



[美] Joseph Howse 著 张弢 译

# OpenCV项目开发实战

OpenCV for Secret Agents

清华大学出版社



# OpenCV 项目开发实战

[美] Joseph Howse 著

张 骏 译

清华大学出版社

北 京

## 内 容 简 介

本书详细阐述了与 OpenCV 移动应用程序相关的基本解决方案, 主要包括搜索世界范围内的豪华建筑、利用姿势控制应用程序、车载后视摄像头和危险警告、利用运动放大相机查看心率、根据素描画创建物理模拟等内容。此外, 本书还提供了相应的示例、代码, 以帮助读者进一步理解相关方案的实现过程。

本书适合作为高等院校计算机及相关专业的教材和教学参考书, 也可作为相关开发人员的自学教材和参考手册。

Copyright © Packt Publishing 2015. First published in the English language under the title *OpenCV for Secret Agents*.

Simplified Chinese-language edition © 2016 by Tsinghua University Press. All rights reserved.

本书中文简体字版由 Packt Publishing 授权清华大学出版社独家出版。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。

北京市版权局著作权合同登记号 图字: 01-2016-5194

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目 (CIP) 数据

OpenCV 项目开发实战 / (美) 约瑟夫·豪斯 (Joseph Howse) 著; 张弢译. —北京: 清华大学出版社, 2016

书名原文: OpenCV for Secret Agents

ISBN 978-7-302-44265-3

I. ①O… II. ①约… ②张… III. ①图像处理软件-程序设计 IV. ①TP391.41

中国版本图书馆 CIP 数据核字 (2016) 第 153184 号

责任编辑: 贾小红

封面设计: 刘超

版式设计: 牛瑞瑞

责任校对: 王云

责任印制: 沈露

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈: 010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 装 者: 清华大学印刷厂

经 销: 全国新华书店

开 本: 185mm×230mm 印 张: 15.25 字 数: 315 千字

版 次: 2016 年 9 月第 1 版 印 次: 2016 年 9 月第 1 次印刷

印 数: 1~3000

定 价: 59.00 元

# 译者序

当今移动市场，Android、iOS 操作系统风头正劲。然而，计算机视觉同样无处不在，移动设备、安全系统、制造检验系统、医学图像分析、无人机等都可以见到它的踪影。

OpenCV 的全称是 Open Source Computer Vision Library。OpenCV 是一个开源的跨平台计算机视觉库，可以运行在 Linux、Windows 和 Mac OS 等操作系统上，且具有轻量、高效等特征——由一系列 C 函数和少量 C++ 类构成，同时提供了 Python、Ruby、MATLAB 等语言的接口，实现了图像处理和计算机视觉方面的很多通用算法。OpenCV 致力于真实世界的实时应用，通过优化的 C 代码对其执行速度带来了可观的提升，并且可以通过购买 Intel 的 IPP 高性能多媒体函数库（Integrated Performance Primitives）得到更快的处理速度。

本书以案例的形式（其中包括 6 个开发项目）介绍 OpenCV 的新特性和 C++ 新接口，案例中包含具体的代码与详细的说明，并很好地平衡了基础知识与进阶内容，要求读者具有基础的 C++ 知识。

在本书的翻译过程中，除张弢之外，米玥、潘冰玉、皮雄飞、史云龙、孙年果、王典、王梅、程聪、朱利平、解宝香、张欣欣、李宝金等人也参与了本书的翻译工作，在此一并表示感谢。

译者

# 前 言

计算机视觉系统的身影出现于多种场合，例如在夜间定位北冰洋中的冰山；针对亚马逊雨林的火灾、瘟疫或树木砍伐情况拍摄的航空地图；在港口或机场，视觉系统可对嫌疑犯和违禁品进行监视；在深暗的马里亚纳海沟，视觉系统可对潜艇的行进路线进行导航；在手术室中，该系统可实现手术进程的可视化效果，并监视病人的健康状态；另外，此类系统还可作为热追踪防空导弹的转向系统。

出于各方面的原因，人们很难亲身经历上述各种场合。然而，精彩的故事往往可假设某些极端环境，在一些苛刻的条件下，人们需要特殊工具的辅助。恰当地说，当代小说中深受人们喜爱的人物和普通人并无两样（英俊但并不夸张，聪明却非遥不可及），他们通常身着漂亮的西服并为英国政府工作，通常只会选同一种饮料，欣赏同一类型的女人，甚至说话口气也从未改变。他们装备齐全并被派往执行危险的任务。

“我是邦德，詹姆斯·邦德”。

本书题材源自谍报小说 007，并讨论其中所涉及的科技与技术。小说的主要角色詹姆斯·邦德擅长于侦查、伪装、智能设备的应用、图像捕捉技术，甚至涉及某些专业的计算机视觉技术。在本书丰富的想象力以及技术支持下，读者同样有机会晋升为下一代的装备专家，并与詹姆斯·邦德的御用工程师 Q 博士一争高下。

## 本书内容

第 1 章讨论 Python 开发环境，Windows 环境下的 Android 开发环境，Mac 或 Linux 系统。本章还将在 Windows 或 Mac 操作系统中安装 Unity 开发环境。

第 2 章将根据色彩方案对建筑图像进行分类，包括豪华酒店或斯大林时期的公寓建筑。本章采用了搜索引擎中的分类器，并对搜索到的图像结果进行标记。

第 3 章讨论人脸或猫脸的检测和识别技术，并将其作为一种警示方法。在本章中，任务之一便是是否找到了 Ernst Stavro Blofeld（电影 007 中著名的奸角），以及他怀中的那只蓝眼安哥拉猫。

第 4 章讨论运动检测和姿势识别技术，并可作为智能手机上的一种游戏控制方式。即使在无人知晓的情况下，手机设备也可检测到詹姆斯·邦德的点头动作。

第 5 章将对汽车头灯进行检测，对其颜色分类，估算与车灯之间的距离，并向驾驶

员提供信息反馈。例如，后方车辆是否在紧紧跟随我们的汽车？

第 6 章采用实时方式对视频摄入内容中的运动行为进行放大，以使心跳或气息处于可见状态，并以此考查某人的心智状态。

第 7 章将对物理运动进行模拟。其中，物理和时控机制将作为重点内容加以讨论。

## 背景知识

本书支持多种操作系统作为开发环境，包括 Windows XP 及其后续版本，Mac OS X 10.6 及其后续版本，Debian Wheezy、Raspbian、Ubuntu 12.04 及其后续版本，Linux Mint 13 及其后续版本，Fedora 18 及其后续版本，CentOS 7 及其后续版本，openSUSE 13.1 及其后续版本。

本书根据下列需求实现了 6 个项目，其中包括：

- 4 个项目可运行于 Windows、Mac 或 Linux 操作系统中，并需要网络摄像头的支持。作为可选项，这一类项目还可使用 Raspberry Pi 系统，或者其他基于 Linux 操作系统的单片机上。
- 1 个项目可运行于 Android 2.2 及其后续版本中，并需要前置摄像头的支持（大多数 Android 设备均可满足这一条件）。
- 1 个项目可运行于 Android 2.3 及其后续版本中，并需要后置摄像头以及重力传感器的支持（大多数 Android 设备均可满足这一条件）。在开发过程中，该项目需要使用 Windows 或 Mac 操作系统。另外，其中涉及的游戏开发软件大约需要花费 75 美元。

本书提供了所需库和工具安装的全部信息，另外还包括 Raspberry Pi 的安装信息。

## 本书适用读者

本书引领读者将计算机视觉技术付诸于实践，这一过程充满了乐趣。本书假设读者已经熟悉了 2D 图形学概念、面向对象语言、GUI、网络机制以及命令行操作，且并不要求读者了解特定库或操作系统平台。另外，本书将从开发环境构建至应用程序发布等多个方面对 OpenCV 予以介绍。

如果读者计划学习多种技术并将其进行有机的结合，本书将引领读者深入理解多种系统类型，以及与计算机视觉相关的应用领域，并通过多种方案检测、识别并增强人脸、物体和运动行为。

## 本书约定

本书涵盖了多种文本风格，进而对不同类型的信息加以区分。下列内容展示了对应示例及其具体含义。

文本中的代码、数据库表名称、文件名称、文件名、文件扩展名、路径名、伪 URL、用户输入以及推特用户定位采用如下方式表示：

"You can edit /etc/modules to check whether bcm2835-v4l2 is already listed there."

代码块则通过下列方式设置：

```
set PYINSTALLER=C:\PyInstaller\pyinstaller.py

REM Remove any previous build of the app.
rmdir build /s /q

rmdir dist /s /q

REM Train the classifier.
python HistogramClassifier.py
```

命令行输入和输出的显示方式如下所示：

```
$ echo "bcm2835-v4l2" | sudo tee -a /etc/modules
```

 图标表示较为重要的概念，而  图标则表示提示或相关操作技巧。

## 读者反馈和客户支持

欢迎读者对本书的建议或意见予以反馈，以进一步了解读者的阅读喜好。反馈意见对于我们来说十分重要，以便改进我们日后的工作。

对此，读者可向 [feedback@packtpub.com](mailto:feedback@packtpub.com) 发送邮件，并以书名作为邮件标题。

若读者针对某项技术具有专家级的见解，抑或计划撰写书籍或完善某部著作的出版工作，则可阅读 [www.packtpub.com/authors](http://www.packtpub.com/authors) 中的 **author guide** 一栏。

作为本书的读者，我们将对每一名用户提供竭诚的服务。

## 资源下载

读者可访问 <http://www.packtpub.com> 下载本书中的示例代码文件；或者访问 <http://www.packtpub.com/support>，经注册后可直接通过邮件方式获取相关文件。另外，读者还可访

问作者的个人网站，并获取最新的示例代码，对应网址为 <http://nummist.com/opencv/>。

## 勘误表

尽管我们在最大程度上做到尽善尽美，但错误依然在所难免。如果读者发现谬误之处，无论是文字错误抑或是代码错误，还望不吝赐教。对于其他读者以及本书的再版工作，这将具有十分重要的意义。对此，读者可访问 <http://www.packtpub.com/submit-errata>，选取对应书籍，单击 `ErrataSubmissionForm` 超链接，并输入相关问题的详细内容。经确认后，输入内容将被提交至网站，或添加至现有勘误表中（位于该书籍的 `Errata` 部分）。

另外，读者还可访问 <http://www.packtpub.com/books/content/support> 查看之前的勘误表。在搜索框中输入书名后，所需信息将显示于 `Errata` 项中。

## 版权须知

一直以来，互联网上的版权问题从未间断，Packt 出版社对此类问题异常重视。若读者在互联网上发现本书任意形式的副本，请告知网络地址或网站名称，我们将对此予以处理。

关于盗版问题，读者可发送邮件至 [copyright@packtpub.com](mailto:copyright@packtpub.com)。

对于作者的爱护，我们表示衷心的感谢，并于日后向读者呈现更为精彩的作品。

## 问题解答

若读者对本书有任何疑问，均可发送邮件至 [questions@packtpub.com](mailto:questions@packtpub.com)，我们将竭诚为您服务。除此之外，读者还可直接与本书作者进行联系，作者邮箱为 [josephhowse@nummist.com](mailto:josephhowse@nummist.com)；或者访问其个人网站 <http://nummist.com/opencv/>，并查看与本书相关的常见问题解答。



# 目 录

第 1 章 准备工作 .....	1
1.1 构建开发环境 .....	2
1.1.1 Windows 操作系统 .....	3
1.1.2 Mac 操作系统 .....	8
1.1.3 Debian Wheezy 及其衍生工具 .....	12
1.1.4 Fedora 及其衍生工具 .....	14
1.1.5 openSUSE 及其衍生工具 .....	14
1.1.6 Tegra Android 开发包 .....	15
1.1.7 Unity 引擎 .....	23
1.2 构建 Raspberry Pi .....	25
1.3 获取 OpenCV 的文档、帮助和更新 .....	30
1.4 Raspberry Pi 的替代方案 .....	31
1.5 本章小结 .....	32
第 2 章 搜索世界范围内的豪华建筑 .....	33
2.1 规划 Luxocator 应用程序 .....	33
2.2 创建、比较并存储直方图 .....	35
2.3 利用参考图像训练分类器 .....	40
2.4 从 Web 中获取图像 .....	41
2.5 从 Bing 图像搜索中获取图像 .....	43
2.6 针对应用程序准备图像和资源 .....	48
2.7 整合至 GUI 中 .....	50
2.8 构建 Luxocator 发布版 .....	58
2.9 本章小结 .....	61
第 3 章 人物和猫脸识别 .....	62
3.1 理解机器学习 .....	63
3.2 规划交互式识别应用程序 .....	64
3.3 理解 Haar 级联和 LBPH .....	65

3.4	实现交互式识别器应用程序 .....	69
3.5	规划猫脸检测模型 .....	82
3.6	针对猫脸检测模型实现训练脚本 .....	84
3.7	规划 Angora Blue 应用程序 .....	96
3.8	实现 Angora Blue 应用程序 .....	97
3.9	构建 Angora Blue 发布版本 .....	103
3.10	进一步讨论 .....	103
3.11	本章小结 .....	104
<b>第 4 章</b>	<b>利用姿势控制应用程序 .....</b>	<b>105</b>
4.1	规划 Goldgesture 应用程序 .....	105
4.2	理解光流 .....	107
4.3	创建 Eclipse 工作区 .....	109
4.4	获取级联文件和音频文件 .....	115
4.5	确定应用程序需求 .....	116
4.6	将摄像头视图设置为主视图 .....	117
4.7	跟踪手势 .....	118
4.8	播放音频片段 .....	120
4.9	捕捉图像并跟踪脸部内容 .....	124
4.10	本章小结 .....	140
<b>第 5 章</b>	<b>车载后视摄像头和危险警告 .....</b>	<b>141</b>
5.1	规划 Living Headlights 应用程序 .....	142
5.2	光源的斑点检测 .....	143
5.3	距离估算（简单方案） .....	146
5.4	实现 Living Headlights 应用程序 .....	148
5.5	测试 Living Headlights 应用程序 .....	161
5.6	测试车载 Living Headlights 应用程序 .....	164
5.7	本章小结 .....	168
<b>第 6 章</b>	<b>利用运动放大相机查看心率 .....</b>	<b>169</b>
6.1	规划 Lazy Eyes 应用程序 .....	170
6.2	理解欧拉影像放大算法 .....	171
6.3	利用快速傅里叶转换从视频中获取重复信号 .....	172

---

6.4	利用图像金字塔合成两幅图像 .....	176
6.5	实现 Lazy Eyes 应用程序 .....	177
6.6	针对各种运动行为配置并测试应用程序 .....	185
6.7	其他光源 .....	190
6.8	本章小结 .....	191
<b>第 7 章</b>	<b>根据素描画创建物理模拟 .....</b>	<b>192</b>
7.1	规划 Rollingball 应用程序 .....	193
7.2	检测圆形和直线 .....	195
7.3	针对 Unity 设置 OpenCV .....	197
7.4	配置并构建 Unity 项目 .....	200
7.5	在 Unity 中生成 Rollingball 场景 .....	201
7.6	创建 Unity 资源并将其添加至场景中 .....	204
7.6.1	编写着色器并创建材质 .....	204
7.6.2	创建物理材质 .....	206
7.6.3	创建预置组件 .....	208
7.6.4	编写第一个 Unity 脚本 .....	211
7.6.5	编写 Rollingball 主脚本文件 .....	212
7.7	整合与测试 .....	231
7.8	本章小结 .....	231

# 第1章 准备工作

银幕中的詹姆斯·邦德从未赤手空拳地与敌人战斗，观众可看到他驾驶水陆两用车的英姿，或者肩背着火箭背包。邦德总会展示某些最新的技术装备，且从不惧怕将其摧毁，这也令 Q 博士和工程师感到万分沮丧。

作为 21 世纪初的软件开发人员，我们曾见证了新平台应用的爆发阶段。家庭中，Windows、Mac、iOS 以及 Android 设备随处可见，父母的工作场所也会采用不同的操作平台。另外，孩子们可能会拥有 3 种不同平台的游戏机，如果算上移动版本，这一数字会达到 5 种之多。除此之外，儿童们手中的 LeapFrog 平板学习电脑也屡见不鲜。其他诸如智能控温杯也越来越多地出现于人们的日常生活中。

对于新平台的尝试，我们不必感到恐惧，且应致力于了解其全新的组合方式，毕竟大多数用户已经参与其中。

本书主要讨论多平台开发，展示了奇特而有趣的应用程序，并出现于许多意想不到的地方。其中使用了多种计算机传感器，特别是计算机视觉，进而使得不同的设备组合展现出了新的活力。

在特工 007 展示其装备之前，通常会受命于 Q 博士的作战指示，本章即饰演 Q 博士这一角色，并讨论系统的构建过程。

在本章结束时，针对 Windows 平台利用 C++ 语言或 Python 语言，抑或针对 Android 平台采用 C++ 语言或 Java 语言，读者可获取到全部开发工具。另外，读者还可成为 Raspberry Pi 单板计算机的新用户（这一附加应为可选项），并对 Unity 这一整合至 OpenCV 中的游戏引擎有所了解。

读者不必对本章涉及的操作范围有所担心，可以确定的是，实际应用中并不会使用全部开发工具，某一项目中也不会采用全部组合。虽然 Q 博士和我会一次性地介绍多种技术的构建过程，但读者依然可暂时略过本章，并在项目实际应用中再对本章内容进行回顾。



由于涉及基础 OpenCV 的构建过程以及相关参考资料，本章会涵盖某些入门书籍中的专家意见，例如 OpenCV Computer Vision with Python 和 Android Application Programming with OpenCV，分别由 Packt 出版社出版。全部内容均经过重新测试并予以更新，且最终扩展至较新的 OpenCV 版本以及附加的操作系统。除此之外，在附加硬件和游戏引擎方面也会包含全新的内容。

## 1.1 构建开发环境

读者可在桌面级平台、笔记本甚至是廉价的 Raspberry Pi 设备（稍后将对此类设备进行介绍）上开发自己的 OpenCV 应用程序，大多数应用程序的内存空间占用不会超过 128MB，因而依然可运行于较早的或小概率设备上（但运行速度会较慢）。

本书假设读者在开发设备上安装了下列操作系统：

- Windows XP 或后续版本。
- Mac OS 10.6 或后续版本。
- Debian Wheezy 或如下衍生工具：
  - Raspbian。
  - Ubuntu 12.04 或后续版本。
  - Linux Mint 13 或后续版本。
- Fedora 18 或后续版本，或者如下衍生工具：
  - Red Hat Enterprise Linux (RHEL) 7 或后续版本。
  - CentOS 7 或后续版本。
- openSUSE 13.1 或后续版本，抑或某一衍生工具。

除此之外，还可安装其他 UNIX 操作系统，但本书不会全部涉及。

用户可能持有 USB 网络摄像头以及所需的驱动程序，大多数网络摄像头均包含 Windows 和 Mac 平台上的安装驱动程序说明。Linux 发行版通常会绑定 USB Video Class (UVC) Linux 驱动程序，以支持多种网络摄像头，具体内容可访问 <http://www.ideasonboard.org/uvc/#devices>。

同时，本章还将构建下列组件：

- C++ 开发环境。在 Windows 操作系统上，本书将采用 Visual Studio 2010 或其后续版本。关于基于 MinGW 和 Code::Blocks IDE 的 OpenCV 构建过程，读者还可阅读 Kevin Hughes 编写的教程，其网址为 <http://kevinhughes.ca/tutorials/opencv-install-on-windows-with-codeblocks-and-mingw/>。Mac 操作系统中则使用 Xcode，Linux 操作系统上则采用标准配置 GCC。
- 在 Mac 操作系统上，将使用第三方包管理器，以安装相关库及其依赖关系。此处将使用 MacPorts 或 Homebrew。
- Python 2.7 开发环境。在本书编写时，Python 2.7 则是 OpenCV 分支所支持的最

新版本（该分支也支持 Python 2.6）。

- ❑ 常用库，例如 NumPy（用于数值函数）、SciPy（用于数值和科学函数）、Requests（用于 Web 请求）以及 wxPython（用于跨平台 GUI）。
- ❑ PyInstaller 则是一类跨平台工具，用于捆绑 Python 脚本、库以及数据作为重新分发的应用程序，以使用户设备无须安装 Python。在本书中，Python 项目的重新分发操作则可视为一类可选项，第 2 章将对此加以讨论。由于 PyInstaller（类似于其他 Python 捆绑工具）在不同操作系统、Python 版本以及库版本间展现了完全不同的行为，因而用户需要执行自身的测试以及调试。在 Raspberry Pi 或其他 ARM 系统上，PyInstaller 并未得到较好的支持。
- ❑ 针对特定的桌面硬件，基于 C++ 语言和 Python 语言的 OpenCV 支持优化行为。在本书编写时，OpenCV 2.4.x 可视为一类稳定分支，本书将围绕该版本展开讨论。
- ❑ 针对 Android 硬件，基于 C++ 语言和 Java 语言的 OpenCV 支持优化行为，特别地，本书将使用 Tegra Android 开发包（TADP）OpenCV 版本。在本书编写时，TADP 3.0r4 为最新版本。
- ❑ TADP 内嵌了 Android 开发环境，其中包括 Eclipse、ADT、Android SDK 和 Android NDK。
- ❑ Windows 或 Mac 操作系统均可使用 Unity 3D 游戏引擎。



Eclipse 占用较大的内存空间。即使采用 Raspberry Pi 开发桌面和 Pi 应用程序，也需要使用包含较大 RAM 的设备。

下面将分别讨论 3 个平台相关构造步骤、基于 TADP 的平台无关构造步骤，以及对 Unity 的平台无关构造步骤。

### 1.1.1 Windows 操作系统

在 Windows 操作系统中，可选择构建 32 位开发环境（应用程序兼容于 32 位和 64 位 Windows 操作系统）或 64 位开发环境（优化后的应用程序仅兼容于 64 位 Windows 操作系统）。最新的 OpenCV 版本支持 32 位和 64 位 Windows 操作系统。

除此之外，还可选择使用二进制安装包或根据源文件编译 OpenCV。针对本书中的 Windows 应用程序，二进制安装包提供了一切所需内容。但本书依然会对源文件编译选项加以讨论，这可确保支持某些附加特性，例如 Kinect 和 Asus 深度相机，进而完善特殊任务的开发或者满足其他书籍中的项目要求。



对于采用深度相机的 OpenCV 项目，读者可参考本人编写的 OpenCV Computer Vision with Python 一书，该书由 Packt Publishing 出版社出版。

如果不考虑 OpenCV 的获取方式，当前则采用通用 C++ 开发环境，以及通用的 Python 2.7 开发环境，并通过二进制安装包对其构建。

作为本书的 C++ 开发环境，此处将采用 Visual Studio 2010 或其后续版本。读者可使用所购买的安装介质，或者进入 <http://www.visualstudio.com/en-us/downloads/downloadvisualstudio-vs.aspx> 下载界面，并针对下列某一项下载并运行安装包：

- 免费的 Visual C++ 2010 Express。
- 针对 Windows 桌面系统且免费的 Visual Studio Express 2013。
- 任意付费版本，其中包含了 90 天的试用期。

如果安装包显示可选的 C++ 组件，可选择安装全部内容。待安装包运行完毕后，重启设备。



如果计划根据源文件编译 OpenCV，建议使用 Visual Studio 2010（不建议使用其后续版本）。在本书编写时，OpenCV 及其某些可选依赖关系无法通过 Visual Studio 2012 或 Visual Studio 2013 实现简单的编译。

Python 2.7 安装包位于 <http://www.python.org/getit/>，下载并运行最新的 32 位或 64 位 Python 2.7 版本。

若运行 Python 脚本并使用默认最新的 Python 2.7 版本，可编辑系统 Path 变量并添加“;C:\Python2.7”（假设 Python 安装于默认路径），并移除之前的 Python 路径，例如“;C:\Python2.6”。退出并重新进行登录系统（或重启系统）。

相应地，假设针对 NumPy、SciPy 和 wxPython 使用二进制安装包，下载并运行基于 Python 2.7 的、最新且稳定的库版本，此类安装包可能位于下列位置。

- NumPy: 官方安装包仅支持 32 位，并位于 <http://sourceforge.net/projects/numpy/files/NumPy/>。非官方版 64 位则位于 <http://www.lfd.uci.edu/~gohlke/pythonlibs/#numpy>。
- SciPy: 官方安装包仅支持 32 位，并位于 <http://sourceforge.net/projects/scipy/files/scipy/>。非官方版 64 位则位于 <http://www.lfd.uci.edu/~gohlke/pythonlibs/#scipy>。
- wxPython: 对应下载位置为 <http://www.wxpython.org/download.php>。本书中的应用程序均在 wxPython 2.8/2.9/3.0 版本下测试并通过。如果读者对此有所怀疑，则可选择 3.0 版本。另外，如果读者选择 2.8 版本，则应获取其 Unicode 版本。

对于非二进制安装包，可在 <https://github.com/kennethreitz/requests/archive/master.zip> 处下载最新版本的软件包，将其解压至任意位置并记为 <unzip\_destination>。打开 Command Prompt 并运行下列命令：

```
> cd <unzip_destination>
> python setup.py install
```

随后，由于 PyInstaller 可视为工具集而非库，因而可将 PyInstaller 置于任意快捷位置。可从 <http://www.pyinstaller.org/> 处下载最新的发布版本，并将其解压至 C:\PyInstaller 或其他选取位置。

下面将构建 OpenCV，并选择安装其他计算机视觉库。

### 1. 利用二进制安装包安装 Windows 上的 OpenCV

从 <http://opencv.org/downloads.html> 处下载 OpenCV 自解压 ZIP 文件，选取包含 32 位和 64 位库文件的最新版本。双击自解压 ZIP 文件，当出现系统提示时，输入任意目标文件夹，进而可记为 <unzip\_destination>。相应地，名为 <opencv\_unzip\_destination>\opencv 的子文件夹将被创建。

随后，可将 <opencv\_unzip\_destination>\opencv\build\python\2.7\x86\cv2.pyd（32 位）或者 <opencv\_unzip\_destination>\opencv\build\python\2.7\x64\cv2.pyd（64 位）复制至 C:\Python2.7\Lib\site-packages 下（假设已将 Python 2.7 安装至默认位置）。至此，Python 2.7 即可发现 OpenCV。

对应代码示例位于 <unzip\_destination>/opencv/samples 中。

当前已具备了开发 Windows 环境下 OpenCV 开发的一切条件。对于 Android 环境，则需要构建 TADP，稍后将对此予以介绍。

### 2. 利用 CMake 和编译器安装 Windows 上的 OpenCV

为了获取 OpenCV 中的额外特性，可安装多个第三方库，其中包括 OpenNI（支持深度相机）、SensorKinect（向 OpenNI 中添加 Kinect 支持）以及 TBB（支持 Intel 多重处理）。待第三方库安装完毕后，则可配置并构建 OpenCV。最后，还应确保 C++ 和 Python 环境可发现 OpenCV。



OpenCV 的二进制安装包支持 TBB，但并未提供相应的 OpenNI 和 SensorKinect 支持。因此，针对 Windows 环境下的深度相机，需要根据源文件编译 OpenCV。虽然本书并不涉及深度相机这一内容，但 OpenCV Computer Vision with Python 一书中对此有所提及，读者也可将此应用于自己的开发项目中。



下列内容显示了 Windows 环境下 OpenCV 的详细构建过程：

- ❑ 从 <http://www.cmake.org/cmake/resources/software.html> 处下载并安装最新的 CMake 版本。即使采用 64 位版本库和编译器，依然可兼容 32 位 CMake。当安装包请求调整 PATH 时，可针对全部用户选择 Add CMake 于系统 PATH 中，或者针对当前用户以及系统 PATH 选取 Add CMake
- ❑ 作为可选项，可下载并安装 OpenNI 1.5.4.0 的开发版本，对应位置为 <http://www.nummist.com/opencv/openniwin32-1.5.4.0-dev.zip>（32 位版本），或 <http://www.nummist.com/opencv/openni-win64-1.5.4.0-dev.zip>（64 位版本）。除了 OpenNI 的 1.5.4.0 开发版本之外，不建议使用其他版本。根据个人经验，其中的某些版本无法与 OpenCV 实现良好的协同工作。
- ❑ 作为可选项，下载并安装 SensorKinect 0.93（禁止使用其他版本），对应位置为 <https://github.com/avin2/SensorKinect/blob/unstable/Bin/SensorKinect093-Bin-Win32-v5.1.2.1.msi?raw=true>（32 位），或者 <https://github.com/avin2/SensorKinect/blob/unstable/Bin/SensorKinect093-Bin-Win64-v5.1.2.1.msi?raw=true>（64 位）。除 SensorKinect 0.93 之外的其他版本则不在使用范围之列。根据个人经验，某些版本无法与 OpenCV 实现良好的协同工作。
- ❑ 从 <http://opencv.org/downloads.html> 处下载 OpenCV 自解压文件，选取包含 32 位和 64 位库文件的最新版本。双击自解压 ZIP 文件，当出现系统提示时，可输入任意目标文件夹并记为 <opencv\_unzip\_destination>。相应地，将创建名为 <opencv\_unzip\_destination>\opencv 的子文件夹。
- ❑ 从 <https://www.threadingbuildingblocks.org/download> 处下载最新的 TBB 版本，其中包含了 32 位和 64 位库文件。将其解压至任意位置并记为 <tbb\_unzip\_destination>。
- ❑ 打开命令行提示，创建文件夹以存储当前构建内容，如下所示：

```
> mkdir <build_folder>
```

将当前目录调整至新创建的文件夹，如下所示：

```
> cd <build_folder>
```

- ❑ 待全部依赖关系构建完毕后，即可配置 OpenCV 的构建系统。为了进一步理解全部配置选项，读者可阅读 <opencv\_unzip\_destination>\opencv\sources\CMakeLists.txt 中的示例代码。然而，作为示例，此处仅使用发布版本中的选项，包括 Python 绑定、基于 OpenNI 和 SensorKinect 的深度相机支持，以及通过 TBB 实现的多重处理操作。