

广州大学

本科毕业论文(设计)

课题名称 基于机器视觉技术的除草机器人设计

学 院 机械与电气工程学院

专 业 机械设计制造及其自动化

班级名称 机械 122 班

学生姓名 赖浩滔

学 号 1207200070

指导教师 刘镇章

完成日期 2016 年 5 月 25 日

教 务 处 制

基于机器视觉的除草机器人设计

机械设计制造及其自动化专业 机械 122 班 赖浩滔

指导老师：刘镇章

摘要 杂草是农业作物的天敌，目前除草的工作主要还是依靠农药除草或者需要人工参与的机械除草，这两种方式要么对环境不友好要么需要耗费宝贵的人力劳动。本文针对如何实现智能化的机械除草，提出一种基于机器视觉的除草机器人设计。本次设计的机器人整体结构简单实用，利用由 Halcon 开发的机器视觉系统，配合负责获取照片和消息中转的 Android 设备以及机器人运动控制中心 AVR 芯片，可以全程无需人工干预地定位以及拔除杂草。另外，本文还应用了两种在机器人领域应用广泛的通信方式，分别是基于 TCP 的 Socket 通信以及无线蓝牙通信。这两种通信方法的合理使用，有效提升了机器人的工作效率。

关键字 除草机器人，并联机械手，机器视觉，Socket 通信，无线蓝牙通信

ABSTRACT Weeds is the natural enemy of crops. Currently, extirpating weeds mostly rely on chemical method or mechanism which need human to handle. These two methods either do damage to the environment or waste the labor. This study focus on design a kind of intelligent weeding robot based on mechanical vision which can locate the weeds and kill them automatically.

The mechanical vision technique adopted in this study is using Halcon package. The vision data provided by an Android device with a self-development app will be transmitted to the PC to process. After processed, the location of the weeds will be send back to Android who will tell the AVR to control the robot to the right place to weeding.

Besides, this study also apply two kind of popular communication patterns which widely used in robot domain. Precisely, there are TCP Socket communication and Bluetooth communication. With rational use of these two communication methods, the efficiency of the robot will be greatly improved.

KEY WORDS weeding robot; parallel manipulator; mechanical vision; socket communication; Bluetooth communication

目 录

1	前 言	5
1.1	背景介绍.....	5
1.2	研究现状.....	5
1.2.1	机器视觉的研究现状.....	5
1.2.2	除草机器人的研究现状.....	7
2	方案分析.....	8
2.1	功能分析.....	8
2.2	视觉系统可行方案.....	9
2.3	机械系统可行方案.....	10
2.4	电控系统可行方案.....	12
2.5	方案确定.....	13
2.6	本章小结.....	14
3	机械系统设计.....	14
3.1	机器人结构设计.....	15
3.2	并联机械手结构设计.....	16
3.3	并联机械手运动学分析.....	18
3.4	除草夹结构设计.....	22
3.5	除草夹的运动学分析.....	24
4	电控系统设计.....	25
4.1	电路设计.....	25
4.1.1	电子元件简介.....	25
4.1.2	主控芯片的引脚分配.....	28
4.1.3	电路原理图.....	28
4.2	电机驱动.....	29
4.2.1	直流电机驱动.....	29
4.2.2	舵机驱动.....	31

4.2.3	步进电机驱动.....	33
4.3	通信.....	34
4.3.1	Android 与 PC 通信	34
4.3.2	Android 与 Arduino 通信	41
5	视觉处理程序设计.....	44
5.1	图像获取.....	44
5.2	图像处理.....	46
6	结论.....	54
	致谢.....	55
	参考文献.....	56

前言

1.1 背景介绍

我国是一个农业大国，在农田当中杂草是农业生产的大敌，它们直接与农作物争夺养分、水分和阳光。从而导致农作物的产量减少，经济效益下降。全世界范围内，大约有 3000 中杂草，其中 1800 种可能影响农作物的收成。每年，我国都要花费大量的人力、物力和财力对杂草进行拔除。因此，研究如何高效、快速和准确的除草具有重要的现实意义。

目前，我国采取的除草方法大致可以分为三种：人工除草、机械除草和化学除草。其中，人工除草效率低下，需要大量的劳动力，浪费时间较多，除了在部分落后的地区或者面积较小的耕地的情况下，已经被逐渐淘汰。机械除草主要是利用机械工具割草或者拔草，这种方式和人工除草的原理相同，但是利用了机械化的工具来提高效率。最后，化学除草是利用除草类药剂杀灭杂草，虽然效率比较高，但是也容易影响农作物的生长，或者对环境产生污染。21 世纪，农业的发展不单止要追求产量、效率和效益，同样还要顾及和环境、生态、资源和人口的协调。在这种思想的影响下，出于对环境的担心，化学除草的方法已经在逐渐被人们淘汰，重新尝试使用机械除草。而为了减少水土流失，又要尽量地少耕翻土地，在这种情况下，传统的机械设备无法完成除草操作。另外，为了充分利用耕地，根据农作物的情况调整除草作业，为作物生长提供一个良好的环境。以上提出的这些，都是传统的农业机械无法实现的问题。为了解决这些问题，国内外的科研人员开始了新型农业机械的研究。

1.2 研究现状

鉴于本文将以机器视觉以及除草机器人为主要研究方向，故现就机器视觉技术以及除草机器人技术等国内外的研究状况作如下回顾。

1.2.1 机器视觉的研究现状

机器视觉是一门涉及人工智能、神经生物学、心理物理学、计算机科学、图像处理、模式识别等多个领域的交叉学科。机器视觉不仅是人类眼睛的延伸，更重要的是它具有一部分自行思考和决策功能。近年些来，计算机技术飞速发展以及大规模集成

电路的迅速发展，尤其是多媒体技术和数字图像处理及分析理论的成熟，机器视觉技术在全球范围内得到了广泛的应用，取得了巨大的经济效益与社会效益。基于机器视觉的杂草识别技术在较长的一段时间之前就已是国外研究的热点内容之一。在国际上许多著名学术会议中，例如 ASAE、SPEI、精准农业，都有专门关于机器视觉技术的研究和讨论。

与国外的研究发达的国家和机构相比，我国的机器视觉技术起步较晚。目前，机器视觉在杂草识别中的应用尚处于前期的探索阶段，具体的研究情况如下：

- 相阿荣等人^[1]研究了如何利用颜色特征从以土壤为背景的图像中识别杂草的方法以及如何根据植物的形状特征识别杂草的方法。
- 龙满生等人^[2]以玉米苗期的杂草为研究对象，分析了使用不同的颜色指标对背景进行分割的可行性以及应用形状因子区分玉米和杂草的方法。
- 刘敏等人^[3]把自然光照条件下田间草图像中引入分形维数，对植物的纹理进行分析，用分形维数来表征微观杂草叶子纹理以及宏观杂草纹理。根据纹理的粗糙程度的不同，取得了不错的区分识别效果。与传统的纹理分析方法相比，这个方法在时间复杂度和识别效果上具有明显的优势。
- 张健钦等人^[4]在机器视觉理论和方法的基础上，开发了一套拥有测量叶面积的图像采集、图像处理以及叶面积数据库这三个模块的软件系统，同时还实现了既可用于采摘的叶片也可用于活体原位叶面积测量的硬件系统。
- 纪寿文等人^[5]在光照明亮充足的条件下，利用视觉系统实地采集了玉米苗期的田间图像，从图像当中提取了投影面积、最大叶长和最大叶宽这3个形状特征来识别玉米苗中的杂草，试验结果表明：此方法能够有效地识别出玉米苗期(五叶期)在田间生长的细长单子叶杂草。根据田间杂草的投影面积可较有效地判定田间杂草的生长情况。但是，这个算法有局限性，它不适用于玉米的植株比杂草矮小的情况以及田旋花等叶片比较宽大的双子叶杂草的识别，识别准确与否还受到阳光光照强度和土壤种类及其含水量的干扰。
- 尚卓等人^[6]在自然光条件下对旱田作物包括棉花、小麦、玉米等以及果树或道路两旁林木，实时喷药的农药喷洒系统应用图像处理技术及自动控制技术完成初步探索。

1.2.2 除草机器人的研究现状

除草机器人是指可以取代人类进行除草活动的自动化智能农业设备。它可以通过减轻农作物种植人员的劳动强度和减少农业从业人员数量来提高农业生产效率和经济效益。国内，南京林业大学陈勇^[7]最早提出并设计了基于直接施药方法的除草机器人。国外除草机器人的研究则进行得比较早，在1999年美国加州大学戴维思分校生物与农业工程系D.C. Slaughter教授、D.K. Giles教授和W.S. Lee研制出除草机器人原型^[8]，如下图1.1所示。

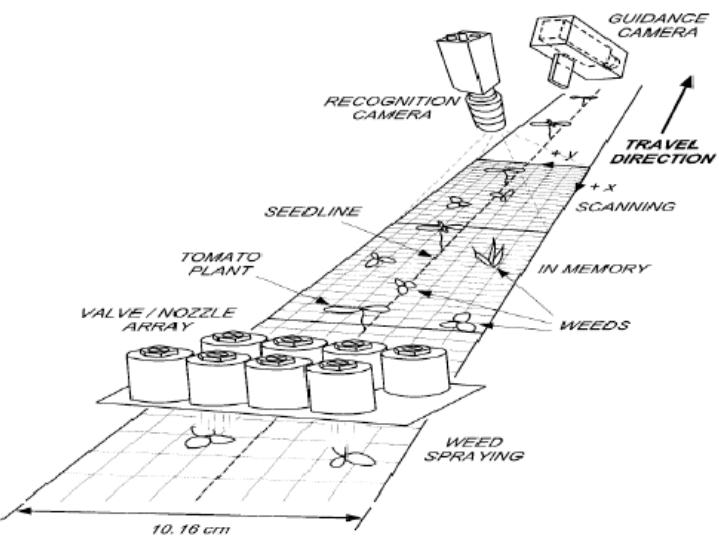


图 0.1 美国加州大学戴维思分校生物与农业工程系制作除草机器人原型^[8]

美国伊利诺依大学农业与生物工程系研究了一款智能除草喷雾机^[9]。这个机器人应用机器视觉技术采集和计算杂草分布特性，通过控制芯片进行施用量决策和控制喷头工作。它能够在3.7米×0.43米范围内将作物、杂草和土壤区别开来，并在0.24秒内做出喷雾决策。另外，该喷雾机器人能以14千米/小时的速度行进，实验得出目标分辨准确率为91%。

澳大利亚科研人员研发出一种在农田使用的智能化施药机械^[10]，它能在田间快速移动，在移动的过程中借助专门的电子传感器来区分周围的庄稼和杂草，一旦发现杂草便对其进行喷洒除草剂。这种只针对杂草的喷洒方式使除草剂的花费只有常规喷洒的1/10，甚至更低。和传统的大面积施药相比，该方法不仅节约了生产成本，还能够减少了对环境的污染。

丹麦科学家研制出一种可用于农田除草的机器人^[11]。这种机器人是四轮运动的轮式，由电能驱动，对地面扫描是使用一台照相机来进行的。它利用机器人上携带着的

识别软件来进行杂草识别。该软件使用 15 种参数，来对杂草的大小和对称性等外部特征进行描述，最后通过 GPS(全球定位系统)来完成杂草的定位。最后，在甜菜田里进行了识别实验。该实验表明，该机器人能够发现并且有目标有选择地使用除草剂杀死杂草，同时可使除草剂用量减少 70%。

日本京都大学 Naoshi Kondo 等人推出了草坪除杂草机器人^[12]，用照相机拍摄图像，利用图像分析自动识别杂草并确定位置和中心，控制操作器自动接近杂草并将其除掉，再放入存草器内。

近几年来，我国江苏大学邱白晶等^[13]进行了割草机器人避障行为以及控制研究，GPS 与数字罗盘组合导航系统研究等。

目前，国际农业林业机器人呈现蓬勃发展的态势，《国家 863 计划先进制造与自动化技术领域机器人技术主题发展战略的若干思考》中强调了在特种机器人专题中有关农林机器人方面的关键技术及原型样机研究工作，从中可以看出尤其是在我国加入 WTO 后开展农业机器人研究的迫切性和严峻性。加快发展我国农业机器人技术对于当前“坚持以人为本，坚持科学发展观”，降低劳动强度，改善劳动环境，提高我国农业机械化和自动化工艺装备的整体技术水平与促进农业的可持续发展，将起到良好的推动作用。

2 方案分析

2.1 功能分析

本文研究一种基于机器视觉的除草机器人，希望它能够在田间全自动地完成除草任务。其功能需求如下：

- 机器人能够准确地确定杂草的位置并将其坐标标定出来；
- 机器人可以通过特定的结构使机器人能够在比较复杂的田间环境下全方向运动并完成除草的动作；
- 机器人能够正确地运动到指定的位置，并精确的拔草而不破坏农作物的生长；

为了能够顺利的完成上述的三个功能，机器人整体可以划分成视觉、机械和电控三大系统，每个系统提供一定的功能，最后将它们组合起来就是一个可以实现所有功能的机器人。系统的划分以及它们具体应该提供的功能如下图 2.1 所示。

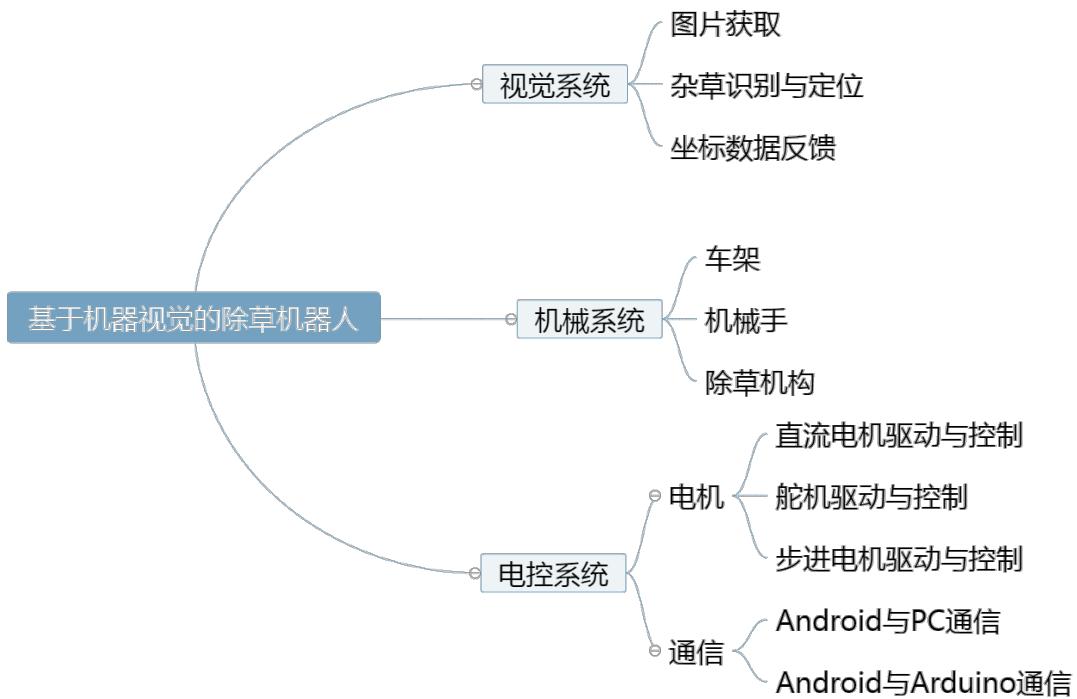


图 2.1 功能模块划分示意图

下面，就上述的三大系统，讨论具体的方案。

2.2 视觉系统可行方案

机器视觉(machine version, mv)，从学科分类上，被认为是人工智能下属科目，经常和另外一门科目一起被提及，计算机视觉(computer version, cv)。这两门学科的目的，都是对一副或者多幅图像进行获取，接着对获取的图像进行特定的处理，最后获得想要的信息。在实际上，计算机视觉更偏重于对算法的研究，对于一副图片，要怎么样处理才能得到想要的信息。而机器视觉则不太关心处理图像的过程，它看重的是结果，因为这个结果往往还关系到机器下一步的动作应该怎么执行。在这种意义上来说，机器视觉可以认为是计算机视觉在工程上的应用。目前，对于这两门科目，主要的开发手段有：OpenCV、Matlab 和 Halcon。

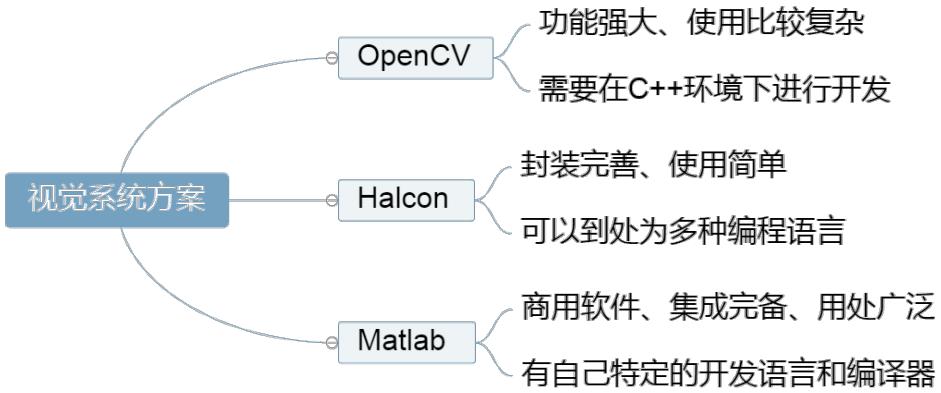


图 2.2 视觉系统方案示意图

OpenCV，是一个在 BSD (Berkeley Software Distribution) 许可（一种开源软件的许可协议）下发行的跨平台计算机视觉软件库，主要用 C++ 语言进行编写，并且以 C++ 语言的形式提供大量的图像处理和计算机视觉方面的通用算法接口。通过调用这些接口，可以比较轻松的完成对图像的处理。但是，需要在 C++ 语言的环境下进行编程。

Halcon，是一套由德国 MVtec 公司开发的专业视觉算法包，它将一个个的图像算法封装成算子，然后把接口提供给用户。在不需要深入了解算法实现，仅仅通过阅读它的说明文档的基础上，就可以进行图像处理，运用不同的算子，集成在特定环境下工作的图像算法。更棒的是，设计好的算法可以自动导出成 C、C++、vb、.net 等不同语言下的具体实现。通过这种方法，我们可以在自己熟悉的平台下进行视觉开发。

Matlab，是一个完整的集成开发环境，是一套商业化的数学软件，它提供了包括编译器，函数库和 Matlab 语言。主要用于算法开发，数据可视化，数据分析以及数值计算等场合，对图像进行处理，只是它的一个部分内容。由于它的体积庞大，用处广泛，相比上述的两款专业软件，它的运算速度会比较缓慢。

2.3 机械系统可行方案

在机械结构上，实现相同的功能可以使用各种不同的机构。本次毕设的机械系统当中，主要需要考虑下图 2.3 所示的几个方面。

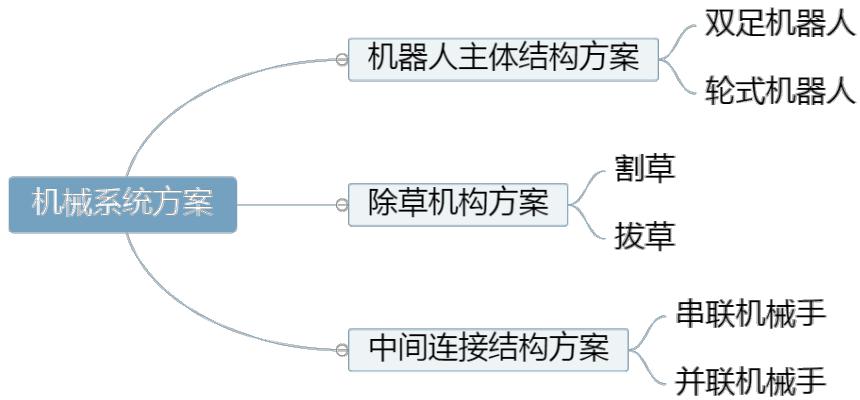


图 2.3 机械系统方案示意图

机器人的主体结构大致可以分为双足机器人和轮式机器人。双足机器人，主要是模仿人类的行走方式，通过伺服电机构成足上的各种关节。然后通过运动学和动力学的计算进行步态分析，进而控制机器人的行走。这种行走方法比较拟人化，控制难度较高，但是可以像通过控制步态以及步伐越过这种障碍。

轮式机器人，是指通过轮子的旋转进行移动的机器人。这种机器人只需要将电机的旋转传递到轮子上即可对机器人进行驱动，不需要尽心复杂的计算。这种方法控制比较简单，机器人的移动比较方便和迅速。但是，对于比较高的障碍无法跨越。

机械除草方式上，大致可以分成割草和拔草两种方式。割草方案，主要是利用高速旋转的刀具对超过某个高度的杂草进行割除。其优点是：操作简单、容易控制、在短期内能有效保护农作物，缺点是：对杂草的清理不完全，杂草很快又会长出来，需要经常性的除草；拔草方案，主要是利用钳子等工具，将杂草从泥土里连根拔起。其优点是：除草彻底，能有效的保护农作物，缺点是：控制比较复杂。

在机器人的主体以及除草机构中间，需要有一个中间的连接结构，向上与机器人主体相连，向下与除草机构相连，并保证除草机构能够准确的去到杂草所在的位置。这个中间结构可以有很多种方式来实现，这里主要讨论两种方式：串联机械手以及并联机械手。串联机械手，又叫关节式机械手。它是工业上的传统机械手，设计思路是用机械的方式模仿人类的手，主要由上下手臂、关节以及末端执行器构成。这种机械手的优点是可以实现多个自由度、动作比较灵活、特别适用于狭窄空间。

并联机械手是上世纪八十年代通过将机床结构和关节型串联机器人结合起来而提出的新式机械手。它的静平台和动平台之间通过两条或者更多的运动链连接，具备两个或者以上的自由度。与串联的机械手相比，并联机械手没有积累误差、精度更高、结构更加紧凑、运动速度、刚度和承载能力够更胜一筹。

2.4 电控系统可行方案

机器人，是自动执行工作的机器装置。科技发展到今天，机器人的种类有很多很多了，但是机器人始终不是人，为了让机器人按照我们的需求工作，我们要对它进行控制。在电控方面，电路、通信和电机都有很多种可以选择的方案，如下图所示。

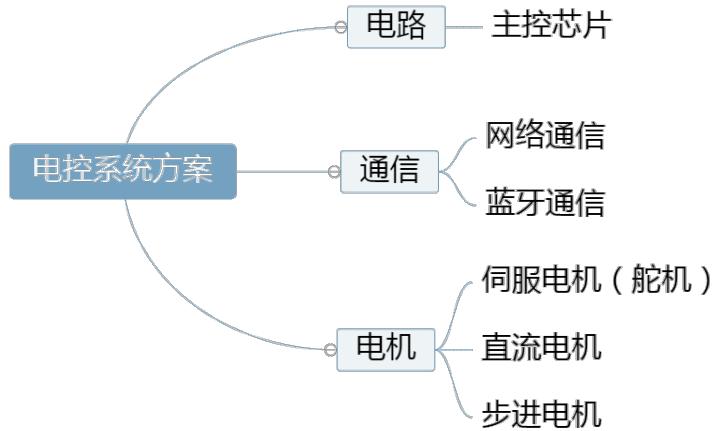


图 2.4 电控系统方案示意图

现代机器人多采用数字电路芯片（即单片机）进行控制。单片机按照寄存器的位数可以分为 8 位单片机、16 位单片机和 32 位单片机等，按照型号可以分为 51、AVR、ARM 等。

在通信方式上，机器人使用最广泛的就是网络通信和无线蓝牙通信。网络通信又包括局域网和广域网通信。前者是指在多个机器人之间，或者在机器人与上位机之间通过路由构建本地网络专供它们通信使用。后者是指通信双方都接入互联网，然后消息通过互联网从一方发送到另一方。这两种通信方式的优点是在带宽足够的前提下，传输速度比较快不受到某些硬件的限制，特别适合传输大容量的数据。

蓝牙通信，是指通信双方在一定的限制范围之内，按照蓝牙传输协议互相收发数据。这种通信方式的一大优点是数据通过无线传输，可以增大机器人的灵活性。但是，传输的速度受到硬件的限制，所以这种方式比较适合传输指令，不适合持续传输大容量的数据。

电机是提供机器人动力装置的，在不同的情况下需要使用不同的电机。伺服电机，又称舵机是可以转动到特定的角度并一直维持这个角度的一种电机。直流电机是一种顺着一个方向一直旋转的电机，在机器人领域一般作为机器人运动的动力输出。步进电机，是一种一次可以按照一定角度转动的电机，一般用于需要精确控制距离的场合。

2.5 方案确定

由 2.2-2.4 的叙述可以知道，OpenCV 只能在 C++语言的环境下进行，但是 C++对界面的编程比较麻烦、不方便使用；Matlab 由于集成的东西比较多所以运算速度相对较慢，故而在本次毕业设计当中 Halcon 是比较好的机器视觉方案，可以在多个语言环境下进行开发。

在机械系统的选择上，双足机器人涉及到步态分析，需要的开发周期比较长，并且很难执行除草任务。在这个项目当中对机器人的外形没有特殊的要求，故而选择轮式机器人是比较好的。另外，并联机械手精度高、运动速度快等特性较之串联机械手也更加合适。至于除草的方法，割草不够彻底，时间一长就会重新长出来。所以，拔草能够根治杂草问题比较合适。

在电控系统上，51 单片机已经比较落后了，但是控制轮式机器人的移动和除草都仅仅是需要控制电机。不涉及到比较复杂的控制方法，选用 ARM 的话会比较大材小用，所以芯片选择 AVR 会比较合适；在电机的类型上，不能单独只选择某一种，在不同的动力需求情况下，要选择不同的方式。

上一段确定了视觉系统方案和机器人的相关方案，但是这还不够，进行视觉处理的前提是必须获取到图片数据，更进一步来说，Halcon 需要在 pc 上运行，也就是说机器人需要把图片发送给 pc，pc 收到图片处理完后，需要把结果反馈给机器人。换句话说，机器人和 pc 之间是需要进行通信的。进一步分析，想要获取图像数据，就必须用摄像头，上文我们已经选定了芯片 8 位的 AVR 单片机，但是它并不足以驱动一个摄像头，而且传输数据比较慢，不适合用来进行图片数据的传输。所以，很显然还需要一些新的设备。

需求条件是：“能够拍照，并且可以接收和发送数据的设备”。通过这样的描述，很容易可以联想到我们生活当中必不可少的一样东西，智能手机。现在的手机，整体上分为两个阵营，分别是以 Google(谷歌)为首的 Android 系统手机和以 Apple(苹果)为首的 IOS 系统手机。从价格上，Android 更胜一筹；从普及率上，还是 Android 比较高；从开发的角度上，Android 是开源，IOS 是封闭的，还是 Android 更合适。所以，可以采用一个 Android 设备作为获取和发送图像，对下沟通机器人，对上沟通 pc 的消息中转站。

使用了 Android 之后，可以确定通信方式。Android 与上位机 PC 之间需要传输

图片，图片的数据量比较大，并且要求较高的传输速度，所以应该选择网络通信。在 Android 与下位机 AVR 之间只需要传达上位机的指令，所以可以选用蓝牙通信。

综上所述，选择 Halcon 作为视觉处理方案，选择 AVR 芯片控制的轮式拔草机器人作为控制方案，选择 Android 设备作为图像获取和数据交换的方案，同时选择网络通信以及蓝牙通信作为通信方案。

2.6 本章小结

本章节首先提出了机器人的功能需求，完成了对机器人的三大功能模块的划分。然后，根据各个功能模块讨论已有的技术方案，并分析每个方案的优缺点。最后，根据机器人的实际情况选定如下图 2.5 所示实现方案。



图 2.5 机器人最终方案示意图

3 机械系统设计

本次毕业设计的机械部分，旨在设计一款能够在田间环境下可以沿着直线方向运动的可以携带摄像设备以及并联除草机械手的轮式机器人，整体结构示意图如下图 3.1 所示。在这个框架下，机械设计分成三大部分，分别是机器人整体、并联机械手和除草夹。下面，将详细叙述这三个部分的设计理念和控制方法。

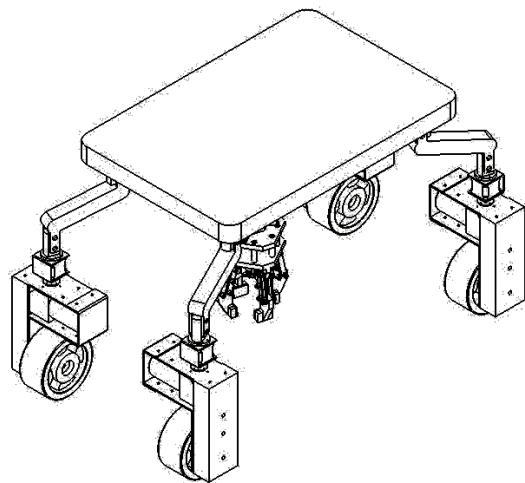


图 3.1 除草机器人整体示意图

3.1 机器人结构设计

机器人骨架的主要功能是支撑起整个机器人并实现机器人的移动。考虑到除草机器人的工作环境，田间的路可能是泥泞的或者崎岖不平的。由于方案选定的章节已经确定机器人是轮式机器人，所以为了使机器人能够在这种环境下运动，应该为机器人的轮子提供足够的动力，这里设计为四轮驱动，另外选择的电机应该能提供较大的扭矩，并且机器人的轮子也应该具有较大的直径。机器人在工作的时候是直线行驶。但是需要考虑到机器人需要驶进和驶出工作区域，所以机器人应该能够全向移动。虽然，通过轮子的正反转也可以实现机器人的转向，但是需要的转弯半径较大，设计上采用步进电机的方式来实现机器人的转向。机器人具体的结构设计如下图 3.2 所示。

其中，(1) 车上平台用于放置控制电路板及其他电子元件，(2) 伸出杆用于安装轮子，扩大机器人底盘的空间，使后面安装的机械手有更大的活动范围，(4) 减速箱用来减低电机的速度，提升扭矩，使机器人可以在更加恶劣的地面上移动。(3) 步进电机和(9) 旋转盘则用来控制机器人的转向。

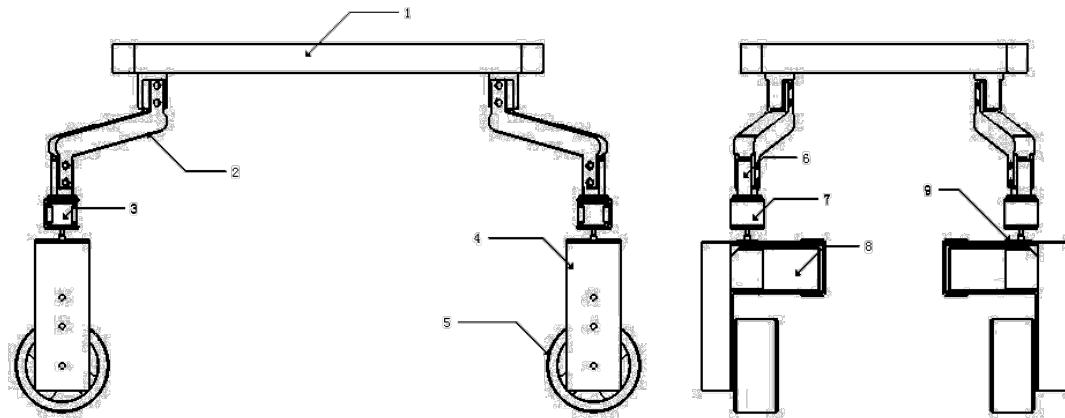


图 3.2 机器人骨架结构示意图

(1) 车上平台 (2) 伸出杆 (3) 步进电机 (4) 减速箱 (5) 车轮

(6) 连接件 (7) 步进电机固定架 (8) 直流电机 (9) 旋转盘。

3.2 并联机械手结构设计

通常来说传统意义上的工业机械手大多是串联机械手，也就是开环结构，即其末端执行器与基座之间有且仅有一条运动链。相较之下，并联机械手的末端执行器至少通过两条运动链与基座连接。同时，每条运动链都是唯一的移动或者转动副。在本次毕业设计当中，采用的是三自由度 delta 并联机械手，即末端执行器与基座有三条运动链，其结构如下图 3.3 所示。

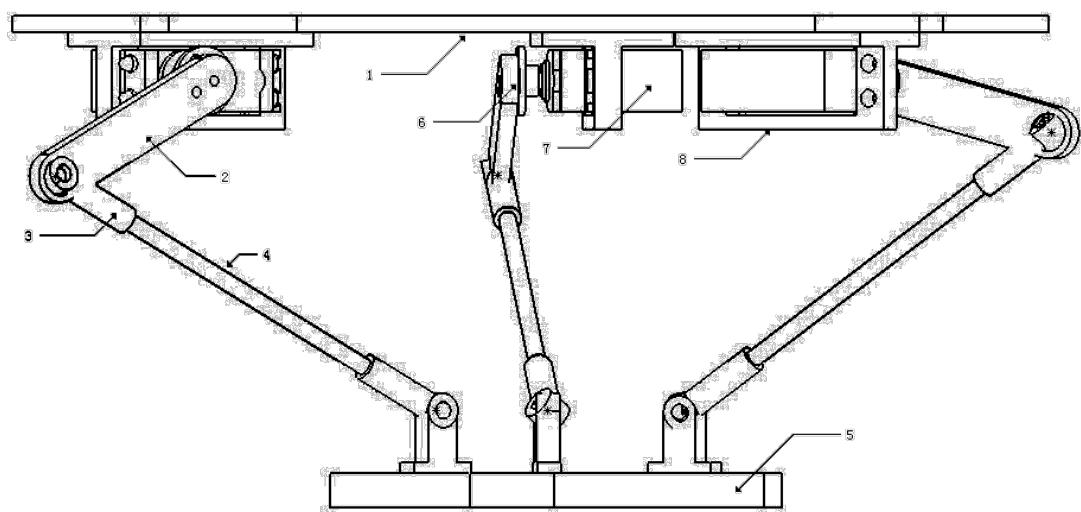


图 3.3 并联机械手结构示意图

(1) 基座 (2) 主动臂 (3) 杆端关节 (4) 从动臂

(5) 末端执行器 (6) 舵盘 (7) 舵机支架 (8) 舵机

由图 3.3 可以看到，基座通过三条轴对称并且均匀分布的运动链分别连接到末端

的执行器上。每条运动链的构成都是相同的，由一条主动臂和一条从动臂组成。主动臂与从动臂之间、从动臂与末端执行器之间各通过一个(3)杆端关节相连。另外还有两个辅助的零件，(7)舵机支架和(6)舵盘。

空间中任意一个不受任何约束的刚体都要六个自由度，分别是3个平动自由度和3个转动自由度。这些刚体通过运动副连接组成机构后，由于它们之间的互相约束，自由度会随之减少。空间机构分为开环机构和闭环机构，其中闭环机构又可以按照闭环数目细分为单闭环和多闭环机构。还有一种混合机构，也就是开环与闭环机构的混合体。由于开环机构自由度计算比较容易，这里主要讨论下闭环机构的自由度问题。目前，在计算空间闭环机构的自由度时，主要采用的是经典的Kutzbach-Grubler公式：

$$M = 6(n - g - 1) + \sum_{i=1}^g f_i \quad (3.1)$$

式子(3.1)当中， M 表示机构的自由度， n 表示机构的活动构件数， g 表示总的运动副个数， f_i 表示第*i*个运动副的自由度个数。

分析上图可知，该并联机构的总的构件个数 $n = 7$ ，总的运动副个数 $g = 9$ ，其中具有3个自由度的球面副有6个，具有1个自由度的转动副有3个，即 $\sum_{i=1}^9 f_i = 3 \times 6 + 3 \times 1 = 21$ ，因此该机构自由度 $M = 6 \times (7 - 9 - 1) + 21 = -18 + 21 = 3$ 。

并联机械手需要与机器人主体连接起来，然后在末端执行器上连接除草结构。下图3.4表示了机械手与机器人主体的装配关系。

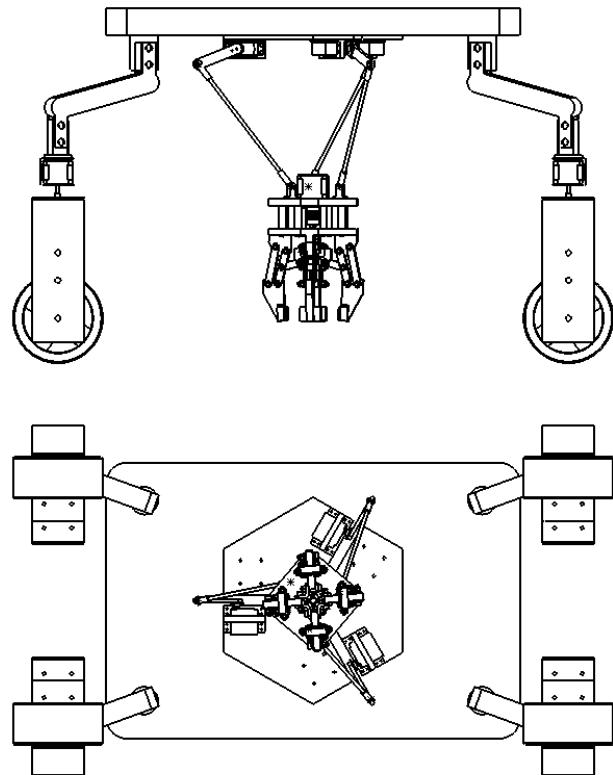


图 3.4 机械手与机器人主体以及除草夹的装配关系示意图

3.3 并联机械手运动学分析

在进行运动学分析之前，首先要对上面提出的结构进行抽象，并建立空间笛卡尔坐标系，完成后得到的结果如下图 3.4 所示。

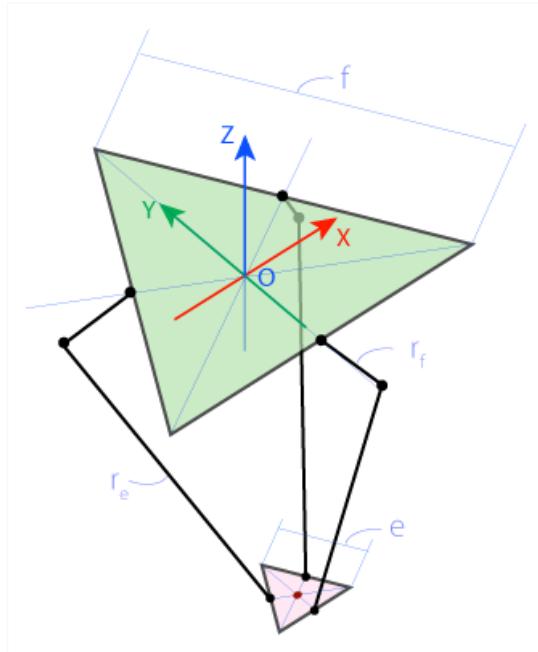


图 3.5 并联机械手结构抽象和建系坐标示意图^[14]

建系的方法从图 3.4 中可以看到，取定平台的中心为坐标原点，取垂直于定平台并且通过原点的直线为 Z 轴，取平行与定平台一边 L 取经过原点的直线为 X 轴，取垂直于 L 且经过原点的直线为 Y 轴。另外，图 3.2 中标示出来的几个参数，它们在后面的运动学逆解计算当中需要用到，分别代表的意思是：

- f : 表示定平台的边长（定平台是一个正三角形）；
- e : 表示动平台的边长（动平台也是一个正三角形）；
- r_f : 表示主动臂的长度；
- r_e : 表示从动臂的长度；

有了上述的坐标系以及四个参数，下面开始进行运动学的逆向求解。对于 delta 机械手来说，逆解就是通过末端执行器位姿，求解主动臂的转动角（即定平台某一条的中线与对应主动臂的夹角），也就是电机的输出角度。

由于定平台与动平台之间的三条运动链是轴对称且均匀分布的，所以对其中一条运动链进行分析即可。定平台与主动臂之间是通过一个转动副相连，所以主动臂必定只能在一个平面内运动。但是，主动臂与从动臂之间是通过球面副相连的，所以若固定从动臂与动平台的连接处 E_1 ，主动臂与从动臂连接点 J_1 的运动轨迹应是一个球体。该球体与主动臂的运动平面所在的平面的交集应是一个圆形的平面，具体的示意如下图 3.5 所示。

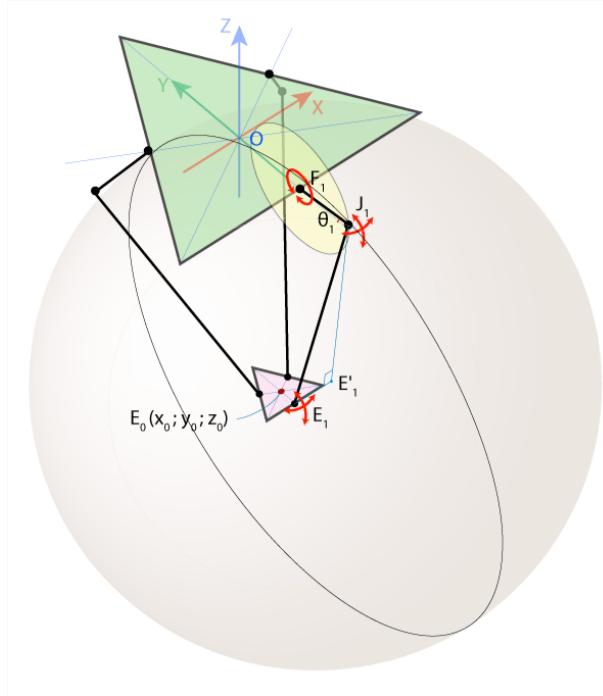


图 3.6 并联机械手结构抽象和建系坐标示意图^[14]

图 3.5 中, 可以确定一些重要的几何关系。首先, 设定末端执行器的中心坐标为已知, 即点 $E_0(x_0, y_0, z_0)$ 。然后, 待求变量是转动的角度 θ_1 , 即主动臂与定平台的夹角。根据约束的关系可以知道, 点 F_1, J_1 在同一个平面内, 且点 E'_1 是点 E_1 在平面 YOZ 上的投影, E'_1 是平面 YOZ 上圆形的圆心。从上面的描述可以注意到, 大多数的已知条件都在 YOZ 平面上, 或者与之有关, 所以可以把注意力放在该平面上, 抽取 YOZ 平面出来可得下图 3.6, 其中左下方是动平台的俯视图。

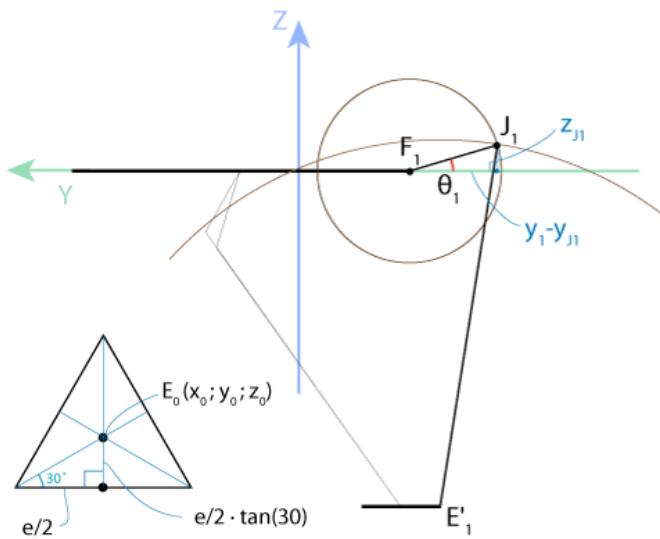


图 3.7 运动学逆解计算辅助图 2^[14]

综合图 3.5 和图 3.6 可以列出下面的式子:

$$EE_1 = \frac{e}{2} \tan 30^\circ = \frac{\sqrt{3}e}{6} \quad (3.2)$$

因为点 E 的坐标已知是 (x_0, y_0, z_0) , 所以 E_1 的坐标也可以用点 E 的坐标来表示, 即 $E_1(x_0, y_0 - \frac{\sqrt{3}e}{6}, z_0)$ 。又因为点 E'_1 是点 E_1 在平面 YOZ 上的投影, 所以 E'_1 点的坐标亦可求得, 经过计算 E'_1 的坐标是 $(0, y_0 - \frac{\sqrt{3}e}{6}, z_0)$ 。

有了 E'_1 的坐标, 且 F_1 很容易求出是 $(0, -\frac{\sqrt{3}f}{6}, 0)$ 。通过图 3.6 所展示的几何关系, 就可以求出 J_1 点的坐标, 进而求出转角 θ_1 了。

$$E_1E'_1 = x_0 \quad (3.3)$$

$$E'_1J_1 = \sqrt{(E_1J_1)^2 + (E_1E'_1)^2} = \sqrt{r_e^2 - x_0^2} \quad (3.4)$$

$$(y_{J1} - y_{F1})^2 + (z_{J1} - z_{F1})^2 = r_f^2 \quad (3.5)$$

$$(y_{J1} - y_{E'1})^2 + (z_{J1} - z_{E'1})^2 = r_e^2 - x_0^2 \quad (3.6)$$

$$\theta_1 = \arctan\left(\frac{z_{J1}}{y_{F1} - y_{J1}}\right) \quad (3.7)$$

式子 (3.5)、(3.6) 和 (3.7) 中 y_x 和 z_x 分别表示 x 点的 y 坐标和 z 坐标。通过上述的式子, 可以计算出 delta 机械手当中一条运动链的转角与末端执行器中点的关系, 但是还有两条运动链的转角需要计算。

所幸的是, delta 机械手的三条运动链是均匀分布的。所以, 只要将坐标系分别旋转 120° 以及 240° , 便可以使用同样的方法计算出其他两条运动链的转角与末端执行器中点的关系。下面以旋转 120° 为例, 叙述计算方法, 如下图所示, 旋转前的坐标系是 XYZ, 旋转后的坐标系是 X'Y'Z。保持点 $E_0(x_0, y_0, z_0)$ 的坐标不变, 设转换后的坐标变为 $E'_0(x'_0, y'_0, z'_0)$, 根据投影关系则有:

$$x' = x \cos(120) + y \sin(120) \quad (3.8)$$

$$y' = -x \sin(120) + y \cos(120) \quad (3.9)$$

将(3.8)、(3.9)建立的映射关系代入上述的公式, 就可以像前面一样求出第二条运动链的关系。最后将这两公式里面的 120 改为 270 , 即可求出第三条运动链的情况。

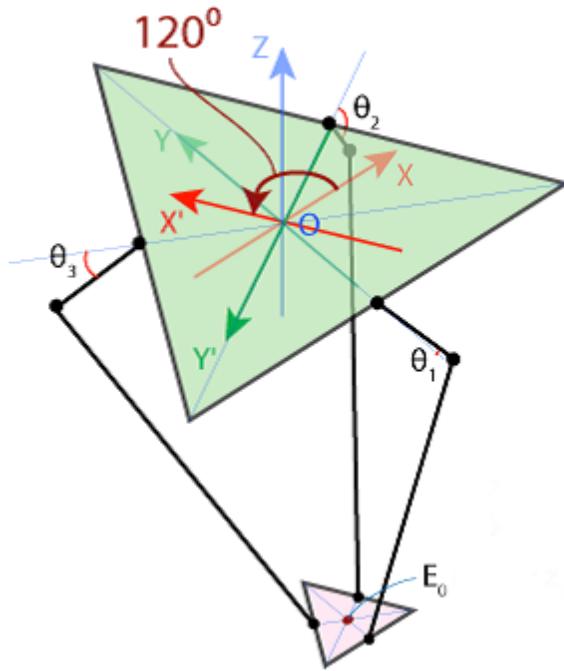


图 3.8 旋转 120°后坐标示意图^[14]

3.4 除草夹结构设计

除草夹是在将杂草的位置确定之后，执行除草动作的机构，它与并联机械手的末端执行器连接在一起。具体的结构设计如下图 3.8 所示，一共有 11 类零件构成。

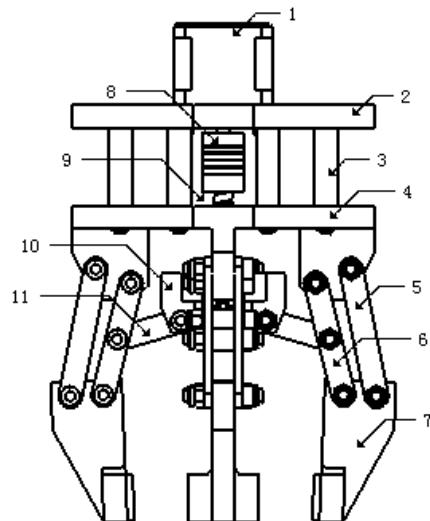


图 3.9 除草夹结构示意图

- (1) 42 步进电机 (2) 上固定板 (3) 支撑套 (4) 基座
- (5) 从动连杆 (6) 中间连杆 (7) 爪子 (8) 联轴器
- (9) 梯形丝杆 (10) 移动平台 (11) 主动连杆。

除草夹的工作原理是这样的：在（10）动平台上连接着丝杆的平台，当（1）步进电机转动时，会通过（8）联轴器将旋转运动传递到（9）丝杆上。由于丝杆时固定的，所以丝杆上的平台会上下运动，它运动将带动动平台做上下运动。由于（11）主动连杆的长度是固定的，当动平台上下运动将会撑远或者拉近（6）中间连杆，从而使（7）爪子放开或者夹紧。

机器人想要完成除草动作，必须综合并联机械手和除草夹，将这两部分装配起来后如下图 3.10 所示。具体的除草流程按照以下 3.9 的步骤进行：

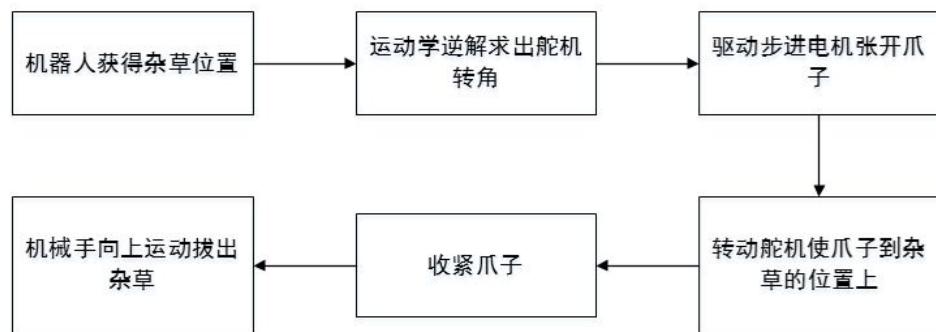


图 3.10 除草流程示意图

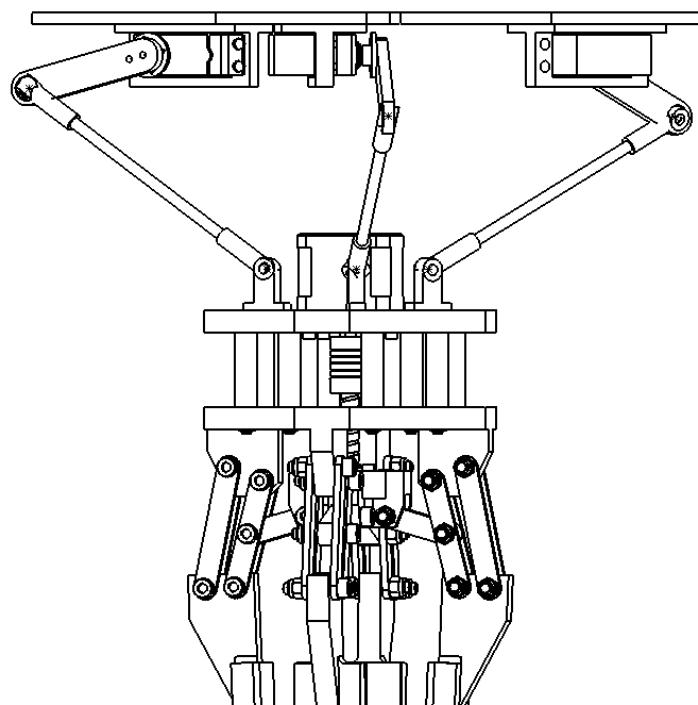


图 3.11 并联机械手和除草夹装配示意图

3.5 除草夹的运动学分析

除草夹的运动由步进电机决定，主要表现为除草夹的张开与收紧。步进电机输出的旋转运动首先传递到丝杆上，使丝杆旋转，由于丝杆被限制成只有一个自由度的，所以丝杆上的平台会在丝杆上做直线运动。平台的直线运动将导致连杆的运动，从而造成除草夹的张开与收紧。下面，建立数学模型，进行运动学分析，模型抽象如下图 3.11 所示。其中，JEABFK 表示夹子收紧状态，IGCDHL 表示夹子张开状态。已知，丝杆平台的直径 $AB = 2r$ ，连杆的长度是 $AE = l$ ，连杆与竖直方向的夹角是 θ 。假设，夹子两爪子的间距是 d ，根据图 3.11 的几何关系可以推导出：

$$JK = EF \quad (3.10)$$

$$AC = AE \cos \theta \quad (3.11)$$

$$CE = AE \sin \theta \quad (3.12)$$

$$EF = EC + CD + DF \quad (3.13)$$

联立(3.10)- (3.13)四式并代入已知条件，可以得到：

$$\theta = \sin^{-1}\left(\frac{d-2r}{2l}\right) \quad (3.14)$$

由被夹物体的形状合理的安排 d 的值，根据上式即可求出夹子在收紧状态下，连杆与水平方向的夹角 θ 。

有了初始条件下的 θ ，设步进电机运动的步数为 s （参见 4.2.3 可知每 400 步电机转过一圈），丝杆的导程是 p ，电机每转一步夹子张开或者夹紧的距离为 y ，应用梯子斜靠墙上逐渐下滑的模型，根据勾股定理可以得到：

$$(y + l \sin \theta)^2 + (l \cos \theta - \frac{sp}{400})^2 = l^2 \quad (3.15)$$

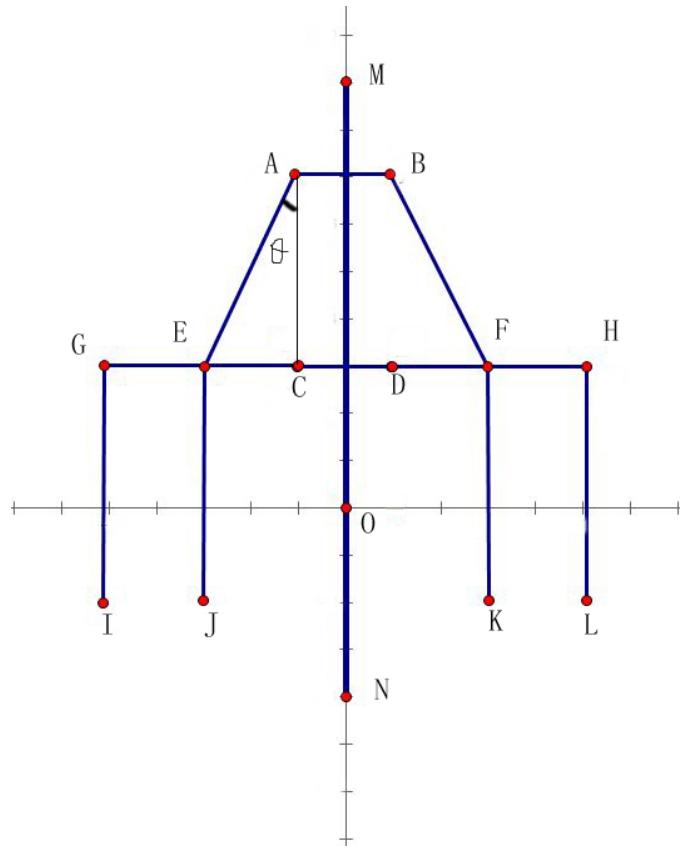


图 3.12 除草夹运动学分析示意图

4 电控系统设计

4.1 电路设计

机械系统的设计完成，标志着机器人具有了完成功能的可能性。但是单纯的机械系统不能自主的产生运动，机器人需要电能来驱动。电能在机器人上的流转需要通过电路来实现，为了保证机器人的运动，电路的设计至关重要。

4.1.1 电子元件简介

在第 2 章的方案选择中已经决定，机器人的控制器由 AVR 单片机来担任。更加具体来说，是 Arduino。这是一款方便、灵活的开源电子平台，它是在 AVR 芯片上添加了底层库的产品，它向下对 AVR 的底层寄存器做了封装，向上对用户提供接口方便用户操作 AVR 的资源。本次毕业设计在对机器人的控制上选用的是 Arduino 系列 Mega 2560 开发板，顾名思义，这是一款增强版，它的核心是 ATmega2560 芯片，提供 56 路数字输入输出管脚（其中的 16 路可以作为 PWM 输出），特别适合需要大量 IO 接口

的设计。



图 4.1 Arduino Mega 2560 开发板

说到底，Arduino 也就是一款 AVR 单片机开发板，所以与它的通信比较方便的方法还是选用串口，Mega 2560 提供四个 USART 硬件串口，其中一个用来和 pc 通信负责烧录程序和调试程序，再用其中一个用来和 Android 进行通信。但是，现在市面上普通的 Android 设备都不会提供串口给用户使用。不过，Android 上基本都会提供蓝牙。在 AVR 单片机上，有很成熟的蓝牙转串口模块，这里我选用的是 hc-06 如下图 4.2 所示。对于这个模块，需要关注的是它的四个引脚：

- VCC 引脚：接电源正极，为模块供电，一般是 3.3v；
- GND 引脚，接电源负极，形成电流回路；
- Tx 引脚，接单片机串口的 Rx，将蓝牙模块接收到的数据送给单片机；
- Rx 引脚，接单片机串口的 Tx，将单片机要发送的数据送给蓝牙模块；



图 4.2 hc-06 蓝牙转串口模块

还有一个很主要的模块，就是各类电机的驱动模块。单片机上的电源一般都是 5v 或者 3.3v，这样的电压及其电流是不足以驱动各类电机的。在本次毕业设计当中，为了实现机器人的移动和除草，从上一章的图纸可以看到，机械人的移动需要四个直流电机来驱动，并联机械手的移动由三个舵机来驱动，除草夹的夹紧与放松有一个步进电机来驱动。所以，需要一些电机驱动模块。我选用的是 L298N，这是一款双 H 桥的电机驱动芯片，其中每一个 H 桥可以提供 2A 的电流，通常情况下，可以功率部分电压是 2.5 到 24v，逻辑电路部分电压是 5v。一个 L298n 模块可以驱动 2 个直流电

机或者一个两相四线的步进电机。综合上述的需求描述，这里一共需要 3 个 L298N 模块。对于这个模块，需要关注以下几个引脚：

- VCC：电机驱动电源正极；
- GND：电机驱动电源负极；
- 5v：外接逻辑电源（如有需要可以接，也可以不接）；
- OUTA/B/C/D：输出引脚，两两一组用于驱动一个直流电机；
- INA/B/C/D：输入引脚，连接单片机的 IO 口用于控制电机方向；
- ENA/B：使能引脚，决定电机的转速（驱动步进电机必须要接 5v）；

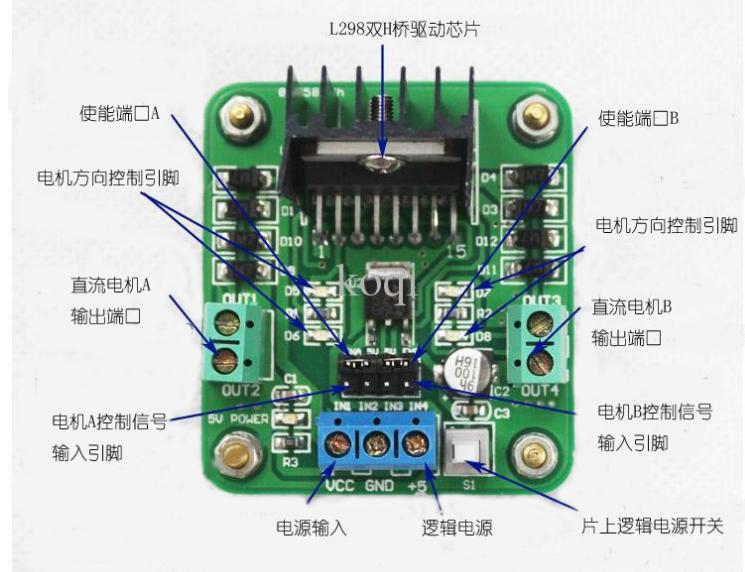


图 4.3 L298N 电机驱动模块

在直流电机方面，因为机器人需要确保直线行走，所以需要带有反馈装置的电机。这里选用带霍尔编码器的直流电机。舵机和步进电机没有特殊的要求，舵机选用普通的 995 舵机，步进电机选择 42 步进电机。



图 4.4 使用到的各类电机

4.1.2 主控芯片的引脚分配

由上述所使用的电子元件可以看到，整体电路应该是以 Arduino Mega 2560 作为核心控制器，外围接有三个 L298N 电机驱动模块及其驱动的四个直流电机和一个步进电机、一个 hc-06 蓝牙转串口模块和四个霍尔编码器。通过 Mega 2560 的 I/O 口，实现对整个机器人整体的控制。由于 I/O 口资源是有限的，不同的外设需要不同功能的 I/O 口进行控制。为了合理利用 I/O 引脚，本次毕设中引脚的分配如下图 4.5 所示。

arduino 2560 引脚安排			
功能	引脚	功能	引脚
蓝牙Rx	18	蓝牙Tx	19
左前电机方向	49	右前电机方向	42
左前电机方向	48	右前电机方向	38
左前电机速度	8	右前电机速度	10
左后电机方向	47	右后电机方向	41
左后电机方向	43	右后电机方向	40
左后电机速度	9	右后电机速度	11
左前电机编码器	2	右前电机编码器	20
左后电机编码器	3	右后电机编码器	21
舵机1	5	舵机2	6
舵机3	7		
步进电机A	31	步进电机B	33
步进电机C	35	步进电机D	37
外部中断0	2	外部中断2	3
外部中断1	20	外部中断3	21
串口1Rx	19	串口1Tx	18
备注：			
1. 速度引脚是pwm功能引脚			
2. 方向引脚是没有复用功能的引脚			

图 4.5 Arduino Mega 2560 引脚分配图

4.1.3 电路原理图

由前两小节可知，机器人以 Arduino Mega2560 为主控芯片，用 3 个 L298N 分别驱动四个直流电机以及一个步进电机，用 4 个编码器获取直流电机的实时转速，用 3 个舵机控制并联机械手的三条运动链，用 1 个蓝牙转串口模块负责和上位机的通信。故，综合以上的情况设计电路原理图如 4.6 所示：

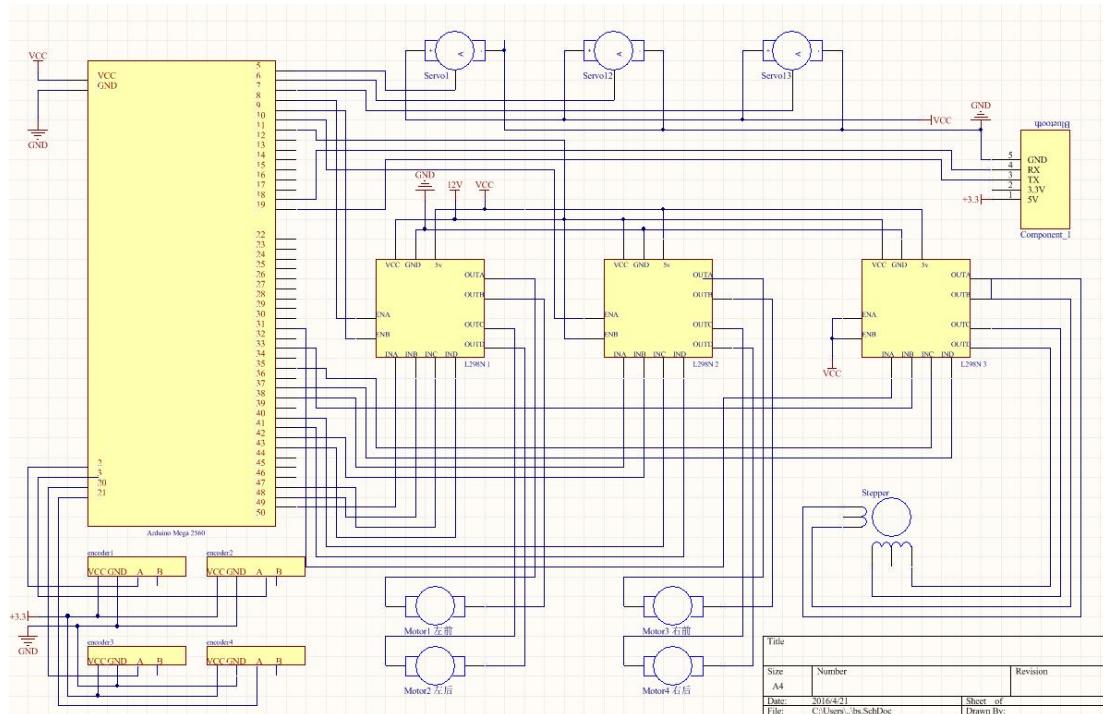


图 4.6 电路原理图

4.2 电机驱动

4.2.1 直流电机驱动

直流电机的构造比较简单，它的工作原理是通电导体在磁场中会受到力的作用。一般的直流电机会有两个引脚，只要往这两个引脚上提供足够的电压和电流，电机就能够转动。但是，这样直接在电机上接电源的方法只能让电机向一个方向转动，并且转速也不能控制。上文也有提到，在本次毕业设计当中，直流电机的驱动是依靠 L298N 模块来实现的。该模块主要依靠 H 桥来进行工作，H 桥的示意图如下图 4.7 所示，下面叙述它的工作原理。

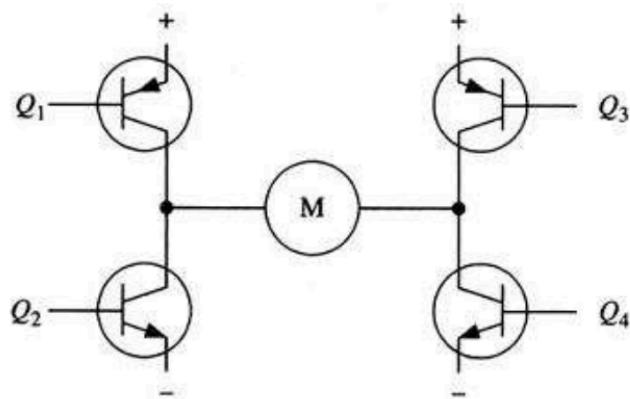


图 4.7 H 桥电路示意图

图 4.7 中，4 个三极管组成了 H 的两边，电机 M 在 H 的横杠部分，H 桥因此而得名。由图可知，若是需要电机转动，则必须导通对角线上的一对三极管。根据不同三极管的导通情况，电流可能从左往右或者从右往左通过电机，从而控制电机的转向。但是，必须保证同一侧的三极管不会同时导通，否则就相当于电源正负极相连，可能会烧坏三极管。同时，为了使能控制和方向逻辑控制，很有必要引入一些其他硬件，并将上图的电路优化成下图 4.8 所示。具体来说，就是增加了四个与门和 2 个非门。

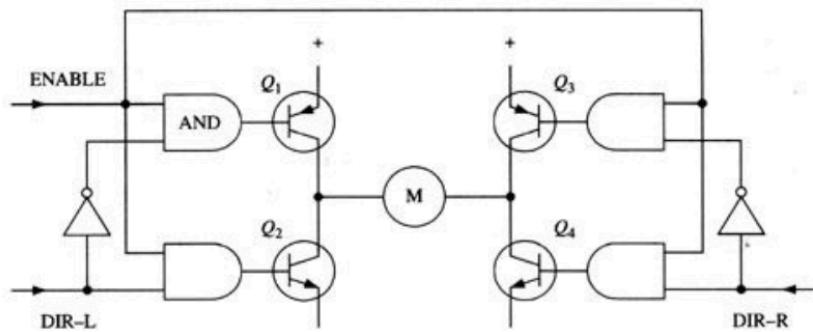


图 4.8 改良后 H 桥电路示意图

图 4.8 中可以看到，4 个与门都与同一个使能信号 ENABLE 相连，故这个信号可以控制整个电路的开关。2 个非门与同一侧的两个三极管相连，保证了它们不会同时导通。采用了这种方案，只需要三个引脚就能完成一个直流电机的控制了。（注意：以上的连接方法并不是一个完整的电路，只是一个逻辑的示意图，与门直接与三极管相连是不能工作的。）

在直流电机这一块，还涉及到编码器的问题。使用编码器的原因是为了形成一个反馈信号，令主控芯片可以对电机的转速进行 PID 的控制，从而确保机器人沿着直线运动。本次毕业设计选用的是霍尔编码器，即磁性编码器。

磁性编码器原理是通过磁力形成脉冲序列，进而产生信号，它的特征是将未硫化的橡胶中混合稀土类磁性粉末形成磁性橡胶坯子，硫化粘附在加强环上，形成磁性橡胶环，在该磁性橡胶环上以圆周状交替着磁，产生 S 极和 N 极。同时采用新型的 SMR 或霍尔效应传感器作为敏感元件，其信号稳定、可靠。用比较通俗的话来概括上面的原理，就是电机每转过一定的角度，编码器就会产生一个脉冲信号，产生一个信号的角度越小，编码器的分辨率就越高，通过这样的编码器计算出来的转速就会越准确。

对于单片机来说，使用编码器主要是关注的就是如何捕捉它的脉冲，然后通过计数来确定电机的转速。Arduino 的核心是 AVR，作为一种单片机，外部中断这个资源是必须具备的。利用外部中断，就可以很容易的捕捉到编码器输出的脉冲信号了。具体的步骤如下：

- 定义一个 long 类型的全局变量 count 用来表示捕捉到的脉冲次数；
- 选定一个采样周期，本次毕设的采样周期是 100ms；
- 把编码器的输出引脚接到 Arduino 的外部中断引脚上，并开启中断，配置中断出发模式为下降沿触发；
- 每一个采样周期结束后，读取 count 的数值，便可计算出转速等参数，最后重新将其置零；

4.2.2 舵机驱动

舵机，又称为伺服电机。主要是指由直流电机、减速齿轮组、位置传感器和控制电路组成的一个可以输出特定转角的电机。它通过接收一定时间长度的脉冲信号来指定输出轴旋转角度。一般而言，舵机都有最大旋转角度（本次毕设当中选用的舵机最大角度是 180° ），它与普通直流电机的区别主要是直流电机是一圈圈不停转动的，而舵机只是在一定角度内转动，不能一圈圈地转（数字舵机除外，数字舵机可以在舵机模式和电机模式中进行来回切换）。不带编码器的普通直流电机无法反馈转动的角度信息，但是舵机可以。它们两者用途也不同，普通直流电机一般是整圈转动，作为动力输出。舵机是控制某物体转动一定角度，进一步实现某种需求。

舵机的机械结构比较简单，如下图 4.9 所示，总共只有 5 种类型的零件，分别是：直流电机、电位器、电机控制板、齿轮组和外壳体。它们的作用是：

- 直流电机：动力来源；
- 电位器：作为反馈信号，根据旋转后产生的电阻变化来确定舵机是否转动到

指定的位置；

- 电机控制板：提供硬件电路；
- 齿轮组：将电机的速度减慢，提高扭矩，使小功率的电机产生较大的扭矩；
- 外壳体：提供固定的孔位以及保护内部的零件；



图 4.9 构成舵机的零件示意图

有了对舵机结构的认识，下面将叙述如何对舵机进行控制。舵机的伺服系统由可变宽度的脉冲来进行控制，也就是常说的 PWM。舵机有三根引线，分别是控制线，VCC 和 GND。本次毕设选用的是型号为 995 的舵机，VCC 只需提供 5V 的电压即可工作，注意要将控制板的 GND 和舵机的 GND 连接起来即可。信号线，顾名思义，就是控制脉冲的输入线了，应该连接在控制板的 PWM 引脚上。

一般而言，脉冲的参数有最小值、最大值和频率。舵机控制的基准信号通常是周期为 20ms，宽度是 1.5ms。此基准信号定义为舵机的中间位置，在本次毕设当中选用的是 180° 的舵机，故中间位置是 90°。角度是由控制线的持续脉冲所产生的，脉冲的长度决定舵机转角的大小，如下图 4.10 所示。当控制芯片发出指令，让舵机转动到指定的角度并保持这个位置时(即维持控制线引脚的 PWM 不变)，外力的作用，并不能使舵机的角度发生变化。但是，这个外力的大小是有上限的，这个上限就叫做舵机的最大扭力。

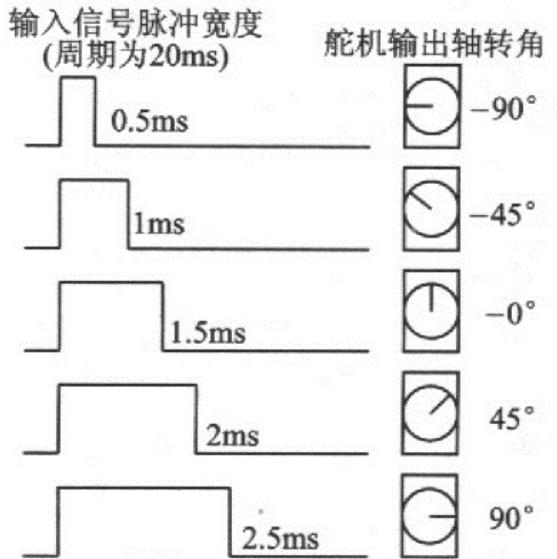


图 4.10 舵机控制脉冲示意图

在 Arduino 上控制舵机就非常简单了，因为已经有专业的工程师将舵机的控制集成一个库文件。在这里只需要简单的几句代码，就可以实现对舵机的精确控制。

4.2.3 步进电机驱动

步进电机有很多种，本次毕业设计当中，选用的是两相四线的 42 步进电机。两相步进电机在定子上有两个绕组，共四根出线，整步为 1.8° ，半步为 0.9° 。在驱动器中，只要对两相绕组电流通断和电流方向进行控制就可以了。同样的，在这里步进电机的驱动器仍然是 L298N，它的介绍不再赘述。驱动该步进电机，可以采用四拍或者八拍的方式，它们的不同是：

- 四拍的驱动顺序为 A-B-C-D；
- 八拍得驱动顺序为 A-AB-B-BC-C-CD-D-DA；
- 驱动的最终结果是步进角不一样，八拍可以精确到半步，四拍则只能精确到一步；

本次毕设，驱动步进电机采用的是八拍的时序。由上文已经可以看到八拍的驱动顺序，所以驱动方式只需要按照上面叙述的顺序给出电流的方向以及使能和失能即可。为了按照八拍的驱动方式驱动电机，L298N 上的电机方向控制引脚应该按照一定的时序输出不同的电平状态，在一个八拍内其电平状态依照顺序如下图 4.11 所示（图中 1 为高电平 0 为低电平）：

两相四线步进电机八拍驱动				
电平状态	INA	INB	INC	IND
A	1	0	0	0
AB	1	1	0	0
B	0	1	0	0
BC	0	1	1	0
C	0	0	1	0
CD	0	0	1	1
D	0	0	0	1
DA	1	0	0	1

图 4.11 L298N 八拍驱动两相步进电机电平逻辑示意图

由于 Arduino 的步进电机库只适合驱动四相的步进电机，所以这里需要自己用 c 实现一个步进电机的驱动程序。为了降低程序的耦合性和提高程序的可维护性和拓展性，单独把这个驱动写成一个.c 文件，然后在主程序的文件当中通过#include 加载，具体请参见附录。

4.3 通信

通信是指发信者通过某种媒体以某种格式来传递信息到收信者已达到某个目的。在机器人领域往往各个模块之间需要协同工作，很难通过某个单独的模块获取所有信息完成特定的功能。在这种情况下，通信就显得非常重要了。

4.3.1 Android 与 PC 通信

在本次毕业设计当中，Android 扮演的是一个消息中转站与图像获取的角色。消息中转，顾名思议是要进行消息的传递；而图像获取之后，需要将获取到得图像要发送到 PC 端进行处理，其逻辑如下图 4.12 所示。从这里可以看出，Android 端必须要与 PC 端实现数据的交换，也就是我们常说的通信了。

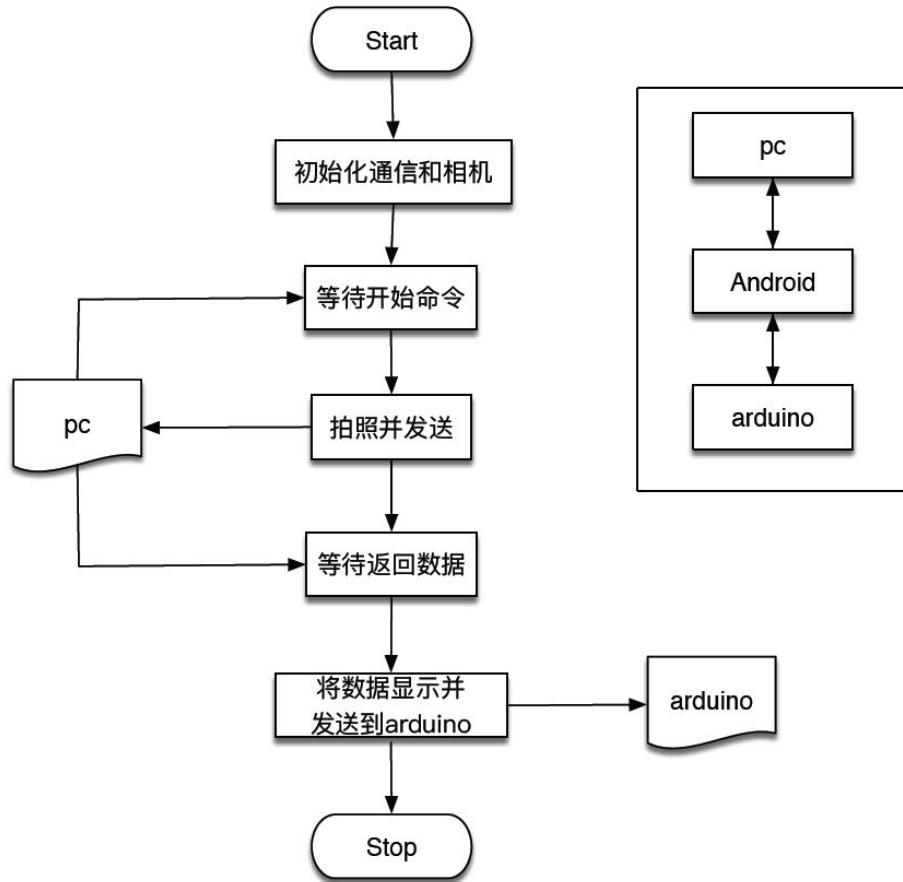


图 4.12 Android 与 PC 通信逻辑示意图

4.3.1.1 通信原理

本次毕业设计中设定，机器人通电之后需要上位机的控制器发送一个开始命令，机器人才开始运行。同时，pc 端需要接收 Android 获取的图片进行处理。

现代的 Android 设备，提供给我们很多种方式与其他设备进行通信。这里，我们选择最为常见的方式，网络通信。也就是通过互联网或者局域网，实现 Android 与 PC 端的通信（本次毕业设计当中 PC 端运行的操作系统是 Window 7）。由于涉及到网络通信，那肯定绕不开经典的网络通信协议，TCP/IP 协议。

TCP/IP 协议，是网络当中最基本的协议，是国际互联网络的基础，由网络层的 IP 协议和传输层的 TCP 协议组成，层级示意图如下 4.13 所示。其中，IP 协议负责给因特网当中的每一台设备分配一个唯一的识别地址，使得该设备能够在网络中能够被准确的定位；TCP 协议则负责传输数据，当数据包在传输的过程中出现问题时，它可以发出信号，要求重新发送数据包，以保证数据在一台设备传输到另一台设备后

能够保持完整性。本次毕业设计当中使用的是在 TCP 协议下的 Socket 来实现 Android 与 PC 的通信。下面将详细叙述，它的实现原理。

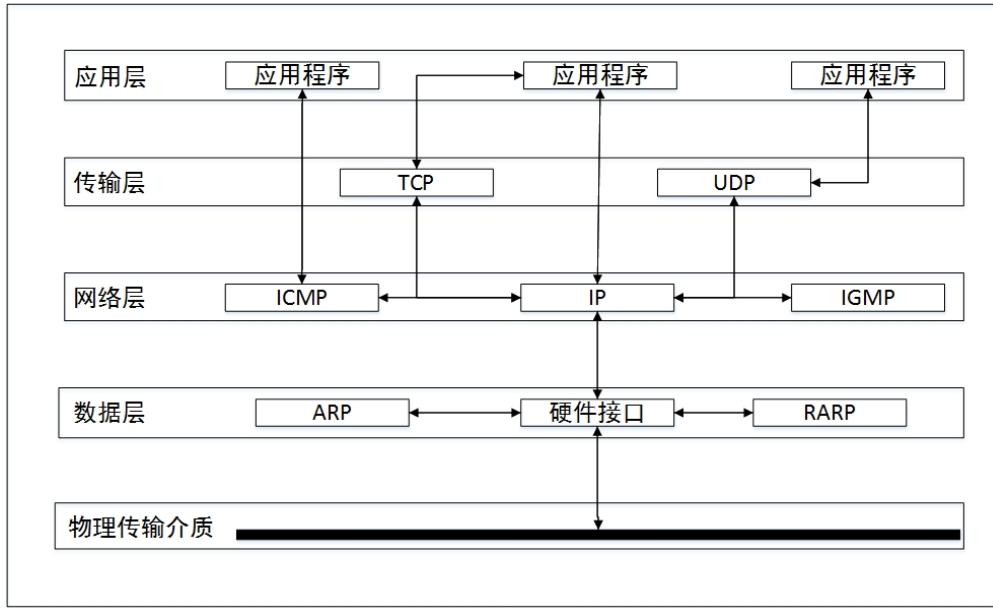


图 4.13 IP/TCP 协议层级示意图

Socket，中文译作套接字，是美国伯克利大学为了方便开发网络程序而开发的一种最初在 Unix 上运行的应用程序访问通信协议的操作系统调用。它的出现使程序开发者可以很方便的访问 TCP/IP，而不用过多的关注底层。在网络通信当中，被动等待其他设备来连接的一端叫服务端，主动发起连接的一端叫客户端，Socket 可以理解成客户端或者服务端用于描述 IP 地址和通信端口的一个特殊对象，在 Socket 当中，通过流来进行数据的交互。

Socket 的通信机制应该是这样的：

- 服务端设置自己的 IP 地址绑定一个端口，堵塞线程等待客户端来连接；
- 客户端发起连接请求，服务端接收到该请求后决定是否建立连接，如果需要建立连接，则开启一个子线程维持该连接，主线程继续等待其他客户端的请求。
- 服务端、客户端各自获取自己的输入流和输出流，通过数据流进行通信；

原本只是在 Unix 系统下运行的 Socket 由于方便和好用，很快就普及在了各种计算机系统上。由于是建立在 Unix “一切皆是文件”的哲学之下，所以 Socket 通信的实质很容易理解，就是服务端和客户端共同维护一个文件。数据的发送方将数据写在这个文件上，接收方则读取这个文件上的数据。

4.3.1.2 通信协议

虽然有了数据，但是如果需要发送方和接收方都知道这些数据代表的是什么意思的话，还需要令双方都遵循一套共同的标准。在本次毕业设计当中，由于通信的过程并不复杂，发送命令只需通信双方商定即可，发送图片由于数据量相对较大，所以这里自行定义了一套简单的图片传输协议，具体的流程如下图 4.14 所示：

1. 每次发送的数据包长度为 1029 个字节（数据包是一个有符号的字节数组，下标从 0 开始到 1028 结束）
2. 第一个字节是标志位，该位为 1 时，表示后面还有数据；该位为 2 时，表示这是最后一包数据
3. 数据包的[1 : 4]表示该数据包的有效数据长度，因为数据包最大可以存放 1024 字节的数据，所以[1 : 4]每一位分别表示“个十百千”位上的数字。除去最后一包数据，其他包都应该是放满的，最后一包的数据则很大可能都不满 1024（配合流的 read 方法可以避免接收冗余数据）。

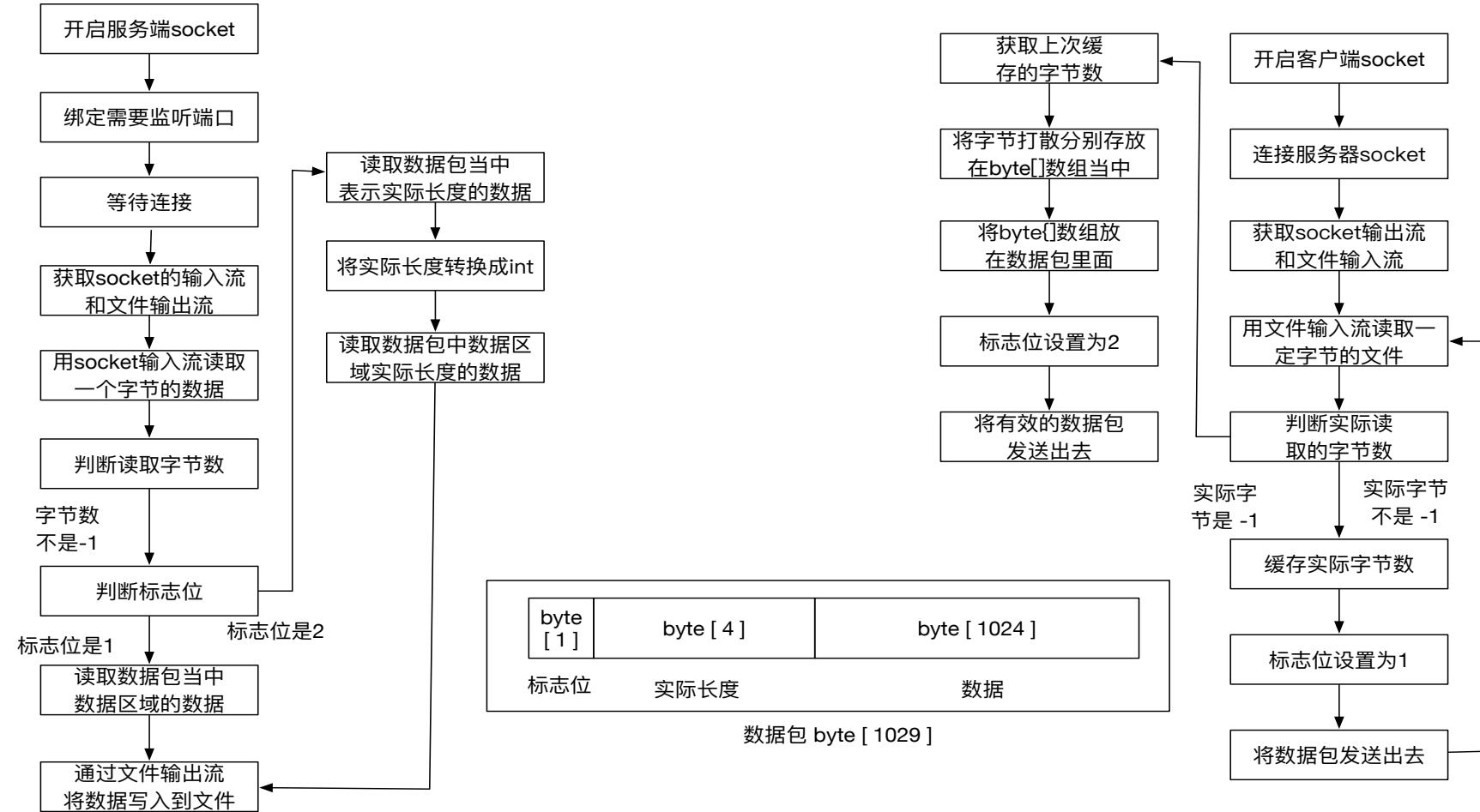


图 4.14 自定义通信协议以及数据包结构示意图

有了上述的通信协议，可以保证从 Android 发往 PC 的图片得以完整的发送。但是，图片经过 PC 的处理之后需要将得到的杂草位置数据从 PC 发往 Android。下面阐述一下，在本次毕业设计当中位置数据从 PC 发往 Android 端的策略。由上文可知，在 PC 端位置数据分别存储在两个浮点数组当中。当需要将数据发送到 Android 上的时候，PC 的流程是这样设计的：

1. 从 distancePix 数组当中获取数据的长度并保存到临时变量当中；
2. 开始同时遍历两个 distancePix 数组（分别是 x 的坐标和 y 的坐标）；
3. 构造发送字符串，将同一点的 x 和 y 坐标从“，”分隔，不同的点之间用“；”分隔；
4. 将发送字符串转换成字节数组，通过 socket 的 OutputStream 发送出去；

4.3.1.3 通信实现

为了实现上述的通信，需要在 PC 端和 Android 端编写各自的软件。在本次毕业设计当中，PC 端用 vb.net 编写了一套上位机软件，通过调用 Halcon 的算子对图像进行处理，通过 TCP 协议的 Socket 机制与 Android 进行数据通信，其界面如下图 4.15 所示。其中的功能键作用如下：

- IP 地址选择框：按下右边的下拉按键，软件会检测计算机目前所处的网络并列出可以接到本机的所有 IP 地址；
- 端口号输入框：定义本软件在 IP/TCP 协议当中所使用的端口号，默认是 9998，可以按照需要修改（注意不要和系统服务的端口冲突）；
- 打开连接按键：按下该按钮将开启服务器并监听网络请求；
- 关闭连接按钮：按下该按钮将关闭正在运行的服务器；
- 开始除草按钮：下位机通电并与上位机建立连接后，按下该按钮机器人将开始除草作业；
- 方向功能键：下位机通电并与上位机建立连接后，按下该系列功能键机器人将按照希望的方向进行运动一定的距离；
- 消息显示框：显示当前的机器人实时状态以及图像处理后的结果；



图 4.15 上位机程序界面示意图

Android 端用 Java 编写一个应用，通过调用硬件摄像头获取图像，通过 TCP 协议的 Socket 机制与 PC 进行通信，其界面如下图 4.16 所示。使用的方法如下：

- 将上位机给出的 IP 地址以及端口填入输入框里；
- 点击开启连接按钮；
- 建立了连接之后，上位机发送过来的数据会在消息显示区域显示出来；
- 后面给出的几个功能按钮是在调试阶段方便进行单独的各个模块的测试；



图 4.16 Android 端程序界面示意图

4.3.2 Android 与 Arduino 通信

在 Android 与 Arduino 的通信过程当中，Arduino 上的蓝牙模块已经做好了所有的底层工作，我们需要做的仅仅是像操作串口那样，把数据取出来或者放进去，其逻辑比较简单如下图 4.17 所示，这里不在详细叙述。重点把目光放在 Android 端上，Android 给我们提供的只是蓝牙硬件和 SDK 当中一些操作蓝牙的接口，我们需要用这些接口组成一套适合本次毕业设计的蓝牙通信方式。

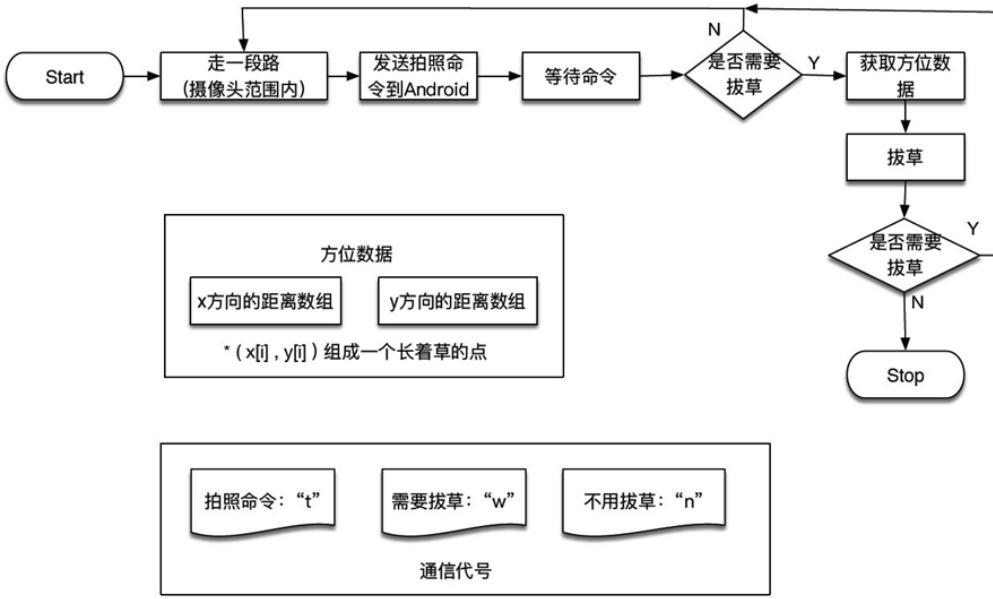


图 4.17 Arduino 上的通信逻辑示意图

首先，介绍一下 Google 提供的 Android SDK 当中的蓝牙操作接口。在本次毕业设计当中使用的有下面三个类：

- **BluetoothAdapter:** 本地的蓝牙配适器，所有蓝牙交互操作的“起始点”。通过它，我们可以发现其他蓝牙设备、查询并列出绑定过的设备、从物理地址实例化一个蓝牙设备并且创建 socket 来监听来自其他设备的信息。
- **BluetoothDevice:** 远程蓝牙设备，通过它可以请求一个远程设备的蓝牙 socket 或者获取该设备的基本信息。比如名字，物理地址，绑定信息等。
- **BluetoothSocket:** 蓝牙的 socket 接口（类似一个 TCP Socket），这个连接可以允许应用于其他的蓝牙设备通过输入流和输出流进行数据的交换。

有了上面三个类，可以完成很多蓝牙的操作。为了使代码更加简洁，使用前采用单例模式将蓝牙的操作封装成一个工具类命名为 BluetoothUtil，下面叙述具体的实现逻辑：

1. 私有化构造函数，并暴露一个调用的借口；
2. 新建一个 Handler 负责处理蓝牙接收到的数据；
3. 通过系统服务获取 BluetoothAdapter 实例，检测本地的蓝牙配适器是否开启，若否，则开启之；

4. 通过物理地址从 BluetoothAdapter 实例获取 BluetoothDevice 实例;
5. 通过已经封装好的 BluetoothUtil 建立与远程设备（上一步获取的 BluetoothDevice 实例）的连接；

有了 BluetoothUtil，仅仅需要上述的五步就可以实现蓝牙的功能。这一节的最后，叙述对 BluetoothUtil 的封装过程。先给出这个类的成员变量以及方法。

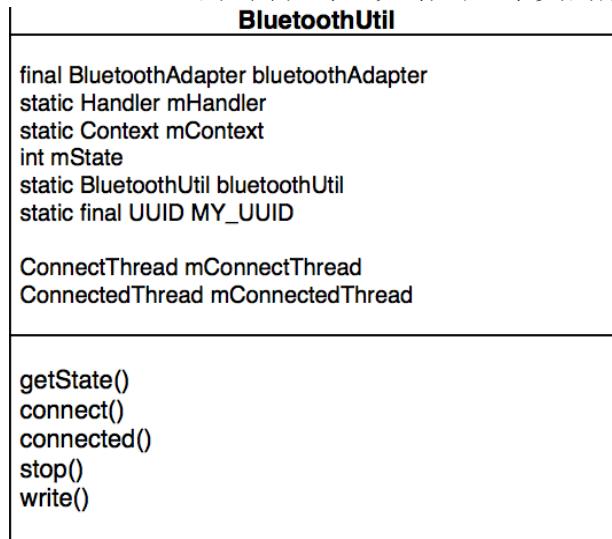


图 4.18 BluetoothUtil 类结构示意图

这个类当中有两个内部类 ConnectThread 和 ConnectedThread。从名字可以看到这个是两个线程，它们分别用于建立蓝牙连接和管理蓝牙连接。使用子线程的好处是可以避免阻塞主线程，造成“失去响应”的错误。从成员变量可以看到，这个类的构造器要求传入的参数有当前的上下文 mContext，以及蓝牙的数据处理器 mHandler。剩下的几个成员变量分别是：用来指示当前的蓝牙状态、特定的蓝牙模块的 UUID、本类的实例，以及蓝牙配适器实例。

从方法名可以看到，提供出来的方法不多。它们的作用依次分别是：获取当前的蓝牙状态、开启蓝牙连接、停止当前的蓝牙连接、通过蓝牙向远程设备写数据。下面分别介绍一下：

- 获取当前的蓝牙状态：每次改变状态都会更新 int 类型的变量，需要获取状态时只需返回这个 int 类型代表的状态即可；
- 开启蓝牙连接：调用 ConnectThread 的 start 方法开启连接子线程，成功连接之后，调用 ConnectedThread 的 start 方法开启管理子线程，最后更新蓝牙状态。
- 停止当前的蓝牙连接：关闭输入输出流，将对应的实例指向 null，更新

- 蓝牙状态；
- 通过蓝牙向远程设备写数据：首先检查是否已经与远程设备建立了连接，如果没有建立连接，则抛出错误。如果已经建立了连接，则通过调用 ConnectedThread 的 BluetoothSocket 的 write 方法向流当中输出数据；

5 视觉处理程序设计

5.1 图像获取

在智能手机被广泛运用的今天，利用 Android 设备获取照片在日常生活当中是一件很平常的事情，基本当下的大部分 Android 手机或者平板上都有提供摄像头硬件。在本次毕业设计当中，需要运用摄像头来获取地面的照片，供视觉处理系统发现以及定位杂草。本文的实验，使用小米平板 2 上的自带摄像头。这一小节主要探讨如何利用 Google 提供的 Android SDK 进行摄像头的定制开发，使其能够依照上位机的命令进行拍照的动作。

首先，来了解一下 Android 的摄像头接口架构，如下图 5.1 所示。整个 Camera2 由一个 CameraManager 来进行统一管理，通过 Context 的 getSystemService 方法可以实例化 CameraManager，然后该类主要通过三个类来对 Camera 进行操作。下面分别介绍一下：

- CameraDevice：描述一个照相机设备，一个 Android 设备可能会有多个摄像头，通过 CameraId 可以进行区别。它最主要有一个相机状态的回调函数，当下达打开相机的命令后，若相机正确的打开便会回调该函数。
- CameraCharacteristic：某个照相机设备的具体参数，主要用到它提供的输出格式（即输出数据的格式）。
- CameraCaptureSession：相机捕获会话，通过这个类可以和相机进行对话（预览还是单张拍照还是录像等）。这里有两个回调函数，捕获状态的回调，和捕获数据的回调。

图 5.1 左上部分所示的是 Android 设备和 camera 设备的通信情况，两者之间通过 pipeline（管道）进行数据交换。当需要进行不同的操作时，将 CameraCaptureRequest 通过管道传给 camera，接收到请求后，camera 做出相应的反应，将获取到得数据 CmaeraMetadata 通过管道传回给 Android 设备。

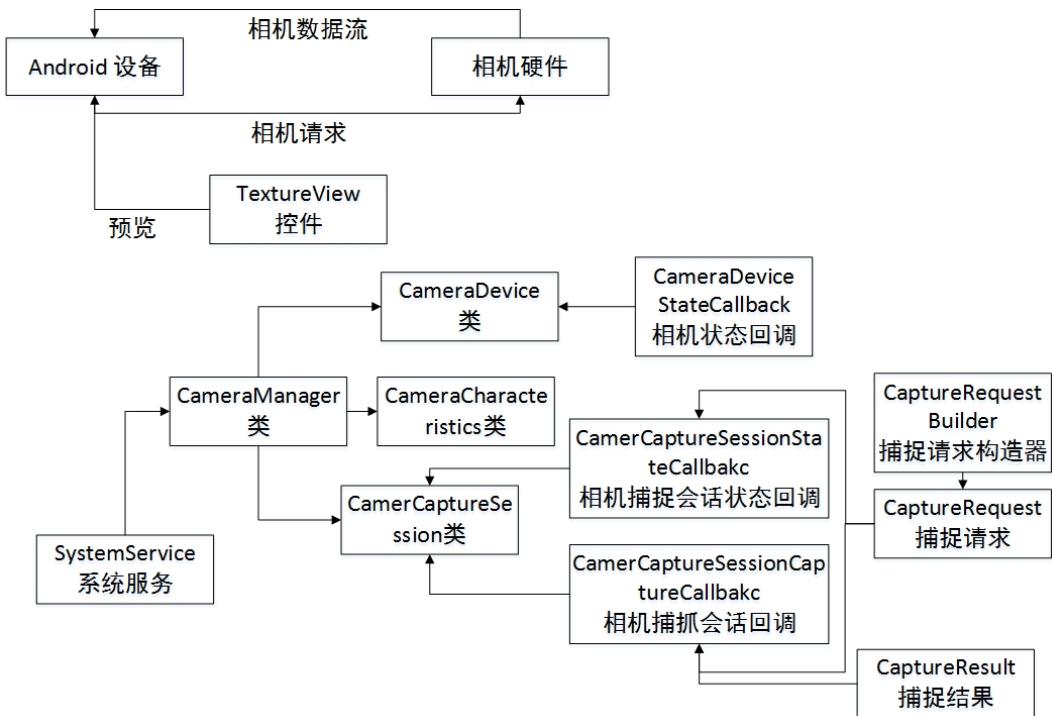


图 5.1 Camera 类代码流转示意图

在本次毕设中，用一个 activity 承载一个 fragment。所有的代码都写在 fragment 里面，重写了 fragment 的几个生命周期函数：

- `onCreateView`: 加载 fragment 的布局文件；
- `onViewCreated`: 实例化布局控件；
- `onActivityCreated`: 在 SD 卡的目录下建立 jpg 文件等待将拍到的照片写进去；
- `onResume`: 开始照相机线程，执行一些逻辑判断；
- `onPause`: 关闭照相机，停止照相机线程；

代码的整体流程如下图 5.2 所示，activity 将需要的 fragment 加载进来后，开始加载显示预览的控件 `texture`，当控件加载完毕会执行一个回调函数 `onSurfaceTextureAvailable()`，在这个回调函数里面，打开摄像头（即执行 `openCamera()`）。`openCamera()` 里面，首先要配置相机的输出，预览图像和拍照的图片要作不同的处理，然后根据当前的设备屏幕环境，判断是否需要进行数据的转换，最后通过 `cameraManager` 打开摄像头（调用 `cameraManager.openCamera()` 方法）。

摄像头打开后，首先进入的是预览模式，也就是显示摄像头实时看到影像。但是，在毕设当中，我们并不需要看到实时影响，而且需要在界面上做一些其他

的操作。所以，需要用一种方法将 Texture 控件隐藏起来。这里采用的方法是将它的宽度和高度都设置为 1dp，虽然它在画面上，但是肉眼并无法看到，也不影响其他的交互操作。

最后，只需将拍照的命令封装成一个方法。当收到上位机发来的拍照命令后，执行一次该方法即可。

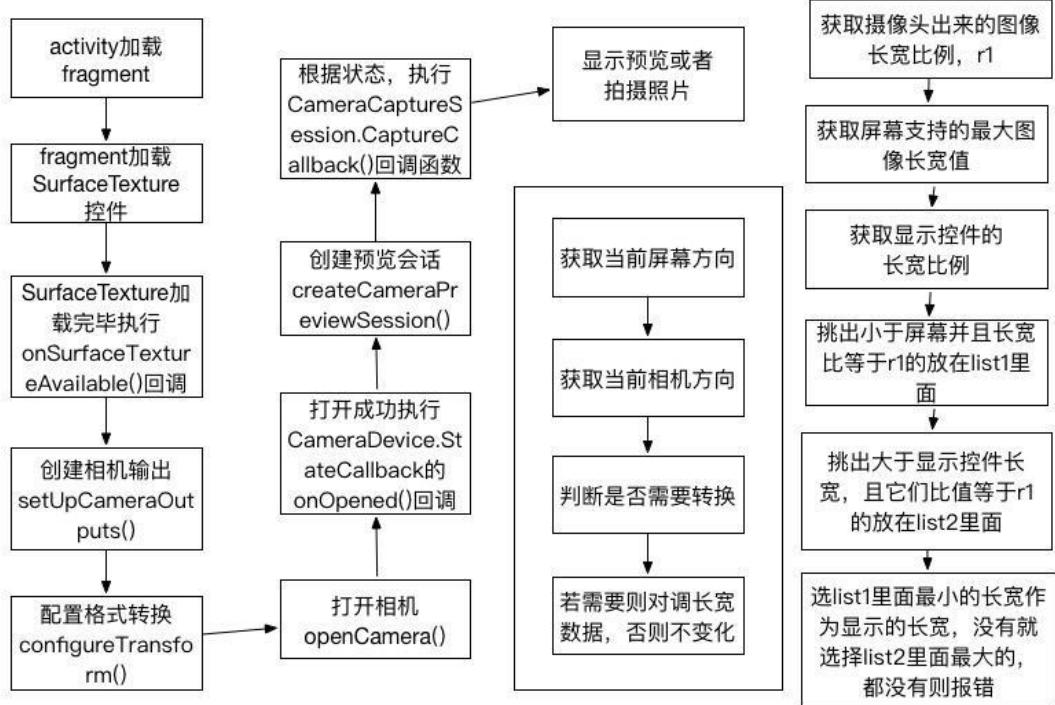


图 5.2 Android 获取照片流程示意图

5.2 图像处理

图像处理，是指用计算机对图像数据进行分析，以达到所需结果的技术，一般是指数字图像处理。图像处理的步骤一般是：提取原始图像、灰度处理、滤波/边缘检测/图像分割、提取特征值。图像数据的格式有很多种，常见的有二值图像（表示该图像的二维数据仅有 1 和 0 组成）、灰度图像（又名 256 灰度图像，它的二维数据元素的取值范围是 [0, 255]，每个值代表一个灰度等级，其中 0 为黑色，255 为白色）和 RGB 图像（图像的每一个像素都由 RGB 三个颜色分量数值组成）等。数字化图像的存储方式有两种，一种是位图存储，一种是矢量存储。位图存储，是将图像的每一个像素点转成一个数据；矢量存储，是存放图像信息经过复杂的分析演算之后得出图像的轮廓。

现代图像资料大多是彩色的，所以现在进行的大部分图像处理的原始图像都

是彩色图像。所以有重要的两个彩色图像的颜色模型需要了解一下，分别是 RGB 颜色模型和 HSV 颜色模型。

RGB 颜色模型，是在工业界广泛应用的一种颜色标准。它通过对红(R)、绿(G)、蓝(B)三个通道的颜色叠加来得到自然界当中各种不同的颜色。目前，这个标准几乎囊括人类视力所能感知到的所有颜色，是使用最为广泛的一种颜色系统。它的模型可以用一个立方体来表示，如下图 5.3 所示。

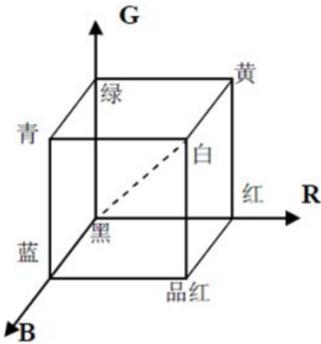


图 5.3 RGB 颜色模型示意图

HSV(Hue, Saturation, Value)是根据颜色的直观特性由 A. R. Smith 在 1978 年创建的一种颜色空间，也称六角锥体模型(Hexcone Model)。这个模型中用三个参数来对一种颜色进行描述，它们分别是：

- 色调 (H)，用角度来度量，取值范围是从 0° 到 360° 。从红色开始按逆时针方向计算，红色为 0° ，绿色为 120° ，蓝色为 240° ；
- 饱和度 (S)，它表示某种颜色接近光谱色的程度，可以看成是某种光谱色与白色混合的结果。其中光谱色所占比例越大，颜色接近光谱色的程度就越高，颜色的饱和度也就越高。饱和度高，颜色则深而艳；
- 明度 (V)，明度表示颜色明亮的程度，对于光源色，明度值与发光体的光亮度有关；对于物体色，此值和物体的透射比或反射比有关。通常取值范围为 0% (黑) 到 100% (白)。

这种颜色模型的示意图如下图 5.4 所示。

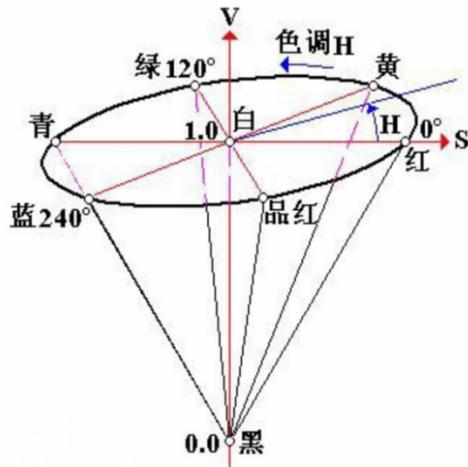


图 5.4 HSV 颜色模型示意图

在本次毕业设计当中，图像处理时由 Halcon 来实现的，下面先介绍一下 Halcon 环境的一些特定用语；

- 图像变量 Image：Halcon 当中用来存放图像数据的变量类型；
- 图像变量 Region：Halcon 当中符合某些性质的像素的子集；
- 数组 Tupe：Halcon 当中的数据集合，类似 C 语言当中的数组；
- 算子：对某个特定算法的实现，并将其封装成一个可以调用的形式，类似 C 语言当中的函数；
- 控制流：Halcon 提供的对图像处理程序流程进行控制的语句，类似 C 语言当中的控制语句；
- 阈值分割：一种基于区域的图像处理手段，通过设定不同的特征阈值，把图像像素点分为若干类常用的特征包括：直接来自原始图像的灰度或彩色特征；由原始灰度或彩色值变换得到的特征。

有了上述的认识，下面开始叙述在本次毕业设计当中使用的图像处理方法及其步骤。由于经费和实验条件的限制，项目当中没有使用真正的草，而是使用不同的方块来模拟农作物的生长环境，其中绿色的方块就代表杂草。整体的图像处理流程如下图 5.5 所示。

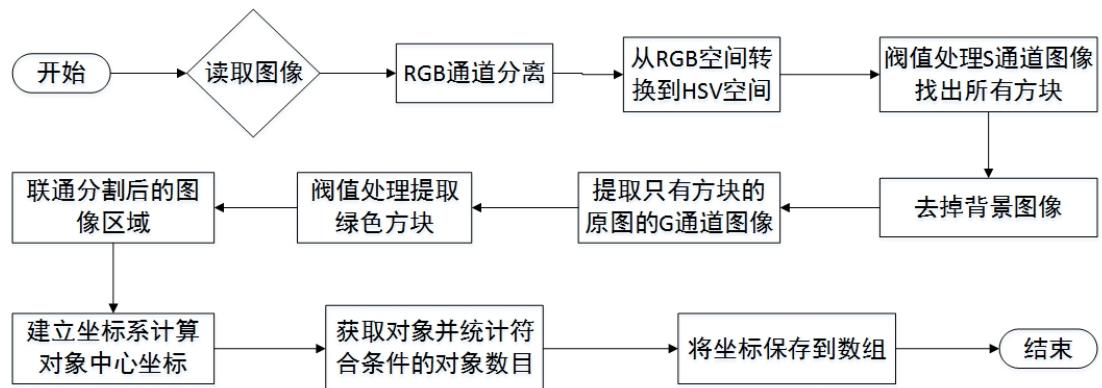


图 5.5 图像处理步骤示意图

下面详细叙述每一步所使用到的算子及其作用：

1. 读取图像：read_image(:Image:FileName)。这个算子的作用是从FileName 的路径下读取符合名字的图片，并将它存放到名字为 Image 所规定的图像变量里面，后续的处理就是对这个图像变量展开的。用于实验的图片如下 5.6 所示。



图 5.6 原始图片示意图

2. RGB 通道 分 离 ， decompse3(MultiChannelImage:Image1, Image2, Image3)。这个算子的作用是将一副多通道的图片，分离成若干单通道的图片。第一个参数，传入上一步读取的图像变量；后面三个参数分别代表分离后的代表 RGB 单通道图像。从这一步执行完毕后，可以看到图 5.7，仍然比较难分辨出方块和背景，所以需要进一步的处理。

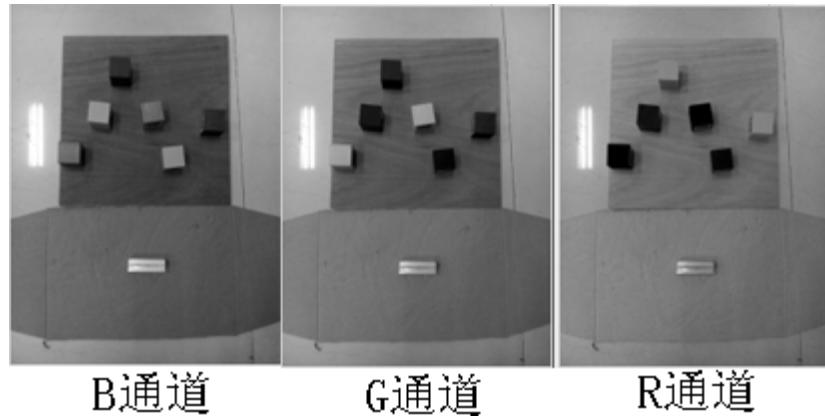


图 5.7 RGB 通道分离处理示意图

3. 从 RGB 空间转换到 HSV 空间, `trans_from_rgb(ImageRed, ImageGreen, ImageBlue:ImageResult1, ImageResult2, ImageResult3: "hsv")`。这个算子的作用是将图像从 RGB 颜色空间转换到 HSV 空间, 由下面的结果图 5.8 可以看到, 从 HSV 空间的 S 通道图像当中, 可以排除掉背景的影响, 准确的识别出方块。

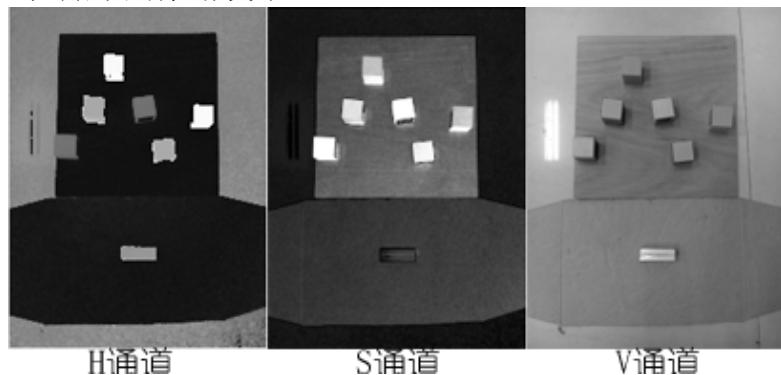


图 5.8 RGB 空间转换到 HSV 空间处理示意图

4. 原图减去方块的图, `reduce_domain(Image, Region:ImageReduced::)`。这个算子的意思是, 从 Image 当中减去 Region 的区域, 把结果存放到 ImageReduced 变量中。处理完后的结果如下 5.9 所示。

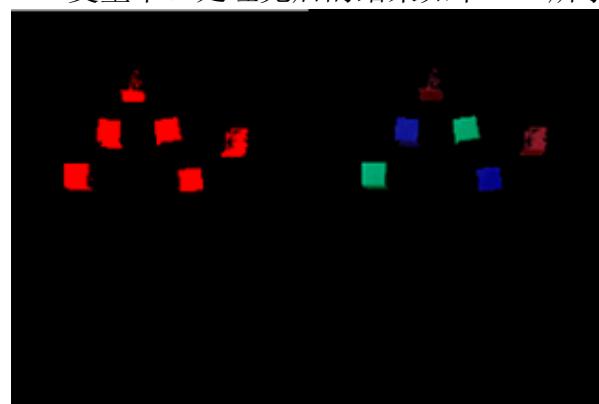


图 5.9 过滤背景处理示意图

5. 在上一步处理的基础上提取方块原图的 G 通道，同第 2 部的算子，处理结果如下图 5.10 所示。

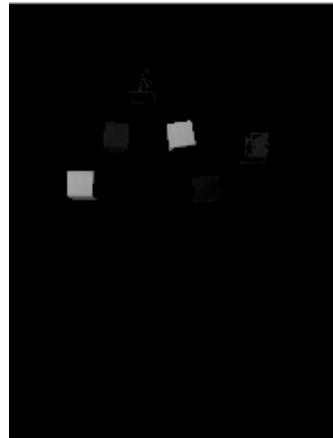


图 5.10 提取需要的物体操作示意图

6. 阈值处理，threshold(ImageReduce:Region:MinGray, MaxGray)。这个算子的作用是对一个 Image 变量进行处理，提取其中灰度值在 [MinGray, MaxGray] 当中的区域，将这个区域存放在 Region 当中。具体在本次毕业设计当中，就是将所有绿色方块的区域提取出来。结果如下图 5.11 所示。

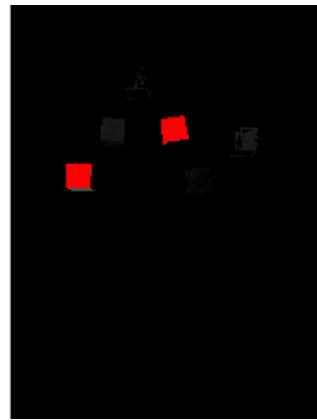


图 5.11 阈值处理操作示意图

7. 区域连通，connection(Region:ConnectedRegions)。这个算子的作用是将上一步选择出来的结果连接成一个 Region，方便后面的对象处理。第一个参数是需要连接的区域，第二个参数是连接后得到的 Region。具体到本次毕业设计当中，就是将提取出来的绿色方块区域组成一个新的 Region，等待后面的处理。
8. 根据面积获取区域，select_shape(Regions:SelectedRegions:Features, Operation, Min, Max)。这个算子的作用是在目前的区域当

中选择面积在[Min, Max]当中的区域，并把它们按照 Operation 的方式组成一个区域，放到 SelectedRegions 当中。具体到本次毕业设计当中，就是在图中将每一个单独绿色方块提取出来。

9. 统计对象，count_obj(Region:Number)。这个算子的作用是统计 Region 的范围当中的对象个数。具体到本次毕业设计当中，就是统计绿色方块的数目，方便后面对每一个方块进行定位坐标计算。结果如下图 5.12 所示。

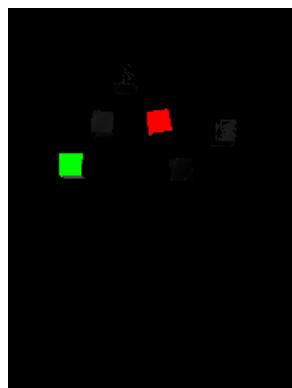


图 5.12 统计对象操作示意图

10. 对每一个符合条件的对象进行坐标计算并保存到 Tupel 当中。本次毕业设计当中，以图像最底部一行的中点为坐标原点，建立笛卡尔坐标系，然后将每一个绿色方块的中心点在坐标系当中标注出来。结果如下图 5.13 所示。

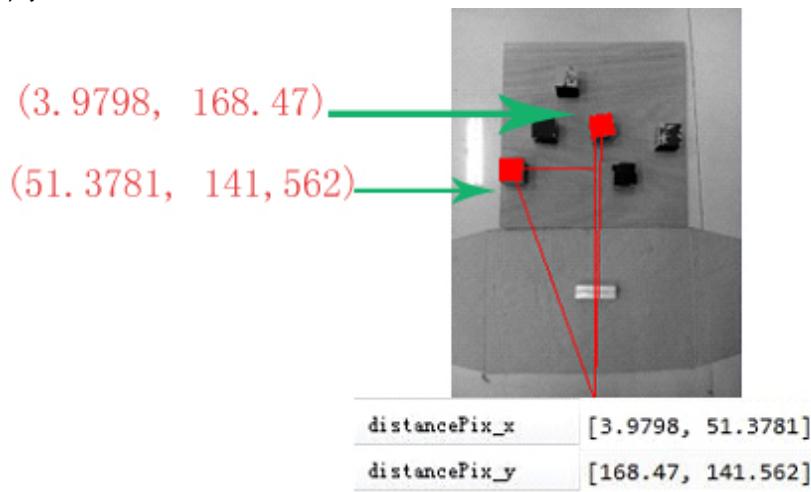


图 5.13 处理结果示意图

以上，是一次对 Android 发送过来的图片的处理过程。从上图可以看到，杂草的位置数据已经获取到，并存在了数组当中。名字为开头 point 的两个数组分

别存放的是杂草中心位置的横纵坐标。名字开头为 `distancePix` 的两个数组，顾名思义，就是以像素为单位的距离数据，它的原点是图片最下方一行像素数据的中点。

结论

在现代农业逐渐走向机器化智能化的时代背景下，本文研究了一种基于机器视觉的除草机器人。该机器人可以在极少人力参与或者完全无人操作的前提下（只需按下开始键），实现对田间生长的杂草的拔除，以达到解放劳动力、节省人力成本、提高农作物经济效益的目的。目前，取得得主要研究成果如下：

1. 设计了一个利用 halcon 机器视觉技术的 vb.net 软件，可以完成对杂草的识别以及定位；
2. 设计了一个 Android 上运行的 app，可以完成图像获取、图片传输的功能和消息中转；
3. 设计了一个 Arduino 的机器人控制程序，可以使机器人自动根据已经识别出来的杂草位置完成除草操作；
4. 设计了一款搭载并联机械手的轮式智能除草机器人，在 Solidworks 上建立机器人的三维模型，并制造出试验样机；
5. 探究并应用了 TCP 网络协议下的 Socket 通信方式；
6. 探究并应用了 Android 设备与 Arduino 的无线蓝牙通信方式；

本次毕业设计虽然完成并实现了一定的功能，但是在如何提高图片传输的速度，如何在没有网络覆盖的地方利用更加方便的方式数据传输方式进行数据传输等方面都还存在很大的提升空间。另外，在机器人的机架结构以及除草结构的设计上也还有很大的发展空间。

致谢

四年的本科生活即将结束，大一刚入学时的许多情景还历历在目，不得不感慨时光的飞逝。

首先要感谢悉心指导我毕业设计的刘镇章老师，本次的毕业设计的顺利完成离不开刘老师的细心指导与帮助。从论文的选题到结稿的各个环节，刘老师都倾注了大量心血，并提出了许多中肯的建议。

其次，要感谢广州大学实验中心的各位老师们，从大二开始一直提供各种机会让我能够接触并且参与到不同的项目当中，为顺利地完成毕业设计积累了不少的项目经验。最后，还要感谢我的舍友陈昊同学和李志聪同学在我遇到困难的时候能够给予种种鼓励和帮助。

另外，还要感谢我的父母和亲人。他们一直以来的支持，是我克服困难不断前进的动力，正是因为如此我才能顺利完成学业。在此，我向他们致以最崇高的敬意。

赖浩滔

2016年5月25日

参考文献

- [1] 相阿荣. 识别杂草和土壤背景物的图像处理方法研究[D]. 北京:中国农业大学, 2001.
- [2] 龙满生. 玉米苗期杂草识别的机器视觉研究[D]. 杨凌:西北农林科技大学, 2002.
- [3] 刘敏. 基于分形的田间杂草图像分析与识别的研究[D]. 北京:北京工业大学, 2002.
- [4] 张健钦. 计算机视觉技术在杂草识别中的应用研究[D]. 河北:河北农业大学, 2003.
- [5] 纪寿文, 王荣本, 陈佳娟, 等. 应用计算机图像处理技术识别玉米苗期田间杂草的研究[J]. 农业工程学报, 2000, 17(2): 154—155.
- [6] 尚卓. 图像处理与自动控制技术在农药喷洒机中的研究[D]. 北京:中国农业大学, 2003.
- [7] 陈勇. 自动施药机器人及可变量控制系统研究[D]. 南京林业大学 2005.
- [8] W. S. Lee, D. C. Slaughter and D. K. Giles. Robotic Weed Control System For Tomatoes precision Agriculture[J], 1999, (1):95~ 113.
- [9] Lei Tian. Sensor-based Precision Chemical Application Systems[A]. Proceedings of the World Congress of Computers in Agriculture and Natural Resources, Iguacu Falls, Brazil, pp. 279–289, 2002.
- [10] 《上海农村经济》 [J] 2003[5]:48.
- [11] 丹麦: 研制出农田除草机器人. 环球农业, 2004.
- [12] Usman Ahmad, Naoshi Kondo, Seiichi Arima, Mitsuji Monta and Kentaro Mohri: Weed Detection in Lawn Field Using Machine Vision – Utilization of Textural Features in Segmented Area-, Journal of JSAM, Vol. 61, No. 2, 61–69(1999).
- [13] 邱白晶, 钱国宏, 周宁等. 割草机器人避障控制 [J]. 江苏大学学报 (自然科学版), 2005, 26(4):285~ 288.
- [14] <http://forums.trossenrobotics.com/tutorials/introduction-129/delta-robot-kinematics-3276/>

- [15] 毛文华 基于机器视觉的田间杂草识别技术研究[D] 中国农业大学, 2004.
- [16] 候学贵 除草机器人杂草识别与视觉导航技术研究[D] 南京林业大学, 2007.
- [17] 葛艳艳 对靶变量施药系统控制软件的研究[D] 莱阳农业大学, 2006.
- [18] 郭伟斌 除草机器人设计与控制[D] 南京农业大学, 2008.
- [19] 王巧华 文友先 刘俭英我国机器视觉技术的发展前沿[J] 《农机化研究》 TP391. 41, 2002.
- [20] 李嘉秀 基于 arduino 平台的足球机器人在 RCJ 中的应用[J] 《物联网技术》, 2015.