Python Image Processing Cookbook

Book · April 2020

CITATIONS
0

READS
2,608

1 author:

Sandipan Dey
University of Maryland, Baltimore County
13 PUBLICATIONS **115** CITATIONS

SEE PROFILE

Python Image Processing Cookbook

Over 60 recipes to help you perform complex image processing and computer vision tasks with ease

Sandipan Dey

Packt
www.packt.com

Python Image Processing Cookbook

Over 60 recipes to help you perform complex image processing and computer vision tasks with ease

Sandipan Dey

Packt

BIRMINGHAM - MUMBAI

Python Image Processing Cookbook

Commissioning Editor: Pravin Dhandre
Acquisition Editor: Devika Battike
Content Development Editor: Athikho Sapuni Rishana
Senior Editor: Sofi Rogers
Technical Editor: Manikandan Kurup
Copy Editor: Safis Editing
Project Coordinator: Aishwarya Mohan
Proofreader: Safis Editing
Indexer: Pratik Shirodkar
Production Designer: Joshua Misquitta

First published: April 2020

Production reference: 1170420

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

ISBN 978-1-78953-714-7

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.

I dedicate this book to my beloved parents and all the teachers in my life from all the schools:

- Ballygunge Government High School, Kolkata, India
- Jadavpur University, Kolkata, India
- University of Maryland, Baltimore County, USA
- MOOC platforms like Coursera and edX

Contributors

About the author

Sandipan Dey is a data scientist with a wide range of interests, covering topics such as machine learning, deep learning, image processing, and computer vision. He has worked in numerous data science fields, working with recommender systems, predictive models for the events industry, sensor localization models, sentiment analysis, and device prognostics. He earned his master's degree in computer science from the University of Maryland, Baltimore County, and has published in a few IEEE data mining conferences and journals. He has earned certifications from 100+ MOOCs on data science, machine learning, deep learning, image processing, and related courses. He is a regular blogger (sandipanweb) and is a machine learning education enthusiast.

About the reviewers

Srinjoy Ganguly holds a master's in Artificial Intelligence from the University of Southampton, UK and has over 4 years of experience in AI. He completed his bachelor's in Electronics and Communications from GGSIPU, Delhi, and gained a thorough knowledge of signal processing algorithms and neural networks. He has actively pursued the theory and principles of machine learning, deep learning, computer vision, and reinforcement learning. His research interests include Bayesian machine learning, Bayesian deep learning, and quantum machine learning. He is also an active contributor to Julia language. He strongly believes that this book will be an extremely valuable asset into the hands of beginners in the field of AI and computer vision.

Shankar Jha is a software engineer who has helped a lot of beginners get to grips with Python through his articles and talks. He writes about Python, Django, tackling climate change, and poverty alleviation. He lives by the philosophy of Richard Feynman and Swami Vivekananda. His latest endeavor is to help facilitate the use of environment-friendly products. He believes this book will give you enough hands-on experience to jump start your career in computer vision.

Packt is searching for authors like you

If you're interested in becoming an author for Packt, please visit `authors.packtpub.com` and apply today. We have worked with thousands of developers and tech professionals, just like you, to help them share their insight with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Table of Contents

Preface

With advancements in wireless devices and mobile technology, there's an increasing demand for digital image processing skills to extract useful information from the ever-growing volume of images. This book provides comprehensive coverage of tools and algorithms, along with guiding you through analysis and visualization for image processing.

With the help of over 60 cutting-edge recipes, you'll address common challenges in image processing and learn how to perform complex tasks such as image detection, image segmentation, and image reconstruction using large hybrid datasets. Dedicated sections will also take you through implementing various image enhancement and image restoration techniques such as cartooning, gradient blending, and sparse dictionary learning. As you advance, you'll get to grips with face morphing and image segmentation techniques. With an emphasis on practical solutions, this book will help you apply deep learning-based techniques such as transfer learning and fine-tuning to solve real-world problems.

By the end of this Python book, you'll be proficient in applying image processing techniques effectively to leverage the capabilities of the Python ecosystem.

Who this book is for

This Python cookbook is for image processing engineers, computer vision engineers, software developers, machine learning engineers, or anyone who wants to become well-versed with image processing techniques and methods using a recipe-based approach. Although no image processing knowledge is expected, prior Python coding experience is necessary to understand key concepts covered in the book.

What this book covers

`Chapter 1`, *Image Manipulation and Transformation*, is where you will learn how to use different Python libraries (NumPy, SciPy, scikit-image, OpenCV, and Matplotlib) for image manipulation and transformation. You will learn how to write Python code to do point transforms (log/gamma transform, Gotham filter, colorspace transformation, and increasing brightness/contrast) and geometric transforms (swirl transform, perspective transform, and homography).

`Chapter 2`, *Image Enhancement*, is where you will learn how to use different Python libraries (NumPy, SciPy, scikit-image, OpenCV, PyWavelet, and MedPy) to denoise images (using linear/nonlinear filters, **Fast Fourier transform** (**FFT**), and autoencoders). You'll learn how to implement image enhancement techniques such as histogram equalization/matching, sketching/cartoonizing, pyramid blending/gradient blending, and edge detection with zero crossing.

`Chapter 3`, *Image Restoration*, is where you will learn how to implement image restoration (using NumPy, scikit-image, OpenCV, and scikit-learn) with deconvolution (inverse/weiner/LMS) filters. You'll learn how to implement image restoration with inpainting, variational methods, and sparse dictionary learning. You'll also learn how to implement steganography/steganalysis techniques with pysteg.

`Chapter 4`, *Binary Image Processing*, is where you will learn how to use different Python libraries (NumPy, SciPy, scikit-image, and OpenCV) for binary image processing (with mathematical morphology). You'll learn how to implement morphological operators, filters, and pattern matching and how to apply them in segmentation, fingerprint enhancement, counting objects, and blob separation.

`Chapter 5`, *Image Registration*, is where you will learn how to use different Python libraries (NumPy, scikit-image, OpenCV, and PyStasm) for image matching/registration/stitching. You'll learn how to implement image registration techniques with warping/feature (SIFT/SURF/ORB)-based methods and the RANSAC algorithm. You'll also learn how to implement panorama image creation, and face morphing, as well as how to implement a basic image search engine.

`Chapter 6`, *Image Segmentation*, is where you will learn how to use different Python libraries (NumPy, scikit-image, OpenCV, SimpleITK, and DeepLab) for image segmentation. You'll learn how to implement image segmentation techniques with graph-based methods/clustering methods, super-pixelation, and machine learning algorithms. You'll also learn how to implement semantic segmentation with DeepLab.

`Chapter 7`, *Image Classification*, is where you will learn how to use different Python libraries (scikit-learn, OpenCV, TensorFlow, Keras, and PyTorch) for image classification. You'll learn how to implement deep learning-based techniques such as transfer learning/fine-tuning. You'll learn how to implement panorama image creation and face morphing. You'll also learn how to implement deep learning-based classification techniques for hand gestures and traffic signals.

`Chapter 8`, *Object Detection in Images*, is where you will learn how to use different Python libraries (scikit-learn, OpenCV, TensorFlow, Keras, and PyTorch) for object detection in images. You'll learn how to implement classical machine learning (HOG/SVM) techniques as well as deep learning models to detect objects. You'll also learn how to implement barcode detection and text detection from images.

`Chapter 9`, *Face Detection and Recognition*, is where you will learn how to use different Python libraries (scikit-learn, OpenCV, dlib, TensorFlow, Keras, PyTorch, DeepFace, and FaceNet) for face detection in images. You'll also learn how to implement facial keypoint recognition and facial/emotion/gender recognition with deep learning.

To get the most out of this book

Basic knowledge of Python and image processing is required to understand and run the code, along with access to a few online image datasets and the book's GitHub link.

Python 3.5+ (Python 3.7.4 was used to test the code) is needed with Anaconda preferably installed for the Windows users, along with Jupyter (to view/run notebooks).

All the code was tested on Windows 10 (Pro) with 32 GB RAM and an Intel i7-series processor. However, the code should require little/no change to be run on Linux.

You will need to install all the required Python packages using pip3.

Access to a GPU is recommended to run the recipes involving training with deep learning (that is, training that involves libraries such as TensorFlow, Keras, and PyTorch) much faster. The code that is best run with a GPU was tested on an Ubuntu 16.04 machine with an Nvidia Tesla K80 GPU (with CUDA 10.1).

A basic math background is also needed to understand the concepts in the book.

Software/hardware covered in the book	**OS requirements**
Python 3.7.4.	Windows 10.
Anaconda version 2019.10 (py37_0).	Windows 10.
For the GPU, you will need an NVIDIA graphics card or access to an AWS GPU instance (`https://docs.aws.amazon.com/dlami/latest/devguide/gpu.html`) or Google Colab (`https://colab.research.google.com/`).	Windows 10/Linux (Ubuntu 16).

If you are using the digital version of this book, we advise you to type the code yourself or access the code via the GitHub repository (link available in the next section). Doing so will help you avoid any potential errors related to the copying and pasting of code.

To access the notebooks and images, clone the repository from this URL: `https://github.com/PacktPublishing/Python-Image-Processing-Cookbook`.

Install Python 3.7 and the necessary libraries as and when required. Install Anaconda/Jupyter and open the notebooks for each chapter. Run the code for each recipe. Follow the instructions for each recipe for any additional steps (for instance, you may need to download a pre-trained model or an image dataset).

Some additional exercises are provided for most of the recipes in a *There's more...* section to test your understanding. Perform them independently and have fun!

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/Python-Image-Processing-Cookbook`. 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: `https://static.packt-cdn.com/downloads/9781789537147_ColorImages.pdf`.

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: "Implement a `bilinear_interpolate()` function, which interpolates over every image channel."

A block of code is set as follows:

```
def get_grid_coordinates(points):
  xmin, xmax = np.min(points[:, 0]), np.max(points[:, 0]) + 1
  ymin, ymax = np.min(points[:, 1]), np.max(points[:, 1]) + 1
  return np.asarray([(x, y) for y in range(ymin, ymax)
         for x in range(xmin, xmax)], np.uint32)
```

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

```
$ pip install mtcnn
```

Bold: Indicates a new term, an important word, or words that you see onscreen. For example, words in menus or dialog boxes appear in the text like this. Here is an example: "**Face alignment** is a data normalization process—an essential preprocessing step for many facial recognition algorithms."

Warnings or important notes appear like this.

Tips and tricks appear like this.

Sections

In this book, you will find several headings that appear frequently (*Getting ready*, *How to do it...*, *How it works...*, *There's more...*, and *See also*).

To give clear instructions on how to complete a recipe, use these sections as follows:

Getting ready

This section tells you what to expect in the recipe and describes how to set up any software or any preliminary settings required for the recipe.

How to do it…

This section contains the steps required to follow the recipe.

How it works…

This section usually consists of a detailed explanation of what happened in the previous section.

There's more…

This section consists of additional information about the recipe in order to make you more knowledgeable about the recipe.

See also

This section provides helpful links to other useful information for the recipe.

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`.