

Learning OpenCV 4 Computer Vision with Python 3

Third Edition

Get to grips with tools, techniques, and algorithms for computer vision and machine learning



Packt>

www.packt.com

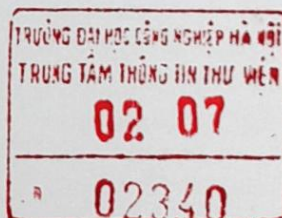
Joseph Howse and Joe Minichino

Learning OpenCV 4 Computer Vision with Python 3

Third Edition

Get to grips with tools, techniques, and algorithms for computer vision and machine learning

Joseph Howse
Joe Minichino



Packt

BIRMINGHAM - MUMBAI

Learning OpenCV 4 Computer Vision with Python 3

Third Edition

Copyright © 2020 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: Richa Tripathi
Acquisition Editor: Alok Dhuri
Content Development Editor: Digvijay Bagul
Senior Editor: Rohit Singh
Technical Editor: Ketan Kamble
Copy Editor: Safis Editing
Project Coordinator: Francy Puthiry
Proofreader: Safis Editing
Indexer: Pratik Shirodkar
Production Coordinator: Shraddha Falebhai

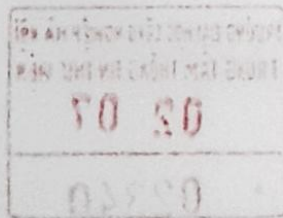
First published: April 2013
Second edition: September 2015
Third edition: February 2020

Production reference: 1190220

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

ISBN 978-1-78953-161-9

www.packt.com





Packt.com

Subscribe to our online digital library for full access to over 7,000 books and videos, as well as industry leading tools to help you plan your personal development and advance your career. For more information, please visit our website.

Why subscribe?

- Spend less time learning and more time coding with practical eBooks and Videos from over 4,000 industry professionals
- Improve your learning with Skill Plans built especially for you
- Get a free eBook or video every month
- Fully searchable for easy access to vital information
- Copy and paste, print, and bookmark content

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.packt.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.packt.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.

Table of Contents

| | |
|---|----|
| Preface | 1 |
| Chapter 1: Setting Up OpenCV | 7 |
| Technical requirements | 8 |
| What's new in OpenCV 4 | 9 |
| Choosing and using the right setup tools | 9 |
| Installation on Windows | 10 |
| Using a ready-made OpenCV package | 11 |
| Building OpenCV from source | 11 |
| Installation on macOS | 15 |
| Using Homebrew with ready-made packages | 15 |
| Using Homebrew with your own custom packages | 17 |
| Installation on Debian, Ubuntu, Linux Mint, and similar systems | 17 |
| Using a ready-made OpenCV package | 18 |
| Building OpenCV from source | 19 |
| Installation on other Unix-like systems | 21 |
| Running samples | 22 |
| Finding documentation, help, and updates | 23 |
| Summary | 24 |
| Chapter 2: Handling Files, Cameras, and GUIs | 25 |
| Technical requirements | 26 |
| Basic I/O scripts | 26 |
| Reading/writing an image file | 26 |
| Converting between an image and raw bytes | 29 |
| Accessing image data with numpy.array | 32 |
| Reading/writing a video file | 34 |
| Capturing camera frames | 36 |
| Displaying an image in a window | 38 |
| Displaying camera frames in a window | 39 |
| Project Cameo (face tracking and image manipulation) | 41 |
| Cameo – an object-oriented design | 42 |
| Abstracting a video stream with managers.CaptureManager | 42 |
| Abstracting a window and keyboard with managers.WindowManager | 48 |
| Applying everything with cameo.Cameo | 49 |
| Summary | 52 |
| Chapter 3: Processing Images with OpenCV | 53 |
| Technical requirements | 53 |
| Converting images between different color models | 54 |
| Light is not paint | 55 |

| | |
|--|-----|
| Exploring the Fourier transform | 56 |
| HPFs and LPFs | 56 |
| Creating modules | 60 |
| Edge detection | 60 |
| Custom kernels – getting convoluted | 62 |
| Modifying the application | 64 |
| Edge detection with Canny | 66 |
| Contour detection | 68 |
| Bounding box, minimum area rectangle, and minimum enclosing circle | 69 |
| Convex contours and the Douglas-Peucker algorithm | 73 |
| Detecting lines, circles, and other shapes | 75 |
| Detecting lines | 76 |
| Detecting circles | 77 |
| Detecting other shapes | 78 |
| Summary | 79 |
| Chapter 4: Depth Estimation and Segmentation | 81 |
| Technical requirements | 82 |
| Creating modules | 82 |
| Capturing frames from a depth camera | 83 |
| Converting 10-bit images to 8-bit | 85 |
| Creating a mask from a disparity map | 88 |
| Modifying the application | 89 |
| Depth estimation with a normal camera | 92 |
| Foreground detection with the GrabCut algorithm | 99 |
| Image segmentation with the Watershed algorithm | 103 |
| Summary | 106 |
| Chapter 5: Detecting and Recognizing Faces | 107 |
| Technical requirements | 108 |
| Conceptualizing Haar cascades | 108 |
| Getting Haar cascade data | 109 |
| Using OpenCV to perform face detection | 110 |
| Performing face detection on a still image | 111 |
| Performing face detection on a video | 113 |
| Performing face recognition | 117 |
| Generating the data for face recognition | 117 |
| Recognizing faces | 119 |
| Loading the training data for face recognition | 120 |
| Performing face recognition with Eigenfaces | 122 |
| Performing face recognition with Fisherfaces | 124 |
| Performing face recognition with LBPH | 125 |
| Discarding results based on the confidence score | 125 |
| Swapping faces in the infrared | 126 |
| Modifying the application's loop | 127 |

| | |
|---|-----|
| Masking a copy operation | 130 |
| Summary | 133 |
| Chapter 6: Retrieving Images and Searching Using Image Descriptors | 135 |
| Technical requirements | 138 |
| Understanding types of feature detection and matching | 138 |
| Defining features | 137 |
| Detecting Harris corners | 137 |
| Detecting DoG features and extracting SIFT descriptors | 140 |
| Anatomy of a keypoint | 143 |
| Detecting Fast Hessian features and extracting SURF descriptors | 143 |
| Using ORB with FAST features and BRIEF descriptors | 145 |
| FAST | 146 |
| BRIEF | 147 |
| Brute-force matching | 147 |
| Matching a logo in two images | 148 |
| Filtering matches using K-Nearest Neighbors and the ratio test | 151 |
| Matching with FLANN | 155 |
| Performing homography with FLANN-based matches | 160 |
| A sample application – tattoo forensics | 164 |
| Saving image descriptors to file | 164 |
| Scanning for matches | 166 |
| Summary | 169 |
| Chapter 7: Building Custom Object Detectors | 171 |
| Technical requirements | 172 |
| Understanding HOG descriptors | 172 |
| Visualizing HOG | 173 |
| Using HOG to describe regions of an image | 175 |
| Understanding NMS | 176 |
| Understanding SVMs | 177 |
| Detecting people with HOG descriptors | 178 |
| Creating and training an object detector | 181 |
| Understanding BoW | 182 |
| Applying BoW to computer vision | 183 |
| k-means clustering | 184 |
| Detecting cars | 184 |
| Combining an SVM with a sliding window | 193 |
| Detecting a car in a scene | 194 |
| Saving and loading a trained SVM | 202 |
| Summary | 203 |
| Chapter 8: Tracking Objects | 205 |
| Technical requirements | 206 |
| Detecting moving objects with background subtraction | 206 |

| | |
|--|-----|
| Implementing a basic background subtractor | 208 |
| Using a MOG background subtractor | 211 |
| Using a KNN background subtractor | 216 |
| Using GMG and other background subtractors | 218 |
| Tracking colorful objects using MeanShift and CamShift | 221 |
| Planning our MeanShift sample | 222 |
| Calculating and back-projecting color histograms | 223 |
| Understanding the parameters of cv2.calcHist | 226 |
| Understanding the parameters of cv2.calcBackProject | 227 |
| Implementing the MeanShift example | 228 |
| Using CamShift | 231 |
| Finding trends in motion using the Kalman filter | 232 |
| Understanding the predict and update phases | 233 |
| Tracking a mouse cursor | 234 |
| Tracking pedestrians | 237 |
| Planning the flow of the application | 237 |
| Comparing the object-oriented and functional paradigms | 238 |
| Implementing the Pedestrian class | 240 |
| Implementing the main function | 242 |
| Considering the next steps | 246 |
| Summary | 247 |
| Chapter 9: Camera Models and Augmented Reality | 249 |
| Technical requirements | 250 |
| Understanding 3D image tracking and augmented reality | 250 |
| Understanding camera and lens parameters | 253 |
| Understanding cv2.solvePnPRansac | 258 |
| Implementing the demo application | 261 |
| Importing modules | 261 |
| Performing grayscale conversion | 262 |
| Performing 2D-to-3D spatial conversions | 263 |
| Implementing the application class | 265 |
| Initializing the tracker | 266 |
| Implementing the main loop | 274 |
| Tracking the image in 3D | 275 |
| Initializing and applying the Kalman filter | 279 |
| Drawing the tracking results and masking the tracked object | 283 |
| Running and testing the application | 286 |
| Improving the 3D tracking algorithm | 291 |
| Summary | 292 |
| Chapter 10: Introduction to Neural Networks with OpenCV | 293 |
| Technical requirements | 294 |
| Understanding ANNs | 295 |
| Understanding neurons and perceptrons | 297 |
| Understanding the layers of a neural network | 298 |

| | |
|--|-----|
| Choosing the size of the input layer | 299 |
| Choosing the size of the output layer | 299 |
| Choosing the size of the hidden layer | 299 |
| Training a basic ANN in OpenCV | 300 |
| Training an ANN classifier in multiple epochs | 303 |
| Recognizing handwritten digits with an ANN | 308 |
| Understanding the MNIST database of handwritten digits | 308 |
| Choosing training parameters for the MNIST database | 310 |
| Implementing a module to train the ANN | 310 |
| Implementing a minimal test module | 315 |
| Implementing the main module | 315 |
| Trying to improve the ANN's training | 321 |
| Finding other potential applications | 323 |
| Using DNNs from other frameworks in OpenCV | 324 |
| Detecting and classifying objects with third-party DNNs | 325 |
| Detecting and classifying faces with third-party DNNs | 329 |
| Summary | 336 |
| Appendix A: Bending Color Space with the Curves Filter | 339 |
| Other Book You May Enjoy | 349 |
| Index | 351 |

Preface

Now in its third edition, this is the original book on OpenCV's Python bindings. Readers will learn a great range of techniques and algorithms, from the classics to the state-of-the-art, and from geometry to machine learning. All of this is in aid of solving practical computer vision problems in well-built applications. Using OpenCV 4 and Python 3, we adopt an approach that is accessible to computer vision novices, yet also informative for experts who want to expand and update their skills.

We start with an introduction to OpenCV 4 and explain how to set it up with Python 3 on various platforms. Next, you'll learn how to perform basic operations such as reading, writing, manipulating, and displaying still images, videos, and camera feeds. You'll learn about image processing and video analysis, along with depth estimation and segmentation, and you'll gain practice by building a simple GUI application. Next, you'll tackle two popular problems: face detection and face recognition.

As we advance, we'll explore concepts of object classification and machine learning, enabling you to create and use object detectors and classifiers, and even track objects in movies or video camera feeds. Then, we'll extend our work into 3D tracking and augmented reality. Finally, we'll learn about **artificial neural networks (ANNs)** and **deep neural networks (DNNs)** as we develop applications to recognize handwritten digits, and to classify a person's gender and age.

By the end of this book, you will have acquired the right knowledge and skills to embark on your own real-world computer vision projects.

Who this book is for

This book is intended for people interested in learning computer vision, machine learning, and OpenCV in the context of practical real-world applications. The book will appeal to computer vision novices as well as experts who want to get up to date with OpenCV 4 and Python 3. Readers should be familiar with basic Python programming, but no prior knowledge of image processing, computer vision, or machine learning is required.