1. Record Nr. UNINA9910955187003321

Autore	Vipina Kumara
Titolo	Robot Operating System cookbook : over 70 recipes to help you master advanced ROS concepts / / Kumar Bipin
Pubbl/distr/stampa	Birmingham, UK : , : Packt Publishing, , 2018
ISBN	9781783987450 1783987456
Edizione	[1st edition]
Descrizione fisica	1 online resource (1 volume) : illustrations
Disciplina	629.892
Soggetti	Robots Robots - Programming Personal robotics Robots - Control systems
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	Cover -- Title Page -- Copyright and Credits -- www.PacktPub.com -- Contributors -- Table of Contents -- Preface -- Chapter 1: Getting Started with ROS -- Introduction -- Installing ROS on desktop systems -- ROS distribution -- Supported operating systems -- How to do it… -- Configuring Ubuntu repositories -- Setting up the source.list file -- Setting up keys -- ROS Kinetic Installation -- ROS Melodic installation -- Initializing rosdep -- Setting up the environment -- Getting rosinstall -- Build Farm Status -- Installing ROS on a virtual machine -- How to do it… -- Using ROS from a Linux container -- Getting ready -- How to do it… -- Installing Docker -- Adding the Docker repository to APT sources -- Getting and using ROS Docker images -- See also -- Installing ROS on an ARM-based board -- Getting ready -- How to do it… -- Setting system locale -- Setting up sources.list -- Setting up keys -- Installing the ROS packages -- Adding individual packages -- Initializing rosdep -- Environment setup -- Getting rosinstall -- Chapter 2: ROS Architecture and Concepts I -- Introduction -- Exploring the ROS filesystem -- Getting ready -- How to do it… -- There's more… -- Analyzing the ROS computation graph -- Getting ready -- How it works... -- Associating with the ROS community --

Getting ready -- Learning working with ROS -- Getting ready -- How to do it… -- How it works… -- Creating ROS nodes -- Building the ROS node -- Creating ROS messages -- Creating ROS services -- See also -- Understanding the ROS launch file -- Chapter 3: ROS Architecture and Concepts - II -- Introduction -- Understanding the parameter server and dynamic parameters -- Getting ready -- How to do it… -- Understanding the ROS actionlib -- Getting ready -- How to do it… -- Understanding the ROS pluginlib -- Getting ready -- How to do it… -- Creating plugins.
Compiling the plugin library -- Plugins registration -- Making the plugins available to the ROS toolchain -- The Plugin XML File -- Exporting plugins -- Using a plugin -- Running the code -- Understanding the ROS nodelets -- Getting ready -- How to do it… -- Creating a nodelet -- Plugin description -- Building and running nodelets -- There's more… -- Understanding the Gazebo Framework and plugin -- Getting ready -- How to do it… -- Hello World plugin -- Compiling the plugin -- Using a plugin -- Understanding the ROS transform frame (TF) -- Getting ready -- Using view_frames -- Using rqt_tf_tree -- Using tf_echo -- Using RViz and TF -- How to do it… -- Writing a TF broadcaster -- Writing a tf listener -- Compiling and running the TF -- Understanding the ROS Visualization tool (RViz) and its plugins -- Getting ready -- Display panel -- RViz toolbar -- View panel -- Time panel -- Developing an RViz plugin for IMU Display -- How to do it… -- Exporting the plugin -- Building and working with the plugin -- Chapter 4: ROS Visualization and Debugging Tools -- Introduction -- Debugging and profiling ROS nodes -- Getting ready -- How to do it… -- Logging and visualizing ROS messages -- Getting ready -- How it works… -- There's more… -- Inspecting and diagnosing the ROS system -- Getting ready -- How to do it… -- Visualizing and plotting scalar data -- Getting ready -- How to do it… -- There's more… -- Visualizing non-scalar data - 2D/3D images -- Getting ready -- How it works… -- Recording and playing back ROS topics -- Getting ready -- How it works… -- There's more… -- Chapter 5: Accessing Sensors and Actuators through ROS -- Introduction -- Understanding the Arduino-ROS interface -- Getting ready -- How to do it… -- How it works… -- Interfacing 9DoF Razor IMU-Arduino-ROS -- Getting ready -- How to do it… -- How it works… -- Using a GPS system - Ublox.
Getting ready -- How to do it… -- How it works… -- Interfacing servomotors - Dynamixel -- How to do it… -- How it works… -- Using a Laser Rangefinder - Hokuyo -- Getting ready -- How to do it… -- How it works… -- Working with the Kinect sensor to view objects in 3D -- Getting ready -- How to do it… -- How it works… -- Using a joystick or a gamepad in ROS -- How to do it… -- How it works… -- Chapter 6: ROS Modeling and Simulation -- Introduction -- Understanding robot modeling using URDF -- Getting ready -- How it works… -- Understanding robot modeling using xacro -- Getting ready -- How it works… -- Understanding the joint state publisher and the robot state publisher -- Getting ready -- How it works… -- There's more… -- Understanding the Gazebo architecture and interface with ROS -- Getting ready -- How to do it… -- Integrating sensors -- Using map -- Controlling the robot -- Chapter 7: Mobile Robot in ROS -- Introduction -- The navigation stack in ROS -- Getting ready -- How it works… -- Transform Frames -- Sensors -- Odometry -- Base controller -- Map -- Interfacing the mobile robot to the navigation stack -- Getting ready -- How to do it… -- How it works… -- Common parameters -- Global costmap -- Local costmap -- Configuring the planner -- Creating a launch file for the navigation stack -- Getting

ready -- How it works… -- Setting up RViz for the navigation stack - visualization -- Getting ready -- How it works… -- 2D pose estimate -- 2D nav goal -- Static map -- Particle cloud -- Robot's footprint -- Local costmap -- Global costmap -- Global plan -- Local plan -- Planner plan -- Current goal -- There's more… -- Robot localization - Adaptive Monte Carlo Localization (AMCL) -- Getting ready -- How it works… -- Configuring navigation stack parameters with rqt_reconfigure -- How it works….
Autonomous navigation of mobile robots - avoiding obstacles -- Getting ready -- How it works… -- Sending goals -- Getting ready -- How it works… -- Chapter 8: The Robotic Arm in ROS -- Introduction -- Dangerous workspaces -- Repetitive or unpleasant work -- Human-intractable workspaces -- Basic concepts of MoveIt! -- MoveIt -- Motion planning -- Perception -- Grasping -- Getting ready -- DoFs for manipulation -- Grippers -- Motion planning using graphical interfaces -- Getting ready -- MoveIt architecture -- How it works… -- Basic motion planning -- How it works… -- MoveIt! planning scene -- MoveIt! kinematics handling -- MoveIt! collision checking -- There's more… -- Moving the real robot -- Performing motion planning using control programs -- Getting ready -- How to do it… -- Planning a trajectory -- Planning to a joint space goal -- Getting useful information from motion planning -- Executing a trajectory -- Adding perception to motion planning -- Getting ready -- How to do it… -- Adding perception to MoveIt -- How it works… -- There's more… -- See also… -- Grasping action with the robotic arm or manipulator -- Getting ready -- Creating a pick and place task -- How to do it… -- How it works… -- See also -- Chapter 9: Micro Aerial Vehicles in ROS -- Introduction -- Overview of MAV system design -- Getting ready -- A generic mathematical model of an MAV/drone -- Getting ready -- Forces and moments on quadcopter -- Hover and upward and downward motion -- Rotation (yaw) motion -- Linear (pitch and roll) motion -- Simulation of an MAV/drone using RotorS/Gazebo -- Getting ready -- Simulator overview -- How to do it… -- Hovering -- State estimation -- Sensors mounting -- Evaluation -- How it works… -- Developing a custom controller -- There's more… -- See also -- Creating custom sensors -- Autonomous navigation framework for an MAV/Drone -- Getting ready.
Collision avoidance -- Path planning -- How to do it… -- How it works… -- Working with a real MAV/drone - Parrot, Bebop -- Getting ready -- How to do it… -- Executing the trajectory with the real MAV/drone -- How it works… -- Chapter 10: ROS-Industrial (ROS-I) -- Introduction -- Understanding ROS-I packages -- Getting ready -- 3D modeling and simulation of an industrial robot and MoveIt! -- Getting ready -- URDF modeling for an industrial robot -- How to do it… -- Controlling the robot in the simulation -- Working with ROS-I packages - Universal Robots, ABB robot -- Getting ready -- Universal Robots -- ABB Robots -- ROS-I Robot support packages -- Getting ready -- ROS-I Robot client package -- Getting ready -- ROS-I robot driver specification -- Getting ready -- Developing a custom MoveIt! IKFast plugin -- Getting ready -- OpenRAVE installation -- How to do it… -- ROS-I-MTConnect integration -- Getting ready -- How to do it… -- Future of ROS-I - hardware support, capabilities, and applications -- Other Books You May Enjoy -- Index.

Sommario/riassunto

Leverage the power of ROS to build exciting collaborative robots. About This Book Delve into an open source, meta-operating system for your robot Get acquainted with tools and libraries for building and running code on multiple platforms Use Gazebo to model your robot and create a virtual environment Who This Book Is For If you're a researcher or

engineer with an interest in the problems, solutions, and future research issues that you may encounter in the development of robotic applications, this book is for you. Basic knowledge of C++ and Python programming with the GNU/Linux environment is strongly recommended to assist with understanding the key concepts covered in the book. What You Will Learn Explore advanced concepts, such as ROS pluginlib, nodelets, and actionlib Work with ROS visualization, profiling, and debugging tools Gain experience in robot modeling and simulation using Gazebo Understand the ROS Navigation Stack for mobile robots Configure a MoveIt! package for a manipulator robot Develop an autonomous navigation framework for MAV using ORB SLAM and MoveIt Integrate sensors, actuators, and robots into the ROS ecosystem Get acquainted with the ROS-Industrial package with hardware support, capabilities, and applications In Detail This book will leverage the power of ROS with an introduction to its core and advanced concepts through exciting recipes. You will get acquainted with the use of different synchronous and asynchronous communication methods, including messages, services, and actions. You will learn how to use the various debugging and visualization tools used in development and how to interface sensors and actuators with the ROS framework. Firstly, you will get to grips with ROS simulation frameworks, such as Gazebo and RotorS for modeling and simulating any physical robot and virtual environment. You will also cover mobile robotics, micro-aerial vehicles, and robotic arms, which are the leading branches of robotic applications. Robot Operating System Cookbook will also guide you in the development of an autonomous navigation framework for both mobile robots and micro-aerial vehicles. Finally, you will explore ROS-Industrial, an open source project that extends the advanced capabilities of ROS software to manufacturing industries. Style and approach The goal of this book is to provide an integrated overview of the concepts and techniques. ROS is not an operating system in the traditional sense of process management and sched...