

Table of Contents

Preface	1
Chapter 1: Getting Started with Image Processing	7
What is image processing and some applications	8
What is an image and how it is stored on a computer	8
What is image processing?	10
Some applications of image processing	11
The image processing pipeline	11
Setting up different image processing libraries in Python	13
Installing pip	14
Installing some image processing libraries in Python	14
Installing the Anaconda distribution	15
Installing Jupyter Notebook	16
Image I/O and display with Python	16
Reading, saving, and displaying an image using PIL	17
Providing the correct path to the images on the disk	18
Reading, saving, and displaying an image using Matplotlib	18
Interpolating while displaying with matplotlib imshow	20
Reading, saving, and displaying an image using scikit-image	21
Using scikit-image's astronaut dataset	22
Reading and displaying multiple images at once	23
Reading, saving, and displaying an image using scipy misc	23
Using scipy.misc's face dataset	23
Dealing with different image types and file formats and performing basic image manipulations	25
Dealing with different image types and file formats	25
File formats	25
Converting from one file format to another	26
Image types (modes)	26
Converting from one image mode into another	27
Some color spaces (channels)	28
Converting from one color space into another	28
Data structures to store images	29
Converting image data structures	30
Basic image manipulations	30
Image manipulations with numpy array slicing	31
Simple image morphing - α -blending of two images using cross-dissolving	32
Image manipulations with PIL	33
Cropping an image	34
Resizing an image	34
Negating an image	36
Converting an image into grayscale	36
Some gray-level transformations	37

Some geometric transformations	38
Changing pixel values of an image	40
Drawing on an image	41
Drawing text on an image	42
Creating a thumbnail	43
Computing the basic statistics of an image	43
Plotting the histograms of pixel values for the RGB channels of an image	43
Separating the RGB channels of an image	44
Combining multiple channels of an image	45
α -blending two images	45
Superimposing two images	46
Adding two images	47
Computing the difference between two images	48
Subtracting two images and superimposing two image negatives	49
Image manipulations with scikit-image	49
Inverse warping and geometric transformation using the warp() function	50
Applying the swirl transform	51
Adding random Gaussian noise to images	51
Computing the cumulative distribution function of an image	52
Image manipulation with Matplotlib	52
Drawing contour lines for an image	53
Image manipulation with the scipy.misc and scipy.ndimage modules	53
Summary	53
Questions	54
Further readings	56
Chapter 2: Sampling, Fourier Transform, and Convolution	58
Image formation – sampling and quantization	59
Sampling	59
Up-sampling	60
Up-sampling and interpolation	61
Down-sampling	64
Down-sampling and anti-aliasing	66
Quantization	69
Quantizing with PIL	69
Discrete Fourier Transform	71
Why do we need the DFT?	72
The Fast Fourier Transform algorithm to compute the DFT	73
The FFT with the scipy.fftpack module	73
Plotting the frequency spectrum	74
The FFT with the numpy.fft module	75
Computing the magnitude and phase of a DFT	75
Understanding convolution	79
Why convolve an image?	80
Convolution with SciPy signal's convolve2d	80
Applying convolution to a grayscale image	81
Convolution modes, pad values, and boundary conditions	82
Applying convolution to a color (RGB) image	82
Convolution with SciPy ndimage.convolve	84
Correlation versus convolution	86
Template matching with cross-correlation between the image and template	89

Summary	91
Questions	91
Further reading and reference	92
Chapter 3: Convolution and Frequency Domain Filtering	93
Convolution theorem and frequency domain Gaussian blur	93
Application of the convolution theorem	94
Frequency domain Gaussian blur filter with numpy fft	95
Gaussian kernel in the frequency domain	96
Frequency domain Gaussian blur filter with scipy signal fftconvolve()	99
Comparing the runtimes of SciPy convolve() and fftconvolve() with the Gaussian blur kernel	101
Filtering in the frequency domain (HPF, LPF, BPF, and notch filters)	103
What is a filter?	103
High-Pass Filter (HPF)	105
How SNR changes with frequency cut-off	110
Low-pass filter (LPF)	111
LPF with scipy ndimage and numpy fft	111
LPF with fourier_gaussian	111
LPF with scipy fftpack	113
How SNR changes with frequency cutoff	117
Band-pass filter (BPF) with DoG	117
Band-stop (notch) filter	118
Using a notch filter to remove periodic noise from images	119
Image restoration	121
Deconvolution and inverse filtering with FFT	122
Image deconvolution with the Wiener filter	126
Image denoising with FFT	127
Filter in FFT	129
Reconstructing the final image	130
Summary	131
Questions	132
Further reading	132
Chapter 4: Image Enhancement	133
Point-wise intensity transformations – pixel transformation	134
Log transform	135
Power-law transform	137
Contrast stretching	139
Using PIL as a point operation	140
Using the PIL ImageEnhance module	142
Thresholding	143
With a fixed threshold	144
Half-toning	146
Floyd-Steinberg dithering with error diffusion	147
Histogram processing – histogram equalization and matching	148
Contrast stretching and histogram equalization with scikit-image	149
Histogram matching	154

Histogram matching for an RGB image	156
Linear noise smoothing	158
Smoothing with PIL	158
Smoothing with ImageFilter.BLUR	158
Smoothing by averaging with the box blur kernel	160
Smoothing with the Gaussian blur filter	161
Comparing smoothing with box and Gaussian kernels using SciPy ndimage	162
Nonlinear noise smoothing	163
Smoothing with PIL	164
Using the median filter	164
Using max and min filter	166
Smoothing (denoising) with scikit-image	166
Using the bilateral filter	167
Using non-local means	169
Smoothing with scipy ndimage	171
Summary	172
Questions	173
Further reading	173
Chapter 5: Image Enhancement Using Derivatives	174
Image derivatives – Gradient and Laplacian	175
Derivatives and gradients	175
Displaying the magnitude and the gradient on the same image	178
Laplacian	179
Some notes about the Laplacian	180
Effects of noise on gradient computation	181
Sharpening and unsharp masking	182
Sharpening with Laplacian	183
Unsharp masking	184
With the SciPy ndimage module	184
Edge detection using derivatives and filters (Sobel, Canny, and so on)	187
With gradient magnitude computed using the partial derivatives	187
The non-maximum suppression algorithm	189
Sobel edge detector with scikit-image	190
Different edge detectors with scikit-image – Prewitt, Roberts, Sobel, Scharr, and Laplace	193
The Canny edge detector with scikit-image	195
The LoG and DoG filters	197
The LoG filter with the SciPy ndimage module	201
Edge detection with the LoG filter	202
Edge detection with the Marr and Hildreth's algorithm using the zero-crossing computation	203
Finding and enhancing edges with PIL	204
Image pyramids (Gaussian and Laplacian) – blending images	206
A Gaussian pyramid with scikit-image transform pyramid module	207
A Laplacian pyramid with scikit-image transform's pyramid module	209

Constructing the Gaussian Pyramid with scikit-image transform module's reduce function, and the Laplacian Pyramid from the Gaussian Pyramid and the expand function	211
Reconstructing an image only from its Laplacian pyramid	215
Blending images with pyramids	217
Summary	220
Questions	220
References/further reading	221
Chapter 6: Morphological Image Processing	223
The scikit-image morphology module	224
Binary operations	224
Erosion	224
Dilation	225
Opening and closing	226
Skeletonizing	228
Computing the convex hull	228
Removing small objects	230
White and black top-hats	232
Extracting the boundary	232
Fingerprint cleaning with opening and closing	233
Grayscale operations	235
The scikit-image filter.rank module	237
Morphological contrast enhancement	237
Noise removal with the median filter	238
Computing the local entropy	239
The SciPy ndimage morphology module	241
Filling holes in binary objects	241
Using opening and closing to remove noise	243
Computing the morphological Beucher gradient	245
Computing the morphological Laplace	246
Summary	247
Questions	248
Further reading	249
Chapter 7: Extracting Image Features and Descriptors	250
Feature detectors versus descriptors	250
Harris Corner Detector	252
With scikit-image	252
With sub-pixel accuracy	253
An application – image matching	255
Robust image matching using the RANSAC algorithm and Harris Corner features	255
Blob detectors with LoG, DoG and DoH	259
Laplacian of Gaussian (LoG)	260
Difference of Gaussian (DoG)	260
Determinant of Hessian (DoH)	260

Histogram of Oriented Gradients	262
Algorithm to compute HOG descriptors	262
Compute HOG descriptors with scikit-image	262
Scale-invariant feature transform	263
Algorithm to compute SIFT descriptors	264
With opencv and opencv-contrib	264
Application – matching images with BRIEF, SIFT, and ORB	266
Matching images with BRIEF binary descriptors with scikit-image	266
Matching with ORB feature detector and binary descriptor using scikit-image	267
Matching with ORB features using Brute-Force matching with python-opencv	270
Brute-force matching with SIFT descriptors and ratio test with OpenCV	272
Haar-like features	274
Haar-like feature descriptor with scikit-image	274
Application – face detection with Haar-like features	276
Face/eye detection with OpenCV using pre-trained classifiers with Haar-cascade features	276
Summary	279
Questions	279
Further reading	279
Chapter 8: Image Segmentation	281
What is image segmentation?	281
Hough transform – detecting lines and circles	282
Thresholding and Otsu's segmentation	286
Edges-based/region-based segmentation	288
Edge-based segmentation	289
Region-based segmentation	291
Morphological watershed algorithm	291
Felzenszwalb, SLIC, QuickShift, and Compact Watershed algorithms	295
Felzenszwalb's efficient graph-based image segmentation	295
SLIC	299
RAG merging	300
QuickShift	301
Compact Watershed	303
Region growing with SimpleITK	304
Active contours, morphological snakes, and GrabCut algorithms	309
Active contours	309
Morphological snakes	311
GrabCut with OpenCV	313
Summary	317
Questions	318
Further reading	318
Chapter 9: Classical Machine Learning Methods in Image Processing	319
Supervised versus unsupervised learning	320

Unsupervised machine learning – clustering, PCA, and eigenfaces	321
K-means clustering for image segmentation with color quantization	321
Spectral clustering for image segmentation	324
PCA and eigenfaces	327
Dimension reduction and visualization with PCA	327
2D projection and visualization	328
Eigenfaces with PCA	329
Eigenfaces	333
Reconstruction	333
Eigen decomposition	335
Supervised machine learning – image classification	336
Downloading the MNIST (handwritten digits) dataset	337
Visualizing the dataset	338
Training kNN, Gaussian Bayes, and SVM models to classify MNIST	340
k-nearest neighbors (KNN) classifier	340
Squared Euclidean distance	340
Computing the nearest neighbors	341
Evaluating the performance of the classifier	342
Bayes classifier (Gaussian generative model)	343
Training the generative model – computing the MLE of the Gaussian parameters	344
Computing the posterior probabilities to make predictions on test data and model evaluation	346
SVM classifier	346
Supervised machine learning – object detection	348
Face detection with Haar-like features and cascade classifiers with AdaBoost – Viola-Jones	349
Face classification using the Haar-like feature descriptor	350
Finding the most important Haar-like features for face classification with the random forest ensemble classifier	354
Detecting objects with SVM using HOG features	356
HOG training	356
Classification with the SVM model	357
Computing BoundingBoxes with HOG-SVM	358
Non-max suppression	359
Summary	360
Questions	361
Further reading	362
Chapter 10: Deep Learning in Image Processing - Image Classification	363
Deep learning in image processing	364
What is deep learning?	364
Classical versus deep learning	365
Why deep learning?	367
CNNs	367
Conv or Pooling or FC layers – CNN architecture and how it works	367
Convolutional layer	368
Pooling layer	370
Non-linearity – ReLU layer	370
FC layer	370

Dropout	370
Image classification with TensorFlow or Keras	371
Classification with TF	371
Classification with dense FC layers with Keras	379
Visualizing the network	381
Visualizing the weights in the intermediate layers	382
CNN for classification with Keras	384
Classifying MNIST	384
Visualizing the intermediate layers	387
Some popular deep CNNs	389
VGG-16/19	389
Classifying cat/dog images with VGG-16 in Keras	391
Training phase	391
Testing (prediction) phase	398
InceptionNet	400
ResNet	401
Summary	403
Questions	403
Further reading	404
Chapter 11: Deep Learning in Image Processing - Object Detection, and more	405
Introducing YOLO v2	406
Classifying and localizing images and detecting objects	406
Proposing and detecting objects using CNNs	408
Using YOLO v2	409
Using a pre-trained YOLO model for object detection	409
Deep semantic segmentation with DeepLab V3+	417
Semantic segmentation	417
DeepLab V3+	418
DeepLab v3 architecture	418
Steps you must follow to use DeepLab V3+ model for semantic segmentation	419
Transfer learning – what it is, and when to use it	422
Transfer learning with Keras	423
Neural style transfers with cv2 using a pre-trained torch model	428
Understanding the NST algorithm	428
Implementation of NST with transfer learning	429
Ensuring NST with content loss	430
Computing the style cost	430
Computing the overall loss	431
Neural style transfer with Python and OpenCV	431
Summary	434
Questions	435
Further reading and references	435
Chapter 12: Additional Problems in Image Processing	436
Seam carving	436

Table of Contents

Content-aware image resizing with seam carving	437
Object removal with seam carving	441
Seamless cloning and Poisson image editing	442
Image inpainting	445
Variational image processing	448
Total variation denoising	449
Creating flat-texture cartoonish images with total variation denoising	451
Image quilting	452
Texture synthesis	452
Texture transfer	452
Face morphing	453
Summary	455
Questions	455
Further reading and references	455
Other Books You May Enjoy	457
Index	460
