

# Java Image Processing Recipes

With OpenCV and JVM

Nicolas Modrzyk

Apress®

# *Java Image Processing Recipes*

Nicolas Modrzyk  
Tokyo, Japan

ISBN-13 (pbk): 978-1-4842-3464-8

ISBN-13 (electronic): 978-1-4842-3465-5

<https://doi.org/10.1007/978-1-4842-3465-5>

Library of Congress Control Number: 2018936912

Copyright © 2018 by Nicolas Modrzyk

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr  
Editorial Director: Todd Green  
Acquisitions Editor: Nikhil Karkal  
Development Editor: Matthew Moodie  
Coordinating Editor: Divya Modi

Cover designed by eStudioCalamar

Cover image designed by Freepik ([www.freepik.com](http://www.freepik.com))

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail [orders-ny@springer-sbm.com](mailto:orders-ny@springer-sbm.com), or visit [www.springeronline.com](http://www.springeronline.com). Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail [rights@apress.com](mailto:rights@apress.com), or visit <http://www.apress.com/rights-permissions>.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at <http://www.apress.com/bulk-sales>.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at [www.apress.com/9781484234648](http://www.apress.com/9781484234648). For more detailed information, please visit <http://www.apress.com/source-code>.

Printed on acid-free paper

# Table of Contents

- About the Author .....xi**
- About the Technical Reviewer .....xiii**
- Acknowledgments .....xv**
- Introduction ..... xvii**
  
- Chapter 1: OpenCV on the JavaVM ..... 1**
  - 1-1 Getting Started with Leiningen ..... 2
    - Problem ..... 2
    - Solution ..... 2
    - How it works..... 3
  - 1-2 Writing Your First OpenCV Java Program..... 9
    - Problem ..... 9
    - Solution ..... 9
    - How it works..... 9
  - 1-3 Automatically Compiling and Running Code..... 13
    - Problem ..... 13
    - Solution ..... 13
    - How it works..... 13
  - 1-4 Using a Better Text Editor ..... 15
    - Problem ..... 15
    - Solution ..... 16
    - How it works..... 16

## TABLE OF CONTENTS

1-5 Learning the Basics of the OpenCV Mat Object.....	21
Problem .....	21
Solution .....	21
How it works.....	21
1-6 Loading Images from a File.....	24
Problem .....	24
Solution .....	24
How it works.....	25
1-7 Saving Images into a File .....	28
Problem .....	28
Solution .....	28
How it works.....	28
1-8 Cropping a Picture with Submat.....	30
Problem .....	30
Solution .....	30
How it works.....	30
1-9 Creating a Mat from Submats .....	35
Problem .....	35
Solution .....	35
How it works.....	35
1-10 Highlighting Objects in a Picture .....	40
Problem .....	40
Solution .....	40
How it works.....	41

1-11 Using a Canny Result as a Mask .....	44
Problem .....	44
Solutions.....	45
How it works.....	45
1-12 Detecting Edges with Contours .....	47
Problem .....	47
Solution .....	47
How it works.....	47
1-13 Working with Video Streams .....	51
Problem .....	51
Solution .....	51
How it works.....	52
1-14 Writing OpenCV Code in Scala .....	57
Problem .....	57
Solution .....	58
How it works.....	58
1-15 Writing OpenCV Code in Kotlin.....	66
Problems .....	66
Solutions.....	66
How it works.....	66
<b>Chapter 2: OpenCV with Origami .....</b>	<b>83</b>
2-1 Starting to Code with Origami .....	84
Problem .....	84
Solution .....	84
How it works.....	85

TABLE OF CONTENTS

2-2 Working with Mats..... 106

    Problem ..... 106

    Solution ..... 106

    How it works..... 107

2-3 Loading, Showing, Saving Mats..... 117

    Problem ..... 117

    Solution ..... 117

    How it works..... 117

2-4 Working with Colors, ColorMaps, and ColorSpaces ..... 124

    Problem ..... 124

    Solution ..... 125

    How it works..... 125

2-5 Rotating and Transforming Mats ..... 138

    Problem ..... 138

    Solution ..... 138

    How it works..... 139

2-6 Filtering Mats ..... 148

    Problem ..... 148

    Solution ..... 148

    How it works..... 148

2-7 Applying Simple Masking Techniques ..... 161

    Problem ..... 161

    Solution ..... 161

    How it works..... 161

2-8 Blurring Images .....	166
Problem .....	166
Solution .....	167
How it works.....	167
<b>Chapter 3: Imaging Techniques .....</b>	<b>175</b>
3-1 Playing with Colors.....	176
Problem .....	176
Solution .....	177
How it works.....	177
3-2 Creating Cartoons.....	201
Problem .....	201
Solution .....	201
How it works.....	201
3-3 Creating Pencil Sketches.....	211
Problem .....	211
Solution .....	211
How it works.....	212
3-4 Creating a Canvas Effect .....	221
Problem .....	221
Solution .....	221
How it works.....	221
3-5 Highlighting Lines and Circles .....	226
Problem .....	226
Solution .....	226
How it works.....	227

TABLE OF CONTENTS

3-6 Finding and Drawing Contours and Bounding Boxes .....	239
Problem .....	239
Solution .....	240
How it works.....	240
3-7 More on Contours: Playing with Shapes .....	252
Problem .....	252
Solution .....	252
How it works.....	252
3-8 Moving Shapes .....	260
Problem .....	260
Solution .....	261
How it works.....	262
3-9 Looking at Trees .....	266
Problem .....	266
Solution .....	266
How it works.....	266
3-10 Detecting Blur.....	269
Problem .....	269
Solution .....	269
How it works.....	270
3-11 Making Photomosaics .....	272
Problem .....	272
Solution .....	272
How it works.....	273



<b>Chapter 4: Real-Time Video .....</b>	<b>279</b>
4-1 Getting Started with Video Streaming .....	280
Problem .....	280
Solution .....	280
How it works.....	281
4-2 Combining Multiple Video Streams.....	288
Problem .....	288
Solution .....	289
How it works.....	289
4-3 Warping Video.....	290
Problem .....	290
Solution .....	290
How it works.....	291
4-4 Using Face Recognition .....	293
Problem .....	293
Solution .....	293
How it works.....	294
4-5 Diffing with a Base Image .....	297
Problem .....	297
Solution .....	297
How it works.....	298
4-6 Finding Movement.....	302
Problem .....	302
Solution .....	302
How it works.....	303

TABLE OF CONTENTS

4-7 Separating the Foreground from the Background Using Grabcut ..... 308

    Problem ..... 308

    Solution ..... 308

    How it works..... 309

4-8 Finding an Orange in Real Time..... 317

    Problem ..... 317

    Solution ..... 317

    How it works..... 318

4-9 Finding an Image Within the Video Stream ..... 323

    Problem ..... 323

    Solution ..... 323

    How it works..... 324

**Index.....333**

# About the Author



**Nicolas Modrzyk** is currently Chief Technical Officer of Karabiner Software and a leader of development teams.

He is also an active contributor to the open source software community. As a developer and technical consultant, Nico has been involved over many years in designing large-scale server applications for a video conferencing company, managing enormous clusters of databases through high-performance middleware developed from scratch, enabling Japanese leaders with content management and process management systems, and pushing the boundaries of business processes for leading Asian companies.

Nico is an ardent advocate of Agile methods and is focused on getting the job done right to satisfy clients. He also loves to push friends and team members to challenge themselves and reach their goals. He has lived by those empowering standards in various countries, including France, America, Ireland, Japan, China, and India. Nico is also the author of a few other books on the Clojure programming language, in both English and Japanese.

He is currently based in Tokyo, Japan, where he is often found after hours playing soccer, hiking, performing live concerts with his guitar, and enjoying life with friends and colleagues.

# About the Technical Reviewer



**Aakash Kag** is an AI developer at Manacola Private Ltd. He has two years of experience in big data analytics. He is a postgraduate in Computer Science with a specialization in Big Data Analytics. Aakash has also made contributions to the Microsoft bot builder.

Currently, Aakash is working on problems related to Conversational Bots and Natural Language Understanding.

He is passionate about Machine Learning meetups, where he often presents talks.

# Acknowledgments

It's been the most amazing typing race of my life to get this book out on time, and to beat time and the odds, I got support from so many people that it would take another book just to write the list of names. So ...

Thank you to all my family, brother, sister, friends, Abe-san, all my soccer friends, people still having Guinness pints in Ireland (keep one for me!), the awesome people in America (who sometimes send LP records... when I need them the most), Sawada-san, Chris and the Biners, my French friends (always there for support even when not being asked for it), publisher Apress, Divya for never being impressed and kicking my butt on a regular basis, and ... the people deep in my heart for your NEVER-ENDING support. I never could have finished this without you. I appreciate it so much.

And, of course... thank you to my two beautiful daughters, Mei and Manon, for keeping up and always doing their best even during hard times. You simply rock! I love you.

---

The original version of the book FM was revised. The changes have been made in the editorial section of the copyright page.

# Introduction

My father is a dentist. When I was in my early childhood, he used to repeat the same sentence over and over again, which as far as I can remember and translate properly now was something like:

“Son, get the right tool for the job.”

And as he was looking at me trying to wash the car with the wrong washing product and spending twice the amount of time that I should have, I knew somewhere deep inside of me that he was right.

He did not use a screwdriver to pull out teeth from his patients, and he had what seemed like twenty different brushes to clean each type of tooth. I even thought it was funny at the time.

Fast-forward thirty years later; I was talking about this book with him and he added:

“Well, son, you know, it’s not only about the right tool, it’s about the right tool at the right time.”

And so, this is the philosophy guiding this book.

OpenCV, the computer vision library, has always been one of the tools to work on imaging- and vision-related projects, even more so with every improvement in AI and neural networks. But OpenCV was always taking some time to get the right libraries, and the right build tools, and the right build settings, and so forth.

The vision of the Clojure wrapper *Origami* is to bring you all the power of OpenCV to your hands almost instantly, along with a pleasurable syntax. This way we hope you can focus and spend your time entirely on the job, not on the tool.

Chapter 1 will introduce you to pure OpenCV on the JVM using Java, Scala, and Kotlin and present some of their shortcomings.

## INTRODUCTION

Chapter 2 will present Origami, the Clojure wrapper, and how to use it to perform simple image manipulation.

Chapter 3 will get you up to speed with more advanced concepts of image manipulation, like shape finding, but still in a pleasant syntax.

Finally, Chapter 4 moves to video analysis, with shape finding, transformations, and various techniques to analyze real-time streams with ease.