

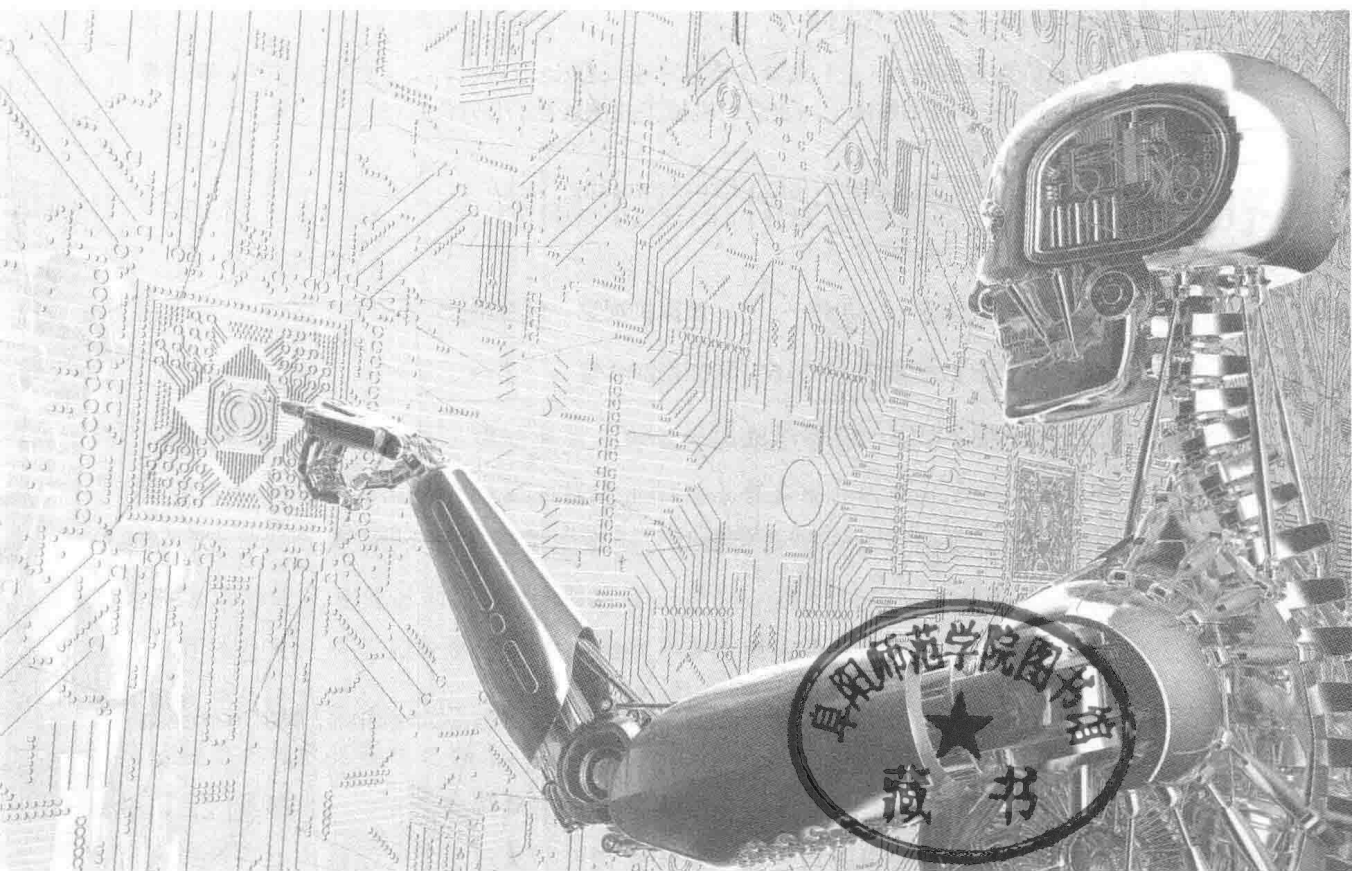
# Arduino 计算机视觉编程

Arduino Computer Vision Programming

[土耳其] 欧森·奥兹卡亚 (Özen Özkaya) 吉拉伊·伊利茨 (Giray Yıllıkçı) 著  
张华栋 译



机械工业出版社  
China Machine Press



# Arduino 计算机视觉编程

Arduino Computer Vision Programming

[土耳其] 欧森·奥兹卡亚 (Özen Özkaya) 吉拉伊·伊利茨 (Giray Yıllıkçı) 著  
张华栋 译



机械工业出版社  
China Machine Press

## 图书在版编目 (CIP) 数据

Arduino 计算机视觉编程 / (土) 欧森·奥兹卡亚 (Özen Özkaya), (土) 吉拉伊·伊利茨 (Giray Yıllıkçı) 著; 张华栋译. —北京: 机械工业出版社, 2016.10  
(机器人设计与制作系列)

书名原文: Arduino Computer Vision Programming

ISBN 978-7-111-55126-3

I. A… II. ①欧… ②吉… ③张… III. 计算机视觉—程序设计 IV. TP302.7

中国版本图书馆 CIP 数据核字 (2016) 第 252209 号

本书版权登记号: 图字: 01-2016-1888

Özen Özkaya, Giray Yıllıkçı: Arduino Computer Vision Programming (ISBN: 978-1-78355-262-7).  
Copyright © 2015 Packt Publishing. First published in the English language under the title “Arduino Computer Vision Programming”.

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。

## Arduino 计算机视觉编程

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 11 月第 1 版第 1 次印刷

开本: 186mm × 240mm 1/16

印张: 11

书号: ISBN 978-7-111-55126-3

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

## The Translator's Words | 译者序

“忍耐很苦，坚持很难”，哪怕是做一件看上去毫不费劲的小事也是如此。在申请翻译本书时，我满以为一个月就能翻译完，实际上却用了整整半年时间。十几年前，看张筑生先生所著的《数学分析新讲》，在后记里面看到“从编写教学改革实验讲义到整理改写成书，前后花费了五年最宝贵的时间”。那个时候年少轻狂，看完以后固然深深为作者的人品打动，却以为写书是件并不怎么费劲的事情，果然是事不经过不知难，翻译尚且如此，反复斟酌几易其稿成书又该是怎么一种心血的煎熬。

随着系统智能化要求的提高，对计算机视觉的要求也越来越多，同时，计算机视觉处理完毕以后，需要有一个合适的载体来做出反应。本书正好就是同时在这两点上着重论述，一方面大刀阔斧地介绍了计算机视觉处理的全流程，并就流程的每一子步骤单独成章介绍常见的方法。另一方面注重实践，带领读者从头开始走完整个建造机器人的全程。正如作者所说，“只要掌握了本书的内容，完全可能创建一个可以看到环境并进行推理的机器人”。并且每一章的介绍都为后续深入开发打下了良好的基础。

本书的一个显著优点是用很简洁的语言带领读者纵览全局，让读者对整个开发有一个全局的掌握，因而非常适合入门。毕竟工作以后再读大部头的书籍，除非真的是卓尔不群，否则往往半途而废。

在此，我想感谢华章公司的编辑缪杰，没有他的帮助、理解和支持，我很难坚持下来。

我还要特别感谢我的爸爸妈妈——张乾光和孟淑华，你们多年来一直辛苦地为我遮风挡雨，给我支持和鼓励，让我有一个非常宽松快乐的成长环境，你们的善良和坚韧也一直深深地影响着我。你们辛苦了！大海的尽头有另外一个世界，希望在以后的岁月里能让你们的生活更加美好！

最后，我要感谢的是我的爱人——崔嘉艺，感谢她对我的支持和信任，相守是最温暖的承诺，我会让她为她的选择感到骄傲！

## 作者简介 | About the Authors

Özen Özkaya 是一名嵌入式系统工程师，在各种各样的计算机视觉应用和嵌入式系统的设计、开发和验证方面有超过 6 年的经验。他非常相信分享知识的力量并持续地在计算机视觉领域中实践。

从业 6 年之后，现在他是西门子公司的一名高级开发工程师，参与开发和研究工业控制设备及工业通信处理器。他还对西门子的软件质量保证项目做了一些贡献。到目前为止，他共申请了 8 个专利，他的所有应用都在进行中。他用了 3 年时间从伊斯坦布尔科技大学 (ITU) 拿到了电子工程的学士学位，并得到了高等荣誉证书。他拥有伊斯坦布尔科技大学的电子工程硕士学位，并且正在攻读伊斯坦布尔科技大学的电子工程博士学位。他在学术研究期间曾在不同的实验室工作过，比如说医疗系统设计实验室、控制和航空电子设备实验室、机器人实验室、模式识别和信号处理实验室、工业自动化实验室，最后是嵌入式系统实验室。

除了做学术研究以外，他现在还是伊斯坦布尔科技大学嵌入式系统实验室的一名导师。

可以通过电子邮件 [ozenozkaya@gmail.com](mailto:ozenozkaya@gmail.com) 或者 [contact@ozenozkaya.com](mailto:contact@ozenozkaya.com) 直接与 Özen 联系。如果你想了解有关他的更多信息，可访问他的网站 <http://www.ozenozkaya.com>。

---

首先我要感谢我的父母 Kıyas 和 Ferah，是他们不停的努力和坚持，才有了我今天的发展。

我的导师 S. Berna Örs Yalcin 副教授，总是支持我并帮助我完成了很多事情，所以我想在这里对她表示感谢。我还要感谢我所有的朋友，可惜名单太长没有办法在这里一一列举。

我还要特别感谢 Nilay Tüfek，在我写作本书的过程中，她鼎力支持我。她真诚地鼓励我接受这一挑战并克服它。

---

Giray Yıllıkçı 专注于嵌入式系统、计算机视觉和机器人技术。他对这些领域的学术和工业方面都有贡献。他喜欢分享他的知识和经验。他认为信息共享才是让生活前进的真正方式。

Giray 目前是 Koç 大学设计实验室的一名高级研究工程师，他负责协调项目的技术过程。此外，他还需要管理技术研究工程师。他正在申请 6 个专利。

他拥有 Koç 大学物理学的学士学位。现在，他在伊斯坦布尔科技大学 (ITU) 为有关卫星通信和遥感的硕士毕业论文而努力。

有关他的更多信息可以访问他的网站 <http://www.girayyillikci.com>，也可以通过 [gyillikci@gmail.com](mailto:gyillikci@gmail.com) 或者 [gyillikci@ku.edu.tr](mailto:gyillikci@ku.edu.tr) 与他联系。

---

---

首先，我要感谢 Özen Özkaya 邀请我合著本书。

我还要向给我们提供支持、意见和帮助的 Packt 员工们表示感谢。我要感谢 Seyed Mousavi，因为他给我们提供了很多有价值的评论，还要感谢 Shweta Pant 成功的内容开发管理。

感谢 Cetin Seren 对我职业方面的指点。他的真知灼见让我在自己的领域中做到最好。

我要感谢 Süreyya Ciliv，他给我们提供了位于 Turkcell 技术应用中心的一个充满智慧和令人振奋的工作环境，在那里我们学到了很多知识。

我还要感谢 Tugce Pakakar 和她的朋友们，她们让我非常享受写作本书的过程。

最后，我要感谢我的母亲、父亲、兄弟，他们在我的生活中支持、鼓励、启迪我。我要对他们表达我最诚挚的感谢。

---

---

# 前言 | Preface

特别是对于现代机器而言，计算机视觉是环境感知发展的下一个阶段。许多现代设备尝试使用有限的资源来模仿人类的感知能力。此外，大多数此类仿制品其实是存在不足的，因为通常情况下它们不能直接模拟人类的感知能力，尤其是在视觉方面。

即使传统传感器的种类有很多，它们仍然无法表现出人类视觉系统的强大，这是人类最复杂的感知功能之一。因此，我们需要一定的视觉信息，使我们的电子系统更加智能化。这就是计算机视觉出现的原因。

相机可以被视为终极视觉传感器，它非常接近于人类的视觉感知系统。但是问题在于，如果使用相机作为视觉传感器，那么操作会非常复杂和困难。本书的目的是通过把一个复杂的问题划分为基本的、可实现的子步骤从而让计算机视觉的任务变得简单。这种方法最好的地方在于我们能把实际生活中的应用也变得很容易！

当我们处理现实生活中的应用时，毫无疑问应该有一种方法来与现实生活进行交互。嵌入式系统正好用来表示这些物理交互。**Arduino** 是最流行的嵌入式系统平台之一，它通过海量的社区支持和学习材料为原型开发提供了一种便捷的方法。同时 **Arduino** 的关键属性也很棒，稍后将进行更详细的讨论，这些使得 **Arduino** 成为视觉系统与现实生活进行交互的最完美的候选。

**Arduino** 在计算机视觉系统中扮演的角色是非常清楚的。在大多数情况下，视觉信息会与来自各种传统的传感器的数据相结合，这些传统的传感器包括温度传感器、压力传感器和加速器等。此外，我们需要一个很好的工具来在视觉处理过程之后采取物理行动，比如驱动电动机和开灯。因此，**Arduino** 在从环境中收集数据并采取行动的过程中非常有用，特别是因为它有大量开发好的库以及社区共享，使得原型开发非常迅速。

开源计算机视觉库 (**OpenCV**) 是一个包含大量优化算法的优秀开源库。**OpenCV** 最重要的优势之一是它可以支持多个平台，它能够在 **Windows**、**Linux**、**Mac OS** 甚至是 **Android** 和 **iOS** 平台上工作。更重要的是它是一个成熟的库，在写作本书时最新版本为 3.0 (和 2.4.11)。出于这些原因，本书中所有的计算机视觉任务都将采用 **OpenCV** 来实现！

本书将通过一种通用的、明确的、可应用的方式来融合 **Arduino** 和计算机视觉的能力。书中的实践和方法可用于任何相关的问题和平台。本书想实现的最终目标是让你可以通过一种通用方法使用各种组件来解决现实生活中任意类型的视觉问题。

在每一章中，你将会找到有关主题在现实生活中实际应用的例子。为了学以致用，我们会一步一步地建立一个视觉机器人。你会发现尽管问题的背景有很大的不同，解决问题的方法都是一样的并且很简单！

## 本书的主要内容

第 1 章通过介绍一种对现实生活视觉问题有效的解决办法来解释设计模块和体系。

第 2 章讲述 **OpenCV** 库的基本原理以及如何在不同的操作系统平台安装和测试 **OpenCV** 库。

第 3 章通过实用的技巧和现实生活的例子来展示从环境中收集数据的有效方法。

第 4 章介绍 **OpenCV** 库中的内置滤波器以及如何根据应用程序的上下文来选择和实现这些滤波器。

第 5 章探索如何从视觉数据中提取有意义的信息以及如何让这些提取出来的数据为识别系统做好准备。

第 6 章通过实践的方式探索人工智能的方法，讨论在一个给定场景中检测和识别物体的方法。

第 7 章揭示 **Arduino** 的通信能力以及在 **OpenCV** 的环境中连接 **Arduino** 的综合性最佳实践。

第 8 章演示如何通过物理行为优雅地与现实世界进行交互。

第 9 章综合前面所学的所有知识来设计和开发一个全功能一体化的机器人。该章还将介绍所提出的方法如何在问题独立的情况下取得成功。

## 阅读本书前的准备

本书的目的是教会你如何开发可以与现实生活进行交互且启用了计算机视觉的系统。书中所有必需的软件都是免费的。当然，根据需要，你也可以使用收费的软件来开发应用程序。

计算机视觉应用程序是使用 **OpenCV** 开发的。**Eclipse**、**Xcode**、**Visual Studio** 作为集成开发环境使用。应用程序的嵌入式系统部分通过使用 **Arduino Uno R3** 和 **Arduino** 的集成编辑器来开发。

有几章中的应用程序需要用到电子元器件，比如传感器或通信模块。这些模块都很便宜



并且很容易找到。应用程序背后的思想是通用的，可以适用于各种相关领域。

## 本书的目标读者

本书主要面向想要在项目中使用计算机视觉来创造更多智能系统的读者。特别是想学习如何使用 **Arduino** 进行计算机视觉编程的用户和对此有兴趣的爱好者。

机器人学是兴趣电子学中最流行的领域之一。我们可以简单地说只要掌握了本书的内容，就完全有可能建立一个可以看到环境并进行推理的机器人。同一方案适用于消费类电子学和兴趣电子学的所有其他领域。任何对这一方案感到兴奋的人都是本书的目标读者。

本书假设读者熟悉 **Arduino** 的基本知识。对于计算机视觉编程方面的知识则没有什么要求。

## 下载示例代码和彩色图片

你可以从华章网站 (<http://www.hzbook.com>) 下载本书的示例代码和彩色图片。

# Contents | 目 录

译者序

作者简介

前言

## 第 1 章 计算机视觉系统综述 ..... 1

- 1.1 计算机视觉系统介绍 ..... 1
- 1.2 解决计算机视觉问题 ..... 2
  - 1.2.1 数据采集 ..... 3
  - 1.2.2 预处理 ..... 4
  - 1.2.3 图像处理的特征提取 ..... 5
  - 1.2.4 后处理和后置滤波 ..... 7
  - 1.2.5 识别或检测 ..... 7
  - 1.2.6 在现实世界中行动 ..... 8
  - 1.2.7 连接器模块 ..... 8
- 1.3 总结 ..... 10

## 第 2 章 OpenCV 的基础与安装 ..... 11

- 2.1 OpenCV 的基础 ..... 11
- 2.2 OpenCV 的安装 ..... 12
  - 2.2.1 在 Linux 上安装 OpenCV ..... 12
  - 2.2.2 在 Mac OS 上安装 OpenCV ..... 17
  - 2.2.3 在 Windows 上安装 OpenCV ..... 18
  - 2.2.4 在 iOS 上安装 OpenCV ..... 20

2.2.5 在 Android 上安装

OpenCV ..... 21

2.3 总结 ..... 24

## 第 3 章 用 OpenCV 和 Arduino

进行数据采集 ..... 25

- 3.1 图像和视频采集 ..... 25
  - 3.1.1 相机选择 ..... 25
  - 3.1.2 图像采集 ..... 29
- 3.2 传感器数据采集 ..... 41
  - 3.2.1 设置 Arduino 环境 ..... 41
  - 3.2.2 传感器基础 ..... 42
  - 3.2.3 从温度传感器中读取数据 ..... 44
- 3.3 总结 ..... 47

## 第 4 章 用 OpenCV 进行数据

过滤 ..... 48

- 4.1 开始过滤 ..... 48
- 4.2 空间域滤波 ..... 49
  - 4.2.1 平滑 ..... 50
  - 4.2.2 锐化 ..... 51
- 4.3 颜色空间转换 ..... 51
  - 4.3.1 灰度化 ..... 52
  - 4.3.2 二值化 ..... 53
- 4.4 形态学滤波器 ..... 55
  - 4.4.1 腐蚀和膨胀 ..... 55

4.4.2	开运算和闭运算	56	6.1.5	使用支持向量机	97
4.5	梯度和边缘检测	58	6.2	总结	98
4.5.1	Canny 边缘检测器	58	<b>第 7 章 用 OpenCV 与 Arduino 进行通信</b> 100		
4.5.2	LoG 滤波器	59	7.1	与 Arduino 进行通信	100
4.5.3	Sobel	61	7.1.1	有线通信	102
4.6	自定义滤波器	63	7.1.2	无线通信	111
4.7	直方图均衡化	65	7.2	用 Java 进行通信	115
4.8	本章的实践项目	66	7.3	用 C++ 进行通信	120
4.9	总结	72	7.4	总结	121
<b>第 5 章 用 OpenCV 进行视觉数据处理</b> 73			<b>第 8 章 将 Arduino 用于真实世界</b> 122		
5.1	提取特征	73	8.1	与电动机的连接	122
5.1.1	使用基本的统计信息	74	8.1.1	驱动直流电动机	123
5.1.2	使用颜色特征	78	8.1.2	驱动步进电动机	125
5.1.3	使用模板进行特征匹配	78	8.1.3	驱动伺服电动机	128
5.1.4	使用轮廓特征	79	8.2	使用显示器	130
5.1.5	使用凸包	80	8.2.1	使用液晶显示器	131
5.1.6	使用矩	81	8.2.2	使用薄膜晶体管显示器	133
5.1.7	使用 Hough 变换	82	8.3	总结	137
5.1.8	使用角点	83	<b>第 9 章 建造一个点击行走机器人</b> 138		
5.1.9	使用尺度不变特征变换	84	9.1	系统总览	138
5.1.10	使用加速鲁棒特征	85	9.2	建造一个机器人	141
5.1.11	使用 ORB	86	9.2.1	构建力学部分	142
5.1.12	使用 blob 分析	86	9.2.2	构建电子电路部分	143
5.2	总结	87	9.2.3	构建机器人控制器和通信系统	146
<b>第 6 章 用 OpenCV 进行识别</b> 88			9.3	构建视觉应用程序	154
6.1	构建一个可以思考的应用程序	88	9.4	总结	163
6.1.1	模板匹配	89			
6.1.2	特征匹配	91			
6.1.3	基于快速近似最近邻搜索库的匹配	94			
6.1.4	使用级联分类器	95			

# 第 1 章

## 计算机视觉系统综述

在本章中，你将会了解计算机视觉系统的基础概念和整体框架。这样当解决计算机视觉问题的时候，你将有一个更广阔的视野。

### 1.1 计算机视觉系统介绍

我们用五种感官（触觉、味觉、嗅觉、听觉和视觉）来观察周围的一切。尽管这五种感官都非常重要，但是有一种感官对于感知有最大的影响。毫无疑问，它就是视觉，这也是本书的主题。

当看到一个场景时，我们能在一个有意义的上下文中理解和解释看到的细节。这看起来很容易，但其实这是一个非常复杂难以建模的过程。是什么让人类的眼睛容易理解视觉而机器难以理解？答案隐藏于人类和机器在感知上的区别。许多研究人员正在试图揭示这个区别。

在计算机视觉发展的道路上，照相机的发明是其中最重要的一个里程碑。尽管照相机是一个保存场景视觉记忆的很好的工具，但是它导致了很多变化，而不仅仅是保存场景。与相机的发明一样，人们总是试图开发设备来让我们的生活更加美好。当前的趋势是开发智能设备，知晓周边的环境肯定是其中重要的一环。这和我们的日常生活体验或多或少是相同的，视觉是最大的影响因素。多亏了科技的发展，使得模拟人类的视觉系统是可能的并且已经在各种各样的设备上实现了。在这个过程中，我们有能力开发可视化设备。

图片和时间序列的图片被称为视频，换句话说就是真实世界的机器表示。任何可视化设备都是通过图片来重建真实的场景。因为通过设备从图片提取解释和隐藏的信息非常复杂，通常使用计算机来达到目的。计算机视觉这个词就是来自让机器用人类的方式理解真

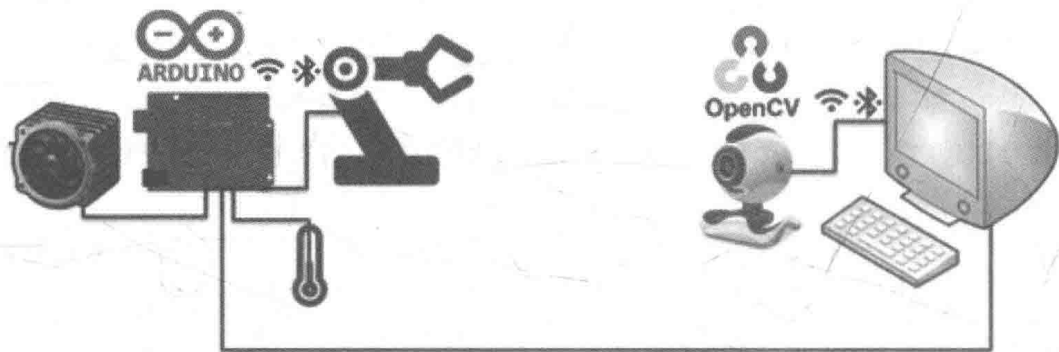
实世界的现代方法。由于计算机视觉对机器的自动化日常工作十分必要，因此它的发展非常迅速，并且大量的框架、工具和库已经被开发出来。

OpenCV (Open Source Computer Vision Library) 的出现对于计算机视觉来说是一个非常大的改变，并且很多人都对它有贡献以便让它变得更好。现在它是一个成熟的库，提供了最先进的设计模块，在后面的章节我们会接触到。因为它是一个易于使用的库，不需要为了完成视觉计算任务而去搞清楚底层到底发生了哪些复杂计算。这种易用性让那些复杂的任务更容易完成，但是即便是这样，你也应该知道如何在解决问题和使用设计工具中找到平衡。

## 1.2 解决计算机视觉问题

解决任何复杂的问题，比如计算机视觉问题，非常重要的一点就是通过理解每一步的目的来把问题分解成简单和可实现的子步骤。本章的目的是向你展示如何解决计算机视觉问题，以及如何用一个通用模型来对问题建模。

本书讲解的实用计算机视觉系统的架构，由 Arduino 系统和 OpenCV 系统组合而成，如下图所示：

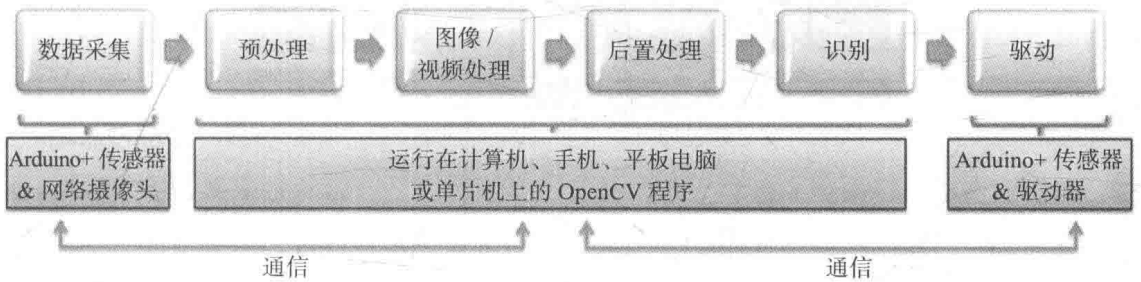


Arduino 只负责从环境中收集感官信息，比如温度、湿度，然后把这些信息发给视觉系统控制器的 OpenCV 系统。视觉系统控制器和 Arduino 之间的通信可以是有线的，也可以是无线的，Arduino 对于这两种方式处理起来都很容易。视觉系统处理完从 Arduino 和网络摄像头传过来的数据后，会检测或者识别出一个结论。比如，它设置可以识别出你的脸。下一步是根据这个结论来给 Arduino 系统发送指令，让 Arduino 采取合适的动作。这些动作可能是开风扇来让环境冷却，移动机械臂来给你递咖啡等。



视觉控制器可以是台式电脑、笔记本电脑、手机，甚至是树莓派或者 Beaglebone 这样的迷你电脑！OpenCV 可以在所有这些平台上运营，所以基本原则在这些平台上都是有效的。迷你电脑也可以做部分 Arduino 的工作。

任何计算机视觉系统都由一系列边界清晰的设计模块组成，它们依次是数据采集、预处理、图像处理、后置过滤、识别（检测）和驱动。本书将针对这些步骤介绍一些详细实用的方案。我们把这些步骤对应到平台上具体实现，就可以得到一个计算机视觉系统的通用流程图。在下图中，你可以看到计算机视觉系统的通用处理流程：



### 1.2.1 数据采集

可以看到，视觉系统处理的第一步是数据采集，通常是收集来自环境的感官信息。从视觉控制器的角度来看，主要有两个数据源：相机和 Arduino 系统。

相机是模仿人类视觉系统的终极传感器并与体系中的视觉控制器直接相连。通过使用 OpenCV 的数据采集功能，视觉控制器可以从相机中读取视觉数据。数据可以是快照图片或者由一帧帧数据按时间组成的视频。相机的种类并没有什么限制。

最根本区分相机类型的方法是根据输出信号是模拟信号还是数字信号进行区分。本书例子中所采用的相机都是数码相机，因为处理环境和处理操作本身都是数字的。图片的基本组成元素被称为像素。在一张数码图片中，一个像素或者说图片元素是光栅化图片上的一个物理点，或者是全屏光点式显示设备上的最小寻址单位。所以它是屏幕上图片的最小控制单位。你可以在网址 <http://en.wikipedia.org/wiki/Pixel> 找到更多的信息。

相机还可以按颜色感应能力进行分类。RGB 相机既能感觉主要颜色成分也能感应这些主要颜色的混合。灰度相机能感应到的场景都是灰色的。因此，这种相机提供的不是颜色信息，而是场景的形状信息。最后，二进制摄像机识别的场景只有黑色和白色。顺便说一下，二进制相机的一个像素只有 2 个值，即黑色和白色。

相机的另外一种分类方法是基于它们的通信接口。比如 USB 相机、IP 相机、无线相机，等等。相机的通信接口也直接影响相机的可用性和功能。在家里我们一般用带 USB 接口的网络摄像头。在使用时通常不需要外部电源或者其他外部设备，所以它很适合用于图像处理。相机也有分辨率等属性，我们会在后面的章节处理相机属性。

最常部署为网络摄像头的 USB 相机提供的是 2D 图片。除了 2D 相机系统，我们现在有可以检测场景中每个元素深度的 3D 相机系统。三维摄像机系统中最著名的可能是 Kinect，如下图所示：



OpenCV 支持各种类型的相机，并且可以通过非常简单的接口从这些相机中读取视觉信息。在随后的章节我们将介绍相关的例子。请记住，图像采集是视觉处理流程最基本的步骤，我们有很多选择。

一般来说，我们还需要获取相机以外的信息来分析我们周围的环境。其中一些信息是和其余四个感官相关的。此外，有时我们需要超出人体感知能力的额外信息。我们可以通过 Arduino 传感器来获得这些信息。

想象下你要构建一个面部识别自动开关门的项目。系统可能会由敲门声或者门铃触发。你需要一个声音传感器来对敲门声或者门铃声做反应。这些信息很容易由 Arduino 收集到。让我们添加一个指纹传感器，使其更加安全！通过这种方式，视觉系统可以结合 Arduino 和相机的数据来得出关于场景的结论。

总之，视觉系统控制器可以通过相机和 Arduino (使用传感器) 来获取详细的环境信息。

## 1.2.2 预处理

预处理的意思是把东西准备好以便进行处理。它可以包含各种各样的子步骤，但是基本原理是一样的。我们现在开始解释什么是预处理以及它的重要性。

首先，让我们明确一件事。这一步的目的是把收集到的视觉数据整理好准备进行处理。计算机视觉系统需要预处理的原因是，原始的数据一般会有很多噪音。图片数据是从

相机得到的，里面有很多区域是不需要的，有时候由于振动、运动等因素会得到一个模糊的图像。在任何情况下，最好都过滤一下图片来让它对我们的系统更加有用。比如，如果你想检测图像中的一个大红球，你就可以删除那些小点，你甚至可以移除那些不是红色的部分。所有这些过滤操作都会让后续操作更简单。

一般来说，相机都在数据获取的时候做了过滤，但是每一个相机都有不同的预处理功能，甚至有一些相机做了防振动处理。但要注意的是内置功能的增加也会带来成本的增加。所以最好在内部通过 OpenCV 进行这些过滤处理。顺便说一句，即使使用的是廉价的设备比如网络摄像头，也能设计出一个鲁棒的视觉系统。

同样的处理也适用于传感器数据。在真实环境中我们得到的总是有噪音的数据，所以噪音应该从传感器的真实数据中移除。这些噪音部分是由环境引起的，部分是由传感器的内部结构引起的。在任何情况下，数据应该被准备好以便进行处理。本书将给出实际可行的方法来达到这一目标。

图像数据的复杂性远远超过任何常规的传感器，比如温度传感器、湿度传感器，这很好理解。用来表征信息的数据维度也是不同的。RGB 图片的每个像素包含三个颜色组件：红色、绿色和蓝色。为了用分辨率  $640 \times 480$  来表征一个场景，一个 RGB 相机需要  $640 \times 480 \times 3 = 921\,600$  字节。乘以 3 的原因是每个像素有三个维度。每个像素对应 3 个字节的数据，1 个字节代表一个颜色。为了表征房间的温度，我们一般需要 4 个字节的数据。这也解释了为什么我们需要更强的设备来处理图像。此外，图像过滤的复杂性也和简单的传感器过滤不同。

但这并不意味着不可以用一个简单的方法来进行复杂的过滤。如果我们知道过滤的目的和过滤参数的含义，就可以更加容易地使用它们。这本书的目的是让你了解过滤的过程和如何更轻松地使用先进的过滤技术。

所以过滤的目的是为了从数据中获取真实的数据信息，这是计算机视觉计算过程中不可或缺的一步。许多计算机视觉项目在开发阶段失败了，就是因为缺少过滤这一层。即使是最好的识别算法，遇到充满噪音和不准确的数据也会失败。所以，请重视数据过滤和预处理。

### 1.2.3 图像处理的特征提取

计算机视觉项目最让人激动的部分大概就是对场景的自动化解释。为了从图像中提取含义，我们应用了各种各样的图像处理技术。有时候一张图片的信息并不够。在这种情况



下，图像帧之间的关系就非常重要。如果这种帧与帧之间的信息会被提取出来，这种处理通常被称为视频处理。许多视频处理应用也包括了对声音信号的处理。因为采用相同的基本原则，视频处理和图像处理区别不大。

理解本章的重点在于逻辑地去看待真实生活中的应用。想象一下你在建一个沿着线行走的视觉机器人。机器人中间上方会有一个相机，它会在白色地板上沿着黑线行走。为了实现这个目标，你需要检测黑线，并找出黑线现在是在机器人的左边还是右边。如果黑线出现在图片的左边部分，你应该让机器人往左走让黑线在中间。同样，如果黑线出现在图片的右边部分，你应该让机器人往右走。如果黑线差不多在图片的中间，你就该让机器人往前走。你也可以检测黑线的方向来提前规划机器人的行进，以便让它的移动更加聪明合理。

在这个例子中，你应该使用图像处理技术来检测图像中的线。你将在本书中看到一系列技术方案，在你解决问题时，这些技术方案将为你指引方向。使用这些技术方案，可以在图像中得到一些可能是线的候选区域。为了有效找到线的候选，你需要在这些候选区域上使用特征提取。通过对特征（属性）的对比，你就可以把真正的线从诸如阴影等噪音中分离出来。

特征提取是一个模式识别和分类的术语，它意味着提取一些可以代表更大数据集的小数据。通过处理图像，我们提取了长度、位置、图像区域等特征。在后面的章节里，我们将用这些特征来检测和识别物体。

从图像中提取有用的小集合数据有一些主要的方法。分割就是针对这种操作的一个术语。

图像分割是把一个数码图像分成多个点集的过程。分割的目的是为了简化问题以及改变图像的表达方式，以一种更容易处理也更有意义的方式表示出来。



更多信息可以在 [http://en.wikipedia.org/wiki/Image\\_segmentation](http://en.wikipedia.org/wiki/Image_segmentation) 找到。

在我们的例子中，线段的候选就是这些图像的分割。

blob 分析是对图像分割进行标记的好方法。它对于找到图像中的闭环非常有用。闭环一般对应着图像中的物体。blob 分析也是一种图像处理技术，我们会在后面讲到。现在最重要的是理解特征拾取的目的。