

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/340844634>

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](#)

Some of the authors of this publication are also working on these related projects:



Temporal Neighborhood discovery using Discretization Methods [View project](#)



Data Hiding Techniques using different number systems [View project](#)

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

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 author(s), 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: 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	1
Chapter 1: Image Manipulation and Transformation	9
Technical requirements	9
Transforming color space (RGB → Lab)	10
Getting ready	10
How to do it...	10
Converting RGB image into grayscale by setting the Lab space color channels to zero	11
Changing the brightness of the image by varying the luminosity channel	12
How it works...	12
There's more...	13
Applying affine transformation	13
Getting ready	14
How to do it...	14
How it works...	16
There's more...	17
Applying perspective transformation and homography	17
Getting ready	18
How to do it...	18
How it works...	21
There's more...	21
See also	21
Creating pencil sketches from images	22
Getting ready	22
How to do it...	22
How it works...	25
There's more...	25
See also	26
Creating cartoonish images	26
Getting ready	27
How to do it...	27
How it works...	28
There's more...	29
See also	29
Simulating light art/long exposure	30
Getting ready	30
How to do it...	31
How it works...	33
There's more...	33

Extended depth of field with mahotas	33
See also	35
Object detection using color in HSV	35
Getting ready	36
How to do it...	36
How it works...	37
See also	39
Chapter 2: Image Enhancement	41
 Applying filters to denoise different types of noise in an image	42
Getting ready	42
How to do it...	42
How it works...	44
There's more...	46
 Image denoising with a denoising autoencoder	46
Getting ready	46
How to do it...	47
How it works...	50
There's more...	53
See also	53
 Image denoising with PCA/DFT/DWT	53
Getting ready	53
How to do it...	54
How it works...	55
There's more...	57
See also	57
 Image denoising with anisotropic diffusion	57
Getting ready	57
How to do it...	58
How it works...	60
There's more...	61
See also	61
 Improving image contrast with histogram equalization	61
Getting ready	62
How to do it...	62
How it works...	63
There's more...	65
 Implementing histogram matching	65
Getting ready	66
How to do it...	66
How it works...	67
There's more...	69
See also	69
 Performing gradient blending	70
Getting ready	71

How to do it...	71
How it works...	72
Edge detection with Canny, LoG/zero-crossing, and wavelets	72
Getting ready	73
How to do it...	73
Canny/hysteresis thresholding	74
LoG/zero-crossing	75
Wavelets	77
How it works...	79
There's more...	81
See also	81
Chapter 3: Image Restoration	83
Restoring an image with the Wiener filter	84
Getting ready	84
How to do it...	84
How it works...	87
See also	88
Restoring an image with the constrained least squares filter	89
Getting ready	89
How to do it...	89
How it works...	93
There's more...	94
See also	94
Image restoration with a Markov random field	95
Getting ready	95
How to do it...	96
How it works...	98
See also	99
Image inpainting	99
Getting ready	100
How to do it...	100
How it works...	101
There's more...	102
Image inpainting with convex optimization	103
See also	104
Image completion with inpainting using deep learning	105
Getting ready	105
How to do it...	105
There's more...	108
See also	109
Image restoration with dictionary learning	109
Getting ready	110
How to do it ...	110
There's more...	113
Online dictionary learning	113

See also	115
Compressing an image using wavelets	115
Getting ready	115
How to do it...	116
How it works...	117
See also	118
Using steganography and steganalysis	118
Getting ready	119
How to do it...	119
How it works...	122
There's more...	123
See also	123
Chapter 4: Binary Image Processing	125
 Applying morphological operators to a binary image	126
Getting ready	127
How to do it...	127
How it works...	130
There's more...	132
See also	134
 Applying Morphological filters	134
Getting ready	134
How to do it...	135
Computing the Euler number, eccentricity, and center of mass with mahotas/scikit-image	135
Morphological image filters with mahotas	136
Binary image filters with SimpleITK	138
Dilation by reconstruction with skimage	141
How it works...	143
There's more...	144
See also	147
 Morphological pattern matching	147
Getting ready	147
How to do it...	148
How it works...	151
There's more...	151
See also	152
 Segmenting images with morphology	152
Getting ready	152
How to do it...	153
Morphological watershed	153
Blob detection with morphological watershed	156
How it works...	158
There's more...	159
Blob detection with LOG scale-space	159
See also	160

Counting objects	160
Getting ready	161
How to do it...	161
Blob separation and detection with erosion	161
Object counting with closing and opening	163
How it works...	164
There's more...	165
See also	167
Chapter 5: Image Registration	169
Medical image registration with SimpleITK	170
Getting ready	171
How to do it...	171
How it works...	174
There's more	174
See also	176
Image alignment with ECC algorithm and warping	177
Getting ready	177
How to do it...	178
How it works...	180
There is more	182
See also	182
Face alignment with dlib	182
Getting ready	183
How to do it...	183
How it works...	186
There is more	187
See also	188
Robust matching and homography with the RANSAC algorithm	189
Getting ready	190
How to do it...	190
How it works...	192
See also	194
Image mosaicing (panorama)	194
Getting ready	195
How to do it...	195
Panorama with OpenCV-Python	201
How it works...	202
There is more	203
See also	203
Face morphing	204
Getting ready	205
How to do it...	206
How it works	211
There is more	212
See also	213

Implementing an image search engine	213
Getting ready	214
How to do it...	214
Finding similarity between an image and a set of images with SIFT	214
Steps to implement a simple image search engine	218
There is more	223
See also	223
Chapter 6: Image Segmentation	225
 Thresholding with Otsu and Riddler–Calvard	226
Getting ready	227
How to do it...	227
How it works...	228
There's more...	229
See also	231
 Image segmentation with self-organizing maps	231
Getting ready	232
How to do it...	233
How it works...	235
There's more...	235
Clustering handwritten digit images with SOM	236
See also	238
 RandomWalk segmentation with scikit-image	239
Getting ready	240
How to do it...	240
How it works...	242
There's more...	243
See also	244
 Human skin segmentation with the GMM-EM algorithm	244
Getting ready	245
How to do it...	245
How it works...	249
See also	250
 Medical image segmentation	250
Getting ready	251
How to do it...	251
Segmentation with GMM-EM	251
Brain tumor segmentation using deep learning	253
Segmentation with watershed	256
How it works...	258
There's more...	258
See also	259
 Deep semantic segmentation	259
Getting ready	260
How to do it...	260
Semantic segmentation with DeepLabV3	260

Semantic segmentation with FCN	264
See also	267
Deep instance segmentation	267
Getting ready	268
How to do it...	269
How it works...	271
See also	272
Chapter 7: Image Classification	273
Classifying images with scikit-learn (HOG and logistic regression)	275
Getting ready	275
How to do it...	276
How it works...	280
There's more...	281
See also	282
Classifying textures with Gabor filter banks	282
Getting ready	283
How to do it...	283
How it works...	287
There's more...	288
See also	288
Classifying images with VGG19/Inception V3/MobileNet/ResNet101 (with PyTorch)	289
Getting ready	290
How to do it...	290
How it works...	292
There's more...	294
See also	294
Fine-tuning (with transfer learning) for image classification	295
Getting ready	296
How to do it...	297
How it works...	302
There's more...	304
See also	304
Classifying traffic signs using a deep learning model (with PyTorch)	305
Getting ready	305
How to do it...	306
How it works...	314
There's more...	315
See also	315
Estimating a human pose using a deep learning model	316
Getting ready	317
How to do it...	318
How it works...	322
See also	322

Chapter 8: Object Detection in Images	323
Object detection with HOG/SVM	324
Getting started	326
How to do it...	326
How it works...	329
There's more...	330
See also	330
Object detection with Yolo V3	331
Getting started	332
How to do it...	333
How it works...	336
There's more...	336
See also	336
Object detection with Faster R-CNN	337
Getting started	338
How to do it...	339
How it works...	341
There's more...	342
See also	342
Object detection with Mask R-CNN	343
Getting started	344
How to do it...	344
How it works...	348
There's more...	348
See also	348
Multiple object tracking with Python-OpenCV	349
Getting started	349
How to do it...	350
How it works...	352
There's more...	352
See also	352
Text detection/recognition in images with EAST/Tesseract	353
Getting started	354
How to do it...	355
How it works...	358
See also	358
Face detection with Viola-Jones/Haar-like features	359
Getting ready	359
How to do it...	360
How it works...	361
There's more...	362
See also	362
Chapter 9: Face Recognition, Image Captioning, and More	363
Face recognition using FaceNet	364

Table of Contents

Getting ready	365
How to do it...	366
How it works...	370
See also	372
Age, gender, and emotion recognition using deep learning models	372
Getting ready	373
How to do it...	373
There's more...	375
See also	375
Image colorization with deep learning	376
Getting ready	376
How to do it...	377
See also	378
Automatic image captioning with a CNN and an LSTM	379
Getting ready	380
How to do it...	381
How it works...	384
See also	384
Image generation with a GAN	384
Getting ready	385
How to do it...	386
How it works...	392
There's more...	392
See also	392
Using a variational autoencoder to reconstruct and generate images	393
Getting ready	394
How to do it...	394
There's more...	400
See also	401
Using a restricted Boltzmann machine to reconstruct Bangla MNIST images	402
Getting ready	403
How to do it...	403
See also	409
Other Books You May Enjoy	411
Index	415

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
- Ziipeg/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.

