Anil Mahtani，Luis Sánchez，
Enrique Fernández，Aaron Martinez 著

ROS高效机器人编程

第3版（影印版）

Effective Robotics Programming with ROS, Third Edition

东南大学出版社
SOUTHEAST UNIVERSITY PRESS

Packt

ROS 高效机器人编程　第 3 版(影印版)

Effective Robotics Programming with ROS, Third Edition

Anil Mahtani, Luis Sánchez,
Enrique Fernández, Aaron Martinez 著

南京　东南大学出版社

图书在版编目(CIP)数据

ROS 高效机器人编程：第 3 版：英文/(西)阿尼尔·马哈塔尼(Anil Mahtani)等著. —影印本. —南京：东南大学出版社，2017.10

书名原文：Effective Robotics Programming with ROS, Third Edition

ISBN 978-7-5641-7365-4

Ⅰ. ①R… Ⅱ. ①阿… Ⅲ. ①机器人—程序设计—英文 Ⅳ. ①TP242

中国版本图书馆 CIP 数据核字(2017)第 196704 号

图字：10-2017-120 号

ROS 高效机器人编程　第 3 版(影印版)

出版发行：东南大学出版社
地　　址：南京四牌楼 2 号　　邮编：210096
出 版 人：江建中
网　　址：http://www.seupress.com
电子邮件：press@seupress.com
印　　刷：常州市武进第三印刷有限公司
开　　本：787 毫米×980 毫米　16 开本
印　　张：29.25
字　　数：573 千字
版　　次：2017 年 10 月第 1 版
印　　次：2017 年 10 月第 1 次印刷
书　　号：ISBN 978-7-5641-7365-4
定　　价：92.00 元

本社图书若有印装质量问题，请直接与营销部联系。电话(传真)：025-83791830

Credits

Authors
Anil Mahtani
Luis Sánchez
Enrique Fernández
Aaron Martinez

Reviewer
Lentin Joseph

Commissioning Editor
Kartikey Pandey

Acquisition Editor
Narsimha Pai

Content Development Editor
Abhishek Jadhav

Technical Editor
Gaurav Suri

Copy Editors
Safis Editing
Dipti Mankame

Project Coordinator
Judie Jose

Proofreader
Safis Editing

Indexer
Pratik Shirodkar

Graphics
Kirk D'Penha

Production Coordinator
Shantanu N. Zagade

Cover Work
Shantanu N. Zagade

About the Authors

Anil Mahtani is a computer scientist who has dedicated an important part of his career to underwater robotics. He first started working in the field with his master thesis, where he developed a software architecture for a low-cost ROV. During the development of his thesis, he also became the team leader and lead developer of AVORA, a team of university students that designed and developed an autonomous underwater vehicle for the Students Autonomous Underwater Challenge – Europe (SAUC-E) in 2012. That same year, Anil Mahtani completed his thesis and his MSc in Computer Science at the University of Las Palmas de Gran Canaria and then became a Software Engineer at SeeByte Ltd, a world leader in smart software solutions for underwater systems. In 2015, he joined Dell Secureworks as a Software Engineer, where he applies his knowledge and skills toward developing intrusion detection and prevention systems.

During his time at SeeByte Ltd, Anil Mahtani played a key role in the development of several semi-autonomous and autonomous underwater systems for the military and oil and gas industries. In those projects, he was heavily involved in the development of autonomy systems, the design of distributed software architectures, and low-level software development and also contributed in providing Computer Vision solutions for front-looking sonar imagery. At SeeByte Ltd, he also achieved the position of project manager, managing a team of engineers developing and maintaining the internal core C++ libraries.

His professional interests lie mainly in software engineering, algorithms, data structures, distributed systems, networks, and operating systems. Anil's main role in robotics is to provide efficient and robust software solutions, addressing not only the current problems at hand but also foreseeing future problems or possible enhancements. Given his experience, he is also an asset when dealing with Computer Vision, machine learning, or control problems. Anil has also interests in DIY and electronics, and he has developed several Arduino libraries, which he has contributed back to the community.

First of all, I would like to thank my family and friends for their support and for always being there when I needed them. I would also like to thank my girlfriend Alex for her support and patience, and for being a constant source of inspiration. Finally, I would like to thank my colleagues Ihor Bilyy and Dan Good, who have taught me a lot, both personally and professionally, during these new steps in my career as a software engineer.

Luis Sánchez has completed his dual master's degree in electronics and telecommunication engineering at the University of Las Palmas de Gran Canaria.

He has collaborated with different research groups as the Institute for Technological Development and Innovation (IDETIC), the Oceanic Platform of Canary Islands (PLOCAN), and the Institute of Applied Microelectronics (IUMA) where he actually researches on imaging super-resolution algorithms.

His professional interests lie in Computer Vision, signal processing, and electronic design applied on robotics systems. For this reason, he joined the AVORA team, a group of young engineers and students working on the development of Underwater Autonomous Vehicles (AUV) from scratch. Inside this project, Luis has started developing acoustic and Computer Vision systems, extracting information from different sensors such as hydrophones, sonar, or camera.

With a strong background gained in marine technology, Luis cofounded Subsea Mechatronics, a young start-up, where he works on developing remotely operated and autonomous vehicles for underwater environments.

Here's what Dario Sosa Cabrera, a marine technologies engineer and entrepreneur (and the cofounder and maker of LPA Fabrika: Gran Canaria Maker Space) has to say about Luis:

"He is very enthusiastic and an engineer in multiple disciplines. He is responsible for his work. He can manage himself and can take up responsibilities as a team leader, as demonstrated at the euRathlon competition. His background in electronics and telecommunications allows him to cover a wide range of expertise from signal processing and software, to electronic design and fabrication."

Luis has participated as a technical reviewer of the previous version of *Learning ROS for Robotics Programming* and as a cowriter of the second edition.

First, I have to acknowledge Aaron, Anil, and Enrique for inviting me to participate in this book. It has been a pleasure to return to work with them. Also, I want to thank the Subsea Mechatronics team for the great experience working with heavy underwater robots, we grew together during these years. I have to mention LPA Fabrika – Gran Canaria Maker Space for the enthusiasm preparing and teaching educational robotics and technological projects; sharing a workspace with kids can be really motivating.

Finally, I will have to thank my family and my girlfriend for the big support and encouragement in every project where I'm involved. I want to dedicate my contribution in this book to them.

Enrique Fernández has a PhD in computer engineering and an extensive background in robotics. His PhD thesis addressed the problem of Path Planning for Autonomous Underwater Gliders, but he also worked on other robotics projects, including SLAM, perception, vision, and control. During his doctorate, he joined the Center of Underwater Robotics Research in the University of Girona, where he developed Visual SLAM and INS modules in ROS for Autonomous Underwater Vehicles (AUVs), and participated in the Student Autonomous Underwater Challenge, Europe (SAUC-E) in 2012, and collaborated in the 2013 edition; in 2012, he was awarded a prize.

During his PhD, Enrique published several conference papers and publications to top robotics conferences, such as the International Conference of Robotics and Automation (ICRA). He has also authored some book chapters and ROS books.

Later, Enrique joined PAL Robotics as a SLAM engineer in June 2013. There he worked with the REEM and REEM-C humanoid robots using ROS software and also contributed to the open source community, mainly to ROS Control repository, being one of the maintainers nowadays. In 2015, he joined Clearpath Robotics to work on the Autonomy team, developing perception algorithms. He has worked on the software that runs on the industrial mobile robots OTTO 1500 and OTTO 100, which has been deployed into the facilities of multiple large industry companies, such as General Electric and John Deere.

I would like to thank the coauthors of the book for their dedication. I also want to say thanks to the members of my research group in Las Palmas de Gran Canaria and the Center of Underwater Robotics Research in Girona. I learned a lot about robotics then, and I started to work with ROS. Thanks also to the ex-colleagues from PAL Robotics, who received me with open hands, and have given me the opportunity to learn even more from ROS and (humanoid) robots. Last by not least, to my current colleagues at Clearpath Robotics, where I have mastered ROS and contributed to the software that runs 24/7 in the self-driving robots we have sold for the Industry 4.0. Finally, thanks to my family and friends for their help and support, especially Eva.

Aaron Martinez is a computer engineer, entrepreneur, and expert in digital fabrication. He did his master's thesis in 2010 at the IUCTC (Instituto Universitario de Ciencias y Tecnologias Ciberneticas) in the University of Las Palmas de Gran Canaria. He prepared his master's thesis in the field of telepresence using immersive devices and robotic platforms. After completing his academic career, he attended an internship program at The Institute for Robotics in the Johannes Kepler University in Linz, Austria. During his internship program, he worked as part of a development team of a mobile platform using ROS and the navigation stack. After that, he was involved in some projects related to robotics; one of them is the AVORA project in the University of Las Palmas de Gran Canaria. In this project, he worked on the creation of an AUV to participate in the Student Autonomous Underwater Challenge-Europe (SAUC-E) in Italy. In 2012, he was responsible for manufacturing this project; in 2013, he helped to adapt the navigation stack and other algorithms from ROS to the robotic platform.

Recently, Aaron created his own company named SubSeaMechatronics, SL. This company works with projects related with underwater robotics and telecontrol systems. They are also designing and manufacturing subsea sensors. The company manufactures devices for other companies and research and development institutes.

Aaron has experience in many fields, such as programming, robotics, mechatronics, and digital fabrication as well as many devices, such as Arduino, BeagleBone, Servers, and LIDAR, and nowadays he is designing in SubSeaMechatronics SL some robotics platforms for underwater and aerial environments.

I would like to thank my girlfriend who has supported me while writing this book and gave me motivation to continue growing professionally. I also want to thank Donato Monopoli, Head of Biomedical Engineering Department at ITC (Canary-Islands Institute of Technology), and all the staff there. Thanks for teaching me all I know about digital fabrication, machinery, and engineering tissue. I spent the best years of my life in your workshop.

Thanks to my colleagues in the university, especially Alexis Quesada, who gave me the opportunity to create my first robot in my master's thesis. I have learned a lot about robotics working with them.

Finally, thanks to my family and friends for their help and support.

About the Reviewer

Lentin Joseph is an author, entrepreneur, electronics engineer, robotics enthusiast, machine vision expert, embedded programmer, and the founder and CEO of Qbotics Labs (`http://www.qboticslabs.com`) in India.

He completed his bachelor's degree in electronics and communication engineering at the Federal Institute of Science and Technology (FISAT), Kerala. For his final year engineering project, he made a social robot that can interact with people (`http://www.technolabsz.com/2012/07/social-robot-my-final-year.html`). The project was a huge success and was mentioned in many forms of visual and print media. The main features of this robot were that it can communicate with people and reply intelligently and has some image processing capabilities, such as face, motion, and color detection. The entire project was implemented using the Python programming language. His interest in robotics, image processing, and Python started with that project.

After his graduation, for 3 years he worked at a start-up company focusing on robotics and image processing. In the meantime, he learned famous robotic software platforms, such as Robot Operating System (ROS), V-REP, Actin (a robotic simulation tool), and image processing libraries, such as OpenCV, OpenNI, and PCL. He also knows about robot 3D designing and embedded programming on Arduino and Tiva Launchpad.

After 3 years of work experience, he started a new company named Qbotics Labs, which mainly focuses on research to build up some great products in domains, such as robotics and machine vision. He maintains a personal website (`http://www.lentinjoseph.com`) and a technology blog named technolabsz (`http://www.technolabsz.com`). He publishes his works on his tech blog. He was also a speaker at PyCon2013, India, on the topic *Learning Robotics using Python*.

Lentin is the author of the books *Learning Robotics using Python* (refer to `http://learn-robotics.com` to find out more) and *Mastering ROS for Robotics Programming* (refer to `http://mastering-ros.com` to find out more) by Packt Publishing. The first book was about building an autonomous mobile robot using ROS and OpenCV. This book was launched in ICRA 2015 and was featured in the ROS blog, Robohub, OpenCV, the Python website, and various other such forums. The second book is for mastering robot operating system; this was also launched ICRA 2016, and it is one of the best seller book in ROS.

Lentin and his team was a winner of HRATC 2016 challenge conducted as a part of ICRA 2016, and he was Also a finalist in the ICRA 2015 challenge, HRATC (`http://www.icra2016.org/conference/challenges/`).

www.PacktPub.com

eBooks, discount offers, and more

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at `www.PacktPub.com` and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at `customercare@packtpub.com` for more details.

At `www.PacktPub.com`, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.

`https://www.packtpub.com/mapt`

Get the most in-demand software skills with Mapt. Mapt gives you full access to all Packt books and video courses, as well as industry-leading tools to help you plan your personal development and advance your career.

Why subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print, and bookmark content
- On demand and accessible via a web browser

Customer Feedback

Thank you for purchasing this Packt book. We take our commitment to improving our content and products to meet your needs seriously—that's why your feedback is so valuable. Whatever your feelings about your purchase, please consider leaving a review on this book's Amazon page. Not only will this help us, more importantly it will also help others in the community to make an informed decision about the resources that they invest in to learn.

You can also review for us on a regular basis by joining our reviewers' club. **If you're interested in joining, or would like to learn more about the benefits we offer, please contact us**: `customerreviews@packtpub.com`.

Preface

Effective Robotics Programming with ROS, Third Edition gives you a comprehensive review of ROS, the Robot Operating System framework, which is used nowadays by hundreds of research groups and companies in the robotics industry. More importantly, ROS is also the painless entry point to robotics for nonprofessionals and students. This book will guide you through the installation process of ROS, and soon enough, you will be playing with the basic tools and understanding the different elements of the framework.

The content of the book can be followed without any special devices, and each chapter comes with a series of source code examples and tutorials that you can run on your own computer. This is the only thing you need to follow the book.

However, we also show you how to work with hardware so that you can connect your algorithms with the real world. Special care has been taken in choosing devices that are affordable for amateur users, but at the same time, the most typical sensors or actuators in robotics research are covered.

Finally, the potential of ROS is illustrated with the ability to work with whole robots in a real or simulated environment. You will learn how to create your own robot and integrate it with a simulation by using the Gazebo simulator. From here, you will have the chance to explore the different aspects of creating a robot, such as perceiving the world using computer vision or point cloud analysis, navigating through the environment using the powerful navigation stack, and even being able to control robotic arms to interact with your surroundings using the MoveIt! package. By the end of the book, it is our hope that you will have a thorough understanding of the endless possibilities that ROS gives you when developing robotic systems.

What this book covers

Chapter 1, Getting Started with ROS, shows the easiest way you must follow in order to have a working installation of ROS. You will see how to install ROS on different platforms, and you will use ROS Kinetic throughout the rest of the book. This chapter describes how to make an installation from Debian packages, compile the sources, and make installations in virtual machines, Docker, and ARM CPU.

Chapter 2, ROS Architecture and Concepts, is concerned with the concepts and tools provided by the ROS framework. We will introduce you to nodes, topics, and services, and you will also learn how to use them. Through a series of examples, we will illustrate how to debug a node and visualize the messages published through a topic.

Chapter 3, Visualization and Debugging Tools, goes a step further in order to show you powerful tools to debug your nodes and visualize the information that goes through the node's graph along with the topics. ROS provides a logging API that allows you to diagnose node problems easily. In fact, we will see some powerful graphical tools, such as rqt_console and rqt_graph, as well as visualization interfaces, such as rqt_plot and rviz. Finally, this chapter explains how to record and play back messages using rosbag and rqt_bag.

Chapter 4, 3D Modeling and Simulation, constitutes one of the first steps in order to implement your own robot in ROS. It shows you how to model a robot from scratch and run it in simulation using the Gazebo simulator. You will simulate sensors, such as cameras and laser range sensors. This will later allow you to use the whole navigation stack provided by ROS and other tools.

Chapter 5, The Navigation Stack - Robot Setups, is the first of two chapters concerned with the ROS navigation stack. This chapter describes how to configure your robot so that it can be used with the navigation stack. In the same way, the stack is explained, along with several examples.

Chapter 6, The Navigation Stack - Beyond Setups, continues the discussion of the previous chapter by showing how we can effectively make our robot navigate autonomously. It will use the navigation stack intensively for that. This chapter shows the great potential of ROS using the Gazebo simulator and RViz to create a virtual environment in which we can build a map, localize our robot, and do path planning with obstacle avoidance.

Chapter 7, Manipulation with MoveIt!, is a set of tools for mobile manipulation in ROS. This chapter contains the documentation that you need to install this package. The chapter also contains example demonstrations with robotic arms that use MoveIt! for manipulation tasks, such as grasping, picking and placing, or simple motion planning with inverse kinematics.

Chapter 8, Using Sensors and Actuators with ROS, literally connects ROS with the real world. This chapter goes through a number of common sensors and actuators that are supported in ROS, such as range lasers, servo motors, cameras, RGB-D sensors, and GPS. Moreover, we explain how to use embedded systems with microcontrollers, similar to the widely known Arduino boards.

Chapter 9, Computer Vision, shows the support for cameras and computer vision tasks in ROS. This chapter starts with drivers available for FireWire and USB cameras so that you can connect them to your computer and capture images. You will then be able to calibrate your camera using the ROS calibration tools. Later, you will be able to use the image pipeline, which is explained in detail. Then, you will see how to use several APIs for vision and integrate OpenCV. Finally, the installation and usage of a visual odometry software is described.

Chapter 10, Point Clouds, shows how to use Point Cloud Library in your ROS nodes. This chapter starts with the basics utilities, such as read or write a PCL snippet and the conversions needed to publish or subscribe to these messages. Then, you will create a pipeline with different nodes to process 3D data, and you will downsample, filter, and search for features using PCL.

What you need for this book

This book was written with the intention that almost everybody can follow it and run the source code examples provided with it. Basically, you need a computer with a Linux distribution. Although any Linux distribution should be fine, it is recommended that you use a version of Ubuntu 16.04 LTS. Then, you will use ROS Kinetic, which is installed according to the instructions given in *Chapter 1, Getting Started with ROS*.

As regards the hardware requirements of your computer, in general, any computer or laptop is enough. However, it is advisable to use a dedicated graphics card in order to run the Gazebo simulator. Also, it will be good to have a good number of peripherals so that you can connect several sensors and actuators, including cameras and Arduino boards.

You will also need Git (the git-core Debian package) in order to clone the repository with the source code provided with this book. Similarly, you are expected to have a basic knowledge of the Bash command line, GNU/Linux tools, and some C/C++ programming skills.

Who this book is for

This book is targeted at all robotics developers, from amateurs to professionals. It covers all the aspects involved in a whole robotic system and shows how ROS helps with the task of making a robot really autonomous. Anyone who is learning robotics and has heard about ROS but has never tried it will benefit from this book. Also, ROS beginners will learn advanced concepts and tools of this framework. Indeed, even regular users may learn something new from some particular chapters. Certainly, only the first three chapters are intended for new users; so those who already use ROS can skip these ones and go directly to the rest.

Conventions

In this book, you will find a number of text styles that distinguish between different kinds of information. Here are some examples of these styles and an explanation of their meaning.

Code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles are shown as follows: "The `rosdep` command-line tool must be installed and initialized before you can use ROS."

A block of code is set as follows:

```
#include <ros/ros.h>
#include <dynamic_reconfigure/server.h>
#include <chapter2_tutorials/chapter2Config.h>
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in bold:

```
dynamic_reconfigure::Server<chapter2_tutorials::chapter2Config>::Call
backType f;

  f = boost::bind(&callback, _1, _2);
```

Any command-line input or output is written as follows:

```
$ sudo apt-get install python-rosdep
$ sudo rosdep init
$ rosdep update
```