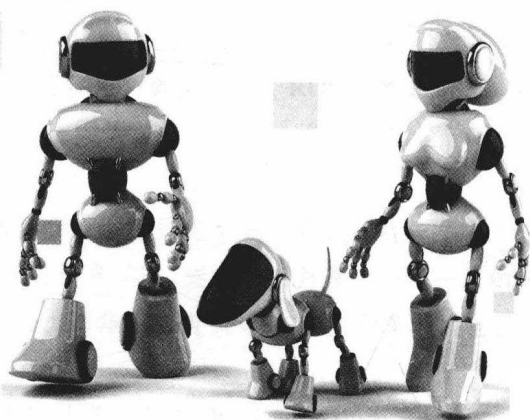


# ROS机器人开发实践

胡春旭 编著



机械工业出版社  
China Machine Press



# ROS机器人开发实践

常州大学图书馆  
藏书章

胡春旭 编著



机械工业出版社  
China Machine Press

## 图书在版编目 (CIP) 数据

ROS 机器人开发实践 / 胡春旭编著. —北京: 机械工业出版社, 2018.5  
(机器人设计与制作系列)

ISBN 978-7-111-59823-7

I. R… II. 胡… III. 机器人—程序设计 IV. TP242

中国版本图书馆 CIP 数据核字 (2018) 第 080884 号

# ROS 机器人开发实践

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 余 洁

责任校对: 殷 虹

印 刷: 中国电影出版社印刷厂

版 次: 2018 年 5 月第 1 版第 1 次印刷

开 本: 186mm×240mm 1/16

印 张: 32.75

书 号: ISBN 978-7-111-59823-7

定 价: 99.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

## | 内容简介 |

本书以实践为重心，讲解大量机器视觉、机器语音、机械臂控制、SLAM和导航、机器学习等多方面ROS应用的实现原理和方法，并配有大量ROS图表、代码等，帮助读者在实现ROS基础功能的同时深入理解基于ROS的机器人开发，从而将书中的内容用于实践。

本书共有14章，可以分为五个部分：第一部分（第1~4章）是ROS基础，了解ROS框架，并且熟悉ROS中的关键概念以及实现方法；第二部分（第5~6章）介绍如何搭建真实或仿真的机器人平台，了解机器人系统的概念和组成，学习如何使用ROS实现机器人仿真，为后续的机器人实践做好准备；第三部分（第7~10章）介绍ROS中常用功能包的使用方法，涉及机器视觉、机器语音、机械臂控制、SLAM和导航等多个机器人研究领域；第四部分（第11~13章）是ROS的进阶内容，介绍了ROS的进阶功能、ROS与机器学习的结合、搭载ROS的机器人平台；第五部分（第14章）介绍了新一代ROS——ROS 2，涉及ROS 2的架构、原理和使用方法。

## | 作者简介 |



胡春旭

网名古月，机器人博客“古月居”博主（[www.guyuehome.com](http://www.guyuehome.com)），华中科技大学自动化学院硕士。目前主要从事机器人控制系统的研究开发。

作为国内最早一批ROS开发者，具备多年开发、应用经验，参与研发过服务机器人、机械臂等机器人产品。曾获开源硬件与嵌入式大赛机器人组全国冠军，飞思卡尔杯智能汽车竞赛全国一等奖，中国人工智能机器人学术会议十佳论文等奖项。



古月居

## Foreword | 推荐序一

“古月”容易让人想到古龙笔下的大侠。

大侠的特质是：开山建宗，随而遁影山林，空余武林纷说大侠的故事。

古月在 CSDN 留下了那些优美的博文——《ROS 探索总结》，启蒙了多少人，开启了多少人对 ROS 的向往？随后，古月却“消失”了，任凭人们谈论着：“古月是谁？”“古月在哪里？”

2017 年年初，我无意间看到古月另建了一个独立网站并发布了一些与 ROS 和机器人相关的博文，马上意识到“古月回来了”。遂邀请古月参加一年一度的 ROS 暑期学校。这样在 2017 年 7 月底，我们有幸在上海华东师范大学一睹古月尊容，也让很多学员在这里接受了 ROS 启蒙。

2018 年 1 月 25 日，上海大雪，路过 2015 年 ROS 暑期学校的举办地——数学馆 201，以及 2016 年和 2017 年的举办地——理科楼 B222，想象着古月踏雪归来。这次他带着这本 ROS 武林秘籍，秉承 ROS 的开放精神，与更多的人分享 ROS 的基础和应用实践，尤其是包含最新的 ROS 2.0 的介绍。我想最可贵的是，此 ROS 武林秘籍通俗易懂。

古月的这本书注定将成为 ROS 江湖人手一本的“武林宝典”。

张新宇博士

华东师范大学智能机器人运动与视觉实验室负责人

机器人操作系统 (ROS) 暑期学校创办人

## 推荐序二 | Foreword

2011年，当国内 ROS 资料还很匮乏的时候，正在读本科的古月同学为了开发一款机器人，一边摸索自学一边总结撰写了《ROS 探索总结》系列博客。由于其博文条理清晰、主题丰富并具有很强的可操作性，文章一经发布便深受广大 ROS 网友的热爱。从此“古月大神”便成为群里热议的话题，“古月大神又出新文章啦！大家快去撸一撸啊！”……

后来，古月同学研究生毕业后就投身于机器人创业的时代浪潮，博客也便沉寂了许久。

你不在江湖，江湖却一直有你的传说！

记得是 2015 年冬天的一个深夜，我已经上床准备睡觉了，突然收到一个群聊信息，原来是机械工业出版社华章公司的张国强先生邀请我一起建议古月出版一本 ROS 相关的图书，我就从多年来学习阅读《ROS 探索总结》的体会以及升级为图书后的风格和思路提了几点简单建议，大家也交换了一些经验和想法，当时古月表示可以考虑出书。说实话，我也深知出书是一件耗时费力的苦差事。当初我曾接受出版社邀约，拉了几个小伙伴团结在一起甚至想合力完成一本书，最后却也未能克服困难坚持下来，更别说一个人写了。尤其是对于古月这样一位创业者，时间成本更是巨大！

在 2016 年创办的星火计划 ROS 公开课以及华东师范大学 ROS 暑期学校等活动中，我们邀请古月一起合作进行了多次授课。他不仅仅讲课深入浅出，每次在实践环节小伙伴们调试机器人时，常常会被各种“坑”折磨得焦头烂额、欲哭无泪，当小伙伴们含泪请教古月老师且自己还没讲清楚情况时，古月仅扫一眼，马上就会胸有成竹地说：“是不是这个现象？你应该这么解决……”此情此景，让我不由得想起了一句话：“今天你遇到的坑，都是我当年走过的路。”可见古月不仅文笔好，实战能力也是超群。

后来也多次得知他创业繁忙，心中也暗自揣度出书的事情恐怕是要搁浅了。然而一直到 2017 年 12 月 26 日，突然收到古月发的一条信息。

“Hi, Top, 还记得两年前筹备的那本书吗？现在终于写完了，希望邀请您写一个推荐序，不知是否方便？”

哇，这可真是大惊喜啊！我连忙打开电脑下载邮件，并将文件打印装订成书，放下手头的工作，重启“ROS 探索之旅”，几个晚上看下来总体感觉如下。

首先，书的内容主要源自古月个人项目开发的经验习得，书名中的“实践”二字恰如其分！其次，书虽是源自《ROS探索总结》系列博文，但也绝不是博文的简单汇集，不仅内容上有了非常大的充实（增加了多个新的章节）和更新（跟踪至2017年12月ROS的最新进展），而且在结构编排上也更适合阅读和上机操作。厚厚的书稿承载着作者满满的诚意，除去国外某两本由ROS论文合集组成的图书，本书也是目前为止国内外已出版的内容最丰富的一本ROS相关图书。可见作者为此书花费了大量的心血！最后，作者对书中的示例代码进行了认真的调试，也做了大量的修改和注释。

作为人工智能的综合实体平台，当前阻碍机器人实现大规模应用的一项主要障碍就是软件的不标准化（只能专用，无法通用）。每每针对某个特定应用场景设计机器人时，都需要花费大量成本和努力来对机器人进行设计和编程。即使完成之后，如果需要对机器人功能进行一个很小的改动，整个系统都需要进行成本很高的重新设计和开发，显然这是不符合可持续和可继承要求的，其限制了机器人的大规模应用与推广。

正如60年前软件行业放弃了从头编写程序的工作模式，ROS的出现是机器人开发的一场革命。如同从软件库和模块开始构建软件一样，通过ROS可以将机器人的标准算法例程化、软件模块化、成果共享化，后人可通过组合软件库和模块来实现十分复杂的功能。ROS有效地降低了工程的复杂度和工作量，让我们不仅可以很快地搭建出机器人系统，而且能够实现大型团队的协同工作与成果复用。这也正是我们努力推广ROS的主要动因。

愿与大家一同享受探索的欢喜！

刘锦涛 (Top) 博士

易科机器人实验室 (ExBot Robotics Lab) 负责人

星火计划联合发起人



## 推荐序三 | Foreword

近年来机器人技术发展越发成熟，越来越多的机器人技术应用在不同的领域。基于机器人技术开发出来的产品推陈出新，如物流机器人、家庭陪护机器人、协作机器人、送餐机器人、清洁机器人、无人机、无人汽车等，可谓百花齐放。大众对机器人的认知及学习的兴趣也不断提升，对机器人相关的技术变得更为关注，而 ROS 就是一个很典型的例子。

ROS 是一个专门针对机器人软件开发而设计的通信框架，源自美国斯坦福大学团队的一个开源项目，目前已有十年的发展历史，其开源以及对商用友好的版权协议使得它很快就得到越来越多的关注与支持。现在的 ROS 已有飞快的发展，越来越多机器人相关的软件工具亦加入 ROS 的行列。国内外也开始出现一些支持 ROS 系统，甚至是基于 ROS 进行开发的商用机器人。相信这个趋势会一直持续下去并且蔓延到全球各地，最终使之成为机器人领域的普遍标准。而我亦通过举办 ROS 推广及培训深深体会到国内对 ROS 的关注也在近年来有显著的上升。

作者是国内最早一批接触 ROS 的人，其 ROS 实战经验非常丰富。我们举办的 ROS 推广活动“星火计划”有幸能邀请到作者作为讲师，学生们对他也是一致好评。而本书的内容亦同样非常精彩，是我现今看过的内容最全面、涵盖层面最广的 ROS 中文入门书籍。从 ROS 1.0 到 ROS 2.0，本书对各种常用的架构、组件及工具等都有完整的叙述，是一本很好的 ROS “入门字典”。其中作者亦把很多人的实战经验融入书中，与网络上的教材相比，必定有另一番收获。

ROS 是机器人软件开发者间一种共同的语言、一个沟通的桥梁。大家可以通过 ROS 的学习及应用，与全球机器人软件开发者进行交流。如你对机器人学已有一定的认识，希望进一步打开机器人软件开发者社群的宝库，这本书你绝对不能错过。

Dr. LAM, Tin Lun 林天麟博士

NXROBO 创始人兼 CEO

## Preface | 前言

2011年年底，笔者第一次接触 ROS。当时实验室的一个师兄在学术会议上听说了 ROS 并意识到它的前景广阔，考虑到笔者当时的研究方向，于是建议笔者进行研究。那时国内外 ROS 的学习环境比较艰苦，几乎只有 Wiki 的基础教程（也没有现在这么完善）。所以一开始，笔者的内心是拒绝的，但还是硬着头皮开始钻研。虽然从拒绝到接受、从未知到熟悉，笔者经历了前所未有的磨难，但同时也收获了前所未有的喜悦。

在这个过程中，笔者也常常思考：ROS 前景无限，但是国内还鲜有人知，即使有人知道，也会被困难吓倒。既然笔者经历过，何不总结一下，让其他人少走弯路。于是，笔者整理了自己学习过程中的一些资料和心得，在 CSDN 上以博客的形式发表，最终形成《ROS 探索总结》系列博文，再后来转移到个人网站——古月居，至今仍保持更新。

2017年11月，ROS 十周岁了！在走过的第一个十年里，ROS 从蹒跚学步的孩童成长为机器人领域的巨人，再华丽蜕变出 ROS 2。如今，大多数知名机器人平台和机器人公司都支持 ROS，越来越多的机器人开发者也选择 ROS 作为开发框架。ROS 已经逐渐成为机器人领域的事实标准，并将逐步从研发走向市场，助力机器人与人工智能的快速发展。

ROS 的重要精神是分享，这也是开源软件的精神，所以才能看到如此活跃的 ROS 社区和众多软件功能包的源码，并且可以在此基础上快速完成二次开发。为了促进 ROS 在国内的发展，现在已经有很多人及组织在积极推广 ROS，比如 ROS 星火计划、ROS 暑期学校，以及网上各种各样的技术分享等，相信未来这个队伍会更加庞大。

本书以《ROS 探索总结》系列博文为基础，重新整理了 ROS 相关基础要点，让读者能够迅速熟悉 ROS 的整体框架和设计原理；在此基础上，本书以实践为重心，讲解大量机器视觉、机器语音、机械臂控制、SLAM 和导航、机器学习等多方面 ROS 应用的实现原理和方法，并且翻译了众多 ROS 中的图表、内容，帮助读者在实现 ROS 基础功能的同时深入理解基于 ROS 的机器人开发，将书中的内容用于实践。

本书共有 14 章，可以分为五个部分。

第一部分是 ROS 基础（第 1 ~ 4 章），帮助了解 ROS 框架，并且熟悉 ROS 中的关键概念以及实现方法。这部分的内容适合初学者，也适合作为有一定经验或者资深开发者的参考手册。

第二部分介绍如何搭建真实或仿真的机器人平台(第5~6章),帮助了解机器人系统的概念和组成,学习如何使用 ROS 实现机器人仿真,为后续的机器人实践做好准备。这部分的内容适合希望自己动手设计、开发一个完整机器人平台的读者。

第三部分介绍 ROS 中常用功能包的使用方法(第7~10章),涉及机器视觉、机器语音、机械臂控制、SLAM 和导航等多个机器人研究领域。这部分的内容适合学习 ROS 基础后希望实践的开发者,以及从事相关领域的机器人开发者。

第四部分是 ROS 的进阶内容(第11~13章),介绍了 ROS 的进阶功能、ROS 与机器学习的结合、搭载 ROS 的机器人平台。这部分的内容适合已经对 ROS 基础和应用有一定了解的读者。

第五部分介绍了新一代 ROS——ROS 2(第14章),涉及 ROS 2 的架构、原理和使用方法。这部分的内容适合对 ROS 有一定了解,希望了解 ROS 2、想要跟上 ROS 进化步伐的开发者。

因此,本书不仅适合希望了解、学习、应用 ROS 的机器人初学者,也适合有一定经验的机器人开发人员,同时也可以作为资深机器人开发者的参考手册。

书中的部分源代码来自社区中的 ROS 功能包,但是笔者在学习过程中对这些代码进行了大量修改,并且为大部分源代码加入了中文注释,以方便国内 ROS 初学者理解。这些代码涉及的编程语言不局限于 C++ 或 Python 中的某一种,编程语言应该服务于具体场景,所以建议读者对这两种语言都有所了解,在不同的应用中发挥每种语言的优势。关于是否需要一款实物机器人作为学习平台,本书并没有特别要求,书中绝大部分功能和源码都可以在单独的计算机或仿真平台中运行,同时也会介绍实物机器人平台的搭建方法并且在实物机器人上完成相应的功能。所以只需要拥有一台运行 Ubuntu 系统的计算机,具备 Linux 工具的基本知识,了解 C++ 和 Python 的编程方法,即可使用本书。

此外,本书创作过程中参考了众多已经出版的 ROS 原著、译著,笔者也将这些内容作为参考资料列出,并向这些著作的作者和译者致敬,希望读者在学习 ROS 的过程中,可以从这些著作中获取更多知识:

- 《Mastering ROS for Robotics Programming》, Lentin Joseph
- 《ROS By Example》(Volume 1/Volume 2), Patrick Goebel
- 《Programming Robots with ROS:A Practical Introduction to the Robot Operating System》, Morgan Quigley, Brian Gerkey & William D. Smart
- 《Learning ROS for Robotics Programming》, Aaron Martinez, Enrique Fernández
- 《A Gentle Introduction to ROS》, Jason M. O’Kane
- 《ROS Robotics Projects》, Lentin Joseph
- 《Effective Robotics Programming with ROS》, Anil Mahtani, Luis Sanchez

在 ROS 探索实践与本书的创作过程中,离不开众多“贵人”的帮助。首先要感谢陪伴笔

者辗转多次并一直无条件支持笔者的妻子，是她给了笔者前进的动力和思考的源泉；其次要感谢笔者的导师何顶新教授，以及为笔者打开 ROS 大门的任慰博士，还有曾与笔者一起彻夜调试的实验室同学顾强、方华启、胡灿、孙佳将、牛盼情、熊泉等；感谢机械工业出版社华章公司对本书的大力支持，以及 Linksprite 姚琪和 ROSClub 李文韬对本书所用硬件平台的赞助；最后要感谢 ROS 探索之路上一同前行的伙伴们，他们是张新宇教授、刘锦涛博士、林天麟博士、王滨海博士、杨帆、田博、张瑞雷、李卓、邱强、林浩鋋等，以及通过博客、邮件与笔者交流的众多机器人爱好者、开发者。要感谢的人太多，无法一一列举，但是笔者都感恩在心。

ROS 成长迅速，机器人系统更是错综复杂，笔者才疏学浅，书中难免有不足和错误之处，欢迎各位读者批评指正，这也是笔者继续前进的动力。本书相关内容的更新和勘误会发布在微信公众号“古月居”和笔者的个人网站 (<http://www.guyuehome.com/>) 上，欢迎各位读者关注或者通过任何形式与笔者交流。

最后分享胡适先生的一句名言，愿你我共勉：怕什么真理无穷，进一寸有一寸的欢喜。

胡春旭

2017 年 12 月于广东深圳

# 目 录 | Contents

推荐序一

推荐序二

推荐序三

前言

## 第 1 章 初识 ROS ..... 1

1.1 ROS 是什么 ..... 1

1.1.1 ROS 的起源 ..... 1

1.1.2 ROS 的设计目标 ..... 2

1.1.3 ROS 的特点 ..... 3

1.2 如何安装 ROS ..... 4

1.2.1 操作系统与 ROS 版本的选择 ..... 4

1.2.2 配置系统软件源 ..... 6

1.2.3 添加 ROS 软件源 ..... 6

1.2.4 添加密钥 ..... 7

1.2.5 安装 ROS ..... 7

1.2.6 初始化 rosdep ..... 8

1.2.7 设置环境变量 ..... 8

1.2.8 完成安装 ..... 9

1.3 本书源码下载 ..... 9

1.4 本章小结 ..... 10

## 第 2 章 ROS 架构 ..... 11

2.1 ROS 架构设计 ..... 11

2.2 计算图 ..... 12

2.2.1 节点 ..... 12

2.2.2 消息 ..... 13

2.2.3 话题 ..... 13

2.2.4 服务 ..... 13

2.2.5 节点管理器 ..... 14

2.3 文件系统 ..... 14

2.3.1 功能包 ..... 14

2.3.2 元功能包 ..... 16

2.4 开源社区 ..... 17

2.5 ROS 的通信机制 ..... 17

2.5.1 话题通信机制 ..... 18

2.5.2 服务通信机制 ..... 19

2.5.3 参数管理机制 ..... 20

2.6 话题与服务的区别 ..... 20

2.7 本章小结 ..... 21

## 第 3 章 ROS 基础 ..... 22

3.1 第一个 ROS 例程——小乌龟

仿真 ..... 23

3.1.1 turtlesim 功能包 ..... 23

3.1.2 控制乌龟运动 ..... 24

3.2 创建工作空间和功能包 ..... 25

3.2.1 什么是工作空间 ..... 25

3.2.2 创建工作空间 ..... 26

3.2.3 创建功能包 ..... 27

3.3	工作空间的覆盖	28	3.9.2	设置 ROS_MASTER_URI	56
3.3.1	ROS 中工作空间的覆盖	28	3.9.3	多机通信测试	56
3.3.2	工作空间覆盖示例	28	3.10	本章小结	57
3.4	搭建 Eclipse 开发环境	30	<b>第 4 章 ROS 中的常用组件</b> 58		
3.4.1	安装 Eclipse	30	4.1	launch 启动文件	58
3.4.2	创建 Eclipse 工程文件	30	4.1.1	基本元素	58
3.4.3	将工程导入 Eclipse	31	4.1.2	参数设置	60
3.4.4	设置头文件路径	31	4.1.3	重映射机制	61
3.4.5	运行 / 调试程序	32	4.1.4	嵌套复用	61
3.5	RoboWare 简介	35	4.2	TF 坐标变换	62
3.5.1	RoboWare 的特点	35	4.2.1	TF 功能包	62
3.5.2	RoboWare 的安装与使用	36	4.2.2	TF 工具	63
3.6	话题中的 Publisher 与 Subscriber	37	4.2.3	乌龟例程中的 TF	65
3.6.1	乌龟例程中的 Publisher 与 Subscriber	37	4.2.4	创建 TF 广播器	67
3.6.2	如何创建 Publisher	37	4.2.5	创建 TF 监听器	68
3.6.3	如何创建 Subscriber	40	4.2.6	实现乌龟跟随运动	70
3.6.4	编译功能包	41	4.3	Qt 工具箱	70
3.6.5	运行 Publisher 与 Subscriber	42	4.3.1	日志输出工具 (rqt_console)	71
3.6.6	自定义话题消息	44	4.3.2	计算图可视化工具 (rqt_graph)	71
3.7	服务中的 Server 和 Client	46	4.3.3	数据绘图工具 (rqt_plot)	72
3.7.1	乌龟例程中的服务	46	4.3.4	参数动态配置工具 (rqt_reconfigure)	73
3.7.2	如何自定义服务数据	47	4.4	rviz 三维可视化平台	73
3.7.3	如何创建 Server	48	4.4.1	安装并运行 rviz	74
3.7.4	如何创建 Client	49	4.4.2	数据可视化	75
3.7.5	编译功能包	51	4.4.3	插件扩展机制	76
3.7.6	运行 Server 和 Client	51	4.5	Gazebo 仿真环境	78
3.8	ROS 中的命名空间	52	4.5.1	Gazebo 的特点	78
3.8.1	有效的命名	52	4.5.2	安装并运行 Gazebo	78
3.8.2	命名解析	53	4.5.3	构建仿真环境	81
3.8.3	命名重映射	54	4.6	rosviz 数据记录与回放	82
3.9	分布式多机通信	54			
3.9.1	设置 IP 地址	55			

4.6.1	记录数据	82	5.7.1	rplidar 功能包	110
4.6.2	回放数据	83	5.7.2	PC 端驱动 rplidar	111
4.7	本章小结	84	5.7.3	Raspberry Pi 驱动 rplidar	113
<b>第 5 章 机器人平台搭建</b>		<b>85</b>	5.8	本章小结	113
5.1	机器人的定义	85	<b>第 6 章 机器人建模与仿真</b>		<b>114</b>
5.2	机器人的组成	86	6.1	统一机器人描述格式——URDF	114
5.2.1	执行机构	87	6.1.1	<link> 标签	114
5.2.2	驱动系统	87	6.1.2	<joint> 标签	115
5.2.3	传感系统	87	6.1.3	<robot> 标签	116
5.2.4	控制系统	87	6.1.4	<gazebo> 标签	116
5.3	机器人系统搭建	88	6.2	创建机器人 URDF 模型	116
5.3.1	MRobot	88	6.2.1	创建机器人描述功能包	116
5.3.2	执行机构的实现	88	6.2.2	创建 URDF 模型	117
5.3.3	驱动系统的实现	89	6.2.3	URDF 模型解析	120
5.3.4	内部传感系统的实现	90	6.2.4	在 rviz 中显示模型	122
5.4	基于 Raspberry Pi 的控制系统实现	90	6.3	改进 URDF 模型	124
5.4.1	硬件平台 Raspberry Pi	91	6.3.1	添加物理和碰撞属性	124
5.4.2	安装 Ubuntu 16.04	91	6.3.2	使用 xacro 优化 URDF	125
5.4.3	安装 ROS	93	6.3.3	xacro 文件引用	127
5.4.4	控制系统与 MRobot 通信	94	6.3.4	显示优化后的模型	127
5.4.5	PC 端控制 MRobot	97	6.4	添加传感器模型	128
5.5	为机器人装配摄像头	99	6.4.1	添加摄像头	128
5.5.1	usb_cam 功能包	99	6.4.2	添加 Kinect	130
5.5.2	PC 端驱动摄像头	100	6.4.3	添加激光雷达	132
5.5.3	Raspberry Pi 驱动摄像头	102	6.5	基于 ArbotiX 和 rviz 的仿真器	133
5.6	为机器人装配 Kinect	104	6.5.1	安装 ArbotiX	133
5.6.1	freenect_camera 功能包	104	6.5.2	配置 ArbotiX 控制器	133
5.6.2	PC 端驱动 Kinect	106	6.5.3	运行仿真环境	135
5.6.3	Raspberry Pi 驱动 Kinect	109	6.6	ros_control	136
5.6.4	Kinect 电源改造	109	6.6.1	ros_control 框架	137
5.7	为机器人装配激光雷达	110	6.6.2	控制器	139
			6.6.3	硬件接口	139
			6.6.4	传动系统	140

6.6.5	关节约束	140	7.5.2	源码实现	171
6.6.6	控制器管理器	141	7.6	二维码识别	173
6.7	Gazebo 仿真	142	7.6.1	ar_track_alvar 功能包	173
6.7.1	机器人模型添加 Gazebo		7.6.2	创建二维码	174
	属性	142	7.6.3	摄像头识别二维码	175
6.7.2	在 Gazebo 中显示机器人		7.6.4	Kinect 识别二维码	178
	模型	145	7.7	物体识别	179
6.7.3	控制机器人在 Gazebo 中		7.7.1	ORK 功能包	179
	运动	147	7.7.2	建立物体模型库	181
6.7.4	摄像头仿真	147	7.7.3	模型训练	183
6.7.5	Kinect 仿真	150	7.7.4	三维物体识别	184
6.7.6	激光雷达仿真	153	7.8	本章小结	185
6.8	本章小结	155			
<b>第 7 章 机器视觉</b>		156	<b>第 8 章 机器语音</b>		186
7.1	ROS 中的图像数据	156	8.1	让机器人听懂你说的话	187
7.1.1	二维图像数据	156	8.1.1	pocketsphinx 功能包	187
7.1.2	三维点云数据	158	8.1.2	语音识别测试	188
7.2	摄像头标定	159	8.1.3	创建语音库	190
7.2.1	camera_calibration		8.1.4	创建 launch 文件	192
	功能包	159	8.1.5	语音指令识别	192
7.2.2	启动标定程序	159	8.1.6	中文语音识别	192
7.2.3	标定摄像头	160	8.2	通过语音控制机器人	193
7.2.4	标定 Kinect	162	8.2.1	编写语音控制节点	193
7.2.5	加载标定参数的配置		8.2.2	语音控制小乌龟运动	194
	文件	162	8.3	让机器人说话	195
7.3	OpenCV 库	164	8.3.1	sound_play 功能包	195
7.3.1	安装 OpenCV	164	8.3.2	语音播放测试	195
7.3.2	在 ROS 中使用 OpenCV	164	8.4	人工智能标记语言	196
7.4	人脸识别	166	8.4.1	AIML 中的标签	196
7.4.1	应用效果	167	8.4.2	Python 中的 AIML	
7.4.2	源码实现	168		解析器	197
7.5	物体跟踪	170	8.5	与机器人对话	198
7.5.1	应用效果	170	8.5.1	语音识别	199
			8.5.2	智能匹配应答	201



8.5.3	文本转语音	202	9.5.4	在 Gazebo 中仿真 SLAM	246
8.5.4	智能对话	203	9.5.5	真实机器人 SLAM	247
8.6	让机器人听懂中文	204	9.6	rgbdslam	249
8.6.1	下载科大讯飞 SDK	204	9.6.1	rgbdslam 功能包	249
8.6.2	测试 SDK	206	9.6.2	使用数据包实现 SLAM	250
8.6.3	语音听写	207	9.6.3	使用 Kinect 实现 SLAM	252
8.6.4	语音合成	209	9.7	ORB_SLAM	253
8.6.5	智能语音助手	211	9.7.1	ORB_SLAM 功能包	253
8.7	本章小结	213	9.7.2	使用数据包实现单目 SLAM	254
<b>第 9 章 机器人 SLAM 与自主 导航</b>		214	9.7.3	使用摄像头实现单目 SLAM	256
9.1	理论基础	214	9.8	导航功能包	258
9.2	准备工作	216	9.8.1	导航框架	258
9.2.1	传感器信息	217	9.8.2	move_base 功能包	258
9.2.2	仿真平台	219	9.8.3	amcl 功能包	260
9.2.3	真实机器人	222	9.8.4	代价地图的配置	263
9.3	gmapping	224	9.8.5	本地规划器配置	266
9.3.1	gmapping 功能包	224	9.9	在 rviz 中仿真机器人导航	267
9.3.2	gmapping 节点的配置与 运行	227	9.9.1	创建 launch 文件	267
9.3.3	在 Gazebo 中仿真 SLAM	228	9.9.2	开始导航	268
9.3.4	真实机器人 SLAM	231	9.9.3	自动导航	269
9.4	hector-slam	234	9.10	在 Gazebo 中仿真机器人 导航	277
9.4.1	hector-slam 功能包	234	9.10.1	创建 launch 文件	277
9.4.2	hector_mapping 节点的 配置与运行	236	9.10.2	运行效果	278
9.4.3	在 Gazebo 中仿真 SLAM	237	9.10.3	实时避障	279
9.4.4	真实机器人 SLAM	238	9.11	真实机器人导航	280
9.5	cartographer	240	9.11.1	创建 launch 文件	280
9.5.1	cartographer 功能包	240	9.11.2	开始导航	282
9.5.2	官方 demo 测试	241	9.12	自主探索 SLAM	282
9.5.3	cartographer 节点的配置 与运行	244	9.12.1	创建 launch 文件	282
			9.12.2	通过 rviz 设置探索 目标	283