讯息之后，压缩轻量进程则会进入工作状态。压缩轻量进程的关键作用是将收集轻量进程输而来的视频讯息依照 JPEG 格式标准进行压缩处理，之后传输到视频流讯息传输到缓冲区域的轻量进程队列当中，收集轻量进程的工作程序示意图如图 5.10 所表示。视频传输与接收等相关功能模块收集终端通过 USB 传输视频讯息，同时把它进行组合，形成一帧视频讯息方便解压功能模块运用，接收工作轻量进程流程图呈现为图 5.11。

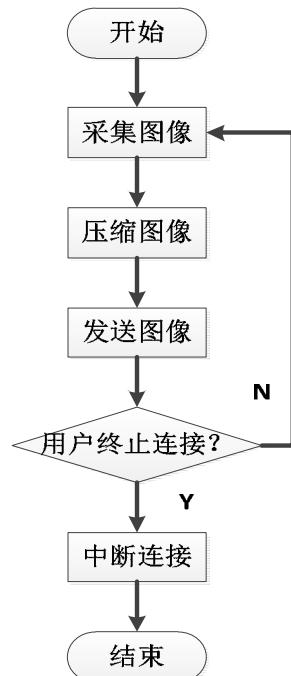


图 5.10 收集线程的工作流程图

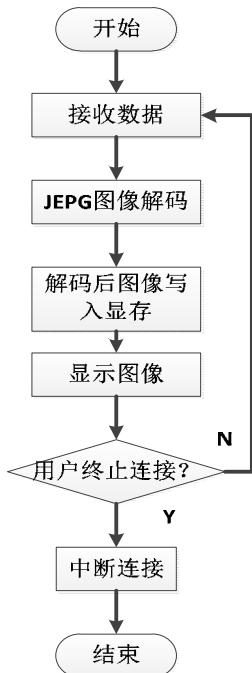


图 5.11 接收工作轻量进程流程图

依据以上办法收集到的视频数据，具有十分庞大的数据量。假如直接对其数据实施保存与传输，则必须占用极大网络带宽，所以必须在此之前对其实施压缩处理。本文所设计系统应用的是 JPEG 类型的图像压缩模式，一般情况下是通常软件设计加以实施的。对于 JPEG 而言，其是一类标准的图像压缩规范，共有三大类，分别是标准 JPEG 模式、渐进式 JPEG 以及 JPEG2000^[56]。JPEG 静态图像压缩规范能够对全部图像帧进行直接处置，压缩比例能够达到原数据的 1/20、1/80，这种压缩格式拥有的分辨率通常是固定的，而且图像处理质量特别高，与初始图像比较也很难区分开来^[57, 58]。

5.5 本章小结

本章是对系统的软件进行设计。针对系统硬件采用 μC/OS-III 操作系统并制定了详细的系统移植方案，并对本系统的一部分外部硬件装置设置了相关的驱动程序，然后加载到该系统控制端构成专门的 PC 上位机，Android 手机应用程序设计做了详细的描述。

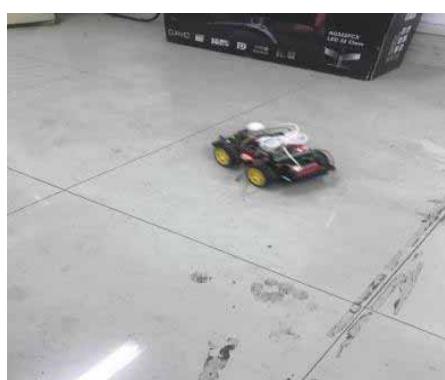
6 系统测试

6.1 系统硬件检测

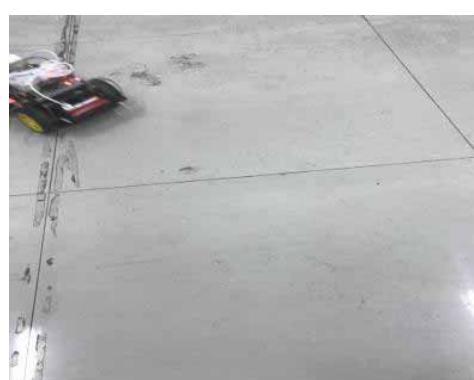
对硬件部分所实施的检测一般包含查看全部系统电路完好与否，线路连接正确与否。第一步需要对电机驱动实施检测，这项检测利用串口调试功能块实施检测，把编制的检测程序加载至主控器芯片当中，查看电机能否正常运转。之后检测小车向左、向右转弯以及向前与向后移动等各种状态是否能正常运行。再用串口检测小车摄像头，查看其工作正常与否，能否可以收集视频讯息，同时利用串口将数据传送到显示界面上呈现出来等相关功能是否正常。

6.1.1 电机驱动的测试

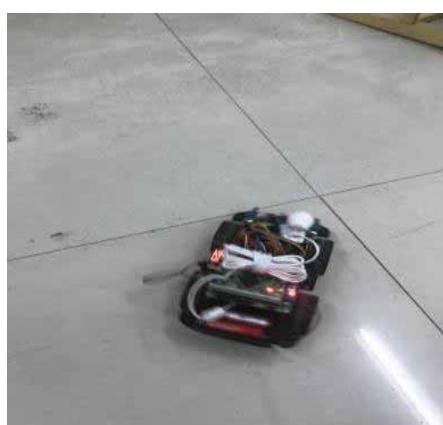
为了测试电机驱动的特性，本文编写一段让小车可以做出花样动作的例程，智能小车花样动作代码见附录 III。系统配置完成后，将编写的代码烧写到智能小车的开发板上，在烧写无误的情况下，即可启动小车。小车花样动作效果图如图 6.1。



(a) 小车向左转



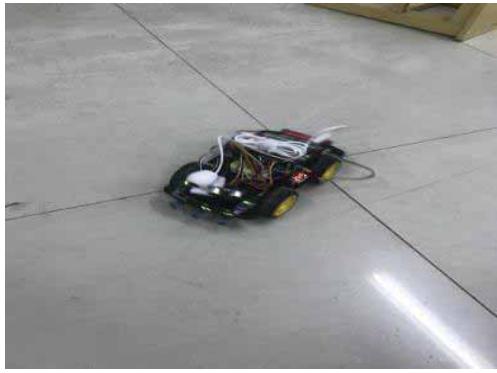
(b) 小车向右转



(c) 小车前进



(d) 小车后退



(e) 小车原地旋转

图 6.1 小车花样动作图

经测试，在智能小车启动后可以很清楚观察到，大约每隔 3s，智能小车就切换到不同的动作，如前进、后退、左转、右转、原地旋转等。经反复测试可知，智能小车电机都正常运转，达到了预期设定的效果。

6.1.2 无线路由器的测试

对无线路由器的测试，采用的网络调试工具是 TCP/UDP-UART 与 NetModuleConfig，其软件界面如图 6.2 和图 6.3。

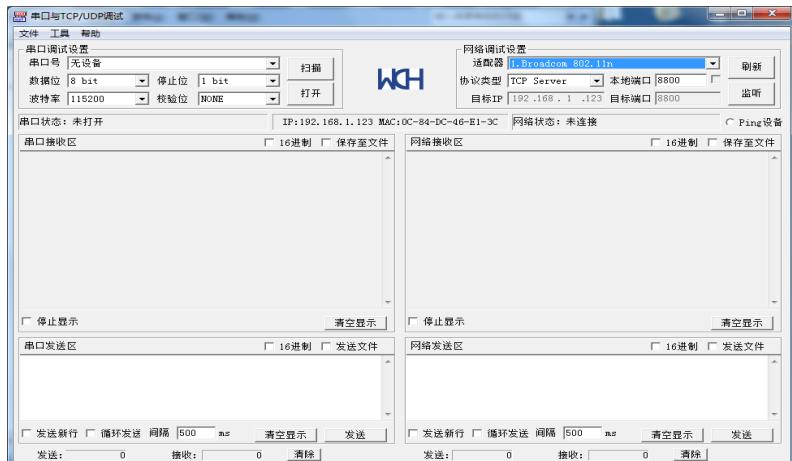


图 6.2 TCP/UDP-UAR 调试工具主界面



图 6.3 NetModuleConfig 调试工具主界面

具体无线通信测试步骤如下：

- ① PC 机上打开 TCP/UDP-UART 工具，可知当前 PC 的 IP 地址和端口（端口号可变），如图 6.4。



图 6.4 获取当前 PC 机的 IP 地址和端口

- ② 使用模块配置工具，搜索查询模块

此时 PC 和模块必须在同一局域网内，使用 NetModuleConfig 软件，点击“搜索设备”，设备列表会显示出子网内的模块；双击列表的设备可获取模块相关信息。如图 6.5。

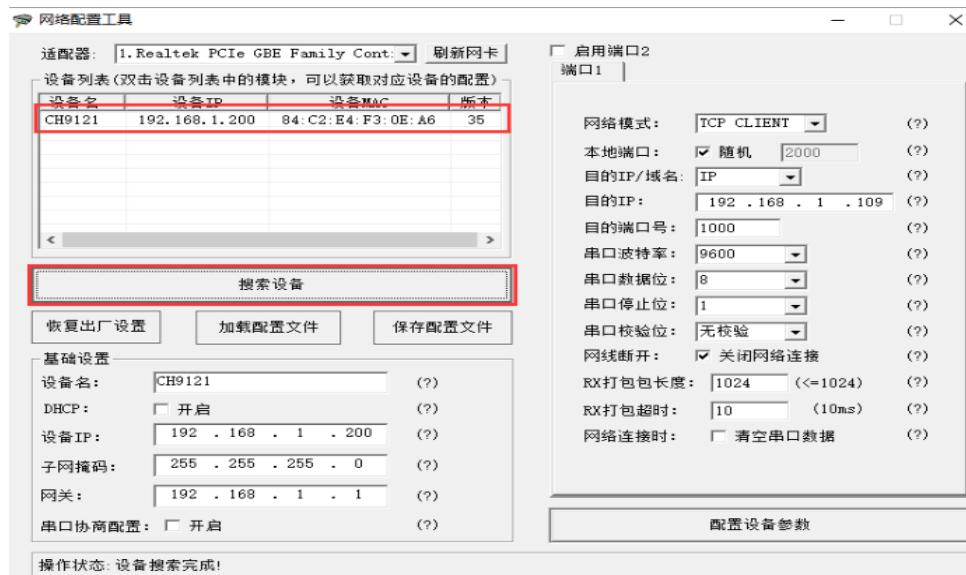


图 6.5 搜索设备

- ③ 配置模块的网络，串口信息。填写好相应的网络和串口信息后，点击“配置设备参数”，如图 6.6。

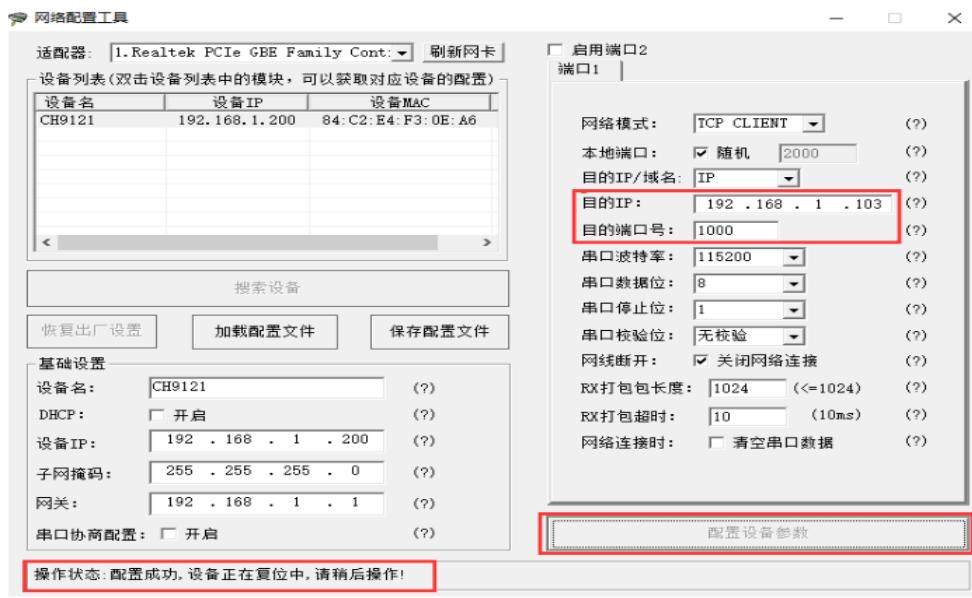


图 6.6 配置设备参数

④ 成功建立 TCP，在调试界面收发信息，在网络发送区发送 HELLO PC! 信息可在串口接收区收到 HELLO PC!，同理在串口发送区发送 HELLO ETHERNE! 信息可在网络接收区收到 HELLO ETHERNE!，测试结果如图 6.7。



图 6.7 通信信息的测试

经反复测试可知，每次在串口发送区发送的消息，大约经过 2s 左后，可在网络接收区收到相同的消息，同理在网络发送区发送的消息可在串口接收区收到相同的消息，表明系统通讯正常。

6.2 系统软件测试

在应用程序测试的环节必须保证 WIFI 信号稳定且信号强，这样系统才能保证数据收发无误。启动无线路由器后首先进行初始化，然后观察 PC 上位机或 Android 手

机控制端网络能否正常接入 WIFI 讯号。当显示界面提示 WIFI 连接成功时，表明无线路由器可以正常运转。具体需要测试的内容是：用户通过 PC 上位机或 Android 手机控制界面能否对小车进行远程控制与视频信息显示。

6.2.1 PC 上位机的测试

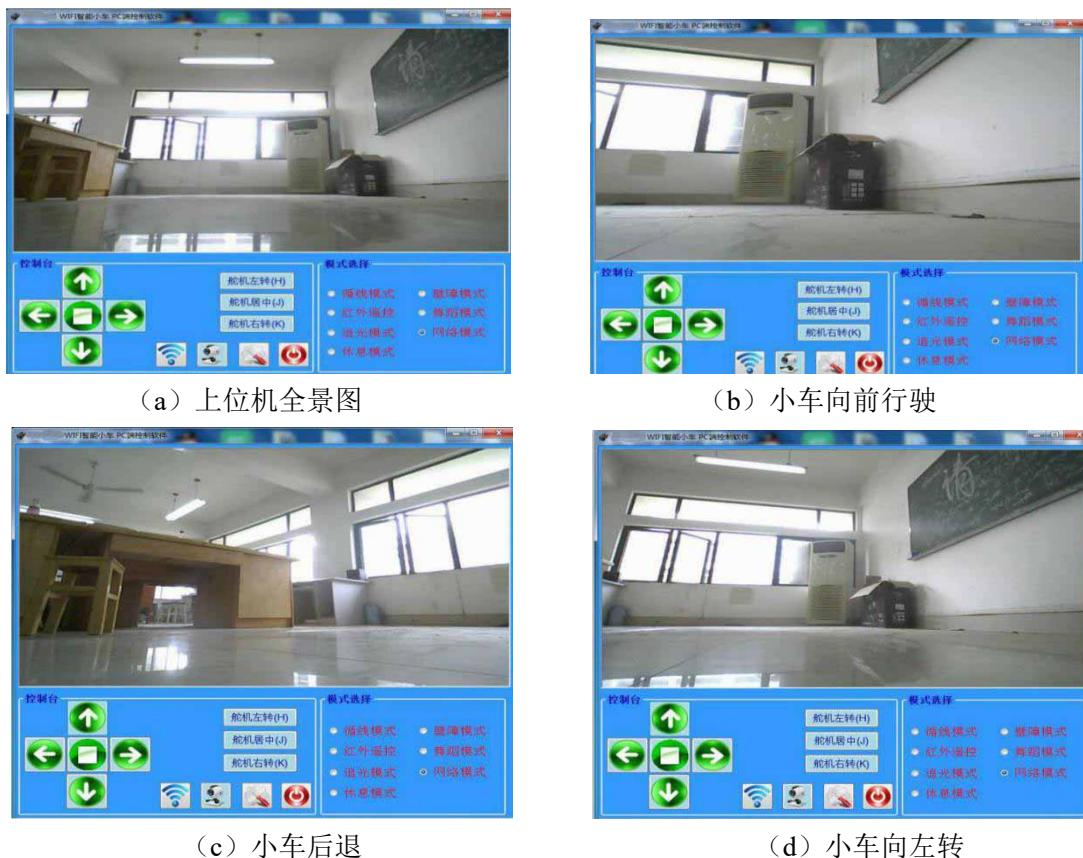
智能小车运动代码见附录 V。将编写的程序代码烧写到小车开发板上，烧写完毕后，将 ST-LINK V2 烧写器与电脑断开。开启智能小车控制板上面的开关，当无线路由器的绿色指示灯常亮，且摄像头蓝光亮，如图 6.8（a）和（b），将电脑与无线路由器通过 WIFI 连接，启动 PC 上位机智能控制软件。上位机测试的过程如图 6.9。



(a) 无线路由器灯亮

(b) 摄像头蓝光亮

图 6.8 路由器与摄像头





(e) 小车向右转

图 6.9 上位机测试过程

上述测试过程中，图（a）所显示的是小车所处的实验环境中；图（b）中表示的是小车接收到在前进的指令，向前行驶；图（c）中表示的是小车接收到在后退的指令，向后倒退；图（d）中表示的是小车接收到在左转的指令，左转弯行驶；图（e）中表示的是小车接收到在右转的指令，向右转弯行驶；通过上述实验可知，在 WIFI 信号稳定的情况下，用户通过上位机控制台对小车发送前进、后退、左转和右转的指令，同时在视频界面可以清晰地观察到小车周围环境信息，小车根据收到的指令可以做到相应的动作。经反复测试，在实验的环境范围内，WIFI 信号比较稳定，各项操作指令均可达到预期设定的效果。

6.2.2 Android 手机端的测试

Android 手机端智能小车测试的过程如下：

- (1) 启动 APP 配置应用程序的参数，智能小车如图 6.10。服务器 IP: 192.168.1.1；视频端口：8080；控制端口：2001
- (2) 设置完毕后点击连接即可进入手机操控界面如图 6.11。



图 6.10 手机 APP 登录界面



图 6.11 手机控制界面

- (3) 对智能小车实时监控测试过程如图 6.12。



图 6.12 手机测试实景图

上述测试过程中，图（a）所显示的是小车所处的实验环境中；图（b）中表示的是小车接收到在前进的指令，向前行驶；图（c）中表示的是小车接收到在后退的指令，向后倒退；图（d）中表示的是小车接收到在左转的指令，左转弯行驶；图（e）中表示的是小车接收到在右转的指令，向右转弯行驶；通过上述实验可知，在较好的 WIFI 讯号环境中，使用者可以利用控制窗口上面的控制按键操控小车的不同运动方向，同时小车前方的摄像头采集的视频信息也可以在视频界面实时显示，可达到预期的效果。

6.3 自动避障功能测试

为使小车拥有躲避障碍物的功能，第一步需要明确周围环境坐标系统。利用机器视觉科技采集目标障碍物相关特征讯息数值，同时采用区域分割算法对小车和目标障碍物的详细位置讯息加以确定。假如前方未出现障碍物抑或目标障碍物未能防碍小车前进时，那么小车就会用特定速度顺着原方向行进；假如目标障碍物影响到小车，则

小车会启动智能避障功能。在躲避障碍物时，小车行进方向会出现偏移，经过障碍物之后再回到原行进道路上来。

6.3.1 避障进程以及躲避障碍物后运行轨迹的复原

小车在道路上运行时，利用机器视觉科技辨别周围障碍物。运用上面简述的区域分割办法，辨别是否存在障碍物及其处位置。

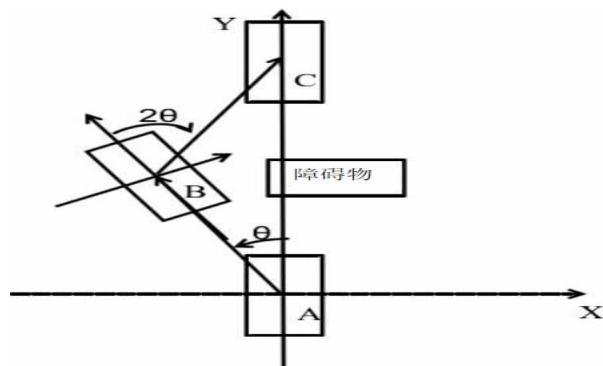


图 6.13 小车避障轨迹

如上图 6.13 所呈示，将小车所处中央点当作原点构建坐标系，Y 轴设定为其前进方向，X 轴设定为垂直于 Y 轴并且通过小车中央点的直线。假设小车起始位置是 A，其运行轨迹为由 A 点至 B 点最后再至 C 点。利用障碍物辨别功能，当其检测出右前方向存在障碍物时，小车则会左转，并且定时器同步开启进行计时。直至分割空间内，区域①与区域②中不存在障碍物为止，则小车不再转弯，停止计时，如上图表示，小车偏转了一个角度 θ 。小车行进 1 秒钟之后，则进行反向转弯，转弯持续时间两倍于前面转弯时间。之后判别有无成功绕过障碍物，假如未能发现障碍物就是持续行进 1 秒钟，然后向着和第一次转弯一样的方向移动，移动时间与上面记录时间相同。如此就顺利运行到 C 点。

此时，将转弯成功之后的行进时间设置成 1s，因为试验进行时，小车摄像头设置位置与角度影响，通过探测发现障碍物的间距往往控制在 30cm 范围内。这时，操控小车进行转弯行进 1s 后便可成功避让前方障碍物。一旦小车在 B 点就探测到由 B 至 C 的行进过程中会碰到障碍物，小车的行动轨迹将转变为 A 至 B 的运行方向，之后行进 1s 前进至另一个新的 B 点。直至行进 N 次之后新 B 至新 C 整个路程中经探测不存在任何障碍物，小车便可由 B 至 C 行进 $N \times 1$ 秒。

6.3.2 避障实验测试

文章中选用的障碍自动辨别小车显示为图 6.14。因为此系统选择的是机器视觉方式对障碍物展开辨别，故而路面本身颜色需尽可能统一，在花纹相对繁复的路面展开实验有可能会将花纹误判为障碍物，进而采取避障措施。考虑到路面环境的原因，此次试验选取的地点是实验室。



图 6.14 小车实物图

在试验过程中需要的各项常量参数分别为：

(1)车身：试验选择小车底盘长为 25cm，加之小车尾部的整个驱动模块长度总计在 $L=30\text{cm}$ 左右；小车整体宽度为 $W=14\text{cm}$ ；所选车轮半径为 $r=3.5\text{cm}$ ；小车速率控制在 $v=0.4\text{m/s}$ ；

(2)摄像头：在摄像头的整个安装期间首先需考量到其的设置高度与倾斜角度，这和摄像头所采集的整个路面环境讯息失真状况息息相关。本文试验中摄像头和地面之间的间隔高速为 $h=15\text{cm}$ ，采取俯仰角为 $\theta = 45$ 度，摄像头进行画面拍照的最远部分和小车最前端之间的间隔是 $a_2=40\text{cm}$ ，最近部分和小车最前端之间的间隔是 $a_1=10\text{cm}$ 。摄像头具体设置示意图显示为下图 6.15。

(3)图像采集频率：整体考量障碍辨识系统的数据处置速率与系统硬件设施，计算机中可以实现 25 帧/s 的图像处置速率，但考量到处置一幅图像基本需花费 0.23s 的时间，所以最终选定 4 帧/s 的图像采集方式。图像和小车最远边长之间的间距为 $h_2=37\text{cm}$ ，和最接近部分边长间隔 $h_1=20\text{cm}$ 。摄像头具体采集范畴属于俯视图，如图 6.16。

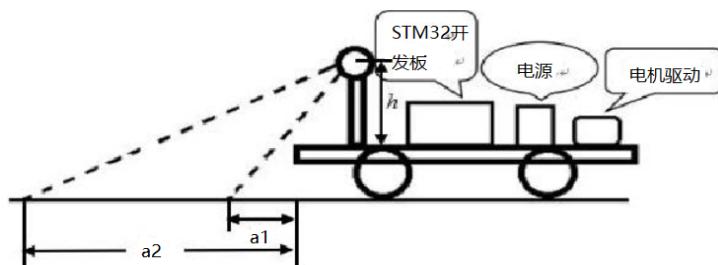


图 6.15 摄像头具体安设部位

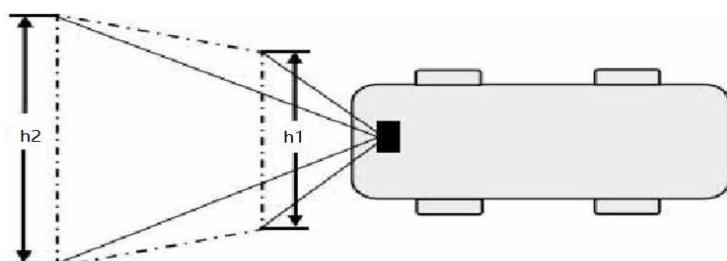


图 6.16 摄像头采集范围俯视图

在试验正式进行之前，第一项工作是于试验场所内设置部分较为常见的物品作为障碍物，障碍物本身形状规则与否不纳入考虑方位。之后，试验期间，小车可以检测到其行驶方向上所存在的障碍物讯息，经过自动识别系统判断障碍物讯息，选取的避

障手法为固定区域分割，逐渐避开目标障碍物。障碍物避让结束之后要再回归至原本的运动轨迹中。小车整个避障流程显示为图 6.17。

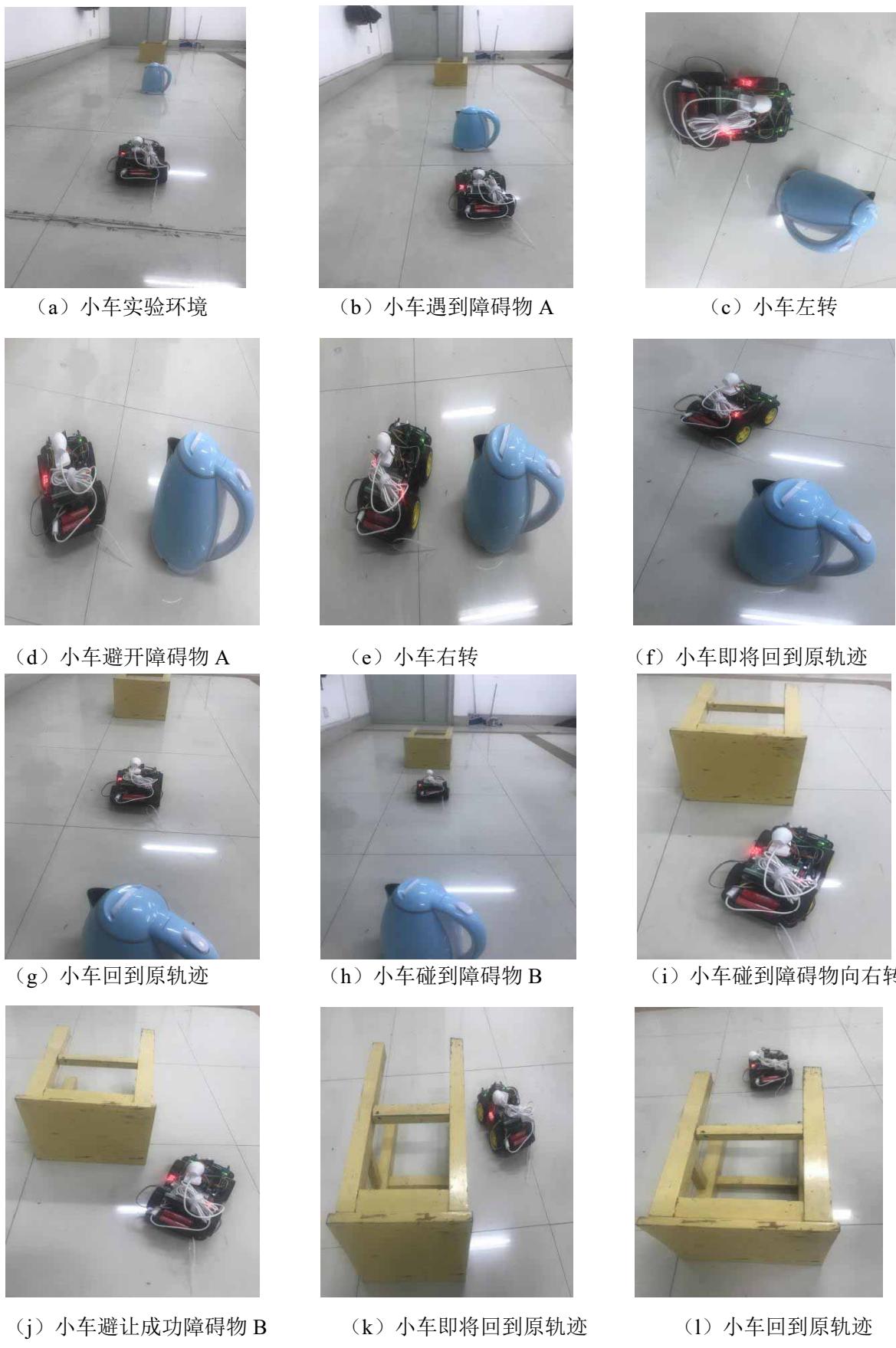


图 6.7 小车自动避障过程图

在上面所进行的小车自动避障整个流程中，图（a）中具体展示了整个避障试验进行的环境；小车往前行驶，前行途中碰到图（b）所显示的障碍物 A，小车通过检测在其右前方发现障碍物；在图（c）中小车对固定区域分割进行了合理运用，选择避障对策，即操控小车进行旋转，向左侧偏移到指定角度；图（d）展示了小车避让障碍物操作成功；在图（e）中小车需再次走回原本的运动轨迹中，故而向右偏移，进行转弯操作；图（f）、图（g）表明小车已经调整到原本的轨迹之上，并进行旋转，向左侧偏移到原本的行进方向上；图（h）说明小车在行使过程中碰到障碍物 B；图（i）表明小车利用检测方式在其左前方发现障碍物，并实施右转策略；图（j）展示了小车将左侧方障碍物避让成功的情况；图（k）表明小车需要再转回至原本的运动轨迹上；图（l）表明小车已经转移到原本的运动方式。通过以上试验能够表明，在此文中所选用的机器视觉技术，运用对固定区域进行分割设计的避障方式，可以快速达成小车的避障需要，这说明在一定程度上此方法的实用价值相对较高。

6.4 本章小结

本章重点就智能小车实施了系统测试，并利用相关试验检验其避障功能。一般来说，系统测试分为两个部分：硬件测试和软件测试。硬件测试的是电机驱动的正常运转和无线路由器通信的工作。软件测试包括通过 PC 上位机和 Android 手机客户端对小车的监控与摄像头采集到的视频信息在客户端界面显示、自动避障功能测试。测试结果表明验证了本文设计的智能小车的有效性。

7 总结与展望

7.1 总结

随着计算机与传感两项技术的迅猛发展，智能小车所涵盖的功能也将愈发强大，于一些特殊的环境中一些新型的智能小车反而比我们人类更适合工作。另外，机器人以及视频监控设备给人们工业生产和生活带来了极大的便利。基于以上需求背景本课题以 STM32 系列开发板作为核心控制器， μ C/OS-III 操作系统为软件平台，设计了一种具有自动避障功能的智能小车。

本文的内容主要工作和成果如下：

本文首先对国内外智能小车的研究现状有了深入的分析，提出了一种自动避障方案，该方案选用图像搜集模块来实时获得路面环境的图像讯息，运用区域生长的方式对目标障碍物的具体数据进行采集。依据智能小车自身与障碍物之间位置关系提出了基于固定区域分割思想的避障算法。其次，阐述了系统软硬件开发平台和系统的测试平台。另外，通过阐述 WIFI 技术的优点，来论证 WIFI 技术能实现远程信号实时传输的可行性。本文设计的是一种具有自动避障功能的智能小车，亮点是本文提出了一种有效的自动避障方案使得智能小车可以自动避开障碍物。另外，用户可通过 PC 上位机或者 Android 手机对小车进行控制和监控。

再者介绍了系统的设计，系统包括系统软件设计和硬件设计等两个方面。其中硬件设计主要包含了电机驱动、视频搜集、无线路由器、STM32 最小系统和电源等多个模块的设计。系统软件的设计包括应用程序的设计、 μ C/OS-III 操作系统的移植、硬件驱动程序的设计和自动避障算法。应用程序的设计是 PC 上位机和 Android 手机控制端的设计，用户可通过这两者中的任意一种客户控制端远程操控小车。

最后系统的测试，测试包括：每个子模块的功能测试、整机测试和自动避障测试。测试结果表明：在稳定的 WIFI 信号情况下，用户可以通过控制终端（PC 上位机或 Android 手机端）实现对小车的实时控制和视频实时动态监控。另外，小车能够自动避开障碍物。

7.2 展望

在本次设计的过程中，由于本文内容所涉及到理论的复杂性以及本人知识的局限性，在智能小车功能的实现上有所欠缺，需要进一步的完善。具体包括下列几点：

(1) 避障小车本身移动速率受到一定限制：因为 STM32 结构处理器与电脑 CPU 处理速率之间存在较大差别，避障算法于 STM32 开发板上实施处置的速率受到了限制，所以智能小车运转期间的速率不可过大，速率太快极易造成小车难以快速闪避障碍物并和障碍物产生碰撞的现象。

而此文在设计中，将小车发现障碍物之后的状态设定为暂停前进，完成转弯之后保持前进的态势，如此设计在一定程度上确保了避障过程中的安全度，却也损耗了很多时间。

(2) 本设计的避障实验所选取的障碍物是静止而非移动的，还未完全能够模拟在真实环境上的避障实验，在这方面还需要进一步的改进实验和完善。

(3) 本论文的用户界面设计不够友好，在以后的计划中，可改善界面的操作性。

(4) 智能小车的障碍自动识别系统的实时性和精确性，还有待进一步的提高。

参考文献

- [1] 徐国华, 谭民. 移动机器人的发展现状及其趋势[J]. 机器人技术与应用, 2015, 8 (03): 7-14.
- [2] 李宁. 基于 MDK 的 STM32 处理器开发应用[M]. 北京: 北京航空航天大学出版社, 2009.
- [3] 董涛, 刘进英, 蒋苏. 基于单片机的智能小车的设计与制作[J]. 计算机测量与控制, 2009, 17 (2) : 380-382.
- [4] 易礼智. 基于机器视觉的避障智能小车系统研究[D]. 湖南: 中南大学, 2012.
- [5] 钱信. 基于 ARM 的移动机器人设计及避障研究[D]. 江西: 南昌大学, 2012.
- [6] 杨敬辉, 洪炳容, 朴松昊. 基于遗传模糊算法的机器人局部避障规划[J]. 哈尔滨工业大学学报, 2004, 36 (7) : 1-3.
- [7] 董宗祥, 石红瑞, 杨杰. 嵌入式智能小车测控系统的设计与实现[J]. 计算机测量与控制, 2013, 7 (02): 357-359.
- [8] 任亚楠, 贾瑞清, 何金田. 基于超声波传感器的移动机器人避障系统研究[J]. 中国测试, 2012, 8 (03): 30-33.
- [9] 刘崇翔, 高美凤. 基于多超声波信息融合的小车避障算法实现[J]. 微计算机信息, 2012, 11 (08) : 60-62.
- [10] 田运暖. 智能家居中无线音视频监控系统的研究与实现[D]. 北京: 北京工业大学, 2015.
- [11] 周航慈, 吴光文. 基于嵌入式实时操作系统的程序设计技术[M]. 北京: 北京航空航天大学出版社, 2008.
- [12] 姚佳. 智能小车的避障及路径规划[D]. 南京: 东南大学, 2005.
- [13] 杨敬辉, 洪炳容, 朴松昊. 基于遗传模糊算法的机器人局部避障规划[J]. 哈尔滨工业大学学报, 2014, 36 (7) : 1-3.
- [14] S.Thrun. Learning Metric-Topological Maps for Indoor Mobile Robot Navigation [J]. Artificial Intelligence, 1998, 99 (1) : 21-71.
- [15] K. Chong, L. Kleeman. Mobile Robot Map Building from an Advanced Sonar Array and Accurate Odometry[J]. Intelligence Journal of Robotics Research, 1999, 30 (18) : 20-36.
- [16] 桂仲成, 吴建东. 全球机器人产业现状趋势研究及中国机器人产业发展预测 [J]. 东方电气评论, 2014, 4 (8) : 4-10.
- [17] Loanno-Perez, Wesley T, M. A . An algorithm for Planning Collision-Free Path among Poly-Hedral Obstacle Common ASS[J]. Computer Math, 1993, 30(22): 560-570.

- [18] Wang Y D, Nie L N. Design of Smart Car Speed Control System Based on Fuzzy Control[C]. IEEE International Conference on Artificial Intelligence and Computational Intelligence, 2010, 2 (6) : 516–519.
- [19] Zhang H, Liu Y G. Control Strategy Design for Smart Car Auto-tracing with Visual[C]. IEEE the 26th Chinese on Control and Decision Conference, 2014, 4 (10) : 4221–4225.
- [20] 王晶. 智能小车运动控制技术系统的研制[D]. 武汉: 武汉理工大学, 2009.
- [21] 付梦印, 李杰, 邓志红. 一种适于车辆导航系统的快速地图匹配算法[J]. 北京理工大学学报, 2005, 25 (3) 225–226.
- [22] 董涛, 刘进英, 蒋苏. 基于单片机的智能小车的设计与制作[J]. 计算机测量与控制, 2009, 17 (2) : 380–382.
- [23] 卢威. 智能小车避障系统的设计与实现[D]. 江西: 南昌大学, 2002.
- [24] 郑海华. 基于视觉的平地探测机器人避障研究[D]. 湖南: 中南大学, 2004.
- [25] 徐有春. 智能车辆视觉的 GPS 综合导航方法的研究[D]. 吉林: 吉林大学, 2001.
- [26] 沈明霞, 姬长英. 基于纹理频谱的农田景物区域检测[J]. 农机化研究, 2000, 8 (3) : 43–47.
- [27] 陆建业. Vision-based Real-time Road Detection in Urban Traffic[C]. Proc. SPLE Real-Time Imaging VI, Nasser Kehtamavaz, 2002, 4666 (6) : 75–82.
- [28] 程洪, 郑南宁. 基于主元神经网络和 K 均值的道路识别算法[J]. 西安交通大学学报, 2003, 37 (8) : 812–815.
- [29] 周欣, 黄席樾, 黎翌. 基于单目视觉的高速公路车道保持与距离测量[J]. 中国图形学报, 2003, 8 (5) : 590–595.
- [30] 刘国栋, 谢宏斌, 李春光. 动态背景中基于遗传算法的移动机器人路径规划的仿真[J]. 西安石油学院学报, 2004, 11 (3) : 24–29.
- [31] 冯琦, 周德云. 极坐标下基于遗传算法的路径规划方法[J]. 机械科学与技术, 2004, 23 (5) : 45–48.
- [32] 周明, 孙树栋, 彭炎午. 使用遗传算法规划机器人路径[J]. 西北工业大学学报, 1998, 16 (4) : 58–83.
- [33] 罗熊, 潘晓平, 易最. 具有大量不规则障碍物的环境下机器人路径规划的一种新型遗传算法[J]. 机器人, 2004, 8 (26) : 23–24.
- [34] Swapnil P. Karmore, Anjali R. Mahajan. Universal Methodology for Embedded System Testing[C]. IEEE International Conference on Universal methodology for embedded system testing Computer Science, 2013, 9 (16) : 567–572.
- [35] 汪波. 智能小车避障与路径优化研究[D]. 重庆: 重庆理工大学, 2015.
- [36] 宋威, 黄进. 基于 WIFI 物联网的家电智能控制系统信息控制端的研究[J]. 信息通信, 2013, 10 (01) : 199–200.

- [37] 赵印伟, 谢胜利. 基于 AT91RM9200 的嵌入式网络摄像头的设计与实现[J]. 微计算机信息, 2006, 8 (04): 26–28.
- [38] 王永虹. 徐伟, 郝立平. STM32 系列 ARM Cortex-M3 微处理器原理与实践 [M]. 北京: 北京航空航天大学出版社, 2008.
- [39] 戈新良, 杨杰. 基于多传感器融合的机器人障碍物检测和识别[J]. 红外技术, 2015, 12 (8): 695–698.
- [40] Shin I, Min SL. Embedded System Design Framework for Minimizing Code Size and Guaranteeing Real-time Requirements[J]. IEEE Real-time Systems Symposium, 2014, 24 (4) : 201–211.
- [41] Hung Sui Aim. An Efficient Localization Algorithm for Mobile Robots Based on RFID System[J]. International joint International on sice-icase, 2007, 25(4): 5945–5950.
- [42] 伊国栋. ARM 的嵌入式远程视频监控系统的设计 [J]. 电子技术与软件工程, 2014, 8(15): 40–45.
- [43] Jean Lab. The Real-time Kernel Second Edition[M]. Lawrence, Kansas: R&D Publications, 2012.
- [44] 周霞, 何建忠, 王宝磊. 多传感器信息融合伺服的移动机器人快速跟随[J]. 计算机工程与设计, 2013, 8(06): 20–22.
- [45] Wang L C, Long H. Algorithm on lane changing and tracking control technology for intelligent vehicle[C]. Proceedings of the 2007 IEEE International Conference on Robotics and Bimetallics, 2007, 30 (6) : 1888–1893.
- [46] 李婕. 基于 STM32 的无线视频监控智能小车设计[D]. 兰州: 兰州理工大学, 2014.
- [47] 张润景, 张丽敏, 王伟. Alter Designer 原理图与 PCB 应用设计[M]. 北京: 电子工业出版社, 2009.
- [48] Atmega48 8-bit micro controller with 8K Byte In-system programmable Flash, data sheet[Z], Oatmeal, 2013, 36 (2) : 120–125.
- [49] 宋威, 黄进. 基于 WIFI 物联网的家电智能控制系统信息控制端的研究[J]. 信息通信, 2013, 27(7): 199–201.
- [50] 程志江, 李剑波. 基于模糊控制的智能小车控制系统开发[J]. 计算机应用, 2008, 14 (12): 350–353.
- [51] 于金霞, 王璐. 位置环境下移动机器人自定位技术[J]. 北京: 电子工业出版社, 2010, 1(6): 357–369.
- [52] 吕霞付, 罗萍. 基于光电传感器的智能车自动循迹系统设计[J]. 压电与声光, 2012, 33 (6): 939–942.

- [53] Li Q. Spring-robot: a prototype Autonomous Vehicle and its Algorithms for Lane Detection[J]. IEEE Transaction on intelligent transportation systems, 2014, 5 (4): 300–380.
- [54] CC1101 Single-Chi o Low cost power RF-transceiver data sheet[Z], Texas instruments, 2012, 02 (8) : 265–270.
- [55] Manchurian R, Matthews L. Terrain Perception for Demo III[C]. Proceedings of the IEEE Conference on Intelligent Vehicle Symposium, 2013, 20 (6) : 326–331.
- [56] 陈宏田, 田恩刚, 彭大成. 基于 Android WIFI 和物联网的智能车设计[J]. 电子世界, 2014, 10(12): 65–70.
- [57] 任明伟. 基于 ARM 的嵌入式视频采集与远程传输系统的设计与实现[D]. 河北: 河北大学, 2011.
- [58] 徐文龙, 郭宝龙. 数字视频压缩编码的国际标准明[J]. 计算机应用研究, 2003, 7(04): 1–5.

致 谢

时光飞逝，二年的研究生生活即将结束，在即将毕业之际，我想对我攻读硕士期间给予关心、帮助、鼓励和支持过我的老师、同学、朋友和亲人致以最诚挚的感谢和最衷心的祝福！在他们全力的支持与尽心的帮助下，我顺利的完成自己二年的研究生学业，谢谢你们！

在这里首先感谢我的导师徐震副教授，在此毕业设计制作和毕业论文撰的写过程中，遇到了很多困难，都是在徐老师的指导和支持下才能顺利的克服这些困难；此外还要感谢其他学科老师，在我研究生期间对我的学业、生活各方面都给予了很大的关心和帮助。我向各位帮助和指导过我的老师表示最真诚的感谢！

感谢我的同学和朋友，我在学校学习期间的很多问题都是同学和朋友帮我解决以及给予我很大的支持，在此由衷的对你们说声谢谢！本文在写作期间参考了大量的文献资料，感谢提供这些文献资料的学者和前辈们。

最后，我要谢谢我的父母，正是他们在背后默默的支持和无私的奉献，我才会顺利的度过研究生生涯，在这里我要将最崇高的敬意献给你们！由于自己个人学术水平有限，所做的设计与所写的论文有不足之处，希望各位老师与前辈们给予批评指导，谢谢！

附录 I

Android 手机端的开发模块程序：

```
public Socket socket ;
String Csite= “192.168.1.1” ;
int Cport=2001;
public void sendMsg(String msg)
{
    try
    {
        socket=new Socket(Csite,cport);
        PrintWrite out =new;
        PrintWrite(socket.getOutputStream());
        out.print(msg);
        out.flush();
        Log.v(“WT”, “发送成功！ ” +msg);
        socket.close();
    }
    catch(Exception)
    {
        Log.v(“WT”, “发送失败！ ” );
    }
}
```

附录 II

电机驱动程序部分代码：

```
void CarMove(void)
{
    // 驱动小车左前轮
    if(front_left_speed_duty > 0) //驱动车轮向前转动
    {
        if(speed_count < front_left_speed_duty)
        {
            FRONT_LEFT_GO;
        } else
        {
            FRONT_LEFT_STOP;
        }
    }
    else if(front_left_speed_duty < 0) //驱动车轮向后转动
    {
        if(speed_count < (-1)*front_left_speed_duty)
        {
            FRONT_LEFT_BACK;
        } else
        {
            FRONT_LEFT_STOP;
        }
    }
    else //停止转动
    {
        FRONT_LEFT_STOP;
    }

    //驱动小车右前轮
    if(front_right_speed_duty > 0) //驱动车轮向前转动
    {
        if(speed_count < front_right_speed_duty)
        {
            FRONT_RIGHT_GO;
        } else //停止
        {
            FRONT_RIGHT_STOP;
        }
    }
    else if(front_right_speed_duty < 0) //驱动车轮向后转动
```

```
{  
    if(speed_count < (-1)*front_right_speed_duty)  
    {  
        FRONT_RIGHT_BACK;  
    } else //停止  
    {  
        FRONT_RIGHT_STOP;  
    }  
}  
else //停止  
{  
    FRONT_RIGHT_STOP;  
}  
...  
...
```

附录 III

智能小车花样动作代码：

```
int main(void)
{
    delay_init();
    IRControlInit();
    TIM2_Init();
    MotorInit();
    ServoInit();
    while(1)
    {
        if(tick_5ms >= 5)
        {
            tick_5ms = 0;
            tick_200ms++;
            if(tick_200ms >= 40)
            {
                tick_200ms = 0;
            }
        }
        CarGo();
        Delay_ms( 2000 );
        CarBack();           //小车后退
        Delay_ms( 2000 );
        CarLeft();          //小车左转
        Delay_ms( 2000 );
        CarRight();         //小车右转
        Delay_ms( 2000 );
        CarStop();          //小车停止
        Delay_ms( 2000 );
    }
}
```

附录 IV

智能小车避障功能代码：

```
void SearchRun(void)
{
    //三路都检测
    if(Search_M_IO==Black_Area    &&    Search_L_IO==Black_Area    &&
Search_R_IO == Black_Area)
    {
        ctrl_comm = COMM_UP;
        return;
    }
    if(Search_R_IO == Black_Area)          //右
    {
        ctrl_comm = COMM_RIGHT;
    }
    else if(Search_L_IO == Black_Area)    //左
    {
        ctrl_comm = COMM_LEFT;
    }
    else if(Search_M_IO == Black_Area)    //中
    {
        ctrl_comm = COMM_UP;
    }
}

...
switch(ctrl_comm)
{
    case COMM_UP:      CarGo();break; //小车前进
    case COMM_DOWN:   CarBack();break; //小车后退
    case COMM_LEFT:   CarLeft();break; //小车左转
    case COMM_RIGHT:  CarRight();break; //小车右转
    case COMM_STOP:   CarStop();break; //小车停止
    default : break;
}
...
```

附录 V

智能小车运动代码：

```
int main(void)
{
    ...
    while(1)
    {
        if(tick_5ms >= 5)
        {
            tick_5ms = 0;
            tick_200ms++;
            if(tick_200ms >= 40)      {tick_200ms = 0;}
            continue_time--;         //200ms 无接收指令就停车
            if(continue_time == 0)    {continue_time = 1;CarStop();}
            //do something
            if(bt_rec_flag == 1)     //接收红外指令
            {bt_rec_flag = 0;
                switch(ctrl_comm)
                {
                    case COMM_UP:      CarGo();break;
                    case COMM_DOWN:    CarBack();break;
                    case COMM_LEFT:    CarLeft();break;
                    case COMM_RIGHT:   CarRight();break;
                    case COMM_STOP:    CarStop();break;
                    default : break;
                }
            }
        }
    }
}
```