ROS Robotics Projects

Second Edition

Build and control robots powered by the Robot Operating System, machine learning, and virtual reality

Ramkumar Gandhinathan and Lentin Joseph

Packt
www.packt.com

ROS Robotics Projects
Second Edition

Build and control robots powered by the Robot Operating System, machine learning, and virtual reality

Ramkumar Gandhinathan
Lentin Joseph

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI
TRUNG TÂM THÔNG TIN THƯ VIỆN
02 07
02339

Packt>

BIRMINGHAM - MUMBAI

ROS Robotics Projects
Second Edition

Copyright © 2019 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the authors, nor Packt Publishing or its dealers and distributors, will be held liable for any damages caused or alleged to have been caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

Commissioning Editor: Vijin Boricha
Acquisition Editor: Meeta Rajani
Content Development Editor: Pratik Andrade
Senior Editor: Rahul Dsouza
Technical Editor: Dinesh Pawar
Copy Editor: Safis Editing
Project Coordinator: Anish Daniel
Proofreader: Safis Editing
Indexer: Rekha Nair
Production Designer: Alishon Mendonsa

First published: March 2017
Second edition: December 2019

Production reference: 1181219

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham
B3 2PB, UK.

ISBN 978-1-83864-932-6

www.packt.com

Table of Contents

Preface

Robot Operating System (**ROS**) is one of the most popular robotics middleware and is used by universities and industries for robot-specific applications. Ever since its introduction, many robots have been introduced to the market and users have been able to use them with ease within their applications. One of its main draws is its open source nature. ROS does not need a user to reinvent the wheel; instead, standardizing robot operations and applications is simple.

This book is an upgrade to the previous edition and introduces you to newer ROS packages, interesting projects, and some added features. This book targets projects in the latest (at the time of writing) ROS distribution—ROS Melodic Morenia with Ubuntu Bionic version 18.04.

Here, you will understand how robots are used in industries and will learn the step-by-step procedure of building heterogeneous robot solutions. Unlike the service call and action features in ROS, you will be introduced to cooler techniques that let robots handle intricate tasks in a smart way. This knowledge should pave the way to far more intelligent and self-performing autonomous robots. Additionally, we will also introduce ROS-2, so you can learn the differences between this version and the previous ROS version and find help in choosing a specific middleware for your application.

Industries and research institutes are focusing primarily on the fields of computer vision and natural language processing. While the previous edition of this book introduced you to some simple vision applications such as object detection and face tracking, this edition will introduce you to one of the most widely used smart speaker platforms on the market, Amazon's Alexa, and how to control robots using it. In parallel, we will introduce new hardware, such as Nvidia Jetson, Asus Tinker Board, and BeagleBone Black and explore their capabilities with ROS.

While people may know how to control robots individually, one of the most common problems faced by users in the ROS community is the use of multiple robots working in synchronization, whether they are of the same type or not. This becomes complicated, as robots may follow similar topic names and may possibly lead to confusion in a sequence of operations. This book helps in highlighting the possible conflicts and suggests solutions.

This book also touches on reinforcement learning, including how it can be used with robotics and ROS. Furthermore, you will find the most interesting projects for building a self-driving car, deep learning with ROS, and building teleoperation solutions using VR headsets and Leap Motion, as they're currently trending and are being researched continuously.

Who this book is for

This book is for students, hobbyists, professionals, and individuals with a passion for learning robotics technology. Additionally, it is aimed at those individuals who are most interested in learning about and writing algorithms, motion control, and perception capabilities from scratch. This might even help a start-up build a new product or help researchers utilize what's already available and create something new and innovative. This book is also intended for those people who would like to work in the software domain or who want to have a career as a robotics software engineer.

What this book covers

Chapter 1, *Getting Started with ROS*, is a basic introductory chapter on ROS for beginners. This chapter will help you get an idea of the ROS software framework and its concepts.

Chapter 2, *Introduction to ROS-2 and Its Capabilities*, introduces you to ROS-2, the newest upgraded framework that helps us use ROS in real-time applications. This chapter is organized in a similar manner to Chapter 1, *Getting Started with ROS*, such that users are able to differentiate between both ROS versions and understand their capabilities and limitations.

Chapter 3, *Building an Industrial Mobile Manipulator*, is where you will learn how to build a mobile robot and a robot arm and combine them both to be used in a virtual environment and control them through ROS.

Chapter 4, *Handling Complex Robot Tasks Using State Machines*, introduces you to techniques in ROS that could be adapted while using robots for continuous and complicated task management.

Chapter 5, *Building an Industrial Application*, is where you will combine the skills acquired in Chapter 3, *Building an Industrial Mobile Manipulator* and Chapter 4, *Handling Complex Robot Tasks Using State Machines*, effectively, to create a user application. Here, we will demonstrate how to use the mobile manipulator to deliver products to houses in a neighborhood.

`Chapter 6`, *Multi-Robot Collaboration*, teaches you how to communicate between multiple robots of the same or different category and control them separately and together in groups.

`Chapter 7`, *ROS on Embedded Platforms and Their Control*, helps you understand the latest embedded controller and processor boards, such as STM32-based controllers, Tinker Board, Jetson Nano, and many more. We will also look at how to control their GPIOs via ROS and control them via voice-based commands through Alexa.

`Chapter 8`, *Reinforcement Learning and Robotics*, introduces you to one of the most commonly used learning techniques in robotics called reinforcement learning. In this chapter, you will understand what reinforcement learning is and the math behind it using examples. Additionally, we will discover how to incorporate this learning technique with ROS by means of simple projects.

`Chapter 9`, *Deep Learning Using ROS and TensorFlow*, is a project made using a trending technology in robotics. Using the TensorFlow library and ROS, we can implement interesting deep learning applications. You can implement image recognition using deep learning, and an application using SVM can be found in this chapter.

`Chapter 10`, *Creating a Self-Driving Car Using ROS*, is one of the more interesting projects in this book. In this chapter, we will build a simulation of a self-driving car using ROS and Gazebo.

`Chapter 11`, *Teleoperating Robots Using a VR Headset and Leap Motion*, shows you how to control a robot's actions using a VR headset and Leap Motion sensor. You can play around with VR, which is a trending technology these days.

`Chapter 12`, *Face Detection and Tracking Using ROS, OpenCV, and Dynamixel Servos*, takes you through a cool project that you can make with ROS and the OpenCV library. This project basically creates a face tracker application in which your face will be tracked in such a way that the camera will always point to your face. We will use intelligent servos such as Dynamixel to rotate the robot on its axis.

To get the most out of this book

- You should have a powerful PC running a Linux distribution, preferably Ubuntu 18.04 LTS.
- You can use a laptop or desktop with a graphics card, and RAM of at least 4 to 8 GB is preferred. This is actually for running high-end simulations in Gazebo, as well as for processing point clouds and computer vision.

- You should have the sensors, actuators, and I/O boards mentioned in the book and should be able to connect them all to your PC. You also need Git installed to clone the package files.
- If you are a Windows user, it will be good to download VirtualBox and set up Ubuntu on it. However, do note that you may have issues while you try to interface real hardware to ROS when working with VirtualBox.

Download the example code files

You can download the example code files for this book from your account at `www.packt.com`. If you purchased this book elsewhere, you can visit `www.packtpub.com/support` and register to have the files emailed directly to you.

You can download the code files by following these steps:

1. Log in or register at `www.packt.com`.
2. Select the **Support** tab.
3. Click on **Code Downloads**.
4. Enter the name of the book in the **Search** box and follow the onscreen instructions.

Once the file is downloaded, please make sure that you unzip or extract the folder using the latest version of:

- WinRAR/7-Zip for Windows
- Zipeg/iZip/UnRarX for Mac
- 7-Zip/PeaZip for Linux

The code bundle for the book is also hosted on GitHub at `https://github.com/PacktPublishing/ROS-Robotics-Projects-SecondEdition`. In case there's an update to the code, it will be updated on the existing GitHub repository.

We also have other code bundles from our rich catalog of books and videos available at `https://github.com/PacktPublishing/`. Check them out!

Download the color images

We also provide a PDF file that has color images of the screenshots/diagrams used in this book. You can download it here: `http://www.packtpub.com/sites/default/files/downloads/9781838649326_ColorImages.pdf`.

Code in Action

Visit the following link to check out videos of the code being run:
`http://bit.ly/34p6hL0`

Conventions used

There are a number of text conventions used throughout this book.

`CodeInText`: Indicates code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles. Here is an example: "Remove the `CMakelists.txt` file."

A block of code is set as follows:

```
def talker_main():
   rospy.init_node('ros1_talker_node')
    pub = rospy.Publisher('/chatter', String)
    msg = String()
    i = 0
```

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

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

Bold: Indicates a new term, an important word, or words that you see on screen. For example, words in menus or dialog boxes appear in the text like this. Here is an example: "Click on **Software & Updates** and enable all of the Ubuntu repositories."

Warnings or important notes appear like this.

Tips and tricks appear like this.

Get in touch

Feedback from our readers is always welcome.

General feedback: If you have questions about any aspect of this book, mention the book title in the subject of your message and email us at `customercare@packtpub.com`.

Errata: Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you have found a mistake in this book, we would be grateful if you would report this to us. Please visit `www.packtpub.com/support/errata`, selecting your book, clicking on the Errata Submission Form link, and entering the details.

Piracy: If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at `copyright@packt.com` with a link to the material.

If you are interested in becoming an author: If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit `authors.packtpub.com`.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions, we at Packt can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about Packt, please visit `packt.com`.