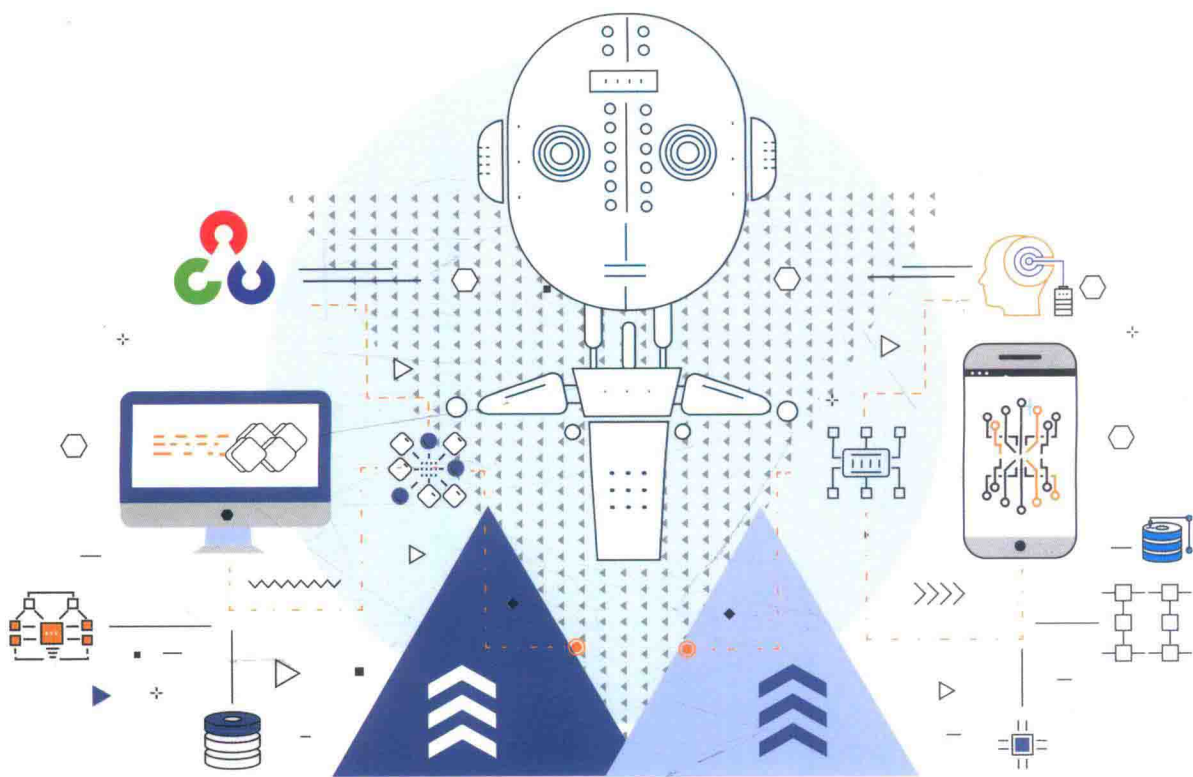


详解OpenCV Android配置与开发，以真实案例展示移动端  
算法流程设计和代码实现

# OpenCV Android 开发实战

贾志刚 著



■ ■ ■ 智能系统与技术丛书

# OpenCV Android 开发实战

贾志刚 著



机械工业出版社  
China Machine Press

## 图书在版编目 (CIP) 数据

OpenCV Android 开发实战 / 贾志刚著. —北京: 机械工业出版社, 2018.6  
(智能系统与技术丛书)

ISBN 978-7-111-60140-1

I. O… II. 贾… III. 移动终端 - 应用程序 - 程序设计 IV. TN929.53

中国版本图书馆 CIP 数据核字 (2018) 第 113043 号

# OpenCV Android 开发实战

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: 张梦玲

责任校对: 殷虹

印刷: 北京市兆成印刷有限责任公司

版次: 2018 年 6 月第 1 版第 1 次印刷

开本: 186mm × 240mm 1/16

印张: 14.75

书号: ISBN 978-7-111-60140-1

定价: 59.00 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

# 前 言

## 为什么要写这本书

2015年，我出版了第一本图像处理方面的图书《Java 数字图像处理：编程技巧与应用实践》<sup>①</sup>，该书主要讲述图像处理的各种基础算法原理与代码实现，基于 Java 语言进行描述，没有太多的工程应用实践案例，是一本编程实践入门级的图像处理图书。因此我一直想再写一本工程实践性比较强的图书，Java 与 Android 程序员可以通过这样的书籍，摆脱底层算法实现难的烦恼，快速学习相关 API 的使用，掌握常见的图像处理技术，快速开发应用，上手计算机视觉应用开发；他们通过学习与参照书中的工程实践案例，可以解决实际需求，提升个人竞争力，为企业和个人在短时间内创造更大的价值。

OpenCV 作为一款开源的计算机视觉框架，封装了超过 1000 个常见的图像处理算法，其 SDK 语言支持 Java、C++、Python 等。借助人工智能兴起的东风，近几年 OpenCV 开发者社区的发展非常迅速，人数成几何级递增，而且已经对 Android 系统有了良好的支持与完备的 SDK 开发接口。在无须了解底层算法实现的情况下，借助 OpenCV 提供的 SDK，Android 开发者可以实现 OCR 识别、图像处理、人脸检测、相机校正、实时视频分析与处理、AR 增强等移动端应用开发。

对大多数 Android 开发者来说，OpenCV 与计算机视觉应用开发都可能显得有点陌生，因为市面上缺乏专业的工程性书籍与文档，OpenCV 社区对 Android SDK 本身也没

---

<sup>①</sup> 该书由机械工业出版社出版，书号为 ISBN 978-7-111-51946-1。——编辑注

有提供完善的 API 文档与代码演示，这让很多 Android 程序员无法顺利使用 OpenCV 框架在移动端开发计算机视觉相关的应用。本书系统性地讲述 OpenCV 如何在 Android 系统上应用开发与工程实践，撰写本书的时候，因为 OpenCV 的很多 API 调用参数缺少文档说明，因此笔者需要通过编程实践一点一点全部尝试之后再总结出来，用实践出真知来形容本书一点也不过分。笔者本人是个地道的程序员，特别理解和了解程序员的视角与工程应用的重点和难点，本书从程序员的视角出发，在思路分析与代码实现上，对每个案例都做了非常清楚的交代与解释，对不同算法函数的应用场景都有详细的代码演示。本书最后三个案例分别涉及 OCR 识别、人脸美颜算法、视频检测与跟踪渲染这些实际落地场景，这三个案例是笔者本人精心挑选的，涵盖了大多数 Android 开发者的工程实践需求与工作需要，力求做到尽善尽美，然“人无完人，金无足赤”，最终还需读者评价。

如果说我的第一本书是对我十年工作的总结，那么本书就是我十年之后再出发的征途起点，“远飞者当换其新羽”，对广大 Android 与 Java 程序员来说，处在人工智能时代，掌握前沿技术，更新自己的技术栈，提升个人竞争力，计算机视觉与 OpenCV 就是个很好的方向与选择。作为技术人员唯有鼎故革新、砥砺前行，才能不负这个最好的时代，本书也是献给广大 Android 与 Java 程序员最好的礼物。

最后，希望通过本书的知识和作者有限的经验，帮助广大 Android 与 Java 程序员，以及众多有志于从事计算机视觉的后来者，借助 OpenCV 框架走上计算机视觉应用开发的道路。本书的顺利出版离不开笔者对 OpenCV 与计算机视觉技术的兴趣，更离不开笔者的毅力与本书写作初衷。希望本书能为国内 OpenCV 框架使用的普及与应用开发实践尽绵薄之力，若能如愿也不枉我的一番努力。

## 读者对象

本书适合于以下读者对象。

- 广大 Android 与 Java 程序员。
- 从事图像处理的工作者。

- 学习图像处理的爱好者。
- 希望提升自我的中高级程序员。
- 计算机专业高年级本科生或者研究生。
- 从事图像处理行业的公司与个人。
- 开设图像处理相关课程的大专院校学生。

## 如何阅读本书

本书共分为两大部分，其中第一部分为第 1 章到第 7 章，系统地介绍了 OpenCV Android 的开发框架及功能。第二部分是本书的案例部分，系统全面地分析了三个实际案例，讲解如何借助 OpenCV 框架解决实际问题。如果你已经对 Java 语言和 Android 系统上的 SDK 开发有基本的认识，那么可以直接开始阅读本书，书中的源代码也是本书的一部分，建议在阅读本书内容的同时，尝试运行与修改本书提供的源代码，这样有助于更加深刻地理解与之相关的 API 参数与算法应用场景。

第一部分为基础篇，由浅入深，从 OpenCV 框架的简单介绍到 OpenCV 与 Android SDK、NDK 的编程应用，系统全面地介绍了 OpenCV 在移动领域的应用、OpenCV 中的核心模块、图像处理模块、特征提取与对象检测模块等。读者在学习与掌握 OpenCV 相关 API 用法的前提下可以学习第二部的实战案例。

第二部分为实战案例部分，由 OCR 识别、人脸美颜、人眼实时跟踪与渲染三个典型案例组成。通过案例学习，读者将学会如何设计算法流程、使用组合算法 API、关注应用的性能与内存问题，以及 NDK 开发技巧、其他图像处理开发相关 API 的使用技巧。

此外，本书的源文件可到 [www.hzbook.com](http://www.hzbook.com) 上搜索本书书名下载，或者到 Github 上下载本书演示工程，网址为 <https://github.com/gloomyfish1998/opencv4android/tree/master/samples/OpencvDemo>。

本书参考资料也可从 Github 上下载，网址为 <https://github.com/gloomyfish1998/opencv4android>。

## 勘误和支持

由于笔者的水平有限，编写的时间也很仓促，书中难免会出现一些错误或者不准确的地方，不妥之处在所难免，恳请读者批评指正。笔者已经把本书配套的源代码上传到 Github，访问地址为 <https://github.com/gloomyfish1998/opencv4android/tree/master/samples/OpencvDemo>，如果有读者想直接提交勘误代码，请先邮件联系笔者，笔者同意以后即可提交，同时笔者也会根据读者反馈更新源代码，所以在阅读本书之前请先从 Github 上获取最新的配套源代码。如果你有更多的宝贵意见，也欢迎发送邮件至邮箱 [57558865@qq.com](mailto:57558865@qq.com)，很期待听到你们的真挚反馈。

## 致谢

OpenCV 能有今天的发展，首先要感谢英特尔当时的开源决策，其次是 OpenCV 社区的巨大贡献，我第一次接触 OpenCV 就被它的开发效率吸引住了，可以说 OpenCV 是计算机视觉应用开发最好用的工具之一，特别是 OpenCV3.0 以后的版本，非常容易学习，所以要感谢那些为 OpenCV 做出过贡献的杰出开发者。在我写作本书的时候，机械工业出版社华章公司的编辑杨绣国老师一直没有向我催稿，反而告诉我要安心创作，认真细致，后期审稿的时候也是逐字逐句推敲，反复修改，感谢你的耐心与严谨，正是你的鼓励、帮助和支持引导我顺利完成本书撰写。

最后感谢我的爸爸、妈妈，感谢你们给予我生命，将我培养成人，感谢我的妻子在我写书的这一年多时间里让我从家务中解脱，给予我支持与鼓励。

谨以此书，献给我最亲爱的两个孩子，以及众多热爱 OpenCV 编程的朋友。

贾志刚

中国，苏州，2018年3月

## 目 录

前言	
<b>第一部分 OpenCV 图像处理 系统学习篇</b>	
<b>第 1 章 OpenCV Android 开发</b>	
<b>框架</b> .....	2
1.1 OpenCV 是什么 .....	2
1.1.1 OpenCV 的历史与发展 .....	3
1.1.2 OpenCV 模块介绍 .....	4
1.1.3 OpenCV Android SDK .....	5
1.2 OpenCV Android 开发环境搭建 .....	6
1.2.1 软件下载与安装 .....	6
1.2.2 环境搭建 .....	7
1.2.3 代码测试 .....	13
1.3 构建演示 APP .....	14
1.4 拍照与图像选择 .....	18
1.5 小结 .....	20
<b>第 2 章 Mat 与 Bitmap 对象</b> .....	21
2.1 Mat 对象 .....	21
2.1.1 加载图像与读取基本 信息 .....	22
2.1.2 Mat 创建与初始化 .....	24
2.1.3 Mat 对象保存 .....	25
2.2 Android 中的 Bitmap 对象 .....	26
2.3 基础形状绘制与填充 .....	29
2.4 Mat 与 Bitmap 的使用与转换 .....	34
2.5 小结 .....	36
<b>第 3 章 Mat 像素操作</b> .....	37
3.1 像素读写 .....	37
3.2 图像通道与均值方差计算 .....	40
3.3 算术操作与调整图像的亮度和 对比度 .....	44
3.4 基于权重的图像叠加 .....	46
3.5 Mat 的其他各种像素操作 .....	48
3.6 小结 .....	52
<b>第 4 章 图像操作</b> .....	53
4.1 模糊 .....	53
4.2 统计排序滤波 .....	59



4.3 边缘保留滤波 .....	62	<b>第 7 章 相机使用</b> .....	144
4.4 自定义滤波 .....	64	7.1 使用 JavaCameraView .....	144
4.5 形态学操作 .....	67	7.2 横屏与竖屏显示 .....	151
4.6 阈值化与阈值 .....	72	7.3 相机预览帧图像处理 .....	153
4.7 小结 .....	78	7.4 在预览帧中实现人脸检测 .....	155
<b>第 5 章 基本特征检测</b> .....	79	7.5 小结 .....	161
5.1 梯度计算 .....	79	<b>第二部分 OpenCV4Android</b>	
5.2 拉普拉斯算子 .....	82	<b>应用实战</b>	
5.3 Canny 边缘检测 .....	83	<b>第 8 章 OCR 识别</b> .....	164
5.4 霍夫直线检测 .....	86	8.1 什么是 OCR .....	164
5.5 霍夫圆检测 .....	89	8.2 开源 OCR 框架 Tesseract .....	167
5.6 轮廓发现与绘制 .....	91	8.3 识别身份证号码 .....	170
5.7 轮廓分析 .....	94	8.3.1 UI 部分实现 .....	171
5.8 图像直方图 .....	98	8.3.2 号码区域查找 .....	172
5.9 模板匹配 .....	107	8.3.3 号码识别 .....	174
5.10 小结 .....	110	8.4 提高 OCR 识别率 .....	177
<b>第 6 章 特征检测与匹配</b> .....	111	8.4.1 使用 Tesseract 命令行	
6.1 Harris 角点检测 .....	111	生成训练数据 .....	177
6.2 Shi-Tomasi 角点检测 .....	114	8.4.2 图像预处理 .....	181
6.3 SURF 特征检测 .....	116	8.5 小结 .....	184
6.4 SIFT 特征检测 .....	121	<b>第 9 章 人脸美颜</b> .....	185
6.5 Feature2D 中的检测器与		9.1 积分图计算 .....	185
描述子 .....	128	9.2 局部均方差滤波 .....	189
6.6 特征匹配查找已知对象 .....	134	9.3 遮罩层生成 .....	192
6.7 级联分类器与人脸检测 .....	137	9.4 高斯权重融合 .....	194
6.8 小结 .....	143		

9.5 边缘提升 .....	196	10.2 人脸检测与跟踪 .....	208
9.6 美颜实现 .....	198	10.3 寻找眼睛候选区域 .....	213
9.7 小结 .....	204	10.4 眼睛检测 .....	214
<b>第 10 章 人眼实时跟踪与渲染 .....</b>	<b>205</b>	10.5 黑眼球定位 .....	217
10.1 界面显示与相机预览 .....	205	10.6 渲染与优化 .....	220
		10.7 小结 .....	222

## 第一部分

# OpenCV 图像处理系统学习篇

第一部分为基础篇，由浅入深、从 OpenCV 框架的简单介绍到 OpenCV 与 Android SDK、NDK 的编程应用、系统全面地介绍了 OpenCV 在移动领域的应用、OpenCV 中的核心模块、图像处理模块、特征提取与对象检测模块等。读者在学习与掌握 OpenCV 相关 API 用法的前提下可以开始学习第二部的实战案例。

# OpenCV Android 开发框架

在开始本书内容之前，笔者假设大家已经有了面向对象语言编程的基本概念，了解了 Java 语言的基本语法与特征，并且尝试过 Android 平台上的应用程序开发。本章将主要介绍 OpenCV 的历史与发展、各个模块的功能说明、如何使用 Android Studio IDE 来建立 OpenCV 的开发环境，以及如何整合配置并成功运行和调用 OpenCV 中的函数实现一个最简单的 OpenCV 程序演示。如果没有特别说明，那么这里使用的 OpenCV 版本都是基于 OpenCV 3.3 Android SDK。

作为使用最为广泛的计算机视觉开源库，OpenCV 在开源社区与英特尔、谷歌等大公司的共同努力之下，发展到今天，已经吸引了全世界各地的开发者编译和使用它实现各种应用程序。而伴随着人工智能时代的到来，作为人工智能眼睛的计算机视觉必然会进一步释放活力，满足市场需要。OpenCV 作为计算机视觉开源框架，其在移动端支持 Android 系统的特性必将进一步深入到移动开发的各种应用场景之中，下面就来开启一段 OpenCV 学习旅程。

## 1.1 OpenCV 是什么

OpenCV 的中文全称是源代码开放的计算机视觉库（Open Source Computer Vision Library），是基于 C/C++ 编写的，是 BSD 开源许可的计算机视觉开发框架，其开源协议

允许在学术研究与商业应用开发中免费使用它。OpenCV 支持 Windows、Linux、Mac OS、iOS 与 Android 操作系统上的应用开发。在笔者动笔写这本书的时候，其最新版本 3.3 刚刚发布不久。

### 1.1.1 OpenCV 的历史与发展

在 OpenCV 孕育发展的过程中，Intel 公司做出了巨大的贡献，OpenCV 最初是 Intel 公司的内部项目，随着时间的推移、OpenCV 的功能算法得到不断的优化与增强，不过是短短十几年的时间，其已经席卷整个业界，得到众多著名 IT 公司的大力支持，其中包括大名鼎鼎的机器人公司 Willow Garage 与搜索引擎公司 Google。下面的时间节点对 OpenCV 的发展都产生过重要影响，具体如下。

- 1999 年，OpenCV 正式立项，那个时候 Android 智能手机的春天还没有到来。
- 2000 年，在 IEEE 的计算机视觉与模式识别大会上 OpenCV 正式发布 Alpha 版本。
- 2001 年~2005 年，Intel 公司陆续发布了最初的 5 个 Beta 测试版本。
- 2006 年，OpenCV1.0 版本正式发布，基于 C 语言接口 SDK 调用。
- 2008 年，OpenCV 获得了当时发展如日中天的机器人公司 Willow Garage 的支持、得到了进一步推广，然而不幸的是，作为机器人业界的传奇公司 Willow Garage 却在 2014 宣布倒闭。
- 2009 年，OpenCV2.0 版本正式发布，这是 OpenCV 发展史上的一个重要里程碑，早期的 OpenCV 是基于 C 语言实现的，在 2.0 的版本中又添加了 C++ 接口，并且对原来的 C 语言代码进行了优化和整合，以期吸引更多的开发者用户。
- 2012 年，Intel 公司决定把 OpenCV 开发者社区正式交给开源社区 [opencv.org](http://opencv.org) 运营与维护。
- 2014 年，OpenCV3.0 版本发布。
- 2016 年，OpenCV3.1 与 OpenCV3.2 版本相继发布，其扩展模块支持集成 Google TensorFlow 深度学习框架。
- 2017 年，OpenCV3.3.x 版本发布，在 Release 开发包中增加了 DNN（深度神经网络）模块支持。

OpenCV 支持 Java 语言开发的 Android SDK 最早是始于 2010 年。在 OpenCV3.x 版本中，OpenCV 更加强调对移动端与嵌入式设备的支持。

### (1) 编程语言

OpenCV 中的这些模块大多数都是基于 C/C++ 完成的，少量的 SDK 接口模块使用 Java、Python 等语言开发。在最新开发的 OpenCV 的核心模块中，C++ 替代 C 成为了开发语言。

### (2) 应用领域

OpenCV 自从 1.0 版本发布以来，立刻吸引了许多公司的目光，被广泛应用于许多领域的产品研发与创新上，相关应用包括卫星地图与电子地图拼接，医学中图像噪声处理、对象检测，安防监控领域的安全与入侵检测、自动监视报警，制造业与工业中的产品质量检测、摄像机标定，军事领域的无人机飞行、无人驾驶与水下机器人等众多领域。

## 1.1.2 OpenCV 模块介绍

OpenCV 分为正式的发布版本与扩展模块，Android SDK 所对应的是 OpenCV 的发布版本，其扩展模块的功能可以通过源代码编译的方式进行集成与开发，关于扩展模块的编译与使用已经超出了本书的讨论范围，这里就不再赘述了。下面以 OpenCV3.3 为例，OpenCV 正式发布版本中包含的核心功能模块具体如下。

- 二维与三维特征工具箱
- 运动估算
- 人脸识别
- 姿势识别
- 人机交互
- 运动理解
- 对象检测
- 移动机器人

- ❑ 分割与识别
- ❑ 视频分析
- ❑ 运动跟踪
- ❑ 图像处理
- ❑ 机器学习
- ❑ 神经网络

除上所述的核心功能模块之外，其扩展模块更加的庞大与繁杂。OpenCV Android SDK 可以从其官方主页上下载获得，下载地址为：<http://opencv.org/opencv-3-3.html>，在最下面就可以发现 Android SDK 的下载链接，点击就可以直接去相关页面上下载最新的 Android SDK。

### 1.1.3 OpenCV Android SDK

OpenCV Android SDK 本质上是使用 Java 语言接口通过 JNI 技术调用 OpenCV C/C++ 代码完成的算法模块。OpenCV4Android 本身并不是一个纯 Java 语言的计算机视觉库，而是基于 OpenCVC++ 本地代码、通过 Java 语言接口定义，基于 JNI 技术实现调用 C++ 本地方法的 SDK 开发包。所以当你下载好 OpenCV Android SDK 之后，在它的 SDK 目录下可以看到如图 1-1 所示的目录结构。

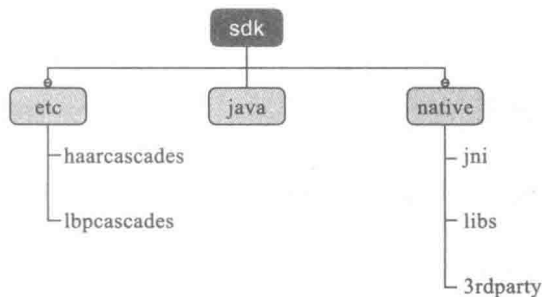


图 1-1

其中，etc 目录里面有两个文件夹，里面都是一些 XML 数据文件，这些 XML 数据

是训练好的 HAAR 与 LBP 级联分类器数据；java 目录里面是 Android SDK 相关文件；native 里面则是基于 C/C++ 编译好的 OpenCV Android 平台支持的本地库文件、JNI 层开发所需要的头文件及 cmake 文件，其中库文件大多数以 \*.a 和 \*.so 结尾。而在与 SDK 同层级的 samples 目录中则包含了 OpenCV Android SDK 的一些应用案例教程，以供初学者参考，但是很不幸的是，直到今天为止，这些教程仍然还是基于 Eclipse 开发环境来演示 OpenCV 功能，不得不说这是一个小小的缺憾，希望 OpenCV 社区在后面的 Open CV 版本中能够更新这些教程，使其基于 Android Studio 来演示。

此外，OpenCV Android SDK 的功能与 OpenCV 对应发布版本中的功能完全相同，唯一不同的是因为 Java 语言的关系，Java 层封装的接口的参数传递和方法调用，与 C++ 的接口相比有一些差异，这些都是为了更适应 Java 语言的特性而做出的改动，使得 Android 开发者更加容易学习与使用 OpenCV 来解决问题。

## 1.2 OpenCV Android 开发环境搭建

当 OpenCV 遇到 Android 时，两者就通过 Java SDK 或者 Android NDK 很好地结合在一起了，可是对于广大 Android 开发者或者 OpenCV 开发者来说，要想成功地在 Android Studio 上运行一个类似于 Hello World 的 OpenCV 程序，还需要做一些工作，下面就一起来完成这些工作，实现开发环境的搭建。

### 1.2.1 软件下载与安装

在搭建开发环境之前，首先需要下载和安装如下几个软件开发包。

- OpenCV Android SDK 3.3 版本
- JDK8: 64 位
- Android Studio
- Android SDK 与 NDK 开发包



这里需要特别说明一下的是，首先应该安装好 JDK，之后再下载安装其他的软件开发包，全部下载安装完毕之后，就可以打开 Android Studio——Android 集成开发环境 (IDE)，配置好 Android SDK 的路径之后，Android Studio (IDE) 工具就可以正常使用了。

所下载的 OpenCV Android SDK 3.3 是一个安装包，只要解压缩到指定磁盘即可，双击解压缩好的目录就可以看到 1.1.3 节中提到的几个目录与层次结构。在本书最后的案例开发中会涉及与使用 NDK 的开发包，这里暂时只要将其安装好即可。

### 版本问题

使用 Android Studio 与 Android SDK、NDK 开发时，同样的代码在不同的版本上运行可能会出现一些兼容性问题。因为项目实际需要或者个人偏好，大家使用的版本可能不尽相同，这里说一下本书所用的版本，具体如下。

- Android Studio 3.0
- Android SDK 26
- Android NDK r13b

## 1.2.2 环境搭建

环境搭建的整个过程可以分为如下四步。

### 1. 新建 Android 项目

打开 Android Studio IDE，选择【File】→【New Project...】，结果如图 1-2 所示。

把项目默认名称修改为 OpencvDemo，然后点击【Next】按钮，结果如图 1-3 所示。

这里支持的最小版本是 Android 14 (Android 4.0 版本)，继续点击【Next】按钮，结果如图 1-4 所示。

默认选择 Empty Activity，点击【Next】→【Finish】，之后就会得到一个新建的默认 Android 版本的 Hello World 程序，如果一切顺利的话，就可以真机运行，查看效果。