

中国科学技术大学

硕士学位论文



基于单目视觉的机器人 避障系统研究

作者姓名：向文辉

学科专业：模式识别与智能系统

导师姓名：汪增福 教授

完成时间：二〇一四年六月二日

University of Science and Technology of China
A dissertation for master's degree



**Obstacle Avoidance System of
Robot Using Monocular Vision**

Author's Name: Wenhui Xiang

Speciality: Pattern Recognition and Intelligence System

Supervisor: Prof. Zengfu Wang

Finished time: June 2nd, 2014

中国科学技术大学学位论文原创性声明

本人声明所呈交的学位论文,是本人在导师指导下进行研究工作所取得的成果。除已特别加以标注和致谢的地方外,论文中不包含任何他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的贡献均已在论文中作了明确的说明。

作者签名: _____

签字日期: _____

中国科学技术大学学位论文授权使用声明

作为申请学位的条件之一,学位论文著作权拥有者授权中国科学技术大学拥有学位论文的部分使用权,即:学校有权按有关规定向国家有关部门或机构送交论文的复印件和电子版,允许论文被查阅和借阅,可以将学位论文编入《中国学位论文全文数据库》等有关数据库进行检索,可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。本人提交的电子文档的内容和纸质论文的内容相一致。

保密的学位论文在解密后也遵守此规定。

☐公开 ☐保密 (____年)

作者签名: _____

导师签名: _____

签字日期: _____

签字日期: _____

摘 要

导航是机器人学科中一个非常重要的研究方向，很多学者在这一领域做出了杰出的贡献，无论采用何种导航方式，避障始终是其主要任务之一。同时，近几年来，以智能手机为代表的移动计算技术的兴起为机器视觉的应用提供了通用的计算平台。本文将机器人、计算机视觉和移动计算技术结合在一起，使地面移动机器人能够利用单目视觉进行实时避障。

本文通过安卓智能手机摄像头获取图像，并利用单目视觉算法进行对图像处理以获得障碍物的位置。该算法这样一个合理假设：场景中所有的物体都垂直于地面，且地面处在图像的底部。地面和非地面物体因为颜色、纹理和亮度的不一致使得地面和非地面的交界处存在着明显的边缘，且该边缘是局部连续和平滑的。根据以上假设，可以利用边缘检测算子计算地面与垂直与地面物体的边界。

在计算图像边缘之前，为了消除阴影所带来的边缘干扰，本文利用 HIS 色彩模型中的 H、S 和 I 分量进行图像融合，实验结果表明该融合结果能够显著地降低阴影带来的影响。

在获得局部连续的地面边界之后，用改进的主动轮廓模型对这些局部连续的边缘进行拟合，最终获得一个平滑地面边界，地面边界上点的深度即为该点的障碍物到小车的距离。机器人在前进中避障时，始终选择地面边界上距离小车最远的点作为前进方向，以达到避障目的。

本文所使用的移动机器人是利用小车改装而来，加装了相应的控制器和传感器。在控制系统的设计上，利用智能手机的惯性传感器计算出小车的姿态，用来控制小车的转弯角度。结合小车速度反馈设计了 PID 控制器，形成小车速度的闭环反馈控制，提高了小车的控制精度。最后的实验结果表明，小车在**快速行驶**的条件下能够实现正确的避障。

关键词：机器人 避障 单目视觉 主动轮廓 传感器融合 移动计算

ABSTRACT

Navigation is a vital research branch of robotics, in which area many scientists have done remarkable work. No matter what kind of navigation method is used, obstacle avoidance is always the primary task. In recent years, the evolution of mobile computing technology, which is represented by smart phone, provides a commonly used computing platform for the application of machine vision. This paper combines robotics with computer vision and mobile computing to enable ground robot to avoid obstacles in real time.

In this thesis, an Android phone is used to capture pictures of the ground, and a monocular vision algorithm is applied to obtain the position of obstacles. Which has several hypothesis: all objects in the scene are perpendicular to the ground and the ground region is in the bottom of the image; Obvious edges exist on the common boundary between ground region and other regions due to the inconsistency of their color and texture. Based on these hypothesis, an edge detector can be used to get the border between the ground region and objects perpendicular to the ground.

Before obtaining the border, image fusion is conducted in the HSI color model to remove the influence of shadow, whose effectiveness is proved by experiment results.

When locally continuous ground border is obtained, it is ameliorated by improved ACM to get a smooth ground border, and the depth of the point on that border is the distance between the car and the very point. The robot will always choose the point on the ground border with a biggest distance from the car as the direction to move forward in order to avoid obstacles.

The robot used in this thesis is a refitted car equipped with a controller and sensors. As for the design of the control system, the car posture is obtained using the inertial sensor of the smart phone to control the rotation of the car. And a PID controller is designed using the car speed to form the closed-loop feedback control system and the performance is improved. The experiment results show that the car can avoid obstacles correctly even when it drives with a high speed.

Key Words: robot, obstacle avoidance, monocular vision, active contour, sensor fusion, mobile computing.

目 录

摘 要	I
ABSTRACT	II
目 录	III
第 1 章 绪论	1
1.1 引言	1
1.2 课题研究背景及意义	1
1.3 课题的国内外研究现状	2
1.4 本文的主要工作	4
1.5 本文的章节安排	5
第 2 章 机器人硬件平台	6
2.1 引言	6
2.2 系统整体框图	6
2.3 智能手机	7
2.4 大脚车	8
2.2.1 电路图	9
2.2.2 微控制器	10
2.2.2 电机的控制	11
2.5 传感器系统	12
2.4.1 图像传感器	13
2.4.2 位置传感器	13
2.4.3 惯性传感器	14
2.6 本章小结	15
第 3 章 图像处理与机器视觉算法	16
3.1 引言	16

3.2	图像算法流程	16
3.3	图像预处理	17
3.3.1	平滑滤波	17
3.3.2	图像融合	19
3.3.3	边缘估计	20
3.3.4	实验结果	22
3.4	地面边界拟合	24
3.4.1	主动轮廓模型综述	24
3.4.2	参数形变模型	25
3.4.3	几何形变模型	29
3.4.4	本文改进模型	31
3.4.5	模型的数值实现	32
3.4.6	实验结果	34
3.5	绝对深度计算	37
3.5.1	深度计算公式	37
3.5.3	实验结果	38
3.6	本章小结	40
第 4 章	机器人控制系统设计	41
4.1	引言	41
4.2	转向角控制	41
4.2.1	惯性传感器特点	41
4.2.1	平衡滤波器	42
4.2.2	实验结果	43
4.3	速度控制	44
4.3.1	控制器设计	45
4.3.2	参数整定	46
4.3.3	实验结果	47
4.4	避障策略	48
4.5	本章小结	49
第 5 章	综合实验及结果	50
5.1	引言	50

5.2 综合实验	50
5.3 结果分析	51
5.4 总结与展望	52
参考文献	53
附录 1 参数轮廓模型变分问题求解过程	58
附录 2 GVF 公式求解及数值实现	59
致 谢	61
在读期间发表的学术论文与取得的其他研究成果	62

第 1 章 绪论

1.1 引言

自机器人诞生以来，机器人技术得到了迅速的发展，机器人的应用范围也扩展到了军事、航空航天、医疗、工业、服务业等各个领域。机器人不知疲倦，操作精度高，可以代替人们做一些危险性的工作，也可以提高产品生产效率，改善产品质量。

尽管目前机器人还不像人们想象的那么强大，但是机器人技术已经渗透到各行各业中，机器人产业也在逐渐变成一个高新技术产业。从某种意义上来讲，一个国家的机器人技术水平的高低反映了这个国家的综合技术实力的高低。在可以预见的将来，机器人将成为人们的得力助手，成为人类朝夕相处的可靠伙伴（谭民，2007）。

1.2 课题研究背景及意义

机器人技术将自动控制工程、机械、电子、仿生、计算数学等等理工科结合在一起，可以说是多学科交叉发展的结晶，也是目前研究最活跃的方向之一。在不同类型的机器人中，地面轮式机器人具有轻、小、快等优点，稳定性好，易于控制，尤其受到人们的青睐。智能化是机器人的一个发展方向，在这之中自主导航又是智能机器人学科中一个重要的研究课题。无论采用何种导航方式，智能移动机器人都要完成路径规划、避障规划与定位等任务（李磊 等，2002）。因此机器人避障系统的研究在移动机器人的研究领域有着重要的意义。

在机器人往智能化方向的发展过程中，研究人员希望机器人能模拟人类的功能，能够感知外部世界，并能有效地理解世界。而在人类的感知系统中，80%以上的信息是通过视觉获取。要想让机器人能够像人类一样感知世界，那么赋予机器人以人类视觉功能对发展智能机器非常重要，由此形成了机器视觉这一门新的学科。

如果从 David Marr 教授在 1977 年正式提出计算机视觉理论以来，计算机视觉也才三十多年的历史。计算机视觉的出现不仅大大推动了智能系统的发展，也将拓宽计算机与各种智能机器的研究范围和应用领域。目前机器视觉在机器人上的应用非常广泛，如智能交通中对行驶车辆进行跟踪，对其驾驶行为进行分析；工业生产线上利用机器视觉对零件进行识别与精确定位；医学图像对核

磁共振图像进行建模分析；国防工业中目标自动识别与目标跟踪等等。

因为视觉信息获取简单，信息量丰富，利用机器视觉对机器人进行导航始终是重要的研究方向。在传统的机器人定位方法中，基于激光扫描仪或双目立体视觉的方法常被用于测量距离信息。但是这些方法造价昂贵、体积大或者标定困难等问题，实用性受到很大的限制。随着计算机视觉技术的发展，基于单目视觉进行导航的方法因为成本低、易于操作等优势受到广泛关注，成为该领域近年来的研究热点之一（向文辉 等，2014）。

处理视觉信息需要强大的计算能力作为支撑。近几年来，包括智能手机、平板电脑在内的移动计算技术正在颠覆传统的个人电脑为代表的桌面计算时代，一台智能手机就是一台小型的电脑。根据市场调研公司尼尔森相关调查(Nielsen, 2013)显示，截至 2012 年，中国有 89%的人拥有手机，其中 66%为智能手机。传统手机用户在减少，他们中的大多数很快将升级为智能手机用户。这意味着不久的将来，仅仅中国就有超过 10 亿人随身携带电脑，这将使得移动电脑成为标准的全球通用计算技术平台（塞勒，2013）。这些计算平台将为机器视觉的广泛应用提供了良好的契机。

面对当今技术的发展趋势，本文意在将具有广阔前景的机器人学科、蓬勃发展的计算机视觉学科、颠覆性的移动计算技术相结合，探寻在移动计算时代来临之际，基于计算机视觉的机器人会有着怎样的发展前景。

1.3 课题的国内外研究现状

定位是移动机器人的一项重要功能，在传统的定位方法中，目前比较流行的室外机器人定位方法是 GPS。实时动态差分法(RTK)GPS 的精度能达到 1cm，但是需要良好的 GPS 信号，不能在室内以及建筑物较多的地方使用。也有学者使用激光（Guivant et al, 2000）和声呐（Tardós et al, 2002）进行周围环境信息扫描，利用扫描的数据创建地图、对机器人导航等。这些传感器都有造价昂贵、设配笨重等问题。

同时，在传统的机器人领域，移动机器人的避障算法根据掌握信息的完整度可以分为障碍物信息已知与障碍物信息未知两种。对于障碍物信息已知这种情况，很多学者已经给出了比较好的解决方案。如可视图法（Janet et al, 1995），栅格法（Khatib, 1986），自由空间法（蔡自兴 等，2000）等。但在实际情况中，障碍物的信息大部分为未知或信息不完全，此时基于全局的路径规划存在缺陷。对此，近年发展出了一些智能的避障方法在某些方面弥补了传统算法的不足。如 Glasius et al（1995）提出的基于神经网络的机器人避障方法，基于遗

传算法的机器人避障算法 (Hu et al, 2004; Awadh et al, 1995), 基于模糊逻辑的机器人避障算法 (Hagras, 2004) 等。

利用机器视觉对机器人进行导航是上个世纪 80 年代以来兴起的一项新技术, 按照视觉传感器来分, 基于机器视觉的导航可以分为三大类:

(1) 360 度全景视觉传感器。该传感器的优势在于可以获得机器人周边 360 度的环境信息。Winters et al (2000) 利用一系列的全景图像, 用机器学习和建立鸟瞰图法实现了机器人的拓扑

导航、视觉路径跟踪。Murillo et al (2007) 则利用 SURF 进行全景图像的特征点匹配法, 实现了室内环境分层的全局定位。但全景视觉传感器的缺点在于图像存在着较大的畸变, 不容易进行特征提取、匹配的相关工作, 并且信息量增多降低了系统的实时性。

(2) 双目视觉传感器。双目视觉模拟了高级生物的视觉系统获取信息的方式, 标定后的双目视觉传感器容易得到目标的深度信息, 因此应用非常广泛。Thorpe et al (1988) 等使用双目视觉识别和检测障碍物, 开发了可以在公路上自主驾驶的 Navlab 系统。2004 年在火星上成功登陆的勇气号火星探测器同样也采用了一套双目视觉系统进行导航 (NASA, 2014)。双目视觉传感器需要进行精确标定, 造价高。

(3) 单目视觉传感器。该方法具有价格低廉, 结构简单, 易于标定等优点, 近些年来, 利用单目视觉进行定位的方法受到了广泛的关注。牛津大学的 Davison 是研究 SLAM (Simultaneous Localization And Mapping) 的代表, 在这方面取得了大量的研究成果 (Davison, 2003; Davison et al, 2007)。他提出了一种基于单目视觉的实时 Mono-SLAM, 该系统可以实时跟踪和处理手持摄像头位姿。但该方法只能在室内小范围内工作。Royer et al (2007) 提出了一种利用单目摄像头和自然路标进行室外机器人的导航的方法, 但该方法需要预先录制场景视频, 进行离线学习。

此外, 利用机器视觉进行导航的方法还有: 光流法 (Santos-Victor et al, 1993), 表观匹配法 (Joulain et al, 1997; Gaussier et al, 1997), 基于目标识别法 (Kim et al, 1998; Kim et al, 1999) 等等。

在利用单目视觉进行机器人避障的相关研究成果还有: Michels et al (2005) 在采集大量户外场景真实深度数据的前提下, 利用机器学习的方法训练分类器。利用该分类器对小车运动过程中采集的单幅彩色图像进行处理, 获得场景的深度信息, 再利用深度信息执行避障动作。优点是速度快, 但是需要大量的训练数据, 只能应用于与训练数据一致的场景中。Santos et al (2010) 利用智能手机上的图像传感器拍摄室内图像, 利用 J2ME 平台进行图像处理, 识别环境中预

先放置好的路标来进行路径规划与避障。缺点是只能在室内有路标的情况下进行避障。

从以上介绍可以看出，计算机视觉在机器人导航等领域的应用广阔，研究非常活跃，成果层出不穷，随着研究技术的进一步发展，计算机视觉与机器人的结合必将迎来更为广阔的发展前景。

1.4 本文的主要工作

本文的主要工作是研究如何利用单目摄像头结合移动计算设备使得机器人避障具备避障能力。本文除了对如何利用单幅图像检测障碍物的算法进行描述，也会介绍本避障系统的硬件平台、传感器系统和控制系统设计等相关内容。最后对整个系统进行实现，并给出避障实验结果。

本文所使用的机器人是在遥控小车上改装而来，在保留了动力的基础上，加装了相应的控制器和传感器，并在小车顶部安装了一部安卓智能手机。本文开发了相应的安卓手机应用程序，使用手机的摄像头作为图像传感器实时地拍摄小车前方的图像，并利用该图像提取地平面的边界。

单幅图像估计地平面的图像算法基于这样一个合理假设：场景中所有的物体都垂直于地平面，地面区域处在图像的底部并且局部连续。在这个假设下，只需要准确估计出地平面和物体的边界，再结合摄像头的标定就可以获得地面上物体的深度信息。

首先，算法利用边缘检测算子获得图像中不连续的边缘。在计算边缘之前，为了排除室外场景下阴影的干扰，用原图的 **HSI** 色彩空间中三个分量进行加权融合，再对融合的结果求边缘。实验结果显示与用原图求边缘相比该方式能够非常明显地去除阴影的干扰。

接着，将对求边缘后获得的不连续的边缘进行拟合，估计出准确的边界。由于地平面与垂直物体的边界具有局部连续、平滑的特点，同时边界的走向又具有大致水平这一更高层次的知识，本文利用主动轮廓模型（**Snakes**）对这些不连续的边界进行拟合。在传统的 **Snakes** 模型的基础上，本文提出的拟合算法结合了 **GVF** 力场和气球力，针对实际情况对 **Snakes** 模型进行了改进，使得 **Snakes** 模型能够快速准确地拟合边界。

最后，当地平面被检测出来后，小车在前进过程中，对当前获得的地平面进行搜索，找出距离小车最远的边界，选择该边界方向作为小车的前进方向，最大程度地远离障碍物，并根据新数据不断更新前进方向，最终达到避障的目的。

此外，为了能够对小车进行精确控制，本文对手机的加速度计、陀螺仪等惯性传感器的数据进行融合，用于控制小车的转向。在小车的传动轴上加装了一个光电编码器，利用测得的小车的前进速度作为反馈信号设计了一个 PID 控制器，形成了对小车速度的闭环反馈控制。

1.5 本文的章节安排

本论文的主要内容将按照如下章节进行安排：

第一章：绪论。主要介绍了利用单目视觉进行机器人避障这项工作的研究背景及其意义，给出了该领域里国内外的相关研究成果及发展趋势，同时在绪论的结尾中对整个论文的结构与内容进行简要介绍。

第二章：机器人硬件平台。本章将对系统进行整体描述，给出整个系统的框图，并且会对机器人避障系统的各个模块进行相关的介绍。

第三章：图像处理与机器视觉算法。本章将对避障系统的图像处理算法进行介绍，分步给出算法的实验结果，并与其它图像算法进行对比。

第四章：机器人避障系统设计。这一章将介绍机器人的控制部分。包括所使用传感器的介绍、传感器信息的融合、控制器的设计以及相应的控制参数的整定。在这一章结束时，将给出相应的控制实验结果。

第五章：综合实验及结果。该章将会对整个系统进行联调，给出最终的实验结果。

第六章：总结与展望。本章将对本论文的内容进行全面总结，分析本避障系统的优缺点，并提出下一步工作重点和研究方向。

第2章 机器人硬件平台

2.1 引言

系统是指为实现预期的目标而将有关元部件互连在一起的集合（Dorf，2002）。为了实现机器人避障这一目标，必然需要设计一个相应的系统。

本章将对机器人避障系统的硬件平台进行介绍。首先将给出系统的整体框图；接着，阐述系统的整个工作流程、相关模块的关系；最后对系统所使用的智能手机、大脚车和传感器系统分别进行介绍。

2.2 系统整体框图

机器人避障系统的整体框图如下图所示：

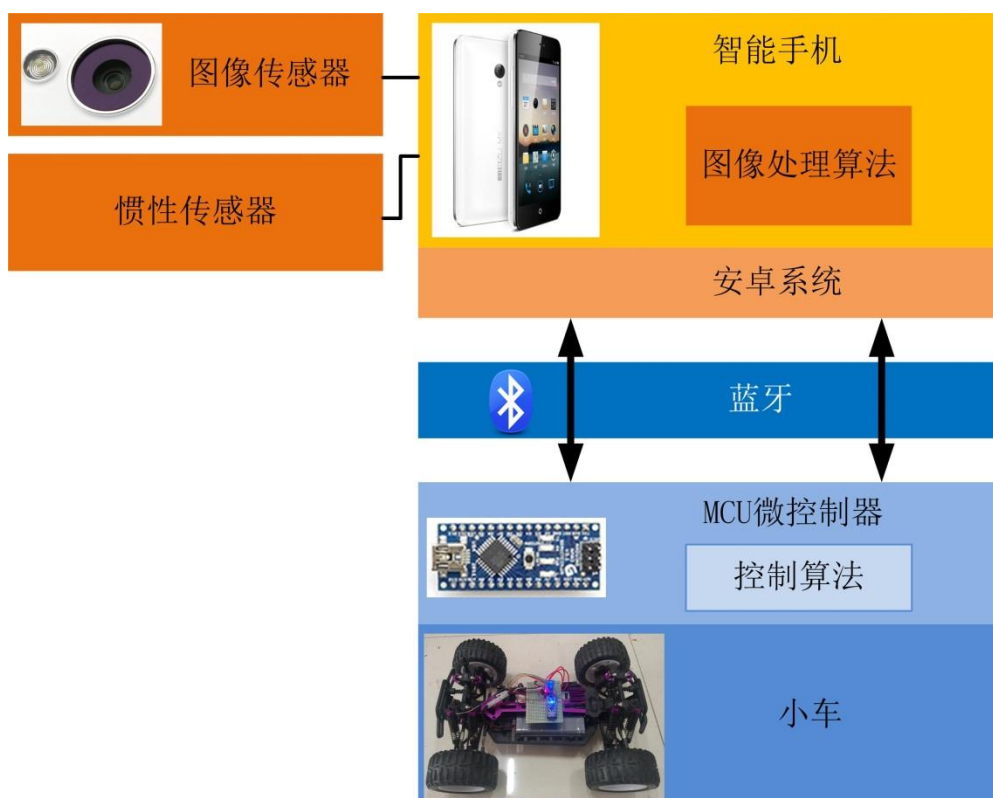


图 2.1 系统框图

按照功能分，整个系统可分为两个部分：

- (1) 安卓智能手机。该手机集成了图像传感器、惯性传感器。负责采集

图像并处理数据，将结果发送到机器人平台。

- (2) 小车平台。该平台属于执行机构，负责接收智能手机发送的命令并按照执行。

整个系统的工作流程如下：

- (1) 将智能手机固定在小车车顶，利用该智能手机上的图像传感器实时采集小车前方的图像。
- (2) 利用本文的图像处理算法对图像进行处理，计算出障碍物的位置并决定小车的前进方向与角度。并通过蓝牙将控制指令发送到小车的微控制器。
- (3) 小车的微控制器在接收到手机的控制指令之后，利用设计的速度、角度控制算法控制小车以指定的速度、角度前进。

在这一过程中，智能手机不断采集当前图像并进行处理，并根据处理结果实时更新小车行进速度、方向，形成一个闭环反馈控制。整个系统分工协作，最终使得小车能够按照指定的方式避开障碍物，实现避障功能。下面将对系统所使用的智能手机、大脚车和传感器系统进行介绍。

2.3 智能手机

移动计算技术在被称为颠覆性的技术，其包括平板电脑，以及最新一代的智能手机。与传统手机不同的是，智能手机是真正的电脑，其操作系统可以安装大量的应用软件。移动计算技术不仅仅是将电话功能加载小型电脑上，它为软件带来了一种全新的模式（应用），一个全新的设计和售卖应用软件的生态系统（应用商店），以及一个全新的人机界面体验（多点触控屏幕）。随着智能手机用户的日益增多，移动电脑最终会成为标准的全球通用的计算技术平台。

目前常见的移动操作系统包括苹果公司的 **IOS**，谷歌公司的 **Android**，微软公司的 **Windows Phone**，黑莓公司的 **BlackBerry** 等。这些操作系统都发布了相关的应用软件开发 **SDK** 包和相应的 **API** 供开发者使用，方便开发者开发手机应用程序，扩展手机的功能。

由于本文进行的是图像与计算机视觉相关的研究，需要图像传感器采集图像，同时也需要对图像进行实时处理，要求处理器有比较强大的性能以保证计算的实时性，智能手机正好满足我们的需求。因此，本系统采用一台 **Android** 智能手机作为计算设备，同时利用了手机的图像传感器与惯性传感器。本文使用的智能手机型号为 **Meizu MX2**，其主要配置如下（该信息来源于魅族公司官网：www.meizu.com，只列出与本文相关信息）：

表 2.1 智能手机配置表

处理器（CPU）	MX5S（A9 四核 1.6GHz，32nm HKMG 工艺）
内存（RAM）	2GB LPDDR2 双通道
显示核心（GPU）	Mali-400MP4（533MHz）
存储空间（ROM）	16GB
传感器	重力感应、电子罗盘、陀螺仪
屏幕	4.4 英寸 1280×800
网络	GSM/TD-SCDMA Bluetooth 4.0 WIFI
操作系统	Flyme 3.0（基于 Android 4.2.1）
尺寸	124.9×64.9×10.2 mm

由上表可以看出，随着科技的进步，智能手机的性能提升速度远超个人电脑。仅仅用了个人电脑一半不到的发展时间，手机厂商就将多核处理器、超大的内存空间等应用到智能手机上。该手机具备四核 1.6GHz 的 ARM A9 处理器，2GB 内存，性能非常强大，但是，该智能手机的配置在目前智能手机日新月异的时代只能算是中等。

2.4 大脚车

本系统所使用的移动机器人是遥控式大脚车，可以通过遥控器远程控制。该车采用四轮驱动，前两轮具备转向功能。为了使手机能够控制该大脚车，在保留了该车的动力系统的基础上进行了改装，加装了微控制器与蓝牙模块。改装后的内部结构如图 2.2 所示：

从功能上分，该机器人可以分为三个部分：

- （1）通信模块。通信模块为蓝牙，能够提供 10 米左右的传输范围。该模块的加入使得机器人能顺利与手机进行双向通信。
- （2）控制器。控制系统为一片 AVR 单片机为主组成的控制系统板。负责解析手机的控制指令并利用内置的控制算法控制整个小车的执行相应动作。
- （3）动力系统。动力部分包括一个 7.2V 直流电机，通过齿轮和传动杆与小车的四轮相连接；一个舵机，负责控制前两轮进行转向；一个蓄电池，电压为 7.4V。



图 2.2 机器人硬件平台

2.2.1 电路图

改装后的机器人电路连接图如图 2.3 所示，图中直流电机的驱动电路为小车自带，输入舵机的控制信号即可控制直流电机的速度以及转动方向，图中本文用一个舵机替代直流电机的驱动电路。

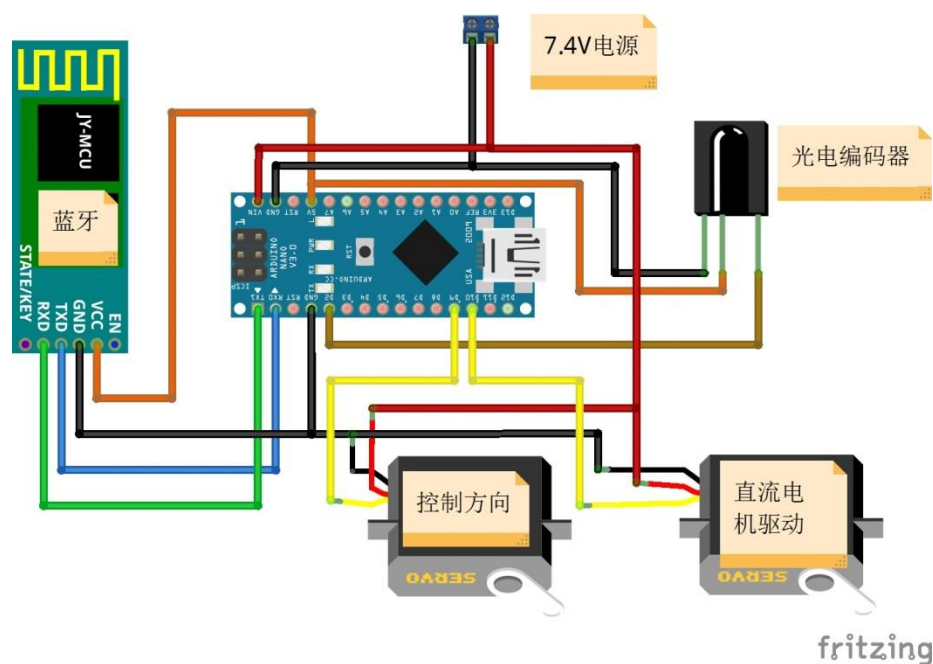


图 2.3 小车电路图

在电路图中，蓝牙模块通过串口与控制器相连接，利用串口协议与 MCU 进行通信。控制方向的舵机与直流电机的驱动电路板的信号线分别连接到 MCU 的 D9、D10 接口，MCU 输出脉冲宽度调制（PWM）信号即可控制舵机旋转角度与直流电机的转速及转动方向。光电编码器则连接到 MCU 的外部中断 1，通

过 MCU 的中断计数来计算小车的行进速度，相关信息将在本章的传感器系统进行介绍。

2.2.2 微控制器

在图 2.3 所示的电路图中所用的控制器为 Arduino Nano 开发板。Arduino 是源自意大利的一款开源的软硬件结合的开发平台，硬件开发板使用的是 Atmel AVR 系列 8 位单片机或者 ARM 系列 32 位处理器。由于其内置了 Bootloader，使得开发板不再需要相应的程序下载器，只需连上电脑 USB 接口即可下载程序调试。图 2.4 即为一些形态、功能各异的 Arduino 开发板。

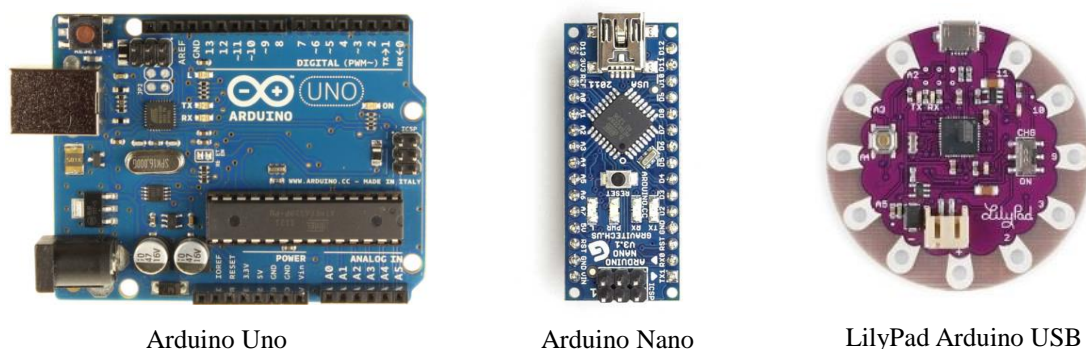


图 2.4 Arduino 开发板

Arduino 的软件集成开发环境（IDE）由 Java 语言编写，具有很好的跨平台特性。程序开发语言风格类似于 C 和 Java，并且提供了很多库，比如舵机的控制、网络通信等，非常方便扩展开发板的功能。

表 2.2 Arduino Nano 参数

主芯片	Atmel ATmega328
工作电压	5V
输入电压	7-12V
数字 IO 口	14 Pins（其中 6 个提供 PWM 波输出）
模拟输入	8 Pins
Flash	32KB
SRAM	2KB
EEPROM	1KB
工作频率	16MHz
尺寸	1.85*4.3（cm）

Arduino 的优势在于软件和硬件的一体化，使得开发者不用重新发明轮子，

不需要特别关注单片机的底层实现部分，而是专注于应用的实现。正是因为这些特点，使得 Arduino 从 2005 年推出后在世界范围内得到了广泛的应用。本文所使用的 Arduino 开发板的型号为 Nano，其主要参数如表 2.2 所示。

2.2.2 电机的控制

该机器人平台上的电机包括两种：一种为舵机，一种为常规直流电机。

(1) 舵机的控制

舵机，也称为伺服电机，是一个糅合了多项技术的科技结晶体，它由直流电机、减速齿轮组、传感器和控制电路组成，是一套闭环反馈自动控制装置。舵机的角度检测器是它的输入传感器，舵机转动的位置变化时位置检测器的电阻值就会跟着变化。通过控制电路读取该电阻值的大小，就能根据阻值适当调整电机的速度和方向，使电机向指定角度旋转。

常用的舵机有一个三线的接口，分别是地线，5V 信号线，和电源线。舵机的控制信号是 PWM 波，PWM 波与舵机的转动关系如下图所示：

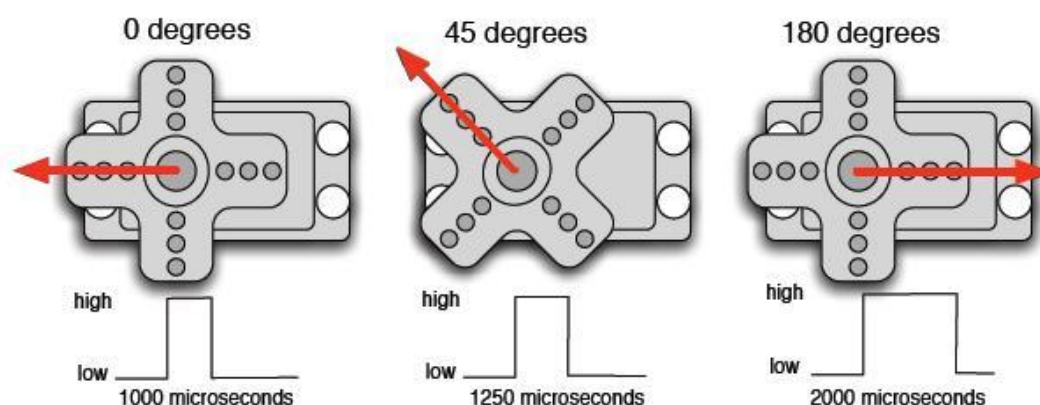


图 2.5 伺服电机控制

在标准的舵机信号中，脉冲的高电平持续 1 到 2 毫秒 (ms)，也就是 1000 到 2000 微秒 (μs)。在 1000 μs 时，舵机左满舵。在 2000 μs 时，右满舵。当然也可以通过调整脉宽来实现更大或者更小范围内的运动。

由前述可知，小车采用一个舵机控制转向，当输入 PWM 波的高电平持续时间为小于 1500 微秒时，舵机向左旋转，小车向左转弯。当输入 PWM 波的高电平持续时间大于 1500 微秒时，舵机向右旋转，小车向右转弯。在这一过程中，小车的转弯角度与舵机旋转角度成正比。当输出 PWM 波的高电平持续时间为 1500 微秒时，舵机处在正中位置，小车直走。

虽然舵机本身是一个闭环控制系统，能够保证转到指定的角度，但是这只

是一个局部的闭环控制系统。在小车的实际行驶过程中，由于机械误差、左右轮不对称、摩擦力不一致等的影响，即使舵机处在正中位置也不一定能保证小车直行，因此需要引入外部传感器反馈，以形成大的小车转向闭环反馈控制系统。与该部分相关的内容传感器将在本章接下来的部分进行介绍，相关的控制系统将在第四章的避障系统的设计中进行介绍。

(2) 电机的控制

直流电机即为依靠直流电驱动的电动机，控制输入的直流电电压大小和电流方向即可控制其转动速度和转动方向。在利用单片机控制直流电机时会加入相应的直流电机驱动电路，通过 PWM 波来控制其转动速度和方向。本文所用的直流电机驱动电路接收标准的舵机信号，并能将该信号的脉宽解析为直流电机的转动速度和转动方向。

通过直流电机驱动电路板对舵机信号的解析，电机转动速度与高电平持续时间 T 有如下关系：

$$V = k(T - 1500) \quad (2.1)$$

当输入的 PWM 波的高电平持续时间小于 1500 微秒时，电机转动速度 V 为负数，电机反转，此时小车向后运动。当输入的 PWM 波的高电平持续时间大于 1500 微秒时，电机转动速度 V 为正数，电机正向转动，小车前进。这一过程中电机的转动速度与 $|T - 1500|$ 成正比，比例系数为 k 。当输出 PWM 波的高电平持续时间为 1500 微秒时，电机静止。

由于小车的传动齿轮与直流电机的齿轮是硬连接，因此小车的运动速度与电机的转动速度也为正比例关系，但是这只是代表了电机空载情况。在小车实际运动过程中，由于小车负载的变动、地形变化等因素会使得小车前进的阻力也随之变化。因此，式 2.1 中的比例系数 k 不是一个定值，而是会随着负载、地形等因素进行变化。为保证小车能以指定速度运动，同样需要利用外部传感器对小车的速度进行测量，并实时反馈到控制器，形成小车的速度闭环反馈控制。相关的传感器将在本章接下来的部分中介绍，控制系统将在第四章的避障系统的设计中进行介绍。

2.5 传感器系统

对于机器人来说，无论是同外部环境进行交互，还是感知自身的姿态，都需要通过传感器来获取相应的信息（谭民 等，2007）。得益于科技的发展，传感器的种类日益丰富，体积、精度相对于多年前无不进步巨大。按照感知信息

对机器人的作用，可以简单地分为内部传感器和外部传感器。内部传感器主要感知与机器人自身参数相关的信息，如速度、角速度、位移、加速度等；外部传感器主要感知机器人以外的外界信息，如：周边环境、障碍物位置、所受外部力等等。对于机器人导航来说，障碍物在环境中的位置与机器人自身的运动状态都是需要利用传感器来获得的信息。下面将对机器人所使用的传感器做一个简单地介绍。

2.4.1 图像传感器

图像传感器，或称感光元件，是一种将光学图像转换成电子信号的设备，它被广泛地应用在数码相机和其他电子光学设备中。根据传感器制造材质不同，可分为感光耦合元件（charge-coupled device, CCD）和互补式金属氧化物半导体有源像素传感器（CMOS Active pixel sensor）两种。在同等条件下，CMOS 图像传感器所用的元件数相对更少，从而功耗较低，数据吞吐速度也比 CCD 更高，因此目前几乎所有的手机所用的摄像头光学元件都采用 CMOS。

图像传感器是小车最主要的传感器之一，下表是小车所采用的摄像头参数信息：

表 2.3 手机摄像头参数

像素	800 万
材质	CMOS 背照式
光圈	F2.4
视频	1080P/720P/480P 30 帧/s
焦距	4mm 定焦镜头
CMOS 大小	1/3.2 inch: 4.54(W)×3.42(H), mm

800 万像素的清晰度对于本文的小车来说已经足够清晰了，为了提高处理速度，在实际采集的图像分辨率为 360*240。

2.4.2 位置传感器

如前所述，为了能够对小车的速度形成闭环反馈控制，需要测量小车的行进速度。在众多的速度测量传感器中，码盘被广泛使用。

从功能分，码盘属于位置传感器，其主要作用是测量位移，包括角位移和线位移。下图为轮式机器人上常用的线位移传感器为增量式旋转编码器：

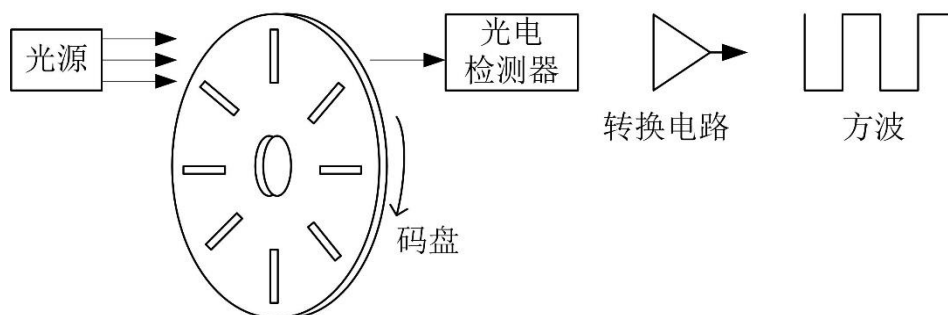


图 2.6 增量码盘原理图

如图所示，中间是一个可透光的光栅圆盘，光源和光敏传感器被安装在码盘的两侧。当光源发出的光线被码盘遮挡时，光敏传感器接收不到信号，输出低电平；当光源发出的光通过码盘，照射到光敏传感器上时，输出高电平。随着码盘的转动，传感器就会连续不断地输出信号，通过对该信号的上升下降沿进行计数就可以测量小车的线位移。对测得的小车的线位移对时间进行微分，即可获得小车的行进速度。

2.4.3 惯性传感器

如前所述，为了能够控制小车的前进方向，需要加入传感器测量小车的姿态，本文所用来测量小车姿态的传感器为陀螺仪和加速度计，均属于惯性传感器。智能手机上集成的惯性传感器采用的是微机电系统（MEMS）技术，即利用半导体制造技术将微型的机械结构、信号采集放大器与处理电路等集成在一

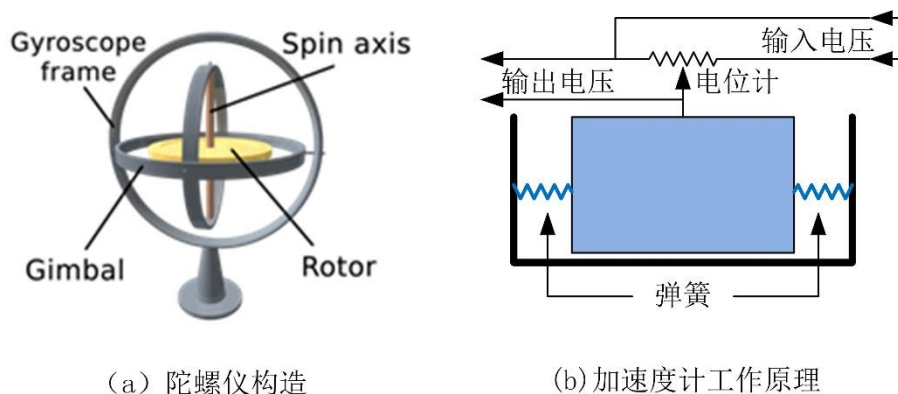


图 2.7 陀螺仪构造和加速度计的工作原理

起，它的操作范围在微米级别。

陀螺仪是一种用来测量运动物体角度、角速度和角加速度的传感器，基于

角动量守恒的理论设计出来的。陀螺仪主要是由一个位于轴心且可旋转的转子构成，如图 2.7（a）中金黄色 Rotor 所示。陀螺仪一旦开始旋转，由于转子的角动量，陀螺仪轴心的转子有抗拒方向改变的趋向，而周边设备的任何姿态的改变就可以检测出来。因此，利用陀螺仪可以测量运动物体的姿态角（航向、俯仰、横滚），精确测量其角运动。

加速度计是用于测量加速度的装置，虽然利用位置传感器测得的位移对时间进行二阶微分可以获得加速度，但在实际应用中想要获得精确的加速度还是需要使用加速度传感器传感器。

加速度计是惯性导航和惯性制导系统的基本测量元件之一，如图 2.7（b）所示，加速度计本质上是一个振荡系统，安装于运动载体的内部，可以用来测量载体的运动加速度。MEMS 类加速度计的工作原理是当加速度计连同外界物体（该物体的加速度就是待测的加速度）一起作加速运动时，质量块就受到惯性力的作用向相反的方向运动。质量块发生的位移受到弹簧和阻尼器的限制，通过输出电压就能测得外界的加速度大小。

2.6 本章小结

本章介绍了机器人避障系统的硬件平台，给出了系统的总体结构图以及该系统各部分之间的相互关系，同时对系统的工作流程进行了阐述。在介绍硬件平台各个子模块之时，不仅给出了各个模块的相关特性，也对各个模块的选型依据进行了讨论。

本文用一台智能手机作为核心模块，在作为计算设备的同时也充当了图像传感器和惯性传感器的作用。此外，本文改装了大脚车作为移动机器人，加装了微控制器，以控制电机；加装了相应的蓝牙模块，以方便地与手机进行数据通信；加装了位移传感器测量小车的速度，以形成对小车速度的闭环反馈控制。本章工作的目的是构建一个快速、准确、稳定地执行避障指令的机器人，为后续机器人避障提供良好的硬件基础。

第3章 图像处理与机器视觉算法

3.1 引言

快速准确的地平面估计是解决移动机器人自主导航问题的关键点，它有助于场景中地表的定位。当准确地估计出地平面后，结合摄像头的标定结果就可以计算出地面上的物体的绝对深度。

在利用单幅图像估计地平面的研究成果中，Hoiem et al (2005) 采用一种基于图的纹理分割算法 (Felzenszwalb, 2004) 将图像场景分割成地面、垂直于地面的物体以及天空，并借助一些假设将这三个部分分别映射到 3D 模型的相应区域中，构成一幅立体图。但是这个方法只适用于有限的一些场景，并且求得的地面信息并不精确，所获得结果更接近于图像分类，无法用于机器人的精确定位。

Cherian et al (2009) 在 Make3D (Saxena et al, 2009) 结果的基础上提出了一种利用图像分割来进行地平面估计的算法，同样利用了基于纹理的分割方法 (Felzenszwalb, 2004)。作者首先将图像分割成纹理相似的小区域；然后利用高斯马尔科夫随机场模型 (Gaussian Markov Random Field) 重建 Make3D 的深度图像，获得每个像素的深度值，该深度值可用来计算分割出来的相邻两个小区域是否共面；最后从图像底部中间位置的小区域（小车所在位置）开始进行横向优先搜索，将与其相邻的并且共面的小块进行合并，获得最终的地面区域。该方法能够获得的结果比较准确，但是这个方法是在 Make3D 的结果上扩展而来，只能应用于结构化的室外场景中，并且其计算量很大，不合适在移动设备上实现。

为了能够求得比较精确的地平面，并且易于在移动设备上实现，本文提出了一种由车载单目摄像头获取的单幅图像估计 3 维地平面的方法，本章将对这一算法进行介绍。

3.2 图像算法流程

该算法的流程图如图 3.1 所示，该算法首先根据图像的特点对图像进行了平滑滤波；然后将图像转换到 HSI 色彩空间，利用了图像中的色调、饱和度信息，对 HSI 空间的 H、S 和 I 分量进行图像融合，降低阴影所带来的干扰；接着利用边缘算子计算出融合后图像中的主要边缘，再利用改进的主动轮廓模型

拟合这些局部连续的边缘得到一条完整平滑的地面边界；最后，结合得到的地面边界信息和摄像头的标定进行深度计算，获得最终的深度信息。

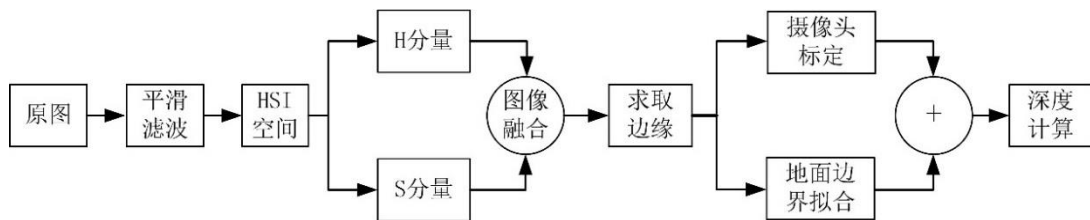


图 3.1 图像算法流程图

本论文图像算法基于以下几个合理假设：

- (1) 场景中所有的物体都与地面有接触，没有悬在地面之上的物体。在该假设下，机器人前方的图像分为地面与垂直于地面的物体两个部分，并且物体与机器人的相对深度可通过物体与地面的接触点到摄像头的距离来。
- (2) 地面处在图像的底部并且边缘平滑且局部连续，基于该假设地面的边缘可以用一条足够平滑的曲线来进行拟合。
- (3) 摄像头安装在小车顶部，其光轴与地平面平行。在该假设下对摄像头进行标定可获得摄像头的内外参数，接着利用第一个假设获得的相对深度结合摄像头参数可计算物体的绝对深度。

本章接下来的部分将对算法流程的每一步分别进行介绍。

3.3 图像预处理

3.3.1 平滑滤波

与静态拍摄的图像不同，由于小车在行进过程中速度比较快、地面不平整带来的摄像头的抖动和图像采集过程中图像的延迟等因素的影响，动态采集的图像会出现一定程度的模糊，模糊的情况主要表现在以下两个方面：

- (1) 图像中会局部模糊。这是由于图像的采集需要一定的时间，在采集过程中小车的运动会使图像有拖影现象出现。
- (2) 图像会出现图像底部清晰，上部分模糊，模糊程度与小车的距离成正比。这是因为摄像头的焦距是固定的，对焦点总是在摄像头的前方，使得近处的图像清晰，远处的图像模糊。

如图 3.2 所示，左上图中红色虚线区域相对于其上下区域而言边缘更为模糊，并且竖直方向应该是直线的黄黑色护栏也因为采集过程中的延迟而导致变

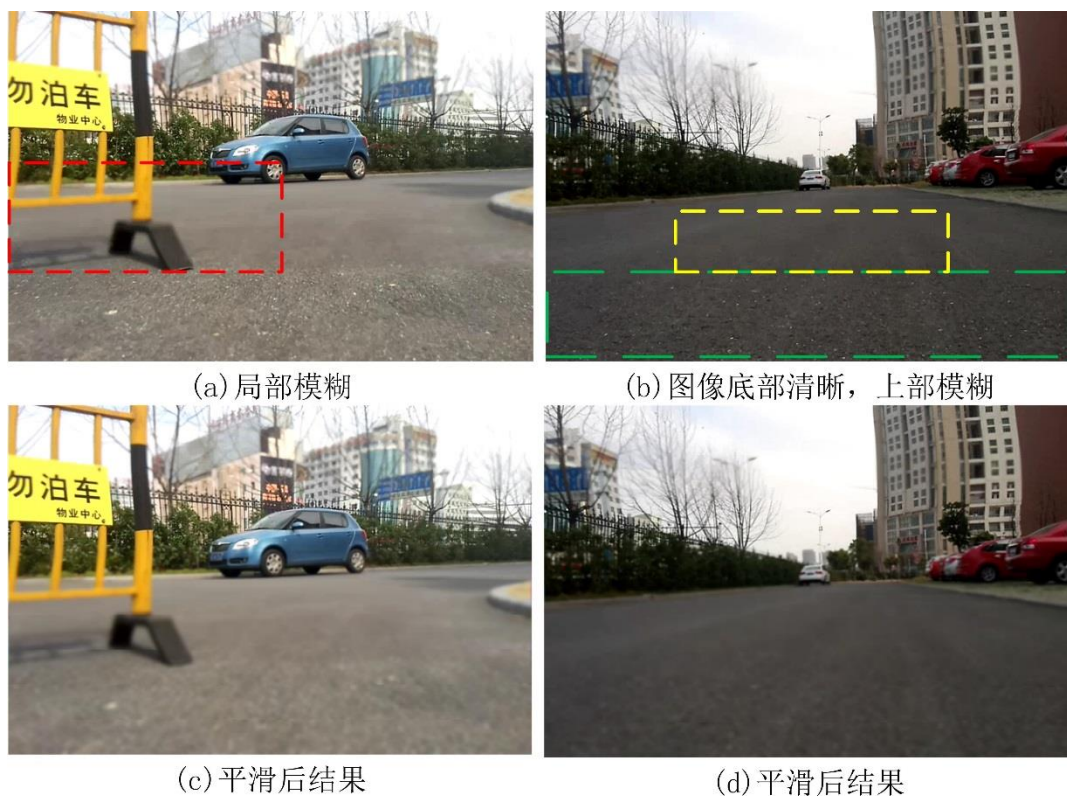


图 3.2 图像平滑滤波

得弯曲。右上图中同样是路面，但是绿色区域中的路面非常清晰，甚至能看到路面的纹理，但是紧随其上的黄色区域中并没有路面纹理出现。因此在进行下一步的处理之前，需要对图像进行平滑处理。

针对这种情况，本文设计了一个二维高斯函数滤波器，标准差为纵坐标的函数，其表达式如下：

$$G(u, v, y) = \frac{1}{2\pi f(y)^2} e^{-(u^2+v^2)/(2f(y)^2)}, \quad (3.1)$$

其中 r 是模糊半径 $r^2 = u^2 + v^2$ ， $f(y)$ 是该正态分布的标准差， $f(y)$ 的表达式如下所示：

$$f(y) = \frac{k}{\sigma\sqrt{2\pi}} e^{-(1-y/h-\mu)/2\sigma^2} \quad y = 0, 1, 2, 3, \dots, h-1, \quad (3.2)$$

式中， k 为比例系数， h 为图像的高度， y 为图像的纵坐标。与一般坐标轴不同的是，该图像坐标轴的零点在图像的左上角，当 $y=0$ 时是图像的第一行， $y=h-1$ 时是图像的最后一行。将高斯函数 3.1 与原图像进行卷积即可得到模糊后的图像：

$$\mathbf{I}_G(x, y) = G(u, v, y) * \mathbf{I}(x, y). \quad (3.3)$$

图 3.2 中(c)(d)即为相应图像进行高斯滤波后的结果($r=9, \sigma=0.3, k=3$), 与图像的上部分区域相比, 图像的下部分使用了更大的标准差, 从图中可以看出近处的路面经过模糊之后减少了细小的纹理, 并且与局部模糊区域的分界线不再明显, 减少了对下一步中求边缘的干扰。

3.3.2 图像融合

对于图像中地面上的阴影而言, 强度信息变化比较明显, 而彩色信息几乎不变。因此, 图像融合的目的是为了将图像的强度信息和彩色信息分离, 以减少阴影的影响。

从图像彩色产生的角度来讲, RGB 色彩空间非常理想, 用彩色摄像机即可同时捕捉 RGB 三原色然后构成一幅彩色图像。并且, RGB 色彩空间与人眼很强烈地感觉红、绿、蓝三原色的事实能很好地匹配。但是这并不能很好地适应实际上人解释的颜色。比如, 首先它没有涉及用组成其颜色的每一原色的百分比给出某一种颜色, 此外也不认为彩色图像是由三幅图像合成了一幅单一图像(冈萨雷斯 等, 2007), 这对彩色描述上的应用有较多限制。因此, 图像用 RGB 三原色来表示图像, 不能很好地分离颜色和亮度信息。

而 HIS 色彩空间是从从人描述色彩的角度出发, 用色调 (Hue)、饱和度 (Saturation) 和亮度 (Intensity) 来描述色彩。色调是用来描述纯色的属性, 如纯黄色、红色或橘红等; 饱和度给出了一种纯色被白光稀释程度的度量。亮度是一个主观描述子, 实际上是不可测量的, 这一点与单色图像的强度 (灰度) 并不一样, 它体现的是无色的强度概念, 是描述彩色感觉的关键参素。这种彩色描述对人而言是自然而直观的。

更为重要的是, HSI 模型可在彩色图像中从携带的彩色信息 (色调和饱和度) 里消去强度分量的影响。使得 HSI 模型成为开发基于彩色描述的图像处理方法的理想工具。

基于以上所述, 为了减少地面上阴影的影响, 本文首先将图像从 RGB 色彩空间转换到 HSI 色彩空间, 并对该图像的 H、S、I 分量进行加权求和, 减弱亮度分量在其中的影响, 实现图像的融合。

给定一幅 RGB 彩色格式的图像, 从 RGB 色彩空间到 HSI 色彩空间的公式可用如下方程表示。其中式 3.4 为色调 H 的计算表达式, 式 3.6 为饱和度 S 的表达式, 式 3.7 为强度 I 的表达式。

$$H = \begin{cases} \theta & B \leq G \\ 360 - \theta & B > G \end{cases} \quad (3.4)$$

其中：

$$\theta = \arccos \left\{ \frac{[(R-G) + (R-B) / 2]}{[(R-G)^2 + (R-G)(R-B)]^{1/2}} \right\}, \quad (3.5)$$

$$S = 1 - \frac{3}{R+G+B} [\min(R, G, B)], \quad (3.6)$$

$$I = \frac{1}{3}(R+G+B). \quad (3.7)$$

将 H、S、I 分量进行加权求和，获得的图像即为融合之后的结果：

$$I_F = w_1 H + w_2 S + w_3 I, \quad (3.8)$$

其中 w_1, w_2, w_3 分别为色调、饱和度和亮度的权重系数， I_F 为归一化的融合图。

3.3.3 边缘估计

当获得了图像的融合结果之后，接下来的工作是估计地面与垂直于地面的物体的边界，该边界是图像中不连续的部分，也称之为图像的边缘。

图像边缘通常反映了图像属性的重要事件和变化，这些变化包括但不限于：深度上的不连续，表面方向不连续，物质属性变化和场景照明变化等。边缘检测是图像处理和计算机视觉中的基本问题，边缘检测的目的是标识数字图像中亮度变化明显的点。

边缘检测通常可通过空域微分算子与原图的卷积来完成，实际应用中数字图像中的求导是利用差分近似微分来完成的（章毓晋，2005）。对于一个连续函数 $f(x, y)$ ，它在位置 (x, y) 的梯度可表示为：

$$\nabla f(x, y) = [G_x, G_y]^T = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]. \quad (3.9)$$

这是一个矢量，其幅度和方向角分别为：

$$\text{mag}(\nabla f) = \|\nabla f_{(2)}\| = [G_x^2 + G_y^2]^{1/2}, \quad (3.10)$$

$$\phi(x, y) = \arctan(G_y / G_x). \quad (3.11)$$

以上各式中的偏导数需要对每个像素位置进行计算，在实际中常用小区域模板卷积来近似计算。对 G_x, G_y 各用一个模板，所以需要两个模板组合起来构

成一个梯度算子。根据模板的大小以及元素的不同，很多学者已经提出了许多种不同的算子，其中比较有名的有罗伯特交叉（Roberts cross）算子，如图 3.3（a）所示，它使用的是两个 2×2 的模板，以及普瑞维特（Prewitt）算子和索贝尔（Sobel）算子，他们都使用了 3×3 的模板，如图 3.3（b）和（d）。算子运算时采用的是卷积的方式，将模板在图像上移动并在每个位置计算对应中心像素的梯度值，对一幅灰度图求梯度的结果是一幅梯度图。

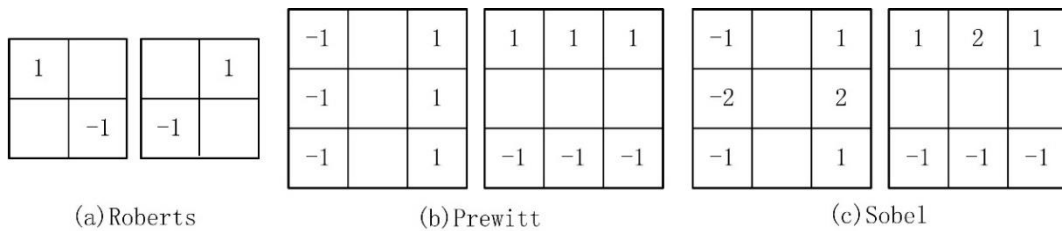


图 3.3 几种常用梯度算子模板

Canny（1986）把边缘检测问题转换为检测单位函数极大值的问题来考虑，他的目标是找到一个最优的边缘检测算法，最优边缘检测的含义是：

- （1）低失误差率：算法尽可能多地表示出图像中的实际边缘，少将非边缘判为边缘。
- （2）高位置精度：标识出的边缘要与实际图像中的实际边缘尽可能地接近。
- （3）最小响应：对每个边缘有唯一的响应，得到的边界为单像素宽度。

为了满足这些要求，Canny 使用了变分法，这是一宗寻找满足特定功能的函数的方法，最优检测使用四个指数函数项的和表示，但是它非常近似于高斯函数的一阶导数。

Canny 算法的步骤如下：

- （1）降噪。

任何边缘检测算法都不可能在未经处理的原始数据上很好地处理，所以第一步是对原始数据与高斯平滑模板做卷积，得到的图像与原始图像相比有轻微的模糊，去掉了单独的一个像素噪声对图像的影响。

- （2）计算梯度大小和方向。

图像中的边缘可能会指向不同的方向，所以 Canny 算法使用 4 个 mask 检测水平、垂直以及对角线方向的边缘。原始图像与每个 mask 所作的卷积都存储起来。

- （3）非极大值抑制。

对于每个点我们都标识在这个点上的边缘模的最大值以及生成的边缘的方向。这样我们就从原始图像生成了图像中每个点亮度梯度图以及亮度梯度的方

向。

(4) 双阈值检测和连接。

较高的亮度梯度比较有可能是边缘，但是没有一个确切的值来限定多大的亮度梯度是边缘多大又不是，所以 Canny 借助了滞后法则选取了两个阈值并最后确定边缘点。两个阈值分别为高阈值和低阈值。认为所有灰度大于高阈值的像素肯定都是边缘像素，而灰度值大于低阈值的像素则要看他们是否与高阈值的像素结合在一起，如果相邻则也认为是边缘像素，这个方法可以减弱噪声在最终边缘图像中的影响。

Canny 算法包含许多可以调整的参数，它们将影响到算法的计算的时间与实效：

高斯滤波器的大小：第一步所用的平滑滤波器的大小将会直接影响 Canny 算法的结果。较小的滤波器产生的模糊效果也较少，这样就可以检测较小、变化明显的细线。较大的滤波器产生的模糊效果也较多，将较大的一块图像区域涂成一个特定点的颜色值。这样带来的结果就是对于检测较大、平滑的边缘更加有效。

阈值：使用两个阈值比使用一个阈值更加灵活，但是它还是有阈值存在的共性问题。设置的阈值过高，可能会漏掉重要信息；阈值过低，将会把细小变化也当成边缘。因此在实际求 Canny 的过程中常使用的是自适应阈值。

由于 Canny 算法能获得单像素宽度的最优边缘，能够适用不同的场合，因此本文的边缘是通过 Canny 算法得到。

3.3.4 实验结果

图 3.4 为对图像融合前后求边缘的结果。如图所示，共有 (a) (b) (c) 三幅带有阴影的图像。其中，第一行图像为原图经过平滑后的灰度图，第二行图像为图像融合的结果，第三行边缘图为原图求边缘的结果，第四行边缘图为融合图求边缘的结果。

为了展示图像融合结果对阴影的影响，选择了三幅不同场景下的阴影图，这三幅图的地面上都有阴影的存在。(a) 图的阴影在图像下部，是一条平直的阴影带；(b) 图和 (c) 图在图像的中部，阴影的边缘比较不规则。更多求边缘相关结果将会在下一节的实验中进行展示。

对比原图和融合图的边缘结果可以看出：针对原图的求边缘的结果中阴影的边缘也被完整地检测出来，该边缘会在下一步的边界拟合中错误地当成地面与垂直于地面的物体的边界。而与此相对的是，对融合图求边缘的结果则比较好地解决了这个问题，在保留了绝大部分真正的边缘的基础上，很大程度上消

除了阴影对边缘结果所带来的影响。

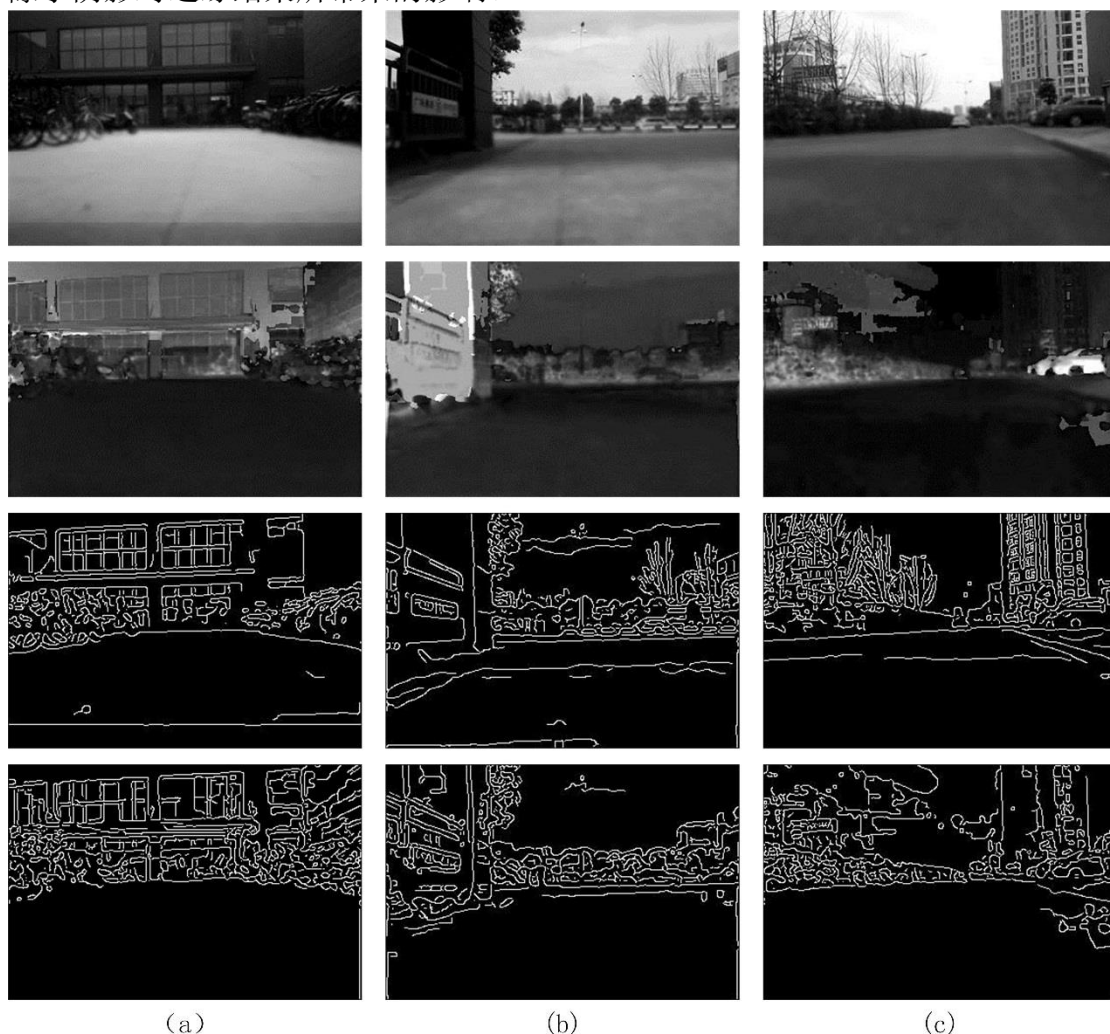


图 3.4 图像融合对求边缘的影响

但是，图像融合的结果在减少了阴影干扰的同时也会带来一定的负面影响。主要表现在以下两个方面：

- (1) 融合图像的边缘图在消除了阴影边缘的同时，也会使得原本连续的边缘变得不连续。

这是因为融合图使用的是颜色信息，要求地面和垂直于地面的物体之间颜色不能太相似，在绝大部分场景下能满足这个要求，但是偶尔也会存在与地面颜色一致的障碍物，这会导致地面边界的不连续。边缘的不连续对边界的拟合提出了更大的挑战，要求拟合过程中能跨过比较大的不连续边缘。

- (2) 在融合图中会出现空洞。

空洞如图 3.4 (c) 第二行右边中部白色区域所示，这是因为从 RGB 到 HSI 空间的转换过程中，若 $S=0$ 时，对应的是无色的点，那么 H 就没有意义。此时定义 H 为 0，当 $I=0$ 时， S 也没有意义。因此，当出现无色的色块时，在融合

图中就会出现空洞，这些空洞的边缘同样会带来一些错误的边缘。

由以上分析可知，虽然图像融合的结果不是非常完美，使得边缘不连续的同时引入了新的错误边界，加大了后期边界拟合的难度。但从总体上来说利大于弊，并且融合图中的 H 分量局部区域无定义的问题可以在后期通过区域生长的方式来避免。

3.4 地面边界拟合

当获得地面与垂直于地面的物体的边缘之后，需要对这些局部连续的边缘进行拟合，使得获得的曲线尽可能地平滑。由前文的假设可知：地面处在图像的底部，边缘平滑且局部连续。因此，地面的边缘可以用一条足够平滑的曲线来进行拟合。基于该假设，本文提出了一种基于主动轮廓模型（Snakes）的地面边界拟合算法，最终获得了一条平滑的连续曲线作为地面边界。

3.4.1 主动轮廓模型综述

活动轮廓模型又被称之为 Snakes，最早由 Kass（1988）在 1987 年为解决图像分割问题而提出。该模型的基本思想很简单，以 2D 图像为例，在图像上创建一条初始曲线，通过曲线的形变，使得曲线在保持自身特征的同时与图像局部特征相匹配，即某种能量函数的最小化，使得曲线贴紧目标的轮廓线而不再形变为止，这样就完成了图像的分割任务。



图 3.5 主动轮廓模型

在 Snakes 这样一个简单的模型下，蕴含了深刻的计算机视觉理论的发展成果（李天庆 等，2005）。以 Marr 为代表的计算机视觉理论，采用自底而上的处理方法，将图像底层特征如边缘、纹理等进行聚集获得 2.5 维简图，最终求解三维模型；而与之相对应的是从心理学发展过来的基于推理的计算机视觉理论，采用自顶而下的处理方法，从人们理解视皮层信息处理的基本原理出发建立相

应的感知模型。边缘、角点、纹理、亮度、光流等都是局部特征，属于底层知识；而人们对于物体的主要认识是来源于其轮廓，属于高层知识。Snakes 的出现很好地融合了这两个知识。图像局部特征与高层知识分别代表了 Snakes 中的外部力和内部力。

由于 Snakes 有着深刻的理论背景，因此提出后很快就成了计算机视觉领域的研究热点。Snakes 衍生出了多种不同的模型，研究人员也提开发出了多种求解算法，被迅速应用到图像分割、视频跟踪、三维重建等各个领域。

与一般的图像模型不同的是，Snakes 是活动的，总是最小化它的能量函数，表现出一种动态行为。基本的形变模型主要包括两种 (Xu et al, 2000)：参数形变模型 (Parametric deformable models) 和几何形变模型 (Geometric deformable models)。参数形变模型中用曲线的参数方程来表达曲线形变过程，这种模型的优势在于用户可以方便地与模型进行交互，并且用参数方程的表达方式更容易用计算机进行快速实现。与之相对应的几何形变模型基于曲线演化和水平集方法，优点是能很好地表达曲线拓扑结构的变化。

为了论述的完整性，本文将对两种模型都进行介绍，并给出参数形变模型的数值实现。

3.4.2 参数形变模型

用于最小化的能量函数是外力和内力的加权和，内力由 Snakes 的形状计算得到，它将使得曲线有弹性并且平滑；而外力由图像获得或是更高层的图像理解过程中得到，会吸引 Snakes 朝着物体的边缘运动。

一条形变曲线可以被定义为：

$$\mathbf{X}(s) = (x(s), y(s)), s \in [0, 1], \quad (3.12)$$

这里的 $x(s), y(s)$ 是轮廓点的 x, y 处的坐标值。这条曲线在图像上运动，以使得如下能量函数最小化：

$$\varepsilon(\mathbf{X}) = \zeta(\mathbf{X}) + \rho(\mathbf{X}), \quad (3.13)$$

该式右边的第一项即为曲线的内部能量项，定义为如下的形式：

$$\zeta(\mathbf{X}) = \frac{1}{2} \int_0^1 \alpha(s) \left| \frac{\partial \mathbf{X}}{\partial s} \right|^2 + \beta(s) \left| \frac{\partial^2 \mathbf{X}}{\partial s^2} \right|^2 ds. \quad (3.14)$$

曲线的一阶微分使得 Snakes 具有弹性，曲线会表现出收缩特性；曲线的二

阶微分使得曲线具有刚度，它会使曲线具有平滑特性。 $\alpha(s), \beta(s)$ 分别表示了弹性和刚度的权重，若刚度的权重为 0，那么表示允许曲线出现角点，在实际应用中， $\alpha(s), \beta(s)$ 取为常数。

式 3.13 右边的第二项为势能函数 $P(\mathbf{X})$ 沿着曲线轮廓的积分：

$$\rho(\mathbf{X}) = \int_0^1 P(\mathbf{X}(s)) ds, \quad (3.15)$$

该势能函数 $P(x, y)$ 可利用图像数据来求得，也可以借助外力建立一个势能场，该势能在图像边缘或其他图像特征位置最小。以常用的图像边缘特征为例建立一个势能函数：

$$P(x, y) = -w_e \left| \nabla [G_\sigma(x, y) * I(x, y)] \right|^2, \quad (3.16)$$

其中 w_e 为大于 0 的权值系数， $G_\sigma(x, y)$ 是标准差为 σ 的高斯模糊函数， $I(x, y)$ 为 2D 灰度图像。 ∇ 为梯度算子， $*$ 为 2D 卷积符号。

当内部能量函数与外部能量函数都确定后，需要求解能量函数 $\varepsilon(\mathbf{X})$ 的局部极小值。为简化表达，将 $\mathbf{X}(s)$ 用 \mathbf{X} 表示。在这个能量函数中 \mathbf{X} 为 s 的函数， $\varepsilon(\mathbf{X})$ 为 \mathbf{X} 的函数，这是一个泛函，求泛函的极小值是变分问题，一般用变分法来解（苏家铎，1993）。

该变分问题即为求解以下形式泛函的极值问题：

$$\mathbf{J}[s] = \int_0^1 \mathbf{F}(s, \mathbf{X}, \mathbf{X}_s, \mathbf{X}_{ss}) ds, \quad (3.17)$$

其中 \mathbf{X}_s 表示 \mathbf{X} 对 s 的一阶偏导， \mathbf{X}_{ss} 表示 \mathbf{X} 对 s 的二阶偏导， \mathbf{F} 的表达式如下：

$$\mathbf{F}(s, \mathbf{X}, \mathbf{X}_s, \mathbf{X}_{ss}) = \frac{1}{2} \left(\alpha \left| \frac{\partial \mathbf{X}}{\partial s} \right|^2 + \beta \left| \frac{\partial^2 \mathbf{X}}{\partial^2 s} \right|^2 \right) + P(\mathbf{X}(s)), \quad (3.18)$$

式 3.17 在 \mathbf{X} 处取得极值的必要条件是 \mathbf{X} 满足欧拉-泊松（Euler-Poisson）方程：

$$\mathbf{F}_\mathbf{X} - \frac{d}{ds} \mathbf{F}_{\mathbf{X}_s} + \frac{d^2}{ds^2} \mathbf{F}_{\mathbf{X}_{ss}} = 0, \quad (3.19)$$

这里 $\mathbf{F}_{\mathbf{X}_s}$ 是 \mathbf{F} 关于 \mathbf{X}_s 的偏导数， $\mathbf{F}_{\mathbf{X}_{ss}}$ 是 \mathbf{F} 关于 \mathbf{X}_{ss} 的偏导数，将式 3.18 代入式 3.19 可得：

$$\frac{\partial}{\partial s} \left(\alpha \frac{\partial \mathbf{X}}{\partial s} \right) - \frac{\partial^2}{\partial^2 s} \left(\beta \frac{\partial^2 \mathbf{X}}{\partial^2 s} \right) - \nabla P(\mathbf{X}) = 0. \quad (3.20)$$

式 3.20 的具体计算过程请参见附录 1。将该式重写为：

$$E_{int} + E_{ext} = 0, \quad (3.21)$$

其中：

$$E_{int} = \frac{\partial}{\partial s} \left(\alpha \frac{\partial \mathbf{X}}{\partial s} \right) - \frac{\partial^2}{\partial^2 s} \left(\beta \frac{\partial^2 \mathbf{X}}{\partial s^2} \right), \quad (3.22)$$

$$E_{ext} = -\nabla P(\mathbf{X}). \quad (3.23)$$

式 3.21 包含了深刻的物理意义， E_{int} 即为前面表述的内力，第二项 E_{ext} 为外力。外力使得曲线往图像特征处运动，但内力阻碍曲线在运动过程中拉伸和弯曲程度。当两个力平衡时，曲线不再运动，泛函取得局部极小值，曲线停下来位置即为 Snakes 的最终解。

为求解式 3.20，将 $\mathbf{X}(s)$ 定义为随参数 s 和时间 t 变化的函数 $\mathbf{X}(s, t)$ ，并定义其对 t 的偏微分等于式 3.20 的左边部分：

$$\gamma \frac{\partial \mathbf{X}}{\partial t} = \frac{\partial}{\partial s} \left(\alpha \frac{\partial \mathbf{X}}{\partial s} \right) - \frac{\partial^2}{\partial^2 s} \left(\beta \frac{\partial^2 \mathbf{X}}{\partial s^2} \right) - \nabla P(\mathbf{X}), \quad (3.24)$$

系数 γ 使得左边的单位与右边一致。该式的意义在于：当曲线固定下来不再随着时间变化时，曲线关于时间的偏微分等于 0，左边项消失，我们获得了式 3.20 的一个解。

Kass 提出的传统主动轮廓有着能够耦合上下层知识、鲁棒性强等优点，但缺点也非常明显：

- (1) 在没有图像力平衡的情况下会导致曲线最终收缩到一个点；
- (2) 要求被分割的物体完全包括在初始曲线内，否则在物体范围内的点无法回到边界；
- (3) 只有梯度很大的边缘点会对曲线的形变产生影响等。

针对这些问题，很多学者提出了改进方法。Cohen et al (1991) 将图像边缘力归一化，并将气球力引入主动轮廓模型：

$$E_p = w_p N(\mathbf{X}) - w_d \frac{\nabla P}{\|\nabla P\|}. \quad (3.25)$$

其中 P 是图像边缘力， N 为曲线上单位法向量， w_p 和 w_d 为气球力和归一化边缘力的权重系数， w_p 的符号能决定曲线是向内收缩还是向外膨胀。在系数选择过程中， w_d 要稍大于 w_p ，使得边缘力能稍大于气球力，让曲线能在边缘

点处停止下来。

气球力的引入让封闭的 Snakes 曲线在没有外部力的情况下能够像一个气球充气一样向外膨胀,使得初始化 Snakes 能够更加随意,可以初始化在目标轮廓的内部,并且在没有图像力的情况下 Snakes 也能向外进行演化。

除此之外,Snakes 的局限性也在于要求初始化要靠近期望的解和对边界的凹形部分的分割存在困难。为了解决这两个问题,Xu et al (1998)提出了梯度矢量场(GVF)的概念,并将其用于 Snake 图像分割。GVF 在边界周围时是一个指向边界的无漩涡的外力场,它在同性图像区域上都朝向图像边界光滑地变化,因此它可以将 Snakes 从更远处拉向图像边界,并能解决物体凹型边界分割的问题。与经典的 Snakes 和气球力法相比,它不会因为平滑边界而无法定位边界,也不需要精细调整气球压力来克服噪声又不至于对突出的图像特征造成影响,Snakes 的初始位置即可在边界内也可在边界外,其初始位置对最终的结果几乎没有影响。

GVF 的求解过程是通过扩散边缘图像的梯度矢量求解一个解耦的线性偏微分方程,从而最小化一个能量泛函。梯度矢量场定义为 $\mathbf{v}(x, y) = [u(x, y), v(x, y)]$,最小能量函数可表示为:

$$E = \iint \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |\mathbf{v} - \nabla f|^2 dx dy, \quad (3.26)$$

其中 μ 是一个正则参数,用来平衡两项权重(当噪声增长时提高 μ)。 f 为图像边缘图(edge map)在物体的边界处有较大的值,可以用任意一种边缘算子求得。

这同样也是一个变分问题,可以用欧拉方程来解,得到梯度矢量场相对于时间 t 的偏微分方程如式 3.27 所示,具体求解过程与数值实现参见附录 2。

$$\frac{\partial \mathbf{v}}{\partial t} = \mu \nabla^2 \mathbf{v} - (\mathbf{v} - \nabla f)(f_x^2 + f_y^2), \quad (3.27)$$

其中 $\mathbf{v}(x, y, 0) = \nabla f$, $\partial \mathbf{v} / \partial t$ 为 $\mathbf{v}(x, y, t)$ 相对于时间 t 的微分, ∇^2 为拉普拉斯算子。对式 3.27 求解后,会在整个图像中产生梯度矢量流,将远离边界的曲线拉向边界。

图 3.6 为上述几种方法的结果比较。图(a)为曲线的初始位置。图(b)

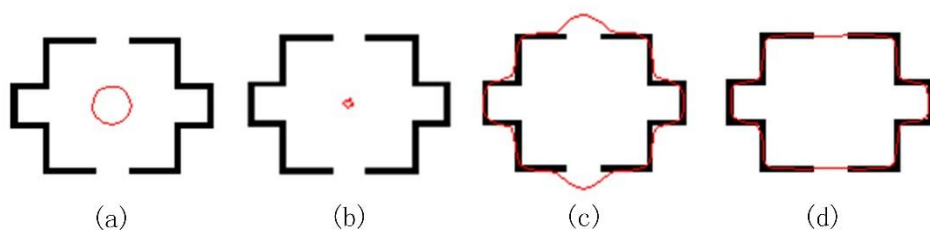


图 3.6 几种 Snakes 结果比较

为传统 Snake 的结果，因为初始化位置距离边缘较远，传统外力没有办法使曲线向边缘靠近，造成曲线最终收缩到一个点。图(c)为加入向外气球力的结果，由于有向外的压力，曲线不能很好地处理缺口。图(d)为 GVF Snake 的结果，可以看出其效果最好，不仅能很好地处理凹口，也能很好地处理缺口，并且还具有从远距离初始化的能力。

3.4.3 几何形变模型

为解决参数形变模型的一些局限性问题，Caselles et al (1993) 和 Malladi et al (1995) 分别独立提出了几何形变模型。该模型主要基于曲线演化理论 (Sapiro et al, 1993; Alvarez et al, 1993; Kimia et al, 1995; Kimmel et al, 1995) 和水平集方法 (Osher et al, 1988; Sethian, 1999)。基于水平集的曲线演化不再依赖于曲线的参数，而是更高一维的水平集函数，演化驱动力仅仅依靠曲线的几何特性，这种与曲线参数无关的表示方式能够很好地表达曲线拓扑结构的改变。

(1) 曲线演化理论

用于图像分割的曲线一般是指简单闭合曲线，在演化过程中的闭合曲线不会发生交叉。一条随时间变化的闭合 2D 曲线的参数化表示为 $\mathbf{X}(s,t)=[X(s,t),Y(s,t)]$ ，其中 s 为曲线的参数， t 为时间。令 \mathbf{N} 为曲线上向内的单位法向量， κ 为曲率，那么曲线沿着其法向量方向演化的过程可以用以下偏微分方程来表示：

$$\frac{\partial \mathbf{X}}{\partial t} = V(x)\mathbf{N}, \quad (3.28)$$

其中 $V(x)$ 决定了曲线各点的演化速度，称为速度函数。该速度函数可以是一个常数，即 $V(x)=V_0$ ，此时曲线各点沿单位法向量的演化速度一致（取正数为收缩力，取负数为膨胀力），该种形变常被称为常数形变 (Constant Deformation)。速度函数也可作为曲率 κ 的函数，即 $V(x)=\alpha\kappa\mathbf{N}$ ，各点的形变速度与该点的曲率相关，此时曲线的驱动力变为长度收缩力，使得曲线沿着曲线弧长度最小化的方向上进行变化，曲线长度不断减小，这与参数形变模型中弹性内力的变形结果是一致的。该种形变被称为曲率形变 (Curvature Deformation)。

(2) 水平集方法

曲线演化可以通过水平集来实现。水平集的方法最初由 Osher 和 Sethian (1988) 共同提出，他们用水平集函数 $\phi(x,y,t)$ 的零水平集来隐式地表示演化的闭合曲线。

水平集定义为有着同样函数值的自变量的点集。如图 3.7 所示,红色的曲面是相应的水平集函数 ϕ 的图像,灰色图形为良性边界包围的有界区域,其边界即为曲线。 ϕ 的某个水平面决定了该灰色图形的形状。假设其中的蓝色平面即为 xy 平面,则形状的边界可以表示为 ϕ 的零水平集,并且该形状是平面上满足 $\phi \geq 0$ 的点的集合(图中灰色图形)。当曲线的拓扑结构改变,分裂成两个闭合的曲线时,曲线的参数方程很难表示这一变化过程,而水平集方法只需更新水平集函数,将红色曲面向下方移动一点就能表达曲线拓扑结构的变化。

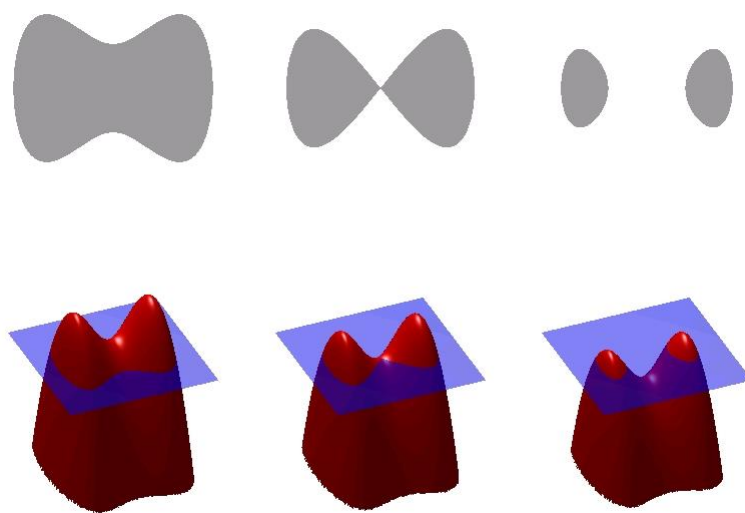


图 3.7 水平集方法

可以用如下水平集函数 ϕ 的零水平集来表达曲线 $\mathbf{X}(s, t)$:

$$\phi[\mathbf{X}(s, t), t] = 0 \quad (3.29)$$

对上式左边求关于 t 的偏微分,并应用链式法则可得:

$$\frac{\partial \phi}{\partial t} + \nabla \phi \frac{\partial \mathbf{X}}{\partial t} = 0, \quad (3.30)$$

其中 $\nabla \phi$ 为函数 ϕ 的梯度。假定零水平集以内 ϕ 为负值,以外为正值,则曲线上向内的单位法向量可用方程 3.31 表示:

$$\mathbf{N} = -\frac{\nabla \phi}{|\nabla \phi|}. \quad (3.31)$$

将式 3.31 代入 3.28, 可以将式 3.30 重写为:

$$\frac{\partial \phi}{\partial t} = V(\kappa) |\nabla \phi|, \quad (3.32)$$

该式即为用水平集方法进行曲线演化的基础。其中零水平集曲线的曲率定义为：

$$\kappa = \nabla \frac{\nabla \phi}{|\nabla \phi|} = \frac{\phi_{xx}\phi_y^2 - 2\phi_x\phi_y\phi_{xy} + \phi_{yy}\phi_x^2}{(\phi_x^2 + \phi_y^2)^{3/2}}. \quad (3.33)$$

(3) 速度函数

速度函数的对曲线的演变意义重大，Caselles et al (1993) 和 Malladi et al (1995) 在提出几何形变模型之时使用的速度函数如下：

$$\frac{\partial \phi}{\partial t} = c(\kappa + V_0)|\nabla \phi|, \quad (3.34)$$

其中：

$$c = \frac{1}{1 + |\nabla(G_\sigma * I)|}. \quad (3.35)$$

当 $V_0 > 0$ 时，曲线收缩，当 $V_0 < 0$ 时曲线膨胀。 c 为图像提供的停止项，能够减缓以致停止曲线的演化，最终获得物体的边界。

除此之外，还有很多速度函数被设计出来，如 Caselles et al (1997) 和 Yezzi et al (1997) 用能量函数最小化来设计速度函数，增加了额外停止项，使得能够将跨过边界的曲线拉回。

3.4.4 本文改进模型

本章的目标是拟合一条分段光滑的曲线作为地面边界，依据前文的分析，该曲线具备以下特征：

- (1) 与传统的主动轮廓模型不同的是，需要拟合的目标轮廓不再是一条闭合的曲线，取而代之的是一条光滑且连续的曲线。
- (2) 该曲线的初始状态为图像最下面的一条直线。
- (3) 拟合曲线在演化过程中没有交叉，也没有拓扑结构的改变。

针对以上特点，本文在参数轮廓模型的基础上进行了改进，将用于闭合曲线的参数轮廓模型应用于一条非闭合的曲线上。同时，为了推动曲线快速向地面与地面物体的边界处运动，将气球力引入 GVF 模型。

该模型的内力即为式 3.14 所示，加入曲线的一阶和二阶微分，使得曲线在具有弹性的同时也具备一定的刚度。当曲线距离边缘较远时，曲线的外力为气球力，促使曲线向图像上方运动，以接近边缘。当曲线离边缘足够近时，外力为 GVF 力场和气球力的合力。这一模型充分运用了气球力和 GVF 的优势，其

特点在于:

- (1) 虽然 GVF 力场具备从远距离初始化的能力,但是计算距离边缘较远区域的力场需要更多的迭代次数,如果只计算边缘附近的 GVF 力场则能成倍地减少计算时间,加快了计算速度。
- (2) 对于远离边缘的区域,利用气球力可以推动曲线向边缘前进。在曲线靠近边缘的时,GVF 力的加入能够加速曲线的停止,防止曲线在气球力的作用下越过边缘。

3.4.5 模型的数值实现

本文的主动轮廓改进模型的数值实现分为两步:

- (1) 曲线初始化。

由于需要拟合的是一条曲线,因此,本文将曲线初始化为图像底部的一条直线,对应图像最下面一行,曲线的长度即为图像的宽度,加入垂直向上的气球力。

- (2) 数值实现。

对于参数形变模型的数值实现,二十几年来,学者们提出了多种方法,其中比较有代表性的有:有限差分法(Kass et al, 1988; Terzopoulos et al 1988)、动态规划法(Amimi et al, 1990)、贪婪算法(Williams et al, 1992)、有限元素法(McInerney et al, 1995; Cohen et al, 1992; Terzopoulos et al, 1990)。

其中比较常用的是有限差分法和有限元素法。虽然有限元素法能够很好地表达形变表面的不规则网格,但是计算比较复杂。相对而言有限差分法只需要进行局部运算,易于实现。因此,本文使用的是有限差分来对主动轮廓模型进行数值实现,下面将给出数值实现的推导公式。

将式 3.24 中的外力用 $E_{ext}(\mathbf{X})$ 表示可得:

$$\gamma \frac{\partial \mathbf{X}}{\partial t} = \frac{\partial}{\partial s} \left(\alpha \frac{\partial \mathbf{X}}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta \frac{\partial^2 \mathbf{X}}{\partial s^2} \right) + E_{ext}(\mathbf{X}). \quad (3.36)$$

首先,对形变曲线 \mathbf{X} 进行离散化表示: $\mathbf{X}_i^n = (X_i^n, Y_i^n) = (X(ih, n\Delta t), Y(ih, n\Delta t))$; 然后,利用该离散化表示的曲线对式 3.36 进行离散化,结果为:

$$\begin{aligned}
 \gamma \frac{X_i^n - X_i^{n-1}}{\Delta t} = & \frac{1}{h^2} [\alpha_{i+1}(X_{i+1}^n - X_i^n) - \alpha_i(X_i^n - X_{i-1}^n)] \\
 & - \frac{1}{h^4} [\beta_{i-1}(X_{i-2}^n - 2X_{i-1}^n + X_i^n) \\
 & - 2\beta_i(X_{i-1}^n - 2X_i^n + X_{i+1}^n) \\
 & + \beta_{i+1}(X_i^n - 2X_{i+1}^n + X_{i+2}^n)] \\
 & + F_{ext}(X_i^{n-1}),
 \end{aligned} \tag{3.37}$$

其中 γ 为阻尼系数, $\alpha_i = \alpha(ih), \beta_i = \beta(ih)$, h 为图像空间步长, Δt 为时间步长。一般来说外部力 F_{ext} 被存储为一个离散的向量场, 对图像的每个像素点该向量场都有一个外部力与之对应。当 \mathbf{X}_i 落在像素之间时, 可以用 \mathbf{X}_i 周围像素的四个点进行双线性差值获得外部力。由于本形变模型不是闭合曲线, 因此曲线 \mathbf{X}_i 上的前几个点与最后几个点不参与迭代, 并且, 第一个点和最后一个点的横坐标始终不变, 分别对应了图像横坐标的起始与终止位置。

将式 3.37 用矩阵的方式表示为:

$$\frac{\mathbf{X}^n - \mathbf{X}^{n-1}}{\tau} = \mathbf{A}\mathbf{X}^n + \tau F_{ext}(\mathbf{X}^{n-1}), \tag{3.38}$$

其中 $\tau = \Delta t / \gamma$, $\mathbf{X}^n, \mathbf{X}^{n-1}$ 和 $F_{ext}(\mathbf{X}^{n-1})$ 为 $m \times 2$ 矩阵, m 为曲线采样点的个数, \mathbf{A} 为 5 元素的 $m \times m$ 对角阵。曲线形变的迭代过程可通过解方程 3.38 来实现, 获得的解如下:

$$\mathbf{X}^n = (\mathbf{I} - \tau \mathbf{A})^{-1} [\mathbf{X}^{n-1} + \tau F_{ext}(\mathbf{X}^{n-1})]. \tag{3.39}$$

其中矩阵 $\mathbf{I} - \tau \mathbf{A}$ 的逆可以用 LU 分解计算以加快迭代速度。

若 α_i, β_i 均为常数, 即: $\alpha_i = \alpha, \beta_i = \beta$, 图像空间步长 $h=1$, 同时由于前几个点与最后几个点不参与迭代, 那么 \mathbf{A} 的形式可用如下矩阵表示:

$$\mathbf{A} = \begin{pmatrix} c & b & a & & & \\ b & c & b & a & & \\ a & b & c & b & a & \\ & a & b & c & b & a \\ & & \ddots & \ddots & \ddots & \ddots \\ & & & a & b & c & b & a \\ & & & & a & b & c & b \\ & & & & & a & b & c \end{pmatrix}, \tag{3.40}$$

其中 $a = \beta, b = -(4\beta + \alpha), c = 6\beta + 2\alpha$ 。

3.4.6 实验结果

为了验证本文算法的有效性，首先在计算机上对算法的结果进行验证，所用的测试集为小车摄像头采集的图像。

图 3.8 为对室外场景图像进行曲线拟合的过程。其中，(a) 为原图，(b) 为进行图像融合的结果，(c) 图为对融合图用 Canny 求边缘的结果。从结果可以看出，因为有平滑滤波和图像融合，在去除了近处的细小纹理的同时，保留了整个场景中大的边缘，使得 Canny 求得的边缘比较理想，为下一步用主动轮廓模型进行拟合提供了比较好的边缘基础。

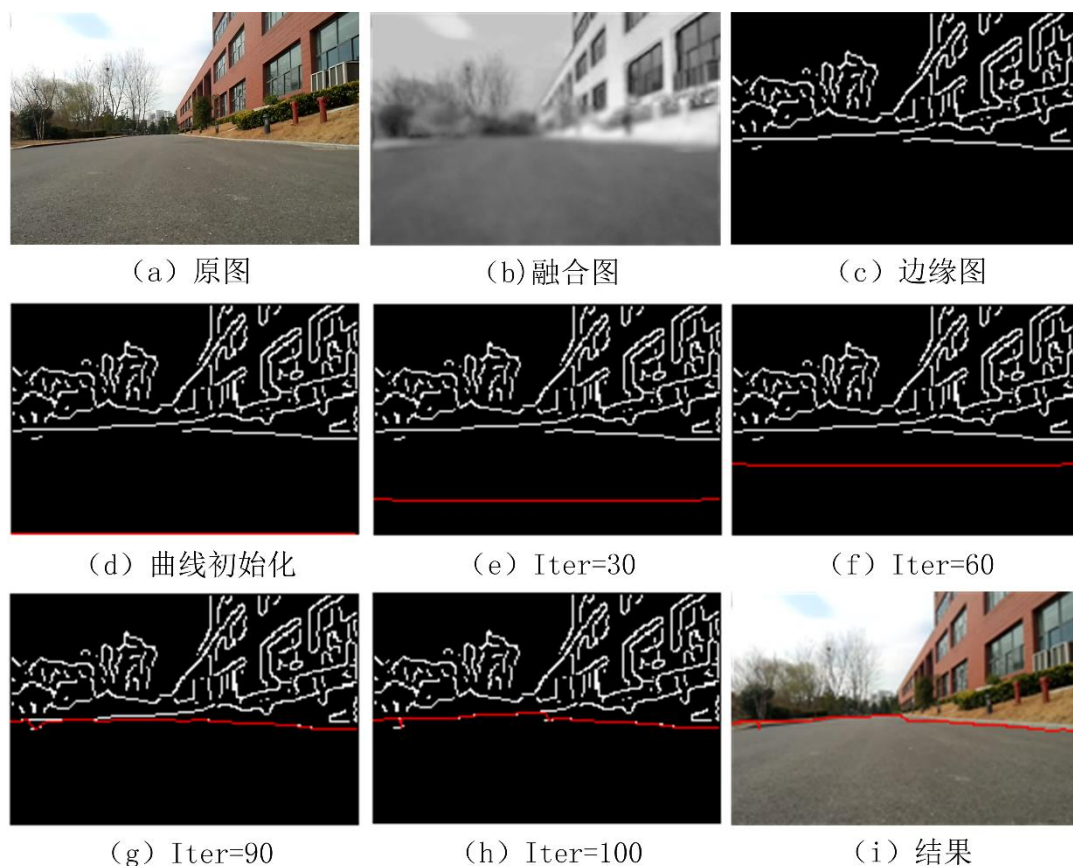


图 3.8 曲线迭代过程

图 3.8 (d) 到图 (h) 为曲线的迭代过程。曲线的初始化状态是一条处在图像底部的直线。开始迭代时，在向上的气球力的作用下向上移动，当接近边缘时，在气球力与 GVF 的共同作用下，贴近地面边缘，拟合出一条平滑曲线，图 3.8(i) 为最终获得的结果。图中红色曲线即为地面与垂直于地面物体的分界线，分界线以下的部分为地面，以上部分为垂直于地面的物体，其深度可根据同地面的接触点来计算获得，深度计算相关内容将在下一节介绍。

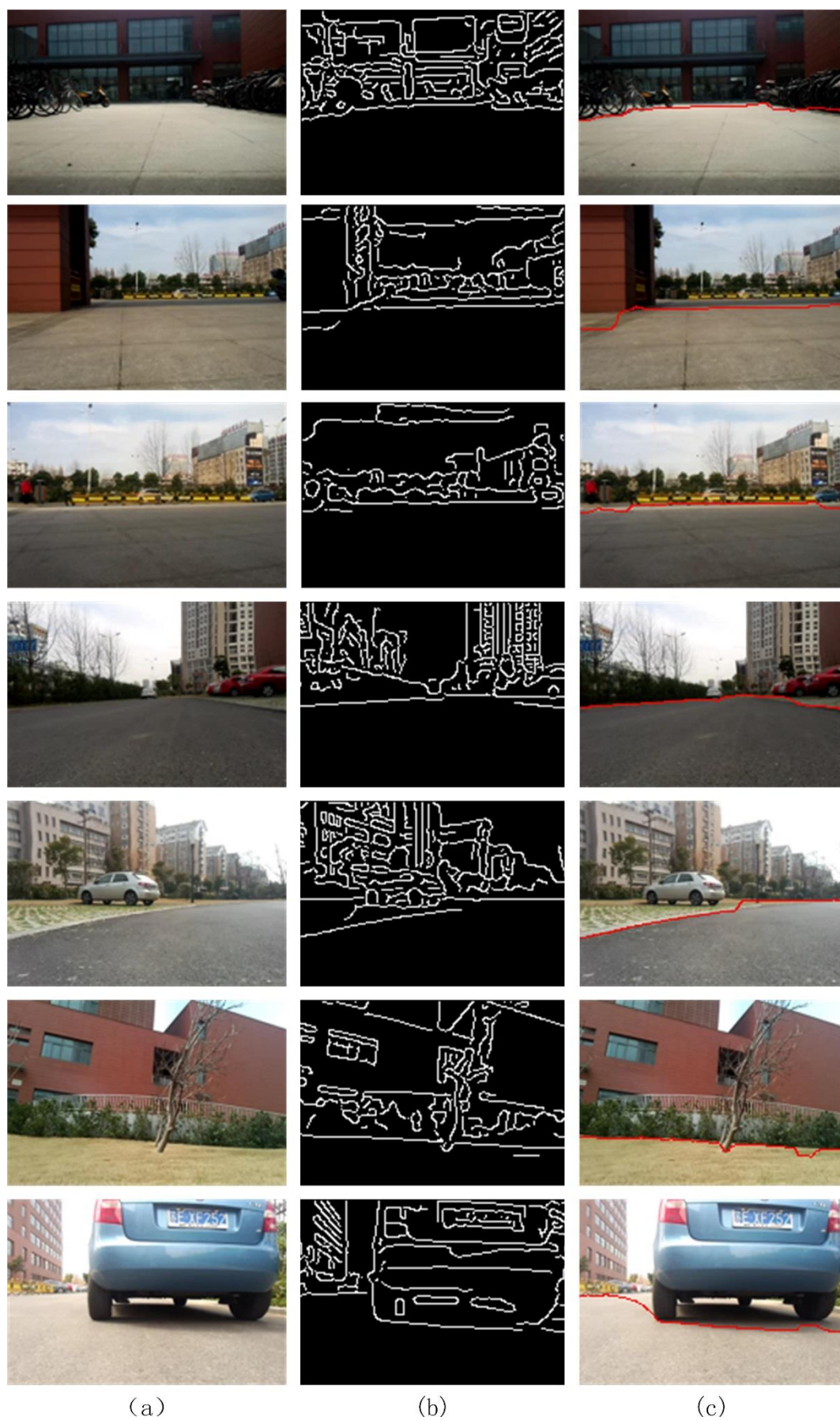


图 3.9 不同场景下地面边界拟合结果

在利用主动轮廓模型对曲线进行迭代拟合时，可以看出，该形变曲线能够在远离边缘的区域向边缘靠近。在向上形变过程中，曲线首尾横坐标不变。在靠近边缘时，能够有效地贴合凹形部分区域，并且能跨过一些小的边缘不连续点，形成一条完整的平滑曲线。

为了测试算法的鲁棒性，本文对各种场景下采集的图像进行了大量的测试，部分结果如图 3.9 所示。图中 (a) 列为原图，(b) 列为求得的边缘图，(c) 列为拟合得到的地面边缘。

从结果中可以看出，本文提出的算法鲁棒性较好，无论是在大理石地面、泊油路面，或者是在普通的草地上，该算法都能比较好地找到地面边界，阴影、细小的纹理对结果影响不大。同时，算法对图像中的障碍物也能很好地识别，如第一幅图中的自行车，最后一幅图中的小汽车等。

为了进一步验证本文算法的准确性，将本文获得的结果与 Cherian (2009) 和 Hoiem (2005) 的结果进行比较，使用的是 Make3D 数据集 (Saxena, 2014)。图 3.10 为本文算法与两种算法进行比较的结果。

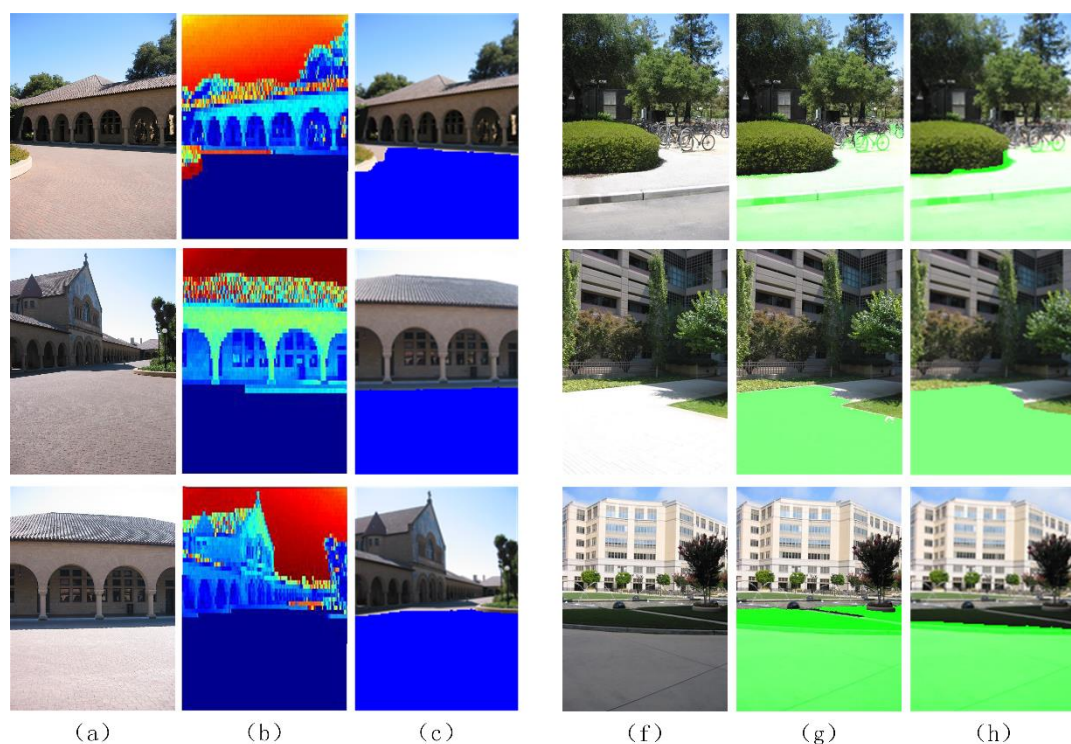


图 3.10 地平面估计结果比较

(a) (f) 为原图，(b) 为 Cherian 算法获得的结果（由于作者没有公布源代码，此处用以比较的结果图为其发表论文上的展示结果），图中底部蓝色部分为他们获得的地平面。(c) 为本文算法结果，为便于比较，将地面部分涂成蓝色。可以看出，Cherian 等的算法和本文的算法在该场景下都能比较好地获得地

平面，相比之下本文算法获得的结果分辨率更高，在边缘的平滑度上有优势。

(g) 为 Hoiem 等的结果，(h) 为本文算法的结果。从图中可以看出，Hoiem 等的方法和本文算法的结果有一定相似度，如第一行和第二行图像所示。但本文的鲁棒性更高，能够较好地分离地面和非地面，比较少受到其他因素的干扰。

为了验证算法的高效性，本文还与 Hoiem 等的结果计算效率进行了比较，实验平台为 AMD Athlon II X4 640 CPU，6GB RAM，操作系统为 Windows 7，编程语言为 MATLAB。表 3.1 为两种算法对 30 幅分辨率为 341×455 的图像进行处理的平均运行时间，从表中可以看出，本文的运算效率远高于 Hoiem 等的 Photo pop-up 算法。

表 3.1 与 Photo pop-up 算法效率比较

算法	所用时间 /s
Photo pop-up	35.45
本文算法	1.03

当然，在实验过程中，也发现本文算法在特定的条件下也有一定的局限性。比较典型的是当地面的纹理信息非常丰富，并且地面组成各块颜色不同时，求边缘的过程中不可避免地会带入大量的干扰边缘，影响地面的拟合。如下图所示，由于停车场有着丰富的颜色与纹理信息，会导致大量的错误边缘出现，最终导致获得的地面边界并不理想。



图 3.11 本文算法的局限性

3.5 绝对深度计算

3.5.1 深度计算公式

在确定了图像中地平面的区域后，可以利用摄像头几何成像原理（zhang, 2000）获得地平面边界的绝对深度信息（Cherian, 2009）。

小车上手机摄像头成像原理如图 3.12 所示。图中，P 为光学中心， h 为光

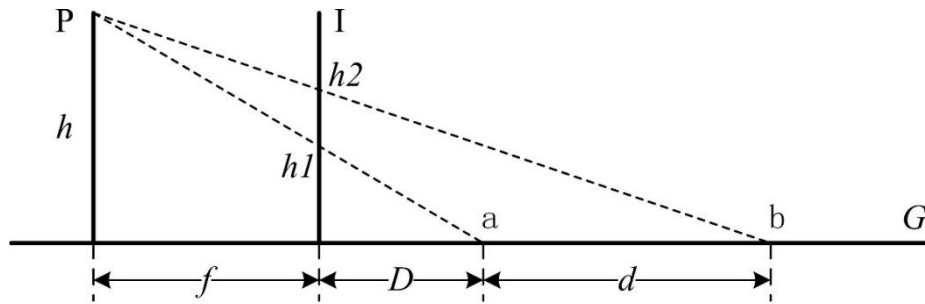


图 3.12 摄像头成像原理图

学中心到地面的高度，I 为图像平面，G 为地平面， f 为摄像头的焦距， D 为图像最下面一行对应的地面点 a 距离图像平面的真实距离， $h1$ 和 $h2$ 为点 a 和点 b 在图像平面的成像点距离地面的高度。可以看出，地面上任意一点 b 相对于点 a 的距离 d ，可以通过 $h1$ 和 $h2$ 之间的高度差来计算。利用相似三角形关系容易推导得到下面公式：

$$d = f \frac{h(h2-h1)}{(h-h1)(h-h2)} = D \frac{h(h2-h1)}{h1(h-h2)}, \quad (3.41)$$

上式中， D 和 h 可以通过实际测量获得， $h1$ 为定值。设图像传感器的 CMOS 的高度为 H_s ，那么 $h1 = h - H_s / 2$ 。 $h2$ 可以用图像的纵坐标 y 来表示：

$$h2 = h1 + yH_{pixel}. \quad (3.42)$$

此处的图像坐标系原点在图像的左下角， $y=0$ 时为图像最下面一行。 H_l 为图像高度， $H_{pixel} = H_s / H_l$ 为每个像素点所占用 CMOS 的高度。将 3.42 代入 3.41 可获得 y 与距离 d 的关系：

$$d = D \frac{hH_{pixel}y}{(h-H_s/2)(H_s/2-H_{pixel}y)}. \quad (3.43)$$

3.5.3 实验结果

为了能够验证本文深度计算公式的准确度，采用如图 3.13 所示实验方式，图中的棋盘格为 8×8 ，每小格的大小为 $14.5\text{cm} \times 14.5\text{cm}$ 。

首先合理放置小车的位置，使得小车摄像头的成像的最下面一行与棋盘格的最下面一排格子的边缘对齐。图 3.13 (a) 即为小车放置的位置，(b) 为摄像头视角获得的图像。然后，测量摄像头平面距离棋盘格边缘的距离 D ，测量摄像头中心点到地面的距离 h 。最后，检测棋盘格的角点在图像上的纵坐标，根据式 3.43 计算距离 d 。由于棋盘格角点的实际距离已知，可以作为参照的真实

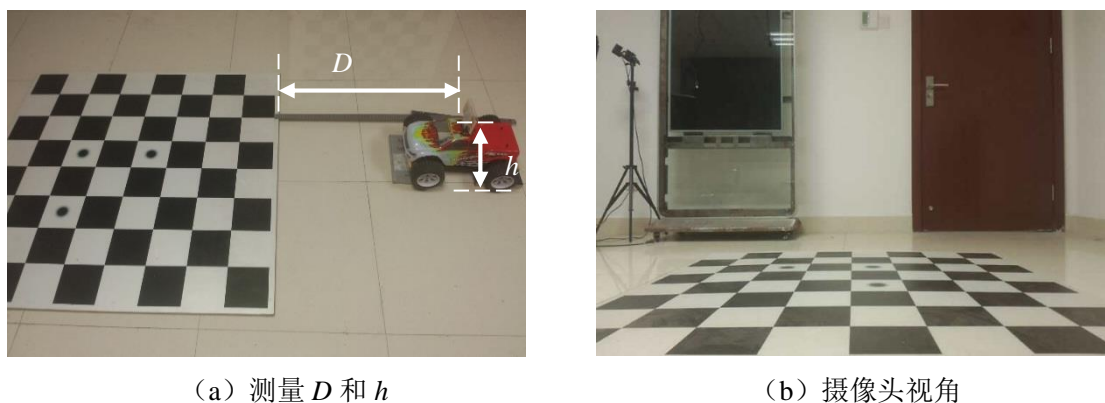


图 3.13 深度计算实验过程

深度。该实验方法能够一定程度上避免手工测量带来的误差。

通过测量，本文小车摄像头距地面高 $h=25\text{cm}$ ， $D=72\text{cm}$ 。同时，由前文可知 $H_s=0.342\text{cm}$ ， $H_l=240$ 。计算距离 d ，并与真实深度进行比较，获得的结果如表 3.2 所示。

表 3.2 绝对深度实验结果比较

	图像纵坐标 y	计算值 d (cm)	真实值 d (cm)	误差 (cm)
1	19	14.54	14.50	0.04
2	33	29.32	29.00	0.32
3	43	43.16	43.50	0.34
4	51	57.12	58.00	0.88
5	63	85.42	87.00	1.58
6	72	115.93	116.00	0.07
				AVG = 0.54

从上表可以看出，用该方法获得的深度值与真实值非常接近，平均误差在 1cm 以下。并且，从结果分析，第 5 组数据表现出来的大误差，很有可能是在检测棋盘格角点所对应成像点过程中带来的误差，在小车避障过程中，这种检测错误是可以避免的。尽管如此，其最大误差依然不超过 2cm 。

同时，本文也对算法估计出来的地面边界计算相应的绝对深度，获得的结果如图 3.14 所示。第一行图像中红色线部分为求得的地面边界，第二行为对应地面边界计算得到的深度，其中横坐标为图像的横坐标，纵坐标为地平面的边界到小车的距离，单位为 cm 。

从结果可以看出，通过该方法计算得到的地平面边界距离与实际场景非常接近。图 3.14 (a) 中，图像的中心部分为无穷远点，地面边界的细微变动也会导致距离的剧变。为了防止小车因地形的变化带来的晃动而导致计算出来的地

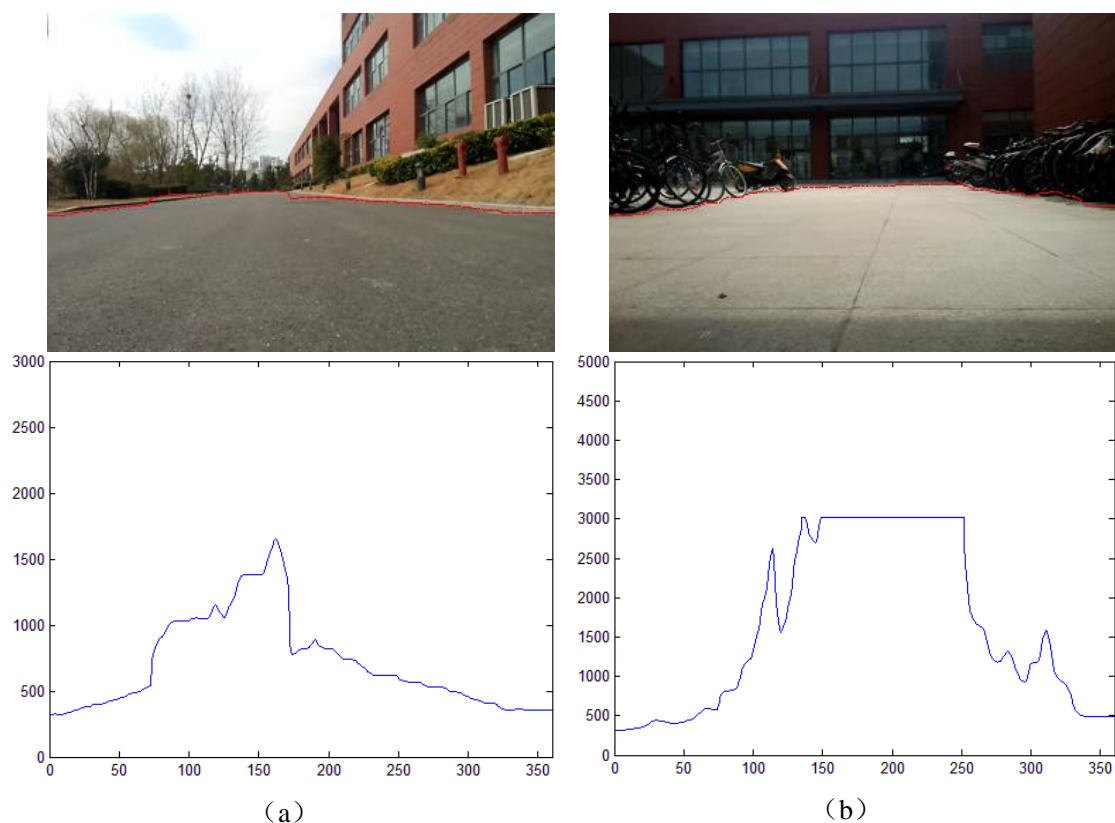


图 3.14 地平面深度实验结果

面边界剧烈变动,在不影响小车导航的情况下对地面边界加入了最远距离限制,结果图 3.14 (b) 所示,当计算出来的距离超过限制时,地面边界为固定值。

3.6 本章小结

本章介绍了一种利用单幅图像求取 3 维地平面的算法。针对智能手机移动摄像头获得的视频具有局部模糊的特点,提出了一个标准差为纵坐标函数的高斯滤波器对图像进行平滑处理。利用了 HSI 色彩空间能够分离色彩与强度分量的特点进行图像融合,获得的结果降低了阴影的影响。在利用 Canny 求得边缘的基础上,利用改进的主动轮廓模型对地面边界进行拟合,最终获得了平滑的地平面边界。

从离线实验结果来看,本算法不仅有着比较好的鲁棒性,能适应各种场景;同时,该算法也具有较高的计算效率,有助于算法在智能手机上的快速实现。

第4章 机器人控制系统设计

4.1 引言

定位与导航是智能机器人学中的一个主要组成部分，机器人自主导航的过程也就是回答 MIT 教授 Leonard 提出的三个问题：“我现在在哪里？”，“我要到哪里去？”，“如何到那里去？”的过程。避障规划则是机器人定位与导航中非常重要的一个问题之一。机器人要根据传感器等其他信息获得障碍物所在的位置，按照一定的方法进行有效地避障，最后达到目标点。

本文第二章介绍了该系统的硬件平台，第三章介绍了图像与机器视觉算法，获得了 3 维地面信息。本章将在此基础上进行控制系统的设计，内容主要包括转向角控制与速度控制两部分。

4.2 转向角控制

为了能准确控制小车转向的角度，需要用到惯性传感器来测量小车的姿态。在实际测量过程中，因为传感器自身的构造特点、存在测量噪声以及测量误差等原因，不可避免地会导致测量结果的偏差。而只有在获取有效信息的基础上，机器人才能做出正确的决策。因此需要对惯性传感器数据进行处理，将相关的传感器信息融合，为控制器提供一个相对稳定的测量结果。

4.2.1 惯性传感器特点

本文与转向角控制相关的惯性传感器包括两种：陀螺仪和加速度计，它们均采用 MEMS 技术，但特点各不相同。

（1）陀螺仪。

陀螺仪能测量三轴的角速度，响应速度很快，准确度高，但是却存在零点漂移。当物体静止时，陀螺仪的输出数据存在着偏差，不一定为零。如果用陀螺仪来计算小车的姿态角，则需要对陀螺仪三个轴的数据相对时间进行积分。此处的积分通过角速度乘以两次测量时间差来得到。但因为陀螺仪存在零点漂移，误差经过积分随着时间不断累加，最终导致数据不可用。

（2）加速度计。

Android 智能手机上的加速度计与电子罗盘集成在一起，利用加速度计提供

的重力矢量和利用电子罗盘提供的方向能计算出手机的姿态。但是这两种传感器的输出并不精确，尤其是电子罗盘，极易受到外界的干扰，造成数据的波动。

从上面的特点可以看出：陀螺仪不容易受到干扰，短时间内的测量数据比较可靠。与之相对应的是，加速度计则能够提供相对准确的长时间的可靠数据，这是因为外界环境的干扰是一个极短的过程，比如小车在行驶过程中的晃动，这些属于高频噪声。因此，结合陀螺仪和加速度计来计算小车的姿态角就非常有必要。

4.2.1 平衡滤波器

针对这些特点，Colton（2007）提出了一种互补型的滤波器，该滤波器如下图所示：

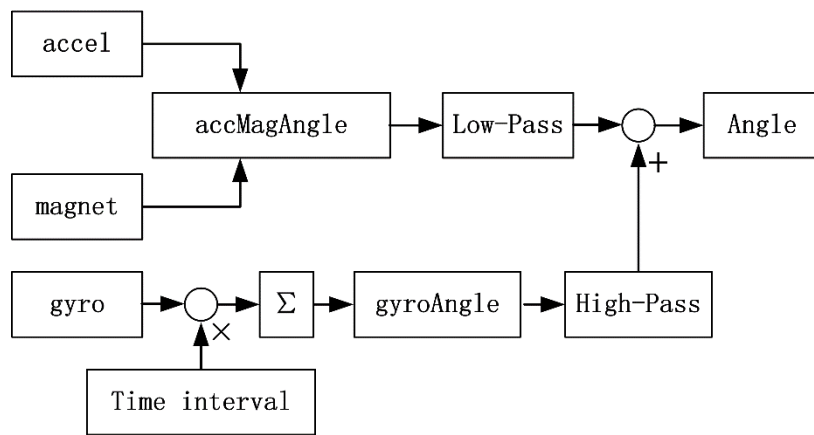


图 4.1 惯性传感器融合示意图

为了抑制加速度计的短时高频噪声，对加速度计和电子罗盘的输出结果加入了一个低通滤波器，该滤波器能够消除高频噪声，保留持续时间较长的数据。同时，为了抑制陀螺仪的零点漂移，在陀螺仪的输出结果加入了一个高通滤波器，该滤波器能够去除属于低频的零点偏差，保留陀螺仪的短时响应结果。

对上图可用数学公式表示为：

$$A_{fused}^t = (1 - \alpha)(A_{fused}^{t-1} + V_{gyro}^t \Delta t) + \alpha A_{accMag}^t \quad (t = 1, 2, 3 \dots) \quad (4.1)$$

上式为低通滤波器和高通滤波器的和，其中等式右边的第一项为高通部分，第二项为低通部分。 t 为采样时刻点， A_{fused}^t 为 t 时刻滤波器的输出， V_{gyro}^t 为 t 时刻陀螺仪测得的角速度， Δt 为采样周期，角速度与采样周期的乘积即为在该采样周期内角度变化量， A_{accMag}^t 为 t 采样时刻通过加速度计和电子罗盘的测量值计算得到的角度。 α 为平滑系数，可通过式 4.2 计算得到：

$$\alpha = \frac{\Delta t}{\tau + \Delta t}, \quad (4.2)$$

其中 Δt 为采样周期, τ 为时间常数, 即该低通滤波器截止频率为 $1/\tau$ 。在手机进行实现时, 采样频率为 33Hz , 即 $\Delta t = 0.03\text{s}$, 设时间常数 $\tau = 0.75\text{s}$, 即可获得 $\alpha = 0.03/(0.75 + 0.03) = 0.038$ 。

当输入信号持续时间比 τ 长时, 加速度计和电子罗盘获得的角度与陀螺仪相比有更大的权重。与之相对的是, 当输入信号持续时间比 τ 短时, 陀螺仪测得的角度对结果有更重要的影响。

假设设备旋转 90° 后短时间内回到原来的位置, 那么平衡滤波器的工作过程如图 4.2 所示 (Lawitzki, 2014)。

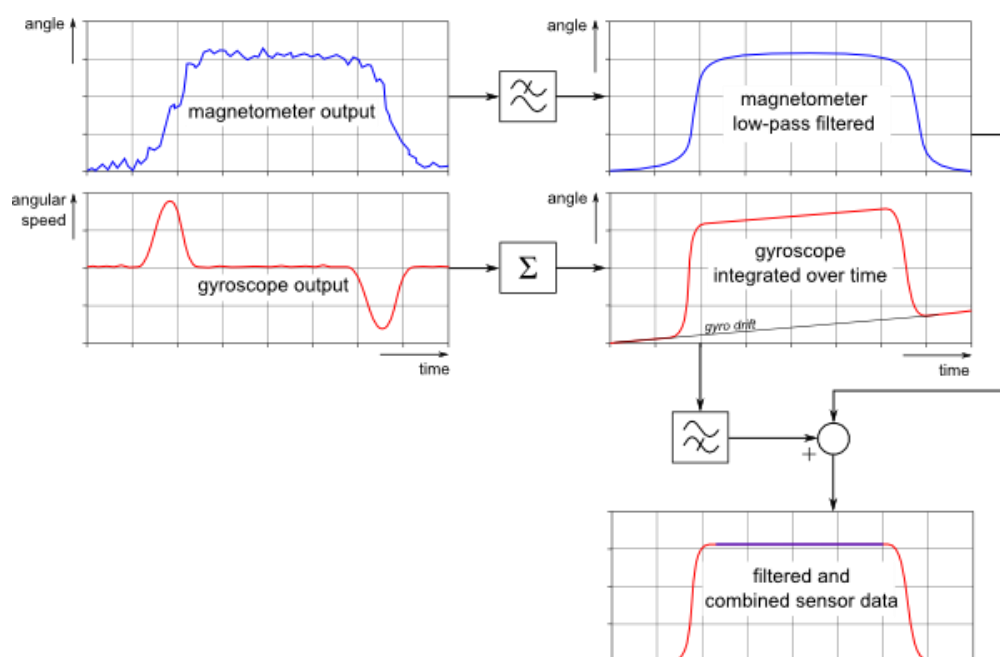


图 4.2 平衡滤波器工作过程

从图中对陀螺仪数据进行积分的结果可以看出, 零点漂移对陀螺仪测量结果的影响非常大, 即使是很细微的原点偏差经过对时间的积分后也会导致比较大的角度误差。平衡滤波器中的低通部分能够去掉电子罗盘数据中的高频噪声, 而高通部分则能消除陀螺仪中零点漂移的影响。

4.2.2 实验结果

首先将手机平放, 接着把手机翻转过来背面朝上, 最后又把手机翻转到最开始的姿态, 同时记录这一过程的所有数据, 其中 z 轴方向上的实验结果如图 4.3 所示。图 4.3 (a) 为加速度计和电子罗盘测得的值, (b) 为陀螺仪测得的结果, (c) 为平衡滤波后的结果。为了方便比较, 将 (a) (b) (c) 三个结果放在

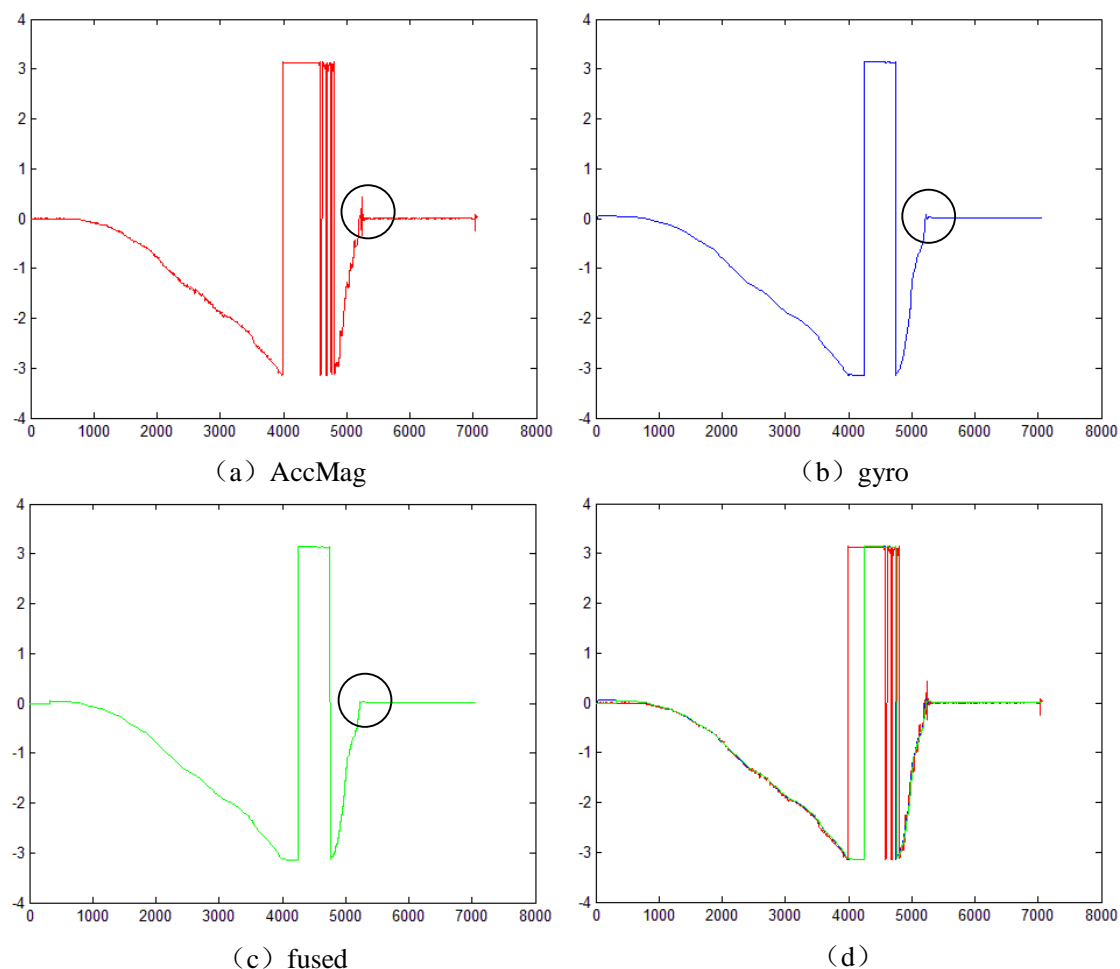


图 4.3 平衡滤波器实验结果

(d)图上显示。从图上可以看出，加速度计获得的结果虽然大致的趋势是对的，但是存在着很多的高频噪声。滤波的结果与陀螺仪获得的角度大致相同，但比陀螺仪要好。如图中黑色圆圈内所示，滤波的结果更为平滑。

当获得了小车姿态角之后，将期望小车转向角度与当前小车的角度的差输入舵机即可控制小车转到相应角度。

4.3 速度控制

控制系统是为了达到预期的目标而设计的，其目的是建立感知系统与执行机构之间的联系，它由相互关联的部件组成，能提供预期的系统响应（Dorf, 2002）。对于小车而言，给定目标速度与转向角度，需要设计一个合适控制算法来控制小车稳定、准确、快速达到这一目标。

根据控制算法的不同，目前机器人的控制可分为 PID 控制、自适应控制、模糊控制、鲁棒控制、神经网络控制、变结构控制等方法。在这众多控制器中，PID 型控制器因为结构简单、鲁棒性高、应用方便等优点，在控制界占据着绝

对的统治地位。即使在各种先进的控制规律广泛使用的今天，基于过程的控制中 95% 以上的控制器都是 PI 或 PID 型 (Ramon, 2012)。

本文设计了一个速度控制的 PID 控制器，并针对该型小车的特点进行参数整定，最后基于参数整定的结果给出一些相应的控制实验结果。

4.3.1 控制器设计

闭环控制系统对实际输出的量进行测量，并将实际输出与预期输出进行比较，将获得的偏差作为控制的依据，并用于实际的控制过程，以使偏差不断地减小。小车的速度闭环反馈控制回路的框图如下图所示：

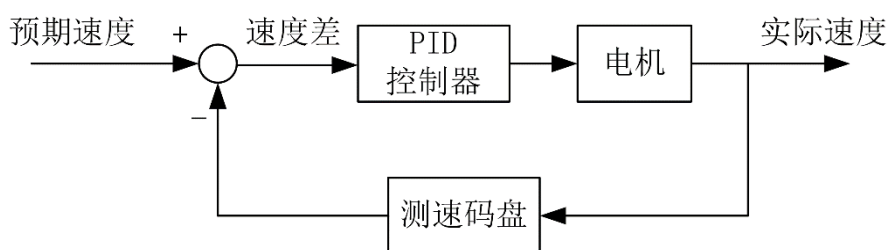


图 4.4 速度闭环反馈控制系统

其中的 PID 控制器可用如下方程来表示：

$$u(t) = K_p e(t) + K_I \int e(t) dt + K_D \frac{de(t)}{dt}, \quad (4.3)$$

$u(t)$ 为控制器输出的控制量， $e(t)$ 为偏差，是由预期的输出响应与实际输出响应的测量值的差，上式右边是比例项、积分项与微分项之和， K_p, K_I, K_D 为相应项的系数，也称为控制增益。

当 K_I, K_D 等于 0 时，即为单纯的比例控制。比例控制下，一旦产生偏差，控制器立即产生控制作用，控制量与偏差成正比，偏差越大控制量也越大。比例控制有着很快的响应速度，稳定性好，但会存在稳态误差。增大 K_p 能降低稳态误差，但系统的震动倾向也会增加。因此，需要在单纯的比例积分之后加入一定的积分项，使得控制量的增加与时间与偏差量成正比。只要偏差一直存在，积分项就不会消失，其基本作用就在于消除系统的稳态误差。当误差为 0 时，比例项的输出也为 0，此时积分项带来的控制量将接替比例控制。

为了使得小车有着更好的响应特性，减少调节时间与超调量，能够更快速地启动与停止，在控制器中加入一定的微分项，其控制量的大小与偏差的变化速率成正比。微分控制能够预测误差变化的趋势，当误差突变时能够及时控制，并能在偏差信号变得太大之前，在系统中引入早期修正信号，从而加快系统的

响应速度。

在数字控制系统中进行的是一种采样控制，只能通过采样时刻的偏差值计算控制量，需要对式 4.3 中的积分控制和微分控制进行离散化处理。设 t 为采样时刻点，离散 PID 控制器的表达式如下：

$$u_t = K_p e_t + K_I \sum_0^t e_t \Delta t + K_D \frac{(e_t - e_{t-1})}{\Delta t} \quad (t = 1, 2, 3 \dots), \quad (4.4)$$

e_t 即为 t 采样时刻的偏差， Δt 为采样周期。该控制器控制量的输出需要对时刻 t 之前所有的偏差进行累加，计算机输出直接对应对象的输出，故称为位置式控制 PID。这种控制方式的缺点在于一旦微处理器出现故障，会出现较大的误动作，对系统影响较大，因而产生了增量式 PID 控制的控制算法，该控制器的输出只是控制量的增量 Δu_t ，由式 4.4 可推导出增量式 PID 控制算法。由递推原理可知：

$$u_{t-1} = K_p e_{t-1} + K_I \sum_0^{t-1} e_{t-1} \Delta t + K_D \frac{(e_{t-1} - e_{t-2})}{\Delta t}. \quad (4.5)$$

用式 4.4 减去式 4.5 可得增量式 PID 控制算法的表达式：

$$\begin{aligned} \Delta u_t &= K_p (e_t - e_{t-1}) + K_I e_t + K_D (e_t - 2e_{t-1} + e_{t-2}) \\ &= (K_p + K_I + K_D) e_t - (K_p + 2K_D) e_{t-1} + K_D e_{t-2}. \end{aligned} \quad (4.6)$$

由于一般计算机采用的是恒定的采样周期，一旦确定了 K_p, K_I, K_D ，只要使用前后三次测量值的偏差，即可由式 4.6 求出控制增量。对应于小车的速度的实际控制量，可通过对增量的累积来实现： $u_t = u_{t-1} + \Delta u_t$ 。

虽然位置式与增量式控制算法并无本质区别，但是还是能带来一些优点，比如误动作影响较小；控制器算式中不需要累加；控制增量易于仅与最近三次的采样值有关，比较容易对采样值进行滤波处理等。

4.3.2 参数整定

对 PID 控制器的设计和应用，核心问题之一是参数的整定，它是根据系统被控过程的特性确定调整 PID 控制器的三个参数，使得系统的过渡过程达到表满意的质量指标。

PID 控制器参数整定的方法很多，概括起来有两大类：

(1) 理论计算整定法。

它主要是依据系统的数学模型，经过理论计算确定控制器参数。这种方法的优点在于能够进行仿真实验，能获得优秀的控制性能。但由于模型不可避免

存在误差，所得到的计算数据还需要通过工程实际进行调整和修改，并且很多情况下建模比较困难，理论计算整定法使用范围比较窄。

(2) 工程整定方法。

它主要依赖工程经验，直接在控制系统的试验中进行参数调整。虽然最终获得的控制器参数不一定是最优解，但该方法简单、易于掌握，在模型未知的情况下也能获得一个比较满意的控制效果，在工程实际中被广泛采用。

本文所用小车的电机参数未知，较难获得准确的系统模型。因此，采用工程整定方法来对小车的 PID 控制器进行参数整定。本文 Ziegler et al (1942) 提出 PID 控制器参数经验公式法起，已经有很多方法被用于 PID 控制器的参数整定 (Åström et al, 1992; 增振平等, 2004; 张立群等, 2005; 张兴华等 2007)。

本文使用的即为 Ziegler 和 Nichols 提出的临界比例度法，该法不依赖与对象的数学模型，而是总结前人的理论和实践经验，通过实验来获得控制器的最佳参数。

表 4.1 ZN 临界比例度法参数整定公式

PID 控制器的参数	K_p	K_I	K_D
P 型控制器	$0.5 K_u$		
PI 型控制器	$0.45 K_u$	$0.83 T_u$	
PID 型控制器	$0.6 K_u$	$0.5 T_u$	$0.125 T_u$

该方法首先将 PID 控制器的积分和微分作用去掉，仅保留了比例控制部分。在系统中加入扰动，若系统响应的震荡幅度不断增大，则减少控制增益；若震荡幅度衰减，则加大控制增益，直到系统做临界等幅周期震荡。记录此时的临界增益 K_u 和临界震荡周期 T_u ，最后利用表 4.1 中的经验公式来求出 P、PI 和 PID 这三种控制器的参数整定值：

4.3.3 实验结果

小车速度的采样间隔为 0.05s，因此，小车的控制周期也设定为 0.05s。将 K_I, K_D 置零，设置不同的 K_p ，对小车输入幅度为 1 的阶跃信号，并记录小车的响应速度。小车在不同 K_p 下的响应如图 (a) (b) (c) 所示。

图中，横坐标为控制周期，纵坐标为小车的响应速度。从图 4.3 (a) 可以看出，小车存在多个震荡周期，在每个震荡周期，系统响应的震荡幅值有减少的趋势。图 4.3 (c) 瞬态响应的振幅在前期有增大的趋势，在后期因为小车不能响应更大的震荡，反而稳定下来，但我们可以认为在该 K_p 值下，系统响应振

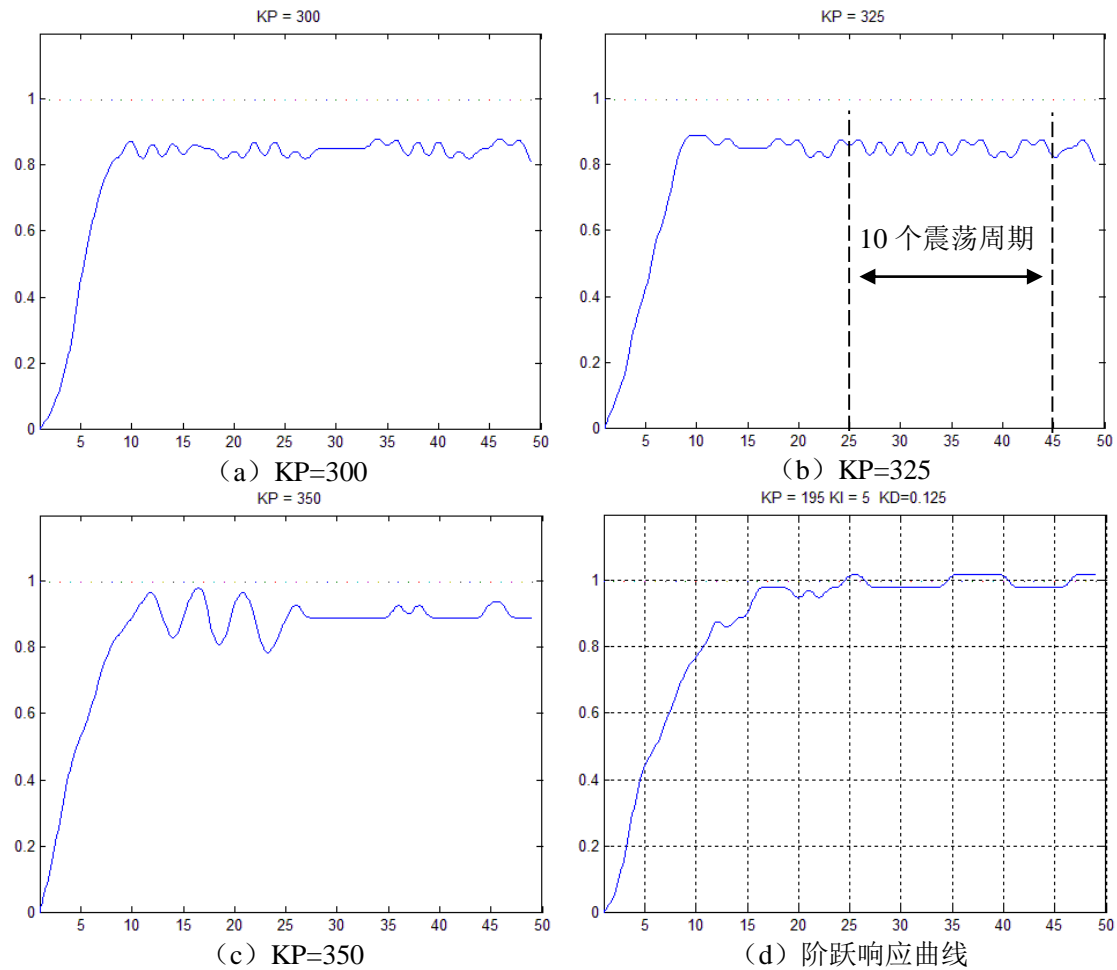


图 4.5 控制器参数整定

幅不断增大。而图 4.3 (b) 的瞬态响应做等幅震荡, 并且由图中可以直接计算得到其等幅震荡周期为 10。由此可以得出 Z-N 法的临界增益 $K_u = 325$ 和临界震荡周期 $T_u = 10$ 。

利用表 4.1 可获得速度 PID 闭环控制的参数为 $K_p = 195, K_i = 5, K_d = 1.25$, 相应的 PID 控制器的阶跃响应如图 4.3 (d) 所示。

因为小车在运行过程中不可避免会有震动, 因此小车响应曲线在上升过程中少许波动是正常现象。同图中可以看出, 小车的上升时间为 0.8s, 超调量为 0, 满足控制系统中快速、准确、稳定的要求。

4.4 避障策略

本文所面对的问题是小区的避障, 暂时并不涉及到小区的导航。小区能获得的信息是地平面的边界, 地面边界之外的物体均为障碍物。本文的目的是机器人能够利用摄像头获得的地面边界信息进行顺利地避障, 因此在这一避障过

程中，机器人不必清楚地知道自身所处的位置，只需对障碍物进行定位，然后选择能绕过障碍物的合理路径，即可完成任务。

小车的避障策略可以简单解释为：在小车的运动过程中，通过图像与机器视觉算法不断地更新地面边界，小车将在地面边界上搜寻距离小车最远的边界，并朝该方向进行运动。这一避障策略将使小车远离障碍物，达到避障目的。

4.5 本章小结

本章从实际问题出发，对机器人的控制系统设计进行了相关的描述。针对陀螺仪存在零点漂移和加速度计的高频噪声问题，使用了平衡滤波器滤除相关噪声，获得了一个准确的小车姿态角。为了精确控制小车的速度和角度，本章设计了一个 PID 控制器，并进行了参数整定，最终形成了速度、角度闭环反馈控制。最后，在第二、三章的基础上，给出了小车的避障策略。

从最终的实验结果来看，小车的控制系统能够准确和快速地控制小车实现指定的控制目标，为最终的综合实验提供了良好的基础。

第 5 章 综合实验及结果

5.1 引言

本文的第二章介绍了避障系统的硬件平台，能够为整个系统提供硬件基础。第三章阐述了图像处理与机器视觉算法，该算法为本系统提供了地平面的边界。第四章对机器人的控制系统进行了相关的介绍，能够保证避障指令很好地执行。但这些都是相互分离的部件，只有相互关联在一起才能发挥系统的作用，实现避障这一目标。

本章将把前述几章的部件有机地结合在一起，并将在实际场景中进行相关实验，以验证本算法的可靠性与稳定性。

5.2 综合实验

5.3 结果分析

5.4 总结与展望

本文设计了一个基于单目视觉的机器人避障系统。

参考文献

- 常健, 吴成东, 李斌. 2010. 移动机器人避障方法综述[J]. 仪器仪表学报, 2010.
- 李天庆, 张毅, 刘志, 等. 2005. Snake 模型综述[J]. 计算机工程, 31(9):1-3.
- 李磊, 叶涛. 2002. 移动机器人技术研究现状与未来[J]. 机器人, 24(5): 475-480.
- 苏家铎. 1993. 泛函分析与变分法[M]. 合肥:中国科学技术大学出版社.
- 谭民, 徐德, 侯增广, 等. 2007. 先进机器人控制[M]. 北京:高等教育出版社.
- 蔡自兴, 周翔. 2000. 基于功能/行为集成的自主式移动机器人进化控制体系结构[J]. 机器人, 22.3 : 169-175.
- 冈萨雷斯, 等著. 阮秋琦, 等译. 2007. 数字图像处理(第二版)[M]. 北京:电子工业出版社.
- 塞勒 著, 邹韬 译. 2013. 移动浪潮[M]. 北京:中信出版社.
- 向文辉, 刘羽, 曹洋, 等. 2014. 基于车载单目图像的三维地平面估计[J]. 机器人, 36(1):76-82.
- 增振平, 陈增强, 袁著祉. 2004. 基于新的误差积分准则的 PID 控制器优化[J]. 控制工程, 11(1):52-54.
- 张立群, 李东海, 薛亚丽, 等. 2005. H_{∞} 混合灵敏度约束的 PID 控制器设计[J]. 清华大学学报, 45(5):700-703.
- 章毓晋. 2005. 图像工程(中册): 图像分析(第二版)[M]. 北京:清华大学出版社.
- 张兴华, 朱筱蓉, 林锦国. 基于量子遗传算法的 PID 控制器参数自整定[J]. 计算机工程应用, 43(21):218-220.
- Alvarez L, Guichard F, Lions P L, et al. 1993. Axioms and fundamental equations of image processing[J]. Archive for rational mechanics and analysis, 123(3): 199-257.
- Amini A A, Weymouth T E, Jain R C. 1990. Using dynamic programming for solving variational problems in vision[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(9): 855-867.
- Åström K J, Hang C C, Persson P, et al. 1992. Towards intelligent PID control. Automatica, 28(1): 1-9.
- Awadh B, Sepehri N, Hawaleshka O. 1995. A computer-aided process planning model based on genetic algorithms[J]. Computers & Operations Research, 22(8): 841-856.
- Canny J. 1986. A computational approach to edge detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8(6): 679-698.
- Caselles V, Catté F, Coll T, et al. 1993. A geometric model for active contours in image processing[J]. Numerische mathematik, 66(1): 1-31.

- Caselles V, Kimmel R, Sapiro G. 1997. Geodesic active contours[J]. *International journal of computer vision*, 22(1): 61-79.
- Chan T F, Vese L A. 2001. Active contours without edges[J]. *IEEE transactions on Image processing*, 10(2): 266-277.
- Cherian A, Morellas V, Papanikolopoulos N. 2009. Accurate 3D ground plane estimation from a single image[C]//*IEEE International Conference on Robotics and Automation*. Kobe: IEEE, 2243-2249.
- Cohen L D. 1991. On active contour models and balloons[J]. *CVGIP: Image understanding*, 53(2): 211-218.
- Cohen I, Cohen L D, Ayache N. 1992. Using deformable surfaces to segment 3-D images and infer differential structures[J]. *CVGIP: Image understanding*, 56(2): 242-263.
- Colton S. 2007. The balance filter[R]. Massachusetts Institute of Technology. Tech. Rep.
- Davison A J. 2003. Real-time simultaneous localisation and mapping with a single camera[C]//*IEEE International Conference on Computer Vision*. France: IEEE, 1403-1410.
- Davison A J, Reid I D, Molton N D, et al. 2007. MonoSLAM: Real-time single camera SLAM[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6): 1052-1067.
- Dorf Richard C, Robert H Bishop. 谢红卫, 等译. 2002. 现代控制系统（第八版）[M]. 北京: 科学出版社.
- Fedkiw S O R. 2003. Level set methods and dynamic implicit surfaces[M]. Springer Verlag.
- Felzenszwalb P F, Huttenlocher D P. 2004. Efficient graph-based image segmentation[J]. *International Journal of Computer Vision*, 59(2): 167-181.
- Gaussier P, Joulain C, Zrehen S, et al. 1997. Visual navigation in an open environment without map[C]// *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Grenoble: IEEE, 2: 545-550.
- Glasius R, Komoda A, Gielen S C A M. 1995. Neural network dynamics for path planning and obstacle avoidance[J]. *Neural Networks*, 8(1): 125-133.
- Guivant J, Nebot E, Baiker S. 2000. Autonomous navigation and map building using laser range sensors in outdoor applications[J]. *Journal of robotic systems*, 17(10): 565-583.
- Hagras H A. 2004. A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots[J]. *IEEE Transactions on Fuzzy Systems*, 12(4): 524-539.
- Hoiem D, Efros A A, Hebert M. 2005. Automatic photo pop-up[J]. *ACM Transactions on Graphics*, 24(3): 577-584.
- Hu Yangrong, Yang S X. 2004. A knowledge based genetic algorithm for path planning of a mobile robot[C]// *IEEE International Conference on Robotics and Automation*. USA: IEEE,

- 5: 4350-4355.
- Lawitzki P. 2014. Android sensor fusion tutorial[EB/OL]. <http://www.codeproject.com/Articles/729759/Android-Sensor-Fusion-Tutorial>.
- Leonard J J, Durrant-Whyte H F. 1992. Directed sonar sensing for mobile robot navigation[M]. Dordrecht: Kluwer Academic Publishers.
- Janet J A, Luo R C, Kay M G. 1995. The essential visibility graph: An approach to global motion planning for autonomous mobile robots[C]// IEEE International Conference on Robotics and Automation. Proceedings. Nagoya: IEEE, 2: 1958-1963.
- Joulain C, Gaussier P, Revel A, et al. 1997. Learning to build visual categories from perception-action associations[C]// IEEE/RSJ International Conference on Intelligent Robots and Systems. Grenoble: IEEE, 2: 857-864.
- Kass M, Witkin A, Terzopoulos D. 1988. Snakes: Active contour models[J]. International journal of computer vision, 1(4): 321-331.
- Khatib O. 1986. Real-time obstacle avoidance for manipulators and mobile robots[J]. The international journal of robotics research, 5(1): 90-98.
- Kim D, Nevatia R. 1998. Recognition and localization of generic objects for indoor navigation using functionality[J]. Image and Vision Computing, 16(11): 729-743.
- Kim D, Nevatia R. 1999. Symbolic navigation with a generic map[J]. Autonomous Robots, 6(1): 69-88.
- Kimmel R, Amir A, Bruckstein A M. 1995. Finding shortest paths on surfaces using level sets propagation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(6): 635-640.
- Kimia B B, Tannenbaum A R, Zucker S W. 1995. Shapes, shocks, and deformations I: the components of two-dimensional shape and the reaction-diffusion space[J]. International journal of computer vision, 15(3): 189-224.
- Saxena A, Andrew Y. Ng. 2014. Make3D: Convert your still image into 3D model[EB/OL]. <http://make3d.cs.cornell.edu/>
- Malladi R, Sethian J A, Vemuri B C. 1995. Shape modeling with front propagation: A level set approach[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(2): 158-175.
- McInerney T, Terzopoulos D. 1995. A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4D image analysis[J]. Computerized Medical Imaging and Graphics, 19(1): 69-83.
- Michels J, Saxena A, Ng A Y. 2005. High speed obstacle avoidance using monocular vision and

- reinforcement learning[C]//Proceedings of the 22nd international conference on Machine learning. USA: ACM, 593-600.
- Murillo A C, Guerrero J J, Sagues C. 2007. Surf features for efficient robot localization with omnidirectional images[C]// IEEE International Conference on Robotics and Automation, Roma: IEEE, 3901-3907.
- NASA. 2014. Spirit[EB/OL]. <http://photojournal.jpl.nasa.gov/spacecraft/Spirit>.
- Nielsen. 2013. The mobile consumer: a global snapshot[EB/OL]. www.nielsen.com.
- Osher S, Sethian J A. 1988. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations[J]. Journal of computational physics, 79(1): 12-49.
- Ramon V, Antonio V. 2012. PID Control in the Third Millennium Lessons Learned and New Approaches[M]. Springer.
- Royer E, Lhuillier M, Dhome M, et al. 2007. Monocular vision for mobile robot localization and autonomous navigation[J]. International Journal of Computer Vision, 74(3): 237-260.
- Santos-Victor J, Sandini G, Curotto F, et al. 1993. Divergent stereo for robot navigation: Learning from bees[C]//IEEE Computer Society Conference on Computer Vision and Pattern Recognition. USA: IEEE, 434-439.
- Santos A C, Tarrataca L, Cardoso J M P. 2010. The feasibility of navigation algorithms on smartphones using J2ME[J]. Mobile Networks and Applications, 15(6): 819-830.
- Sapiro G, Tannenbaum A. 1993. Affine invariant scale-space[J]. International journal of computer vision, 11(1): 25-44.
- Saxena A, Sun M, Ng A Y. 2009. Make3d: Learning 3d scene structure from a single still image[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(5): 824-840.
- Sethian J A. 1999. Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science[M]. Cambridge university press.
- Tardós J D, Neira J, Newman P M, et al. 2002. Robust mapping and localization in indoor environments using sonar data[J]. The International Journal of Robotics Research, 21(4): 311-330.
- Terzopoulos D, Witkin A, Kass M. 1988. Constraints on deformable models: Recovering 3D shape and nonrigid motion[J]. Artificial intelligence, 36(1): 91-123.
- Terzopoulos D, Metaxas D. 1990. Dynamic 3D models with local and global deformations: Deformable superquadrics[C]//Third International Conference on Computer Vision. Osaka: IEEE, 606-615.
- Thorpe C, Hebert M H, Kanade T, et al. 1988. Vision and navigation for the Carnegie-Mellon

- Navlab. IEEE Transactions on Pattern Analysis and Machine Intelligence, 10(3): 362-373.
- Xu Chenyang, Prince J L. 1998. Snakes, shapes, and gradient vector flow[J]. IEEE Transactions on Image Processing, 7(3): 359-369.
- Xu Chenyang, Pham D L, Prince J L. 2000. Image segmentation using deformable models [M]. Handbook of medical imaging, 2: 129-174.
- Williams D J, & Shah M. 1992. A fast algorithm for active contours and curvature estimation[J]. CVGIP: Image understanding, 55(1): 14-26.
- Winters N, Gaspar J, Lacey G, et al. 2000. Omni-directional vision for robot navigation[C]// IEEE Workshop on Omnidirectional Vision. Hilton Head Island: IEEE, 21-28.
- Yezzi Jr A, Kichenassamy S, Kumar A, et al. 1997. A geometric snake model for segmentation of medical imagery[J]. IEEE Transactions on Medical Imaging, 16(2): 199-209.
- Zhang Zhengyou. 2000. A flexible new technique for camera calibration[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11): 1330-1334.
- Ziegler J G, Nichols N B. 1942. Optimum settings for automatic controllers. trans. ASME, 64(11).

附录 1 参数轮廓模型变分问题求解过程

由前文可知，主动轮廓模型的能量极小化问题是一个泛函，其中被积表达式如下：

$$\mathbf{F}(s, \mathbf{X}, \mathbf{X}_s, \mathbf{X}_{ss}) = \frac{1}{2} \left(\alpha \left| \frac{\partial \mathbf{X}}{\partial s} \right|^2 + \beta \left| \frac{\partial^2 \mathbf{X}}{\partial s^2} \right|^2 \right) + P(\mathbf{X}(s)), \quad (1.1)$$

根据变分法，泛函 $\mathbf{J}[s] = \int_0^1 \mathbf{F}(s, \mathbf{X}, \mathbf{X}_s, \mathbf{X}_{ss}) ds$ 在 \mathbf{X} 处取得极值的必要条件是 \mathbf{X} 满足欧拉-泊松(Euler-Poisson)方程：

$$\mathbf{F}_\mathbf{X} - \frac{d}{ds} \mathbf{F}_{\mathbf{X}_s} + \frac{d^2}{ds^2} \mathbf{F}_{\mathbf{X}_{ss}} = 0, \quad (1.2)$$

这里 $\mathbf{F}_\mathbf{X}$ 为 \mathbf{F} 关于 \mathbf{X} 的偏导数， $\mathbf{F}_{\mathbf{X}_s}$ 是 \mathbf{F} 关于 \mathbf{X}_s 的偏导数， $\mathbf{F}_{\mathbf{X}_{ss}}$ 是 \mathbf{F} 关于 \mathbf{X}_{ss} 的偏导数，即： $\mathbf{X}_s = \partial \mathbf{X} / \partial s$ ， $\mathbf{X}_{ss} = \partial^2 \mathbf{X} / \partial s^2$ 。

可以将式 1.1 化为：

$$\mathbf{F}(s, \mathbf{X}, \mathbf{X}_s, \mathbf{X}_{ss}) = \frac{1}{2} \left(\alpha |\mathbf{X}_s|^2 + \beta |\mathbf{X}_{ss}|^2 \right) + P(\mathbf{X}), \quad (1.3)$$

将式 1.3 代入式 1.2，对每一项计算偏导可得：

$$\mathbf{F}_\mathbf{X} = \frac{\partial \mathbf{F}}{\partial \mathbf{X}} = \nabla P(\mathbf{X}) \quad (1.4)$$

$$\frac{d}{ds} \mathbf{F}_{\mathbf{X}_s} = \frac{\partial}{\partial s} (\alpha \mathbf{X}_s) \quad (1.5)$$

$$\frac{d^2}{ds^2} \mathbf{F}_{\mathbf{X}_{ss}} = \frac{\partial^2}{\partial s^2} (\beta \mathbf{X}_{ss}) \quad (1.6)$$

将式 1.4，1.5，1.6 代入式 1.2 可得：

$$\frac{\partial}{\partial s} \left(\alpha \frac{\partial \mathbf{X}}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left(\beta \frac{\partial^2 \mathbf{X}}{\partial s^2} \right) - \nabla P(\mathbf{X}) = 0. \quad (1.7)$$

附录 2 GVF 公式求解及数值实现

设 GVF 矢量场为 $\mathbf{v}(x, y) = [u(x, y), v(x, y)]$, GVF 的最小能量函数可以用下式表示:

$$E = \iint \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |\mathbf{v} - \nabla f|^2 dx dy. \quad (2.1)$$

这是一个二元函数的泛函, 其中被积函数为:

$$\mathbf{F}(x, y, \mathbf{v}, \mathbf{v}_x, \mathbf{v}_y) = \mu(\mathbf{v}_x^2 + \mathbf{v}_y^2) + |\nabla f|^2 |\mathbf{v} - \nabla f|^2. \quad (2.2)$$

求该泛函的最小值可用变分法, 由奥氏方程可知, 当 \mathbf{v} 满足以下条件时, 式 2.1 取得极小值:

$$\mathbf{F}_v - \frac{\partial}{\partial x} \mathbf{F}_{v_x} - \frac{\partial}{\partial y} \mathbf{F}_{v_y} = 0. \quad (2.3)$$

将式 2.2 代入式 2.3 可得:

$$\begin{cases} \mu \nabla^2 u - (u - f_x)(f_x^2 + f_y^2) = 0 \\ \mu \nabla^2 v - (v - f_y)(f_x^2 + f_y^2) = 0 \end{cases} \quad (2.4)$$

其中, ∇^2 为拉普拉斯算子。将 \mathbf{v} 定义为随时间变化的函数 $\mathbf{v}(x, y, t)$, 并定义其对时间的偏导数等于式 2.4 左边部分, 则式 2.4 可变为:

$$\begin{cases} u_t(x, y, t) = \mu \nabla^2 u(x, y, t) - [u(x, y, t) - f_x(x, y)] \cdot [f_x(x, y)^2 + f_y(x, y)^2] \\ v_t(x, y, t) = \mu \nabla^2 v(x, y, t) - [v(x, y, t) - f_y(x, y)] \cdot [f_x(x, y)^2 + f_y(x, y)^2] \end{cases} \quad (2.5)$$

当 u, v 不再随时间变化时, 上式左边等于 0, 即可满足式 2.3 的条件。为表述方便, 将式 2.5 重写为:

$$\begin{cases} u_t(x, y, t) = \mu \nabla^2 u(x, y, t) - b(x, y)u(x, y, t) + c^1(x, y) \\ v_t(x, y, t) = \mu \nabla^2 v(x, y, t) - b(x, y)v(x, y, t) + c^2(x, y) \end{cases} \quad (2.6)$$

其中,

$$\begin{aligned} b(x, y) &= f_x(x, y)^2 + f_y(x, y)^2, \\ c^1(x, y) &= b(x, y)f_x(x, y), \\ c^2(x, y) &= b(x, y)f_y(x, y). \end{aligned}$$

由前文可知, f 为边缘图, 在整个图像上是固定的, 因此, 这几个系数可以先求出来。

设 x, y, t 的步长分别为 $\Delta x, \Delta y, \Delta t$, 对式 2.6 进行离散化为:

$$\begin{aligned} u_t &= \frac{1}{\Delta t} (u_{i,j}^{n+1} - u_{i,j}^n), \\ v_t &= \frac{1}{\Delta t} (v_{i,j}^{n+1} - v_{i,j}^n), \\ \nabla^2 u &= \frac{1}{\Delta x \Delta y} (u_{i+1,j} + u_{i,j+1} + u_{i-1,j} + u_{i,j-1} - 4u_{i,j}), \\ \nabla^2 v &= \frac{1}{\Delta x \Delta y} (v_{i+1,j} + v_{i,j+1} + v_{i-1,j} + v_{i,j-1} - 4v_{i,j}). \end{aligned}$$

将上式代入式 2.6 即可获得求解 GVF 的迭代表达式:

$$\begin{aligned} u_{i,j}^{n+1} &= (1 - b_{i,j} \Delta t) u_{i,j}^n + r(u_{i+1,j}^n + u_{i,j+1}^n + u_{i-1,j}^n + u_{i,j-1}^n - 4u_{i,j}^n) + c_{i,j}^1 \Delta t \\ v_{i,j}^{n+1} &= (1 - b_{i,j} \Delta t) v_{i,j}^n + r(v_{i+1,j}^n + v_{i,j+1}^n + v_{i-1,j}^n + v_{i,j-1}^n - 4v_{i,j}^n) + c_{i,j}^2 \Delta t \end{aligned} \quad (2.7)$$

其中:

$$r = \frac{\mu \Delta t}{\Delta x \Delta y} \quad (2.8)$$

并且, 在迭代过程中由 CFL (Courant–Friedrichs–Lewy) 条件可以推断 Δt 应该满足如下限制:

$$\Delta t \leq \frac{\Delta x \Delta y}{4\mu} \quad (2.9)$$

致 谢

眼前还闪现着刚入学时幼稚懵懂的我，转眼在科大生活学习了七年，想着即将离去，有诸多不舍。

在三年研究学习期间，我有幸得到了三位老师的教导，他们是：我的导师，中国科大汪增福教授，曹洋老师，郑志刚老师。三位老师深厚的学术功底，严谨的工作态度和敏锐的科学洞察力使我受益良多。

汪增福老师尊重每个同学自己的研究兴趣，并给与相应的悉心指导。曹洋老师无论是在学习上还是生活上都对我有着无微不至的关怀，是良师也是益友。郑志刚老师心态平和，宽容待人，让人如沐春风。衷心感谢他们多年来给予我的悉心教导和热情帮助。

感谢女友杨晶的理解和支持，感谢多年同窗和室友刘羽同学在我写论文期间的帮助。在此向所有帮助过我，关心过我的老师，同学，朋友，亲人说一声：谢谢大家了。

最后，谨以此文献给我的父亲和母亲。

2014 年 6 月

在读期间发表的学术论文与取得的其他研究成果

已发表论文:

- [1] 向文辉, 刘羽, 曹洋, 等. 2014. 基于车载单目图像的三维地平面估计[J]. 机器人, 36(1):76-82.
- [2] Xiang Wenhui, Cao Yang, Wang Zengfu. 2012. Automatic take-off and landing of a quad-rotor flying robot[C]//24th Chinese Control and Decision Conference. Taiyuan: IEEE, 1251-1255.