

Android

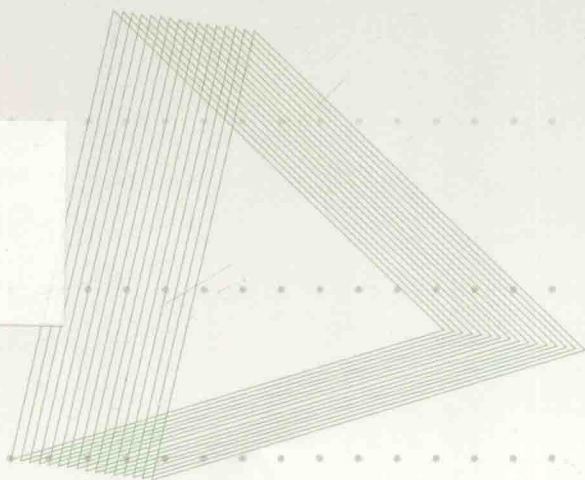
Application Programming with

OpenCV

Android OpenCV应用程序设计

Joseph Howse © 著

赵雷 © 译



清华大学出版社

Android OpenCV 应用程序设计

Joseph Howse 著

赵 雷 译

清华大学出版社

北 京

内 容 简 介

本书详细阐述了与 Android OpenCV 程序设计相关的基本解决方案, 主要包括构建 OpenCV, 与相机帧协同工作, 添加图像效果, 图像识别和跟踪, 图像跟踪与 3D 渲染之间的整合操作等内容。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 2013. First published in the English language under the title *Android Application Programming with OpenCV*.

Simplified Chinese-language edition © 2014 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: 01-2014-3939

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

Android OpenCV 应用程序设计/ (美) 豪斯 (Howse, J.) 著; 赵雷译. —北京: 清华大学出版社, 2015
书名原文: Android application programming with OpenCV
ISBN 978-7-302-38443-4

I. ①A… II. ①豪… ②赵… III. ①移动终端-应用程序-程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字 (2014) 第 260876 号

责任编辑: 钟志芳

封面设计: 刘超

版式设计: 文森时代

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

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 北京密云胶印厂

经 销: 全国新华书店

开 本: 185mm×230mm 印 张: 7.25 字 数: 147 千字

版 次: 2015 年 1 月第 1 版 印 次: 2015 年 1 月第 1 次印刷

印 数: 1~4000

定 价: 29.00 元

产品编号: 060085-01

译 者 序

Android 是一种基于 Linux 的自由及开放源代码的操作系统，主要使用于移动设备，如智能手机和平板电脑，由 Google 公司和开放手机联盟领导及开发。在优势方面，Android 平台首先就是其开发性，开放的平台允许任何移动终端厂商加入到 Android 联盟中来。显著的开放性可以使其拥有更多的开发者，随着用户和应用的日益丰富，一个崭新的平台也将很快走向成熟。

OpenCV 的全称是：Open Source Computer Vision Library。OpenCV 是一个基于（开源）发行的跨平台计算机视觉库，可以运行在 Linux、Windows 和 Mac OS 操作系统上。它轻量级而且高效——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 and 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得到更快的处理速度。

OpenCV 和 Android 库之间的整合过程充分体现了广泛的应用程序特征，本书将会对此予以深入讨论。届时，读者可通过可复用的类库扩展或调整后续的计算机视觉项目，并根据已有的开发环境和知识编写更为丰富的应用程序。

在本书的翻译过程中，除赵雷之外，安娇娇、李芳芳、贾俊琴、高阳、陈西红、顾庆娟、聂芳、陈西红、陈丽丽也参与了本书的翻译工作，在此一并表示感谢。

译 者

前 言

本书讲解了如何在 Android 应用程序中绑定 OpenCV 的 Java 内容，进而显示相机画面、保存-共享照片、操控颜色和轮廓边以及跟踪 2D 或 3D 中的对象。另外，本书还讨论了 OpenGL 的整合方式，读者可据此构建增强现实（AR）程序，进而在相机画面的跟踪对象上绘制虚拟 3D 场景。

作为一类开源、跨平台库，OpenCV 提供了计算机视觉和应用构建模块，以及针对捕捉处理和图像数据显示的高层接口。例如，OpenCV 抽象了相机硬件的细节内容以及数组的内存分配。同时，OpenCV 广泛地应用于教学和软件工业中。

Android 则是一类开源移动操作系统，对于 Java 开发者而言，它提供了称作 Android SDK 的高层应用框架。Android 应用程序是一类模块化程序，并针对数据的发布和共享定义了标准的高层接口。针对照片共享应用，移动、高层抽象和数据共享可视为较好的开始点。

虽然 OpenCV 和 Android 提供了大量的高层抽象内容（以及开源代码），但对于初学者而言并不容易掌握，例如构建适宜的开发环境，将库功能项移至应用程序中。对此，本书旨在强调清晰的构建过程、简洁的应用设计以及各功能项的详细内容。

鉴于 OpenCV 的 Java 和 Android 绑定尚为新鲜事物，因而其文档并不完整，针对 OpenCV 与 Android 的标准相机、媒介以及图形 API 尤其如此。需要说明的是，整合工作是应用程序开发者的重要任务之一，也是本书的讲解重点。

OpenCV 和 Android 库之间的整合过程充分体现了广泛的应用程序特征，本书结束时将会对此予以尝试。届时，读者可通过可复用的类库扩展或调整后续的计算机视觉项目，并根据已有的开发环境和知识编写更为丰富的应用程序。

本书内容

第 1 章讨论 OpenCV 和 Android 开发环境的构建过程，其中包括 Eclipse 和 Android SDK。

第 2 章讨论 OpenCV 与 Android 应用程序之间的整合方式，进而预览、捕捉、保存和共享照片数据。

第 3 章探讨 OpenCV 的功能项，包括颜色通道的控制以及邻接像素问题。除此之外，

还将对现有应用程序进行扩展，以使其包含通道混合滤镜、“曲线”滤镜和黑色轮廓边滤镜。

第 4 章将展示视频画面中既定目标对象（例如油画作品）的识别和跟踪机制。同时，还将对已有应用程序进行适当扩展，并围绕跟踪目标绘制轮廓线。

第 5 章通过确定 3D 环境中目标的位置和旋转改善前述跟踪机制。另外，本章还将构建 OpenGL 3D 场景，并包含与 Android 相机设备相同的透视视角。

背景知识

本书将讨论 OpenCV 和 Android 开发环境的构建过程，其中包括 Eclipse 和 Android SDK。同时，对应软件均包含跨平台特征，包括 Windows、Mac、Linux 以及其他类 UNIX 环境。

读者需要持有一部可运行 Android 2.2 (Froyo) 或后续版本的移动设备，并建议设备包含前置和后置相机。另外，鉴于 OpenCV 使用 Google Play Store 管理安装和共享库的更新操作，因而推荐安装 Google Play Store 应用程序。

本书读者

本书适用于 Java 开发人员，以及致力于计算机视觉应用开发的读者。本书假设读者具备一定的 Java 开发经验（但缺乏相应的 Android 知识），并对图形数据有着基本的理解，例如像素、颜色通道等内容。

本书结构

本书包含了大量不同的文本风格，并以此区分不同种类的信息。下列内容展示了相应的文本风格和对应的解释内容。

文本中的代码字样通常显示为：“编辑系统路径 PATH 并包含<android_sdk>/platform-tools 和<android_sdk>/tools”。

对应代码块如下所示：

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android=
"http://schemas.android.com/apk/res/android"
package="com.nummist.secondsight"
```

```
android:versionCode="1"  
android:versionName="1.0">
```

当强调特定的代码段时，相关内容采用粗体表示，如下所示：

```
mCameraView.enableView();  
mBgr = new Mat();  
mCurveFilters = new Filter[] {  
    new NoneFilter(),  
    new PortraCurveFilter(),  
    new ProviaCurveFilter(),
```

命令行输入和输出如下所示：

```
$ cd /etc/udev/rules.d/  
$ sudo touch 51-android.rules  
$ sudo chmod a+r 51-android.rules
```

另外，某些新增内容和重要内容也采用粗体表示，而屏幕、菜单或对话框中的文本则采用原文显示，例如“clicking on the **Next** button moves you to the next screen”。

最后，“提示”表示一些较为重要的提示；“技巧”则表示相关的操作技巧。

读者反馈和技术支持

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 发送邮件，并以书名作为邮件标题。

若读者意欲查询出版信息，可在 www.packtpub.com 网站的 SUGGEST A TITLE 表单中填写相关信息，或发送邮件至 suggest@packtpub.com。

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 中的 author guide 一栏。

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。

除此之外，读者还可访问作者网站 <http://nummist.com/opencv/> 下载本书示例代码。

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/support>，选取对应书籍，单击 `errata submission form` 超链接，并输入相关问题的详细内容。经确认后，填写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）。同时，读者还可访问 <http://www.packtpub.com/support> 查看当前勘误表。

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。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，我们将竭诚为您服务；或者访问作者网站 <http://nummist.com/opencv/> 获取常见问题的答案。

本书作者和审校人员

Joseph Howse 很可能现在正坐在家中的沙发上撰写某本书籍，或者拎着手提箱处于奔波途中，箱子中被书籍、相机和计算机等设备填满。Joseph Howse 喜欢用这种“观察世界的方式”装备自己，要不就是在执行计算机视觉方面的工作。

Joseph Howse 在加拿大的 Ad-Dispatch 公司工作，并负责 iOS 和 Android 平台的增强现实游戏的开发工作。当与计算机视觉协同工作时，游戏中使用了大量的真实道具，例如孩子的绘画作品、玩具和毛毯。

另外，Joseph Howse 还提供培训和咨询等服务，其中包括基于 OpenCV 人脸识别的嵌入式系统项目。

Joseph Howse 拥有计算机专业、国际发展研究专业以及商务管理专业 3 个硕士

学位（加拿大 Dalhousie 大学），其研究成果曾被 ISMAR 发表。

本书是 Joseph Howse 在 Packt 出版社出版的第二本书籍，另外一本是《*OpenCV Computer Vision with Python*》，该书介绍了 Windows、Mac 以及 Linux 操作系统上的脸部跟踪和深度相机（例如 Kinect）等内容。

下面一段文字引自 Joseph Howse：

“我擅长写作并享受这一过程。这里要感谢和 Sam 在一起的日子，以及父母和我的猫咪，他们是我的精神支柱。”

“感谢本书的编辑和审校人员，在他们的指导下本书方得以出版。他们的专业、友好、良好的判断力以及热忱为本书增加了极大的色彩。”

Rohit Bhat 是一名毕业于印度 BITS Pilani 的本科生，并在业界领先的 Big Data Analytics 公司担任软件工程师一职。他所参与的开发项目涉及多种科技领域，其中包括数据挖掘、Android 开发、OpenCV、群集智能、工作流自动化以及视频会议平台。Rohit Bhat 热衷于最新技术，并可就任何话题与你展开讨论。另外，阅读、创业、经济学以及时下的新闻同样是他关注的话题。他热爱写作并在业余时间撰写自己的博客。

当前，Rohit Bhat 正在为 Packt 出版社编写一本 Bonita Open Solution 方面的书籍，相关技术广泛地应用于工作流自动化和商业处理建模中。

Viral Parekh 是一名计算机科学专业的大学生，同时也是一名经验丰富的移动应用程序开发者，并熟悉多种开源库，例如 OpenCV、OpenNI、FFmpeg。当前，他从事于人机交互和增强现实方面的开发工作。

目 录

第 1 章 构建 OpenCV	1
1.1 系统需求	2
1.2 构建开发环境	2
1.3 获取现有的开发环境——Tegra Android 开发包 (TADP)	2
1.4 逐个整合安装环境	4
1.5 获取预制版本的 OpenCV4Android	6
1.6 构建源 OpenCV	7
1.7 基于 Eclipse 构建 OpenCV 示例	8
1.8 查看文档和帮助文件	18
1.9 本章小结	19
第 2 章 与相机帧协同工作	20
2.1 设计 Second Sight 应用程序	20
2.2 创建 Eclipse 项目	22
2.3 开启相机和磁盘访问功能	25
2.4 创建菜单和字符串资源	27
2.5 在 CameraActivity 中预览和保存照片	29
2.6 删除、编辑和共享 LabActivity 中的照片	40
2.7 本章小结	44
第 3 章 添加图像效果	45
3.1 向当前项目中添加文件	45
3.2 定义滤镜接口	47
3.3 混合颜色通道	47
3.4 曲线色移	50
3.5 使用卷积过滤处理邻接像素	56
3.6 向 CameraActivity 中加入滤镜	58
3.7 本章小结	64

第 4 章 图像识别和跟踪	65
4.1 向当前项目中添加文件	65
4.2 理解图像追踪	66
4.3 编写图像跟踪滤镜	67
4.4 向 CameraActivity 中加入跟踪滤镜	74
4.5 本章小结	80
第 5 章 图像跟踪与 3D 渲染之间的整合操作	81
5.1 向项目中添加文件	81
5.2 定义 ARFilter 接口	82
5.3 在 CameraProjectionAdapter 中构建投影矩阵.....	83
5.4 针对 3D 跟踪调整 ImageDetectionFilter	87
5.5 在 ARCubeRenderer 中绘制立方体.....	92
5.6 将 3D 跟踪和渲染添加至 CameraActivity 中	97
5.7 Android 图形学的更多内容	101
5.8 本章小结	102

第 1 章 构建 OpenCV

本章将快速讲解 Android 和 OpenCV 开发环境的构建方式。另外，还将考察 OpenCV 的示例程序、文档以及开发社区。

在本章结束时，开发环境将涵盖下列组件。

- ❑ Java 开发工具包 (JDK6)：该工具包涵盖了 Java 编程工具。
- ❑ Cyrwin 1.7 或后续版本（仅适用于 Windows）：可视为一个兼容层，并在 Windows 上提供了一个类 UNIX 编程工具。
- ❑ Android 软件开发包 (Android SDK) r21.0.1 或后续版本：该开发包涵盖了 Java 中的 Android 应用程序开发工具。
- ❑ Android 本地开发包 (Android NDK) r8d 或后续版本：该开发包包含了 C++ 中 Android 应用程序的开发工具。虽然本书主要讲解 Java 编程机制，但 OpenCV 同样适用于 C++ 示例程序，并可实现程序的编译和运行。
- ❑ Eclipse 4.2.1 (Juno)：该软件为一个集成开发环境 (IDE)。
- ❑ Java 开发工具 (JDT)：即基于 Java 应用程序设计的 Eclipse 插件（已经内嵌于大多数 Eclipse 版本中）。
- ❑ C/C++ 开发工具 (CDT) 8.1.1 或后续版本：即 C/C++ 应用程序设计的 Eclipse 插件。
- ❑ Android 开发工具 (ADT) 21.0.1 或后续版本：即 Android 应用程序设计的 Eclipse 插件。
- ❑ OpenCV4Android 2.4.3.2 或后续版本：包含了 OpenCV 的 Android 版本，以及 Java 和 C++ 库。

上述组件存在多种安装和配置方式，本书将讨论几种较为常见的构建方案。另外，读者还可查看 OpenCV 的官方文档以获取更多内容，对应网址为 http://docs.opencv.org/doc/tutorials/introduction/android_binary_package/O4A_SDK.html。

1.1 系统需求

基于 Android 和 OpenCV 的开发工具均具有跨平台特征，并支持下列操作系统：

- Windows XP、Windows Vista、Windows 7 或 Windows 8。
- Mac OS 10.6 (Snow Leopard) 或后续版本。
- Ubuntu 10.10 (Maverick) 或后续版本。
- 其他类 UNIX 操作系统（本书并未涉及此类系统）。

为了运行 OpenCV 示例程序以及后续的应用程序，Android 设备应具备下列规范：

- Android 2.2 (Froyo) 或后续版本（必需）。
- 相机设备（必需）或前、后置相机（推荐使用）。
- 自动聚焦功能（推荐使用）。
- Google Play Store（推荐使用）。

这里，并不推荐使用 Android 虚拟设备 (AVD)，OpenCV 的某些部分取决于底层相机访问，且有可能无法使用虚拟相机。

1.2 构建开发环境

可从下列方案中选择开发环境：

(1) 安装包含全部所需组件的预置开发环境。

(2) 分别安装各种组件，经适当配置后使其可协同工作。当采用该方案时，可采用以下两种方式：

- 使用预置的 OpenCV 版本。
- 从资源中配置并构建 OpenCV。

下面将对各种方案进行详细讲解。

1.3 获取现有的开发环境——Tegra Android 开发包 (TADP)

Tegra Android 开发包 (TADP) 针对 Android、OpenCV 以及其他库包含了完整的

预置开发环境。针对 NVIDIA Tegra 处理器，TADP 采用优化方式构建应用程序。尽管如此，对应应用程序同样兼容于其他硬件。

提示：当从零开始搭建 Android 开发环境时，建议使用 TADP，它包含了全部所需软件的最新版本，且构建过程相对简单。

除此之外，TADP 还涵盖了某些附加内容。读者可查看官方文档以获取完整信息，对应网址为 <https://developer.nvidia.com/tegra-android-development-pack>。

设置 TADP 时，读者需要从 NVIDIA 完整下载并安装开发包，步骤如下：

(1) 加入 NVIDIA Registered Developer Program，对应网址为 <https://developer.nvidia.com/user/register>（免费）。

(2) 登录网址为 <https://developer.nvidia.com/user/login>。

(3) 在 https://developer.nvidia.com/user/me/profile/rdp_profile 处填写用户信息。

(4) 应用并加入 Registered Developer Program，对应网址为 <https://developer.nvidia.com/rdp/applications/tegra-registered-developer-program>（免费）。几分钟后，NVIDIA 将会向用户发送一封确认电子邮件。

(5) 访问 <https://developer.nvidia.com/tegra-resources>，并找到 TADP 最新版本的下载链接。在本书编写时，最新版本为 2.0r2，且分别支持 Windows（32 位或 64 位）、Mac 以及 Ubuntu（32 位或 64 位）的安装。相应地，读者可选择并下载相应的安装包。

(6) 当安装包显示这一步骤时，可输入任意目标并称作 <tadp>。默认条件下，<tadp> 表示为 C:\NVPACK（Windows 操作系统）或 ~/NVPACK（Mac 或 Ubuntu）。

(7) 当安装包显示 Installation Options 这一步骤时，可从 Complete、Express 或 Custom 选项中进行选择。与 Express 安装方式相比，Complete 或 Custom 安装则包含附加的 Android SDK 版本，以及 Tegra Android OS 二进制图像，这可是为 NVIDIA 的定制版本。若用户对此难以确定，则可直接选择 Express 选项。

(8) 当安装包显示 Proxy Configuration 这一步骤时，若未使用代理，则可将全部字段留为空白。

(9) 待结束全部安装配置步骤后，须等待 TADP 的下载和安装过程。

至此，安装过程暂告一段落，在执行后续处理之前，应了解 TADP 在相关位置安装了特定的组件。对于 TADP 2.0r2 版本中（在本书编写时为最新版本），对应安装位置如下所示：

- ❑ Android SDK 位于<tadp>/android-sdk-macosx, 可将该位置称作<android_sdk>。
- ❑ Android NDK 位于<tadp>/android-ndk-r8d, 可将该位置称作<android_ndk>。
- ❑ OpenCV4Android 位于<tadp>/OpenCV-2.4.3.2-android-sdk-tadp, 可将该位置称作<opencv>。
- ❑ Eclipse 位于<tadp>/eclipse, 可将该位置称作<eclipse>。

技巧: TADP安装包自动编辑系统PATH, 并包含<android_sdk>/platform-tools和<android_sdk>/tools。另外, TADP还将生成环境变量NDKROOT, 其值为<android_ndk>。

下面将讨论如何通过 Eclipse 构建 OpenCV 示例程序。

提示: 下载示例代码。读者可在<http://www.packtpub.com>并通过购买账号下载PACKT出版社全部书籍的示例代码。另外, 读者也可访问<http://www.packtpub.com/support>, 经注册后可通过电子邮件方式获得相关文件。同时, 本书的示例代码也可通过网址<http://nummist.com/opencv/>得到。

1.4 逐个整合安装环境

若不采用 TADP 作为既定方案, 则可对开发环境进行适当“组装”。总体而言, 该项任务包含如下两个步骤:

- (1) 构建多用途的 Android 开发环境。
- (2) 构建当前环境下的 OpenCV 应用。

下面考察基于多功能 Android 开发环境的构建步骤。此处并不打算详细讲解其过程, 前述链接已给出了相关指示; 另外, 作为一名 Android 开发者, 相信读者对此类构建步骤并不陌生。

提示: 如 Android 开发环境构建完毕, 且正想向其加入组件, 则下列步骤可能并不适用。

对应操作步骤如下所示:

(1) 若使用 Windows 或 Linux 操作系统, 则需要人工获得 JDK 6 (在 Mac 机上, 若 JDK 6 未予提供, 则操作系统在必要时将会自动安装)。针对 Windows 和 Linux 大多数版本, 网址 <http://www.oracle.com/technetwork/java/javase/downloads/jdk6downloads-1902814.html> 提供了相应的安装包。另外, 在 Linux 操作系统中, 须检测 JDK 包的版

本库，并安装 JDK 6。

(2) 下载 Eclipse 并将其解压至目标位置处，此处将其确定为<eclipse>。此处存在多种版本可供选择，Google 提供了称作 Android Developer Tools(ADT)包的 Eclipse 版本，源自 Android SDK 以及预置 ADT 插件。其中，读者可访问 <http://developer.android.com/sdk/index.html> 下载 ADT 包。而 <http://www.eclipse.org/downloads/> 则提供了最新版本的 Eclipse 软件。作为基本的 Android 开发环境，Eclipse 软件可视为移动开发人员的较好选择。

(3) 如果未得到 ADT 包，则需要安装 Android SDK 以及基于 Eclipse 软件的 ADT 插件。对此，读者可访问 <http://developer.android.com/sdk/index.html> 下载 Android SDK Tools。随后，可安装或将其解压至任意位置，即<android_sdk>。打开 Eclipse 并根据官方指示安装 ADT 插件，对应网址为 <http://developer.android.com/sdk/installing/installing-adt.html>。随后，将弹出一个名为 Welcome to Android Development 的窗口，单击 Use Existing SDKs，浏览至<android_sdk>并单击 Next 按钮，最后关闭 Eclipse。

(4) 若使用 Windows 操作系统，则可从 <http://cygwin.com/install.html> 下载并安装 Cygwin。

(5) 从 <http://developer.android.com/tools/sdk/ndk/index.html> 下载 Android NDK，可将其解压至任意位置，该位置称作<android_ndk>。

(6) 编辑系统的 PATH，并包含<android_sdk>/platform-tools 和<android_sdk>/tools。除此之外，还应创建名为 NDKROOT 的环境变量，其值为<android_ndk>。

技巧：在 Windows 操作系统上编辑环境变量

系统的 Path 变量以及其他变量可在 Control Panel 中的 Environment Variables 窗口中进行编辑。在 Windows Vista/7/8 操作系统中，打开 Start 菜单，运行 Control Panel，进入 System and Security | System | Advanced system settings，单击 Environment Variables 按钮。在 Windows XP 操作系统中，打开 Start 菜单并进入 Control Panel | System，单击 Advanced 标签页，并单击 Environment Variables 按钮。当前，在 System variables 中可选取现有环境变量，例如 Path，并单击 Edit 按钮。另外，还可通过单击 New 按钮创建新的环境变量，并在必要时编辑变量的名称和值。例如，若期望向当前 Path 中添加 C:\android-sdk\platform-tools 和 C:\android-sdk\tools，则可向现有 Path 值处追加 “;C:\android-sdk\platform-tools; C:\android-sdk\

tools”。这里，读者应注意分隔符的用法。若期望应用修改结果，可单击OK按钮，返回至Control Panel的主窗口，注销并再次登录。

在Mac机上编辑环境变量

编辑`~/profile`，并设置现有环境变量。例如可添加`export PATH=$PATH:~/android-sdk/platform-tools:~/android-sdk/tools`，这将`~/android-sdk/platform-tools`和`~/android-sdk/tools`添加至当前PATH中。这里，应注意分隔符分号的用法。当创建新的环境变量时，则可添加`export NDKROOT=~/android-ndk`一行，保存变化结果，注销并再次登录。

在Ubuntu上编辑环境变量

编辑`~/pam_environment`，在现有环境变量后添加`PATH DEFAULT=${PATH}:~/android-sdk/platform-tools:~/android-sdk/tools`，这将`~/android-sdk/platform-tools`和`~/android-sdk/tools`追加至当前PATH中。这里，需要注意分隔符分号的用法。当创建新环境变量时，可添加`NDKROOT DEFAULT=~/android-ndk`一行，保存修改结果，注销并再次登录。

至此，Android开发环境设置完毕，但仍需要确定OpenCV。对此，可选择下载OpenCV的预制版本，下面将对对应选项进行讨论。

提示：总体而言，Android应用程序应使用OpenCV的预制版本，其中较为重要的一点是，预制版本可作为共享库供Android用户使用，这将节省硬盘空间并简化更新操作。

本书并不打算构建源OpenCV。出于完整性，本章将对这一选项予以讨论，高级用户可据此对OpenCV进行调整。

1.5 获取预制版本的 OpenCV4Android

读者可访问 <http://sourceforge.net/projects/opencvlibrary/files/opencv-android/> 下载OpenCV4Android的预制版本。其中，可查看包含`opencv-android`名称的文件，例如`OpenCV-2.4.5-android-sdk.zip`（在本书编写时，该版本为最新版本）。随后，可选择最新版本的软件，可将其解压至任意位置，即`<opencv>`。