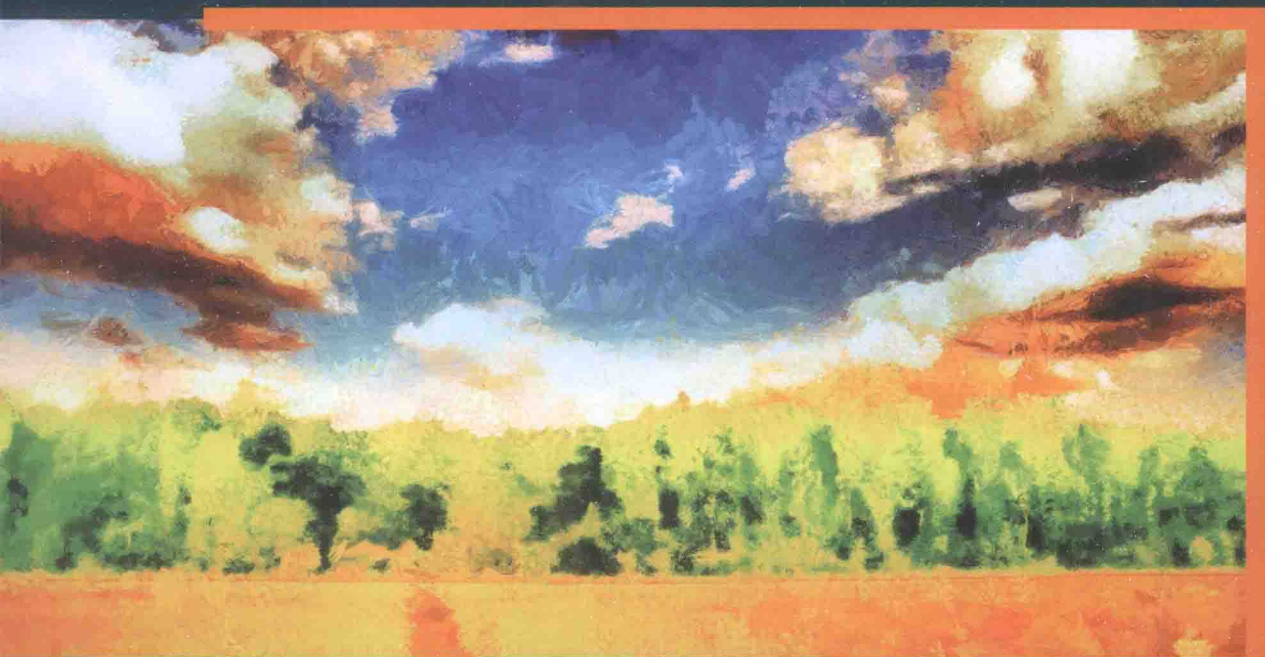


OpenCV 3 计算机视觉

Python语言实现 (原书第2版)

Learning OpenCV 3 Computer Vision with Python

Second Edition



[爱尔兰] 乔·米尼奇诺 (Joe Minichino) 著
[加] 约瑟夫·豪斯 (Joseph Howse)

刘波 苗贝贝 史斌 译



机械工业出版社
China Machine Press

华章程序员书库



Learning OpenCV 3 Computer
Vision with Python
Second Edition

OpenCV 3计算机视觉 Python语言实现

(原书第2版)

[爱尔兰] 乔·米尼奇诺 (Joe Minichino)

[加] 约瑟夫·豪斯 (Joseph Howse)

著

刘波 苗贝贝 史斌 译



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

OpenCV 3 计算机视觉: Python 语言实现 (原书第 2 版) / (爱尔兰) 乔·米尼奇诺 (Joe Minichino) 等著; 刘波, 苗贝贝, 史斌译. —北京: 机械工业出版社, 2016.6

(华章程序员书库)

书名原文: Learning OpenCV 3 Computer Vision with Python, Second Edition

ISBN 978-7-111-53975-9

I. O… II. ①乔… ②刘… ③苗… ④史… III. 图像处理软件—程序设计 IV. TP391.41

中国版本图书馆 CIP 数据核字 (2016) 第 121598 号

本书版权登记号: 图字: 01-2016-1887

Learning OpenCV 3 Computer Vision with Python, Second Edition (ISBN: 978-1-78528-384-0).

Copyright © 2015 Packt Publishing. First published in the English language under the title “Learning OpenCV 3 Computer Vision with Python, Second Edition”.

All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2016 by China Machine Press.

本书中文简体字版由 Packt Publishing 授权机械工业出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

OpenCV 3 计算机视觉: Python 语言实现 (原书第 2 版)

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 陈佳媛

责任校对: 殷虹

印刷: 北京市荣盛彩色印刷有限公司

版次: 2016 年 6 月第 1 版第 1 次印刷

开本: 186mm × 240mm 1/16

印张: 12.5

书号: ISBN 978-7-111-53975-9

定价: 49.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

计算机视觉是一门用计算机模拟生物视觉的学科，更具体地讲，就是让计算机代替人眼实现对目标的识别、分类、跟踪和场景理解。计算机视觉是人工智能的重要分支，也是一门具有很强综合性的学科，涉及计算机科学与工程、信号处理、光学、应用数学、统计学、神经生理学和认知科学等学科。

OpenCV 是开源、跨平台的计算机视觉库，由英特尔公司发起并参与开发，在商业和研究领域中可以免费使用。本书介绍了如何通过 Python 来开发基于 OpenCV 3.0 的应用。作为当前非常流行的动态语言之一，Python 不仅使用非常简单，而且功能强大。通过 Python 来学习 OpenCV 框架，可让读者快速理解计算机视觉的基本概念以及重要算法。

本书分 9 章来介绍计算机视觉的重要概念，所有的概念都融入了一些很有趣的项目。本书首先详细介绍了多个平台下基于 Python 的 OpenCV 安装，继而介绍了计算机视觉应用的基本操作，包括图像文件的读取与显示，图像处理的基本操作（比如边缘检测等），深度估计与分割，人脸检测与识别，图像的检索，目标的检测与识别，目标跟踪，神经网络的手写体识别。可以这样说，本书是一本不可多得的采用 OpenCV 实践计算机视觉应用的好书。

本书的第 1 章由重庆工商大学计算机科学与信息工程学院的刘波博士翻译；第 2 章、第 5 章至第 9 章由苗贝贝翻译；第 3 章和第 4 章由史斌翻译。同时，刘波博士负责全书的技术审稿工作。

翻译本书的过程也是译者不断学习的过程。为了保证专业词汇翻译的准确性，意译部分既不失原著意境又无偏差，在翻译过程中查阅了大量相关资料。但由于时间和能力有限，书中内容难免出现差错。若有问题，读者可通过电子邮件 liubo7971@163.com 和 alingse@foxmail.com 与我们联系，欢迎一起探讨，共同进步。并且，我们也会将勘误信息公布在 <http://www.cnblogs.com/ml-cv/> 上。

本书翻译过程得到如下项目资助：（1）重庆市教委研究项目“多核正则化机器学习理论研究”，项目号为 KJ130709；（2）重庆工商大学研究项目“基于多核学习的高维数据分析研究”，项目号为 2013-56-09；（3）电子商务及供应链系统重庆市重点实验室研究项目“基于迹比率的特征选择及关键技术研究”；（4）大数据稀疏表示判别字典学习及其应用技术研究，项目号为 KJ1400612。

感谢重庆工商大学计算机科学与信息工程学院金融信息化专业杨仕喜同学对本书代码进行验证，并提出了宝贵意见。感谢家人的支持与鼓励，尤其感谢妻子杨雪莉和女儿刘典、刘恩丫，她们的信任与鼓励给我提供了不断前进的动力。

刘波

2016年3月

OpenCV 3 是一种先进的计算机视觉库，可以用于各种图像和视频处理操作，通过 OpenCV 3 能很容易地实现一些有前景且功能先进的应用（比如：人脸识别或目标跟踪等）。理解与计算机视觉相关的算法、模型以及 OpenCV 3 API 背后的基本概念，有助于开发现实世界中的各种应用程序（比如：安全和监视领域的工具）。

本书将从图像处理的基本操作出发，带你开启先进计算机视觉概念的探索之旅。计算机视觉是一个快速发展的学科，在现实生活中，它的应用增长得非常快，因此写作本书的目的是为了帮助计算机视觉领域的新手和想要了解全新的 OpenCV 3.0.0 的计算机视觉专家。

本书的主要内容

第 1 章介绍如何在不同平台下安装基于 Python 的 OpenCV，并给出一些常见问题的解决方法。

第 2 章介绍了 OpenCV 的 I/O 功能，并讨论与项目相关的概念，以及如何针对该项目进行面向对象设计。

第 3 章介绍一些图像变换方法，例如在图像中检测肤色、锐化图像、标记主体轮廓，以及使用线段检测器检测人行横道等。

第 4 章介绍如何利用深度摄像头的数据来识别前景和背景区域，这样就可以限制针对前景或背景的效果。

第 5 章介绍一些 OpenCV 的人脸检测功能和相关的数据文件，这些文件定义了跟踪目标的特定类型。

第 6 章介绍如何用 OpenCV 来检测图像特征，并利用这些特征来匹配和搜索图像。

第 7 章介绍目标检测和目标识别的概念，这是计算机视觉中最常见的问题之一。

第 8 章对目标跟踪进行深入探讨，目标跟踪是对摄像机中的图像或视频中移动的物体

进行定位的过程。

第9章介绍基于 OpenCV 的人工神经网络，并介绍其在现实生活中的应用。

阅读前的准备工作

本书第1章会指导读者安装所有必要软件，你只需准备一台较新的计算机。另外，强烈推荐为计算机安装摄像头，但这并不是必备的。

本书的读者对象

本书针对具有一定 Python 工作经验的程序员以及想要利用 OpenCV 库研究计算机视觉课题的读者。本书不要求读者具有计算机视觉或 OpenCV 经验，但要具有编程经验。

本书体例

本书有很多用来区分不同信息的文本格式，下面给出一些这类格式的样例，并解释它们的含义。

代码块的格式如下：

```
import cv2
import numpy as np

img = cv2.imread('images/chess_board.png')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
gray = np.float32(gray)
dst = cv2.cornerHarris(gray, 2, 23, 0.04)
```

为了提醒读者注意代码块中的特殊部分，会将相关行或相关项设置为**粗体**：

```
img = cv2.imread('images/chess_board.png')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
gray = np.float32(gray)
dst = cv2.cornerHarris(gray, 2, 23, 0.04)
```

命令行的输入或输出的格式为：

```
mkdir build && cd build
cmake D CMAKE_BUILD_TYPE=Release -DOPENCV_EXTRA_MODULES_PATH=<opencv_
contrib>/modules D CMAKE_INSTALL_PREFIX=/usr/local ..
make
```



注意：警告或重要注释以这样的形式出现。



提示：提示和技巧以这样的形式出现。

下载示例代码

读者可登录华章网站 (www.hzbook.com) 本书页面，下载本书示例代码。

作者简介 *About the Authors*

Joe Minichino 是 Hoolux Medical 从事计算机视觉的工程师，他利用业余时间开发了 NoSQL 数据库 LokiJS。他也是重金属歌手 / 作曲家。他是一个充满激情的程序员，对编程语言和技术非常好奇，并一直在使用它们。在 Hoolux，Joe 领导了针对医疗行业的 Android 计算机视觉广告平台的开发。

他出生在意大利瓦雷泽市的 Lombardy，并在那里长大，在米兰 Università Statale 受过哲学教育，最近 11 年 Joe 在爱尔兰的 Cork 度过，在这里他成为 Cork 技术研究所的一名计算机科学研究生。

我非常感谢我的合作伙伴 Rowena，她总是鼓励我，也感谢两个小女儿给我灵感。非常感谢这本书的合作者和编辑，尤其是 Joe Howse、Adrian Roesbrock、Brandon Castellano、OpenCV 社区，以及 Packt 出版社中那些为本书付出劳动的人。

Joseph Howse 生活在加拿大。在冬天，他留着胡子，而他的四只猫留着厚皮毛。他喜欢每天给猫梳毛。有时猫还会抓他的胡子。

自 2012 年以来，他一直在为 Packt 出版社写作，他的著作包括《OpenCV for Secret Agents》《OpenCV Blueprints》《Android Application Programming with OpenCV 3》《OpenCV Computer Vision with Python》以及《Python Game Programming by Example》。

当他不写书或打理萌宠时，他会提供咨询和培训，并通过他的公司（Nummist Media 公司 (<http://nummist.com>)) 进行软件开发服务。

About the Reviewers 审校者简介

Nandan Banerjee 拥有计算机科学学士学位和机器人工程硕士学位。他毕业后在三星电子工作。他在班加罗尔的研发中心工作了一年。为了参加 DARPA 机器人挑战，他还曾在位于 Atlas 的 Boston Dynamics 机器人公司的 WPI-CMU 团队工作过。目前他是 iRobot 公司的一名机器人软件工程师。他是一名嵌入式系统和机器人爱好者，主要喜欢计算机视觉和运动规划。他熟悉各种语言，包括 C、C++、Python、Java 和 Delphi。他在工作中会用到 ROS、OpenRAVE、OpenCV、PCL、OpenGL、CUDA 和 Android SDK。

我要感谢作者和出版商能出版这样精彩的书。

Tian Cao 在美国教堂山的北卡罗来纳州大学攻读计算机科学博士学位，并参与图像分析、计算机视觉和机器学习等项目。

我将这项工作献给我的父母和女友。

Brandon Castellano 来自加拿大的学生，在加拿大伦敦市西安大略大学攻读电气工程硕士学位。在 2012 年，他获得同专业的学士学位。他主要研究实时图像处理算法的并行处理实现和 GPGPU/FPGA 优化。Brandon 也在 Eagle Vision Systems 公司工作过，在这家公司主要专注于机器人应用中的实时图像处理。

虽然他使用 OpenCV 和 C++ 超过 5 年了，但他在研究中一直提倡使用 Python，因为 Python 开发速度快，可与复杂系统实现低级对接。他在 GitHub 上有开源项目，例如，PySceneDetect，这些项目大部分是用 Python 编写的。除了图像 / 视频处理，他还致力于实现三维展示，并提供软件工具来支持这样的开发。

他除了在他的网站 (<http://www.bcastell.com>) 上张贴技术文章和教程外，还参与各种开源和不开源的项目，他在 GitHub 上的用户名为 Breakthrough (<http://www.github.com/Breakthrough>)。他是 Super User 和 Stack Overflow 社区的活跃成员 (其名字仍是

Breakthrough), 可直接通过他的网站与他联系。

我要感谢过去几年所有的朋友和家人的耐心 (尤其是我的父母 Peter 和 Lori 以及我的兄弟 Mitchell), 没有他们持续的爱和支持, 我不可能取得这一切成就。

我还要特别感谢所有致力开源软件库的开发者, 特别是 OpenCV, 这有助于将前沿的软件技术免费带给世界各地的软件开发者。也想感谢那些写文档、提交错误报告和写教程 / 书籍 (尤其是这本书的作者!) 的人们, 你们的贡献对任何开源项目的成功至关重要, 尤指像 OpenCV 这样复杂庞大的开源软件。

Haojian Jin 位于加拿大 Sunnyvale 雅虎实验室的软件工程师 / 研究员。他主要开发移动新设备上 (或最少硬件更改) 的新系统。为了创建当今不存在的事物, 他花费了大量时间来研究信号处理、计算机视觉、机器学习和自然语言处理, 并以有趣的方式来使用它们。可在 <http://shift-3.com/> 上找到更多关于他的信息。

Adrian Rosebrock 一位作家, 也是 <http://www.pyimagesearch.com/> 的博主。他有马里兰大学计算机科学博士学位, 侧重于计算机视觉和机器学习的研究。

他曾在癌症研究所从事通过乳腺图像来自动预测乳腺癌的危险因素的研究。他还写了《Practical Python and OpenCV》一书 (<http://pyimg.co/x7ed5>), 这本书介绍如何利用 Python 和 OpenCV 来构建现实世界中的计算机视觉应用。

About the Translators 译者简介

刘波 博士，重庆工商大学计算机科学与信息工程学院教师，主要从事机器学习理论、计算机视觉和最优化技术研究，同时对 Hadoop 和 Spark 平台上的大数据分析感兴趣，也对 Linux 编程和 Oracle 数据库感兴趣。

苗贝贝 硕士，北京工商大学计算机与信息工程学院研究生，主要从事机器学习理论、时间序列动力学特征分析及应用的研究，对基于 Python 的计算机视觉分析有浓厚的兴趣。

史斌 2015 年本科毕业于电子科技大学计算机学院，目前就职于成都知数科技有限公司，主要从事数据爬取、数据处理、平台运维等工作，熟悉 Python、Linux shell，同时热爱计算机视觉编程，熟悉 Python 下的 OpenCV 编程。

目 录 *Contents*

译者序

前言

作者简介

审校者简介

译者简介

第 1 章 安装 OpenCV	1
1.1 选择和使用合适的安装工具	2
1.1.1 在 Windows 上安装	2
1.1.2 在 OS X 系统中安装	6
1.1.3 在 Ubuntu 及其衍生版本中安装	11
1.1.4 在其他类 Unix 系统中安装	12
1.2 安装 Contrib 模块	13
1.3 运行示例	13
1.4 查找文档、帮助及更新	14
1.5 总结	15
第 2 章 处理文件、摄像头和图形用户界面	16
2.1 基本 I/O 脚本	16
2.1.1 读 / 写图像文件	16
2.1.2 图像与原始字节之间的转换	19
2.1.3 使用 <code>numpy.array</code> 访问图像数据	20
2.1.4 视频文件的读 / 写	22

2.1.5 捕获摄像头的帧	23
2.1.6 在窗口显示图像	24
2.1.7 在窗口显示摄像头帧	25
2.2 Cameo 项目 (人脸跟踪和图像处理)	26
2.3 Cameo——面向对象的设计	27
2.3.1 使用 managers.CaptureManager 提取视频流	27
2.3.2 使用 managers.WindowManager 抽象窗口和键盘	32
2.3.3 cameo.Cameo 的强大实现	33
2.4 总结	34
第 3 章 使用 OpenCV 3 处理图像	36
3.1 不同色彩空间的转换	36
3.2 傅里叶变换	37
3.2.1 高通滤波器	37
3.2.2 低通滤波器	39
3.3 创建模块	39
3.4 边缘检测	40
3.5 用定制内核做卷积	41
3.6 修改应用	43
3.7 Canny 边缘检测	44
3.8 轮廓检测	45
3.9 边界框、最小矩形区域和最小闭圆的轮廓	46
3.10 凸轮廓与 Douglas-Peucker 算法	48
3.11 直线和圆检测	50
3.11.1 直线检测	50
3.11.2 圆检测	51
3.12 检测其他形状	52
3.13 总结	52
第 4 章 深度估计与分割	53
4.1 创建模块	53
4.2 捕获深度摄像头的帧	54

4.3	从视差图得到掩模	56
4.4	对复制操作执行掩模	57
4.5	使用普通摄像头进行深度估计	59
4.6	使用分水岭和 GrabCut 算法进行物体分割	63
4.6.1	用 GrabCut 进行前景检测的例子	64
4.6.2	使用分水岭算法进行图像分割	66
4.7	总结	69
第 5 章	人脸检测和识别	70
5.1	Haar 级联的概念	70
5.2	获取 Haar 级联数据	71
5.3	使用 OpenCV 进行人脸检测	72
5.3.1	静态图像中的人脸检测	72
5.3.2	视频中的人脸检测	74
5.3.3	人脸识别	76
5.4	总结	82
第 6 章	图像检索以及基于图像描述符的搜索	83
6.1	特征检测算法	83
6.1.1	特征定义	84
6.1.2	使用 DoG 和 SIFT 进行特征提取与描述	86
6.1.3	使用快速 Hessian 算法和 SURF 来提取和检测特征	89
6.1.4	基于 ORB 的特征检测和特征匹配	91
6.1.5	ORB 特征匹配	93
6.1.6	K-最近邻匹配	95
6.1.7	FLANN 匹配	96
6.1.8	FLANN 的单应性匹配	99
6.1.9	基于文身取证的应用程序示例	102
6.2	总结	105
第 7 章	目标检测与识别	106
7.1	目标检测与识别技术	106

7.1.1	HOG 描述符	107
7.1.2	检测人	112
7.1.3	创建和训练目标检测器	113
7.2	汽车检测	116
7.2.1	代码的功能	118
7.2.2	SVM 和滑动窗口	122
7.3	总结	134
第 8 章 目标跟踪		135
8.1	检测移动的目标	135
8.2	背景分割器: KNN、MOG2 和 GMG	138
8.2.1	均值漂移和 CAMShift	142
8.2.2	彩色直方图	144
8.2.3	返回代码	146
8.3	CAMShift	147
8.4	卡尔曼滤波器	149
8.4.1	预测和更新	149
8.4.2	范例	150
8.4.3	一个基于行人跟踪的例子	153
8.4.4	Pedestrian 类	154
8.4.5	主程序	157
8.5	总结	159
第 9 章 基于 OpenCV 的神经网络简介		160
9.1	神经网络	160
9.2	神经网络的结构	161
9.2.1	网络层级示例	162
9.2.2	学习算法	163
9.3	OpenCV 中的 ANN	164
9.3.1	基于 ANN 的动物分类	166
9.3.2	训练周期	169
9.4	用神经网络进行手写数字识别	170

9.4.1	MNIST——手写数字数据库·····	170
9.4.2	定制训练数据·····	170
9.4.3	初始参数·····	171
9.4.4	迭代次数·····	171
9.4.5	其他参数·····	171
9.4.6	迷你库·····	172
9.4.7	主文件·····	175
9.5	可能的改进和潜在的应用·····	180
9.5.1	改进·····	180
9.5.2	应用·····	181
9.6	总结·····	181