



Community Experience Distilled

OpenCV Essentials

Acquire, process, and analyze visual content to build full-fledged imaging applications using OpenCV

Oscar Deniz Suarez
Noelia Vázquez Enano
Ismael Serrano Gracia
Jesus Salido Tercero

M^a del Milagro Fernández Carrobles
Gloria Bueno García
Julio Alberto Patón Incertis

[PACKT] open source*
PUBLISHING community experience distilled

OpenCV Essentials

Acquire, process, and analyze visual content to build full-fledged imaging applications using OpenCV

Oscar Deniz Suarez

M^a del Milagro Fernández Carrobles

Noelia Vázquez Enano

Gloria Bueno García

Ismael Serrano Gracia

Julio Alberto Patón Incertis

Jesus Salido Tercero

[PACKT] open source 
PUBLISHING community experience distilled

BIRMINGHAM - MUMBAI

OpenCV Essentials

Copyright © 2014 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 authors, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be 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.

First published: August 2014

Production reference: 1200814

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

ISBN 978-1-78398-424-4

www.packtpub.com

Cover image by Arie Leeuwesteijn (aleeuwesteijn@hotmail.com)

Credits

Authors

Oscar Deniz Suarez
M^a del Milagro Fernández
Carrobles
Noelia Vález Enano
Gloria Bueno García
Ismael Serrano Gracia
Julio Alberto Patón Incertis
Jesus Salido Tercero

Reviewers

Nashruddin Amin
Emmanuel d'Angelo
Karan Kedar Balkar
Arturo de la Escalera

Commissioning Editor

Ashwin Nair

Acquisition Editor

Sonali Vernekar

Content Development Editor

Prachi Bisht

Technical Editor

Novina Kewalramani

Project Coordinator

Sageer Parkar

Copy Editors

Roshni Banerjee
Dipti Kapadia
Gladson Monteiro
Aditya Nair
Karuna Narayanan
Adithi Shetty
Stuti Srivastava

Proofreaders

Simran Bhogal
Bridget Braund
Paul Hindle
Bernadette Watkins

Indexer

Hemangini Bari
Mariammal Chettiyar
Tejal Soni

Graphics

Ronak Dhruv
Disha Haria
Abhinash Sahu

Production Coordinator

Kyle Albuquerque

Cover Work

Kyle Albuquerque

About the Authors

Oscar Deniz Suarez is the author of more than 50 refereed papers in journals and conferences. His research interests are mainly focused on computer vision and pattern recognition. He received the runner-up award for the best PhD work on computer vision and pattern recognition by AERFAI and the Image File and Reformatting Software challenge award by InnoCentive Inc. He has been a national finalist at the 2009 Cor Baayen awards. His work is being used by cutting-edge companies such as Existor, GLIIF, TapMedia, E-Twenty, and others and has also been added to OpenCV. Currently, he works as an associate professor at the University of Castilla-La Mancha and contributes to Visilabs. He is a senior member of IEEE and is affiliated with AAAI, SIANI, CEA-IFAC, AEPIA, AERFAI-IAPR, and The Computer Vision Foundation. He serves as an academic editor for the journal PLOS ONE. He has been a visiting researcher at Carnegie Mellon University, Pennsylvania, USA; Imperial College, London, UK; and Leica Biosystems, Ireland. He has co-authored a book on OpenCV programming for mobile devices.

M^a del Milagro Fernández Carroles received her Bachelor's degree in Computer Science and Master's degree in Physics and Mathematics from the University of Castilla-La Mancha, Spain, in 2010 and 2011, respectively. She is currently a PhD candidate and works at Visilabs. Her research interests include image processing and artificial intelligence, especially in medical imaging.

I would like to thank my parents for their love and support. Without them, I would never have gotten where I am today. I also thank Jorge for his limitless patience.

Noelia Vallez Enano liked computers since childhood, although she didn't have one before her mid-teens. In 2009, she finished her studies in Computer Science through the University of Castilla-La Mancha. She started her work at the Visilabs group through a project about mammography CAD systems and electronic health records. Since then, she has obtained a Master's degree in Physics and Mathematics and has enrolled for a PhD degree. Her work involves using image processing and pattern recognition methods. She also likes to teach and work in other areas of artificial intelligence.

I would like to thank Jose and the rest of my family for all their support throughout these years and specially now that I am writing this book 2200 km away from them.

Gloria Bueno Garca holds a PhD in Machine Vision from Coventry University, UK. She has experience working as a principal researcher in several research centers, such as UMR 7005 research unit CNRS, Louis Pasteur University, Strasbourg, France; Gilbert Gilkes and Gordon Technology, UK; and CEIT San Sebastian, Spain. She is the author of two patents, one registered software, and more than 100 refereed papers. Her interests are in 2D/3D multimodality image processing and artificial intelligence. She leads the Visilabs research group at the University of Castilla-La Mancha. She has co-authored a book on OpenCV programming for mobile devices.

Ismael Serrano Gracia received his Bachelor's degree in Computer Science in 2012 from the University of Castilla-La Mancha. He scored the highest marks in his final degree project about human detection. This application uses depth cameras with OpenCV libraries. He is currently a PhD candidate at the same university, holding a research grant from the Spanish Ministry for Science and Research. He is also working at the Visilabs group as an assistant researcher and developer on different computer vision topics.

Julio Alberto Patón Incertis graduated from the University of Castilla-La Mancha. He started developing computer vision applications as part of his Master's degree project in Computer Science. He has focused on mobile devices, mainly Android; he has created a mobile application for this platform that is capable of locating, tracking, and recognizing text for blind people. This application was later used as a part of a patent-pending indoor positioning system. OpenCV has been an essential part of all his projects.

Thanks to my parents, as their huge efforts allowed me to obtain a degree at the university. I would also like to thank the Visilabs research group for giving me the opportunity to start developing computer vision applications.

Jesus Salido Tercero gained his Electrical Engineering degree and PhD (1996) at Universidad Politécnica de Madrid, Spain. He then spent 2 years (1997/98) as a visiting scholar at The Robotics Institute (Carnegie Mellon University, Pittsburgh, USA), working on cooperative multirobot systems. Since his return to the Spanish university of Castilla-La Mancha, he divides his work time between teaching courses on robotics and industrial informatics and researching on vision and intelligent systems. For the past 3 years, his efforts have been directed to develop vision applications on mobile devices. He has co-authored a book on OpenCV programming for mobile devices.

To my three precious jewels: Dorita, Juan Pablo, and Jacobo.

About the Reviewers

Nashruddin Amin has been programming with OpenCV since 2008. He enjoys learning Computer Vision topics and writing programs using OpenCV for research purposes. He also maintains a blog (<http://opencv-code.com>) where he shares his experiences with OpenCV.

Emmanuel d'Angelo is an image-processing enthusiast who has turned his hobby into a job. After working as a technical consultant on various projects ranging from real-time image stabilization to large-scale image database analysis, he is now in charge of developing Digital Signal Processing applications on low-power consumer devices. You can find more insight about his research and image processing related information on his blog at <http://www.computersdontsee.net/>.

Emmanuel holds a PhD degree from the Swiss Federal Institute of Technology, EPFL, Switzerland, and a Master's degree in Remote Sensing from ISAE, Toulouse, France.

Karan Kedar Balkar has been working as an independent Android application developer for the past 4 years. Born and brought up in Mumbai, he holds a Bachelor's degree in Computer Engineering. He has written over 50 programming tutorials on his personal blog (<http://karanbalkar.com>) that cover popular technologies and frameworks.

At present, he is working as a software engineer. He has been trained on various technologies including Java, Oracle, and .NET. Apart from being passionate about technology, he loves to write poems and to travel to different places. He likes listening to music and enjoys playing the guitar.

Firstly, I would like to thank my parents for their constant support and encouragement. I would also like to thank my friends, Srivatsan Iyer, Ajit Pillai, and Prasaanth Neelakandan for always inspiring and motivating me.

I would like to express my deepest gratitude to Packt Publishing for giving me a chance to be a part of the reviewing process.

Arturo de la Escalera graduated from the Universidad Politecnica de Madrid, Madrid, Spain, in Automation and Electronics Engineering in 1989, where he also obtained his PhD degree in Robotics in 1995. In 1993, he joined the department of Systems Engineering and Automation of Universidad Carlos III de Madrid, Madrid, Spain, where he is an Associate Professor since 1997.

His current research interests include Advanced Robotics and Intelligent Transportation Systems, with special emphasis on vision sensor systems and image data processing methods for environment perception and real-time pattern recognition. He has co-authored more than 30 articles in journals and more than 70 papers in international congresses.

Since 2005, Arturo de la Escalera has headed the Intelligent Systems Laboratory at UC3M (www.uc3m.es/islab). He is a member of the Editorial Board of the International Journal of Advanced Robotic Systems (topic – Robot Sensors), the International Journal of Information and Communication Technology, The Open Transportation Journal, and The Scientific World Journal (topic – Computer Science).

www.PacktPub.com

Support files, eBooks, discount offers, and more

You might want to visit www.PacktPub.com for support files and downloads related to your book.

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.PacktPub.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at service@packtpub.com for more details.

At www.PacktPub.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.



<http://PacktLib.PacktPub.com>

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

Free access for Packt account holders

If you have an account with Packt at www.PacktPub.com, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Table of Contents

Preface	1
Chapter 1: Getting Started	7
Setting up OpenCV	7
Compiled versus precompiled library	8
Configuring OpenCV with CMake	9
Building and installing the library	11
Quick recipe for setting up OpenCV	13
API concepts and basic datatypes	14
Our first program – reading and writing images and videos	17
The qmake project file	18
Reading and playing a video file	22
Live input from a camera	23
Summary	25
Chapter 2: Something We Look At – Graphical User Interfaces	27
Using OpenCV's highgui module	27
Text and drawing	33
Selecting regions	38
Using Qt-based functions	41
Text overlays and status bar	42
The properties dialog	43
Windows properties	46
Qt images	46
Summary	47
Chapter 3: First Things First – Image Processing	49
Pixel-level access and common operations	49
Image histogram	51
Histogram equalization	53
Brightness and contrast modeling	55

Histogram matching and LUT	58
Conversion from RGB to other color spaces	62
Filtering with the retina model	64
Arithmetic and geometrical transforms	67
Arithmetic transform	67
Geometrical transforms	69
Summary	74
What else?	74
Chapter 4: What's in the Image? Segmentation	75
Thresholding	76
Contours and connected components	81
Flood fill	89
Watershed segmentation	92
GrabCut	95
Summary	98
What else?	98
Chapter 5: Focusing on the Interesting 2D Features	99
Interest points	100
Feature detectors	100
The FAST detector	102
The SURF detector	106
The ORB detector	108
The KAZE and AKAZE detectors	110
Feature descriptor extractors	112
Descriptor matchers	114
Matching the SURF descriptors	115
Matching the AKAZE descriptors	118
Summary	121
What else?	121
Chapter 6: Where's Wally? Object Detection	123
Object detection	123
Detecting objects with OpenCV	123
Cascades are beautiful	124
Object detection using cascades	124
Training your own cascade	129
Latent SVM	133
Scene text detection	137
Summary	145
What else?	145

Chapter 7: What Is He Doing? Motion	147
Motion history	147
Reading video sequences	148
The Lucas-Kanade optical flow	150
The Gunnar-Farneback optical flow	153
The Mean-Shift tracker	155
The CamShift tracker	157
The Motion templates	157
The Motion history template	158
The Motion gradient	160
The Background subtraction technique	162
Image alignment	164
Summary	168
What else?	168
Chapter 8: Advanced Topics	169
Machine learning	169
The KNN classifier	170
The Random Forest classifier	172
SVM for classification	176
What about GPUs?	180
Setting up OpenCV with CUDA	180
Configuring the OpenCV build	182
Building and installing the library	183
A quick recipe for setting up OpenCV with CUDA	183
Our first GPU-based program	184
Going real time	186
Performance	190
Summary	190
What else?	190
Index	191

Preface

OpenCV, arguably the most widely used computer vision library, includes hundreds of ready-to-use imaging and vision functions and is extensively used in both academia and industry. As cameras get cheaper and imaging features grow in demand, the range of applications taking advantage of OpenCV is increasing significantly, particularly for mobile platforms.

As a computer vision library, OpenCV provides the following two big advantages:

- It is open source and everyone can freely use it, either on an academic level or for real-life projects
- It arguably contains the most extensive and up-to-date collection of computer vision functions

OpenCV is fed with cutting-edge research in Computer Vision, image and video processing, and machine learning.

The first book published on OpenCV provided a mostly theoretical approach, explaining the underlying computer vision techniques. Subsequent books have adopted the contrary approach, filling pages and pages with large examples (almost complete applications) that are difficult to follow. Large examples are difficult to follow and cannot be easily reused in the reader's projects. Examples taking up several pages are simply not appropriate for a book. We believe that examples should be easy to understand and should also be used as building blocks to reduce the time needed to have a working example for the reader's projects. Consequently, in this book, we also adopt a practical approach, although we aim to cover a larger spectrum of functions with shorter, easy-to-follow examples. From our experience with OpenCV, we can affirm that examples are ultimately the most valuable resource.

What this book covers

Chapter 1, Getting Started, deals with the basic installation steps and introduces the essential concepts of the OpenCV API. The first examples to read/write images and video and capture them from a camera are also provided.

Chapter 2, Something We Look At – Graphical User Interfaces, covers user interface capabilities for our OpenCV-based applications.

Chapter 3, First Things First – Image Processing, covers the most useful image processing techniques available in OpenCV.

Chapter 4, What's in the Image? Segmentation, tackles the all-important problem of image segmentation in OpenCV.

Chapter 5, Focusing on the Interesting 2D Features, covers the functions available for extracting keypoints and descriptors from an image.

Chapter 6, Where's Wally? Object Detection, describes that object detection is a central problem in computer vision. This chapter explains the functionality available for object detection.

Chapter 7, What Is He Doing? Motion, considers more than just a single static image. This chapter deals with motion and tracking in OpenCV.

Chapter 8, Advanced Topics, focuses on some advanced topics such as machine learning and GPU-based acceleration.

What you need for this book

The approach followed in this book is particularly suited for readers who are already knowledgeable in computer vision (or can learn the discipline elsewhere) and want to start developing applications rapidly. Each chapter provides several examples of the key available functions for the most important stages in a vision system. The book is, therefore, focused on providing the reader with a working example as soon as possible so that he/she can develop additional features on top of that.

To use this book, only free software is needed. All the examples have been developed and tested with the freely available Qt IDE. The freely available CUDA toolkit is required for the GPU acceleration examples in *Chapter 8, Advanced Topics*.

Who this book is for

This book is neither a C++ tutorial nor a textbook on computer vision. The book is intended for C++ developers who want to learn how to implement the main techniques of OpenCV and get started with it quickly. Previous contact with computer vision/image processing is expected.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text, folder names, filenames, file extensions, pathnames, system variables, URLs, and user input are shown as follows: "Each module has an associated header file (for example, `core.hpp`)."

A block of code is set as follows:

```
#include "opencv2/core/core.hpp"
#include "opencv2/highgui/highgui.hpp"
using namespace std;
using namespace cv;

int main(int argc, char *argv[])
{
    Mat frame; // Container for each frame
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in bold:

```
#include "opencv2/core/core.hpp"
#include "opencv2/highgui/highgui.hpp"
#include <iostream>


using namespace std;
using namespace cv;


int main(int argc, char *argv[])
{
```

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

```
C:\opencv-buildQt\install
```

New terms and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "Also, the checkboxes labeled as **Grouped** and **Advanced** should be marked in the CMake main window."

[ Warnings or important notes appear in a box like this.]

[ Tips and tricks appear like this.]

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book – what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to feedback@packtpub.com, and mention the book title via the subject of your message.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on www.packtpub.com/authors.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Downloading the example code

You can download the example code files for all Packt books you have purchased from your account at <http://www.packtpub.com>. If you purchased this book elsewhere, you can visit <http://www.packtpub.com/support> and register to have the files e-mailed directly to you.

Downloading the color images of this book

We also provide you a PDF file that has color images of the screenshots/diagrams used in this book. The color images will help you better understand the changes in the output. You can download this file from: https://www.packtpub.com/sites/default/files/downloads/42440S_Graphics.pdf.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books – maybe a mistake in the text or the code – we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting <http://www.packtpub.com/submit-errata>, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from <http://www.packtpub.com/support>.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at copyright@packtpub.com with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at questions@packtpub.com if you are having a problem with any aspect of the book, and we will do our best to address it.

1

Getting Started

This chapter deals with the basic installation steps and settings required to develop applications with the OpenCV library. Also, it introduces the essential concepts in order to use the **Application Programming Interface (API)** provided by the library and the basic datatypes supplied. This chapter includes a section with full examples of code that illustrate how to read/write images and video files, and access images from live cameras. These examples also show how to get access to live input from cameras connected to a computer.

Setting up OpenCV

OpenCV can be downloaded from <http://opencv.org/>, and is available for the most popular operating systems, such as Unix (Linux/Mac), Microsoft Windows (Windows), Android, and iOS. In this book, the last stable release (2.4.9) of OpenCV for Windows 7 (SP1) has been used. For Windows, this release comes in the form of a self-extracting archive (`opencv-2.4.9.exe`), which should be extracted to the desired location (for example, `OPENCV_SRC` for `C:\opencv-src`). It should be noted that in Windows it is strongly recommended to allocate the source and binaries at absolute paths without white spaces because errors might appear later.

After extracting the archive, the obtained files are organized in two subdirectories under `OPENCV_SRC`: `build` and `sources`. The first one (`build`) includes precompiled (binaries) versions with Microsoft Visual C++ compilers (MSVC, v. 10, 11, and 12) for 32- and 64-bit architectures (located in the `x86` and `x64` subdirectories respectively). The `sources` subdirectory contains the source code of the OpenCV library. This code might be compiled with other compilers (for example, GNU `g++`).