

# Table of Contents

<b>About the Author .....</b>	<b>xi</b>
<b>About the Technical Reviewer .....</b>	<b>xiii</b>
<b>Acknowledgments .....</b>	<b>xv</b>
<b>Introduction .....</b>	<b>xvii</b>
<b>Chapter 1: Introduction and Development Environment Setup.....</b>	<b>1</b>
GitHub Repository and Companion Website.....	3
Mathematical Level Required .....	3
Python Development Environment.....	4
Google Colab.....	5
Anaconda.....	9
Docker Image .....	18
Which Option Should You Choose?.....	25
<b>Chapter 2: TensorFlow: Advanced Topics .....</b>	<b>27</b>
Tensorflow Eager Execution.....	28
Enabling Eager Execution.....	29
Polynomial Fitting with Eager Execution .....	30
MNIST Classification with Eager Execution .....	34
TensorFlow and Numpy Compatibility.....	39
Hardware Acceleration.....	40
Checking the Availability of the GPU.....	40
Device Names.....	41



## TABLE OF CONTENTS

Explicit Device Placement .....	42
GPU Acceleration Demonstration: Matrix Multiplication .....	43
Effect of GPU Acceleration on the MNIST Example.....	45
Training Only Specific Layers.....	47
Training Only Specific Layers: An Example.....	48
Removing Layers .....	52
Keras Callback Functions.....	54
Custom Callback Class .....	55
Example of a Custom Callback Class .....	57
Save and Load Models.....	61
Save Your Weights Manually.....	67
Saving the Entire Model .....	68
Dataset Abstraction.....	68
Iterating Over a Dataset.....	71
Simple Batching .....	72
Simple Batching with the MNIST Dataset.....	73
Using <code>tf.data.Dataset</code> in Eager Execution Mode.....	76
Conclusions.....	77
<b>Chapter 3: Fundamentals of Convolutional Neural Networks.....</b>	<b>79</b>
Kernels and Filters .....	79
Convolution .....	81
Examples of Convolution.....	91
Pooling .....	99
Padding .....	104
Building Blocks of a CNN .....	105
Convolutional Layers .....	105
Pooling Layers .....	108
Stacking Layers Together .....	108



Number of Weights in a CNN.....	109
Convolutional Layer .....	109
Pooling Layer.....	110
Dense Layer.....	110
Example of a CNN: MNIST Dataset.....	110
Visualization of CNN Learning.....	115
Brief Digression: keras.backend.function().....	115
Effect of Kernels .....	118
Effect of Max-Pooling .....	121
<b>Chapter 4: Advanced CNNs and Transfer Learning .....</b>	<b>125</b>
Convolution with Multiple Channels.....	125
History and Basics of Inception Networks .....	129
Inception Module: Naïve Version .....	131
Number of Parameters in the Naïve Inception Module.....	132
Inception Module with Dimension Reduction .....	133
Multiple Cost Functions: GoogLeNet .....	134
Example of Inception Modules in Keras .....	136
Digression: Custom Losses in Keras .....	139
How To Use Pre-Trained Networks.....	141
Transfer Learning: An Introduction.....	145
A Dog and Cat Problem .....	149
Classical Approach to Transfer Learning .....	150
Experimentation with Transfer Learning.....	157
<b>Chapter 5: Cost Functions and Style Transfer.....</b>	<b>161</b>
Components of a Neural Network Model .....	161
Training Seen as an Optimization Problem.....	162
A Concrete Example: Linear Regression.....	164



## TABLE OF CONTENTS

The Cost Function .....	165
Mathematical Notation .....	165
Typical Cost Functions.....	166
Neural Style Transfer.....	176
The Mathematics Behind NST .....	178
An Example of Style Transfer in Keras.....	183
NST with Silhouettes .....	190
Masking.....	192
<b>Chapter 6: Object Classification: An Introduction.....</b>	<b>195</b>
What Is Object Localization? .....	196
Most Important Available Datasets .....	199
Intersect Over Union (IoU).....	200
A Naïve Approach to Solving Object Localization (Sliding Window Approach) ..	202
Problems and Limitations the with Sliding Window Approach .....	204
Classification and Localization.....	211
Region-Based CNN (R-CNN).....	213
Fast R-CNN .....	217
Faster R-CNN .....	219
<b>Chapter 7: Object Localization: An Implementation in Python .....</b>	<b>221</b>
The You Only Look Once (YOLO) Method .....	222
How YOLO Works .....	223
YOLOv2 (Also Known As YOLO9000).....	226
YOLOv3 .....	227
Non-Maxima Suppression .....	228
Loss Function .....	228



YOLO Implementation in Python and OpenCV .....231

    Darknet Implementation of YOLO.....231

    Testing Object Detection with Darknet .....233

Training a Model for YOLO for Your Specific Images .....240

Concluding Remarks .....241

**Chapter 8: Histology Tissue Classification .....243**

    Data Analysis and Preparation .....244

    Model Building .....253

    Data Augmentation .....264

        Horizontal and Vertical Shifts .....266

        Flipping Images Vertically.....269

        Randomly Rotating Images.....269

        Zooming in Images .....272

        Putting All Together.....272

    VGG16 with Data Augmentation .....273

        The fit() Function .....273

        The fit\_generator() Function .....274

        The train\_on\_batch() Function.....275

        Training the Network .....276

    And Now Have Fun.....277

**Index.....279**