

Computer Vision with Maker Tech

Detecting People With a
Raspberry Pi, a Thermal
Camera, and Machine Learning

Fabio Manganiello

Apress®

Computer Vision with Maker Tech: Detecting People With a Raspberry Pi, a Thermal Camera, and Machine Learning

Fabio Manganiello
Amsterdam, The Netherlands

ISBN-13 (pbk): 978-1-4842-6820-9

ISBN-13 (electronic): 978-1-4842-6821-6

<https://doi.org/10.1007/978-1-4842-6821-6>

Copyright © 2021 by Fabio Manganiello

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Aaron Black
Development Editor: James Markham
Coordinating Editor: Jessica Vakili

Distributed to the book trade worldwide by Springer Science+Business Media New York, 1 NY Plazar, New York, NY 10014. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at <http://www.apress.com/bulk-sales>.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/978-1-4842-6820-9. For more detailed information, please visit <http://www.apress.com/source-code>.

Printed on acid-free paper

Table of Contents

About the Author	vii
About the Technical Reviewer	ix
Introduction	xi
Chapters at a Glance	xiii
Chapter 1: Introduction to Machine Learning.....	1
1.1 History.....	3
1.2 Supervised and unsupervised learning.....	9
1.3 Preparing your tools.....	11
1.3.1 Software tools	12
1.3.2 Setting up your environment.....	13
1.4 Linear regression	17
1.4.1 Loading and plotting the dataset.....	18
1.4.2 The idea behind regression	19
1.4.3 Gradient descent	22
1.4.4 Input normalization.....	28
1.4.5 Defining and training the model	30
1.4.6 Evaluating your model	34
1.4.7 Saving and loading your model	38

TABLE OF CONTENTS

- 1.5 Multivariate linear regression 41
 - 1.5.1 Redundant features 45
 - 1.5.2 Principal component analysis..... 47
 - 1.5.3 Training set and test set 52
 - 1.5.4 Loading and visualizing the dataset 53
- 1.6 Polynomial regression..... 60
- 1.7 Normal equation 64
- 1.8 Logistic regression..... 67
 - 1.8.1 Cost function 72
 - 1.8.2 Building the regression model from scratch 75
 - 1.8.3 The TensorFlow way 82
 - 1.8.4 Multiclass regression 84
 - 1.8.5 Non-linear boundaries..... 85
- Chapter 2: Neural Networks 87**
 - 2.1 Back-propagation 96
 - 2.2 Implementation guidelines..... 104
 - 2.2.1 Underfit and overfit..... 106
 - 2.3 Error metrics 111
 - 2.4 Implementing a network to recognize clothing items..... 116
 - 2.5 Convolutional neural networks 134
 - 2.5.1 Convolutional layer 138
 - 2.5.2 Pooling layer 144
 - 2.5.3 Fully connected layer and dropout 145
 - 2.5.4 A network for recognizing images of fruits 146

Chapter 3: Computer Vision on Raspberry Pi	159
3.1 Preparing the hardware and the software	161
3.1.1 Preparing the hardware.....	161
3.1.2 Preparing the operating system	166
3.2 Installing the software dependencies	169
3.3 Capturing the images.....	179
3.4 Labelling the images.....	182
3.5 Training the model	185
3.6 Deploying the model	195
3.6.1 The OpenCV way.....	197
3.6.2 The TensorFlow way	201
3.7 Building your automation flows	205
3.8 Building a small home surveillance system.....	208
3.9 Live training and semi-supervised learning.....	216
3.10 Classifiers as a service	219
Bibliography	227
Index.....	231

About the Author



Fabio Manganiello has more than 15 years of experience in software engineering, many of these spent working on machine learning and distributed systems. In his career he has worked, among others, on natural language processing, an early voice assistant (Voxifera) developed back in 2008, and machine learning applied to intrusion detection systems, using both supervised and unsupervised learning;

and he has developed and released several libraries to make models' design and training easier, from back-propagation neural networks (Neuralpp) to self-organizing maps (fsom), to dataset clustering. He has contributed to the design of machine learning models for anomaly detection and image similarity in his career as a tech lead in Booking.com. In the recent years, he has combined his passion for machine learning with IoT and distributed systems. From self-driving robots to people detection, to anomaly detection, to weather forecasting, he likes to combine the flexibility and affordability of tools such as Raspberry Pi, Arduino, ESP8266, and cheap sensors with the power of machine learning models. He's an active IEEE member and open source enthusiast and has contributed to around 100 open source projects over the years. He is the creator and main contributor of Platypush, an all-purpose platform for automation that aims to connect any device to any service, and also provides TensorFlow and OpenCV integrations for machine learning.

About the Technical Reviewer

Vishwesh Ravi Shrimali graduated from BITS Pilani, where he studied mechanical engineering, in 2018. Since then, he has worked with Big Vision LLC on deep learning and computer vision and was involved in creating official OpenCV AI courses. Currently, he is working at Mercedes-Benz Research and Development India Pvt. Ltd. He has a keen interest in programming and AI and has applied that interest in mechanical engineering projects. He has also written multiple blogs on OpenCV and deep learning on Learn OpenCV, a leading blog on computer vision. He has also coauthored *Machine Learning for OpenCV 4* (second edition) by Packt. When he is not writing blogs or working on projects, he likes to go on long walks or play his acoustic guitar.

Introduction

The field of machine learning has gone through a massive growth in the last few years, thanks to increased computing power, increased funding, and better frameworks that make it easier to build and train classifiers. Machine learning, however, is still considered as a tool for IT giants with plenty of resources, data, and computing power. While it is true that models get better when they can be trained with more data points, and computing power surely plays a role in the ability to train complex models, in this book we'll see that it's already possible to build models trained on data points gathered from a smart home environment (like temperature, humidity, presence, or camera images), and you can already use those models to make your home “smarter.” Such models can be used for predictions even on a Raspberry Pi or similar hardware.

After reading this book, you will

- Know the formal foundations of the main machine learning techniques
- Know how to estimate how accurate a classifier is in making predictions and what to tweak in order to improve its performance
- Know how to cleanse and preprocess your data points to maximize the performance of your models
- Be able to build machine learning models using TensorFlow and the standard Python stack for data analysis (numpy, matplotlib, pandas)

INTRODUCTION

- Be able to set up a Raspberry Pi with a simple network of sensors, cameras, or other data sources that can be used to generate data points fed to simple machine learning models, make predictions on those data points, and easily create, train, and deploy models through web services

Chapters at a Glance

Chapter 1 will go through the theoretical foundations of machine learning. It will cover the most popular approaches to machine learning, the difference between supervised and unsupervised learning, and take a deep dive into regression algorithms (the foundation block for most of today's supervised learning). It will also cover the most popular strategies to visualize, evaluate, and tweak the performance of a model.

Chapter 2 will take a deep dive into neural networks, how they operate and “learn,” and how they use computer vision. It will also cover convolutional neural networks (CNNs), a popular architecture used in most of today's computer vision classification models.

Chapter 3 will provide an overview of the most common tools used by makers in IoT today with a particular focus on the Raspberry Pi. We'll see how to use one of these devices to collect, send, and store data points that can be used to train machine learning models, and to train a simple model to detect the presence of people in a room using a cheap camera, and how to use it to make predictions within a home automation flow—for example, turn the lights on/off when presence is detected or send a notification when presence is detected but we are not home. The chapter will also provide an introduction to some strategies for semi-supervised learning and show how to wrap a web service around TensorFlow to programmatically create, train, and manage models.