

## <<OpenCV2计算机视觉编程手册>>

### 图书基本信息

书名：<<OpenCV2计算机视觉编程手册>>

13位ISBN编号：9787030375810

10位ISBN编号：7030375815

出版时间：2013-6-25

出版单位：科学出版社有限责任公司

作者：Robert Laganiere

译者：张静

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<OpenCV2计算机视觉编程手册>>

### 内容概要

《OpenCV2计算机视觉编程手册》以案例的形式介绍OpenCV 2.X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。

本书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。

本书既适合想要学习计算机视觉的C++初学者，也适合专业的软件开发人员。

本书可作为高等院校计算机视觉课程的辅助教材，也可以作为图像处理 and 计算机视觉领域研究人员的参考手册。

## 作者简介

作者：（加拿大）Robert Laganiere 译者：张静Robert Laganiere，加拿大渥太华教授，于1996年获得蒙特利尔INRS-Telecommunications博士学位。

他是计算机视觉领域的研究员，兴趣包括食品分析、智能视觉监控，以及基于图像的建模。他是VIVA实验室的联合创始人，同时还是一家提供云端视频监控服务的公司的首席科学家。他与别人合著了Object-oriented Software Engineering一书，由McGraw-Hill于2011年出版。

## 书籍目录

第1章接触图像 1.1引言 1.2安装OpenCV库 1.3使用MS Visual c++创建OpenCV工程 1.4使用Qt创建OpenCV项目 1.5载入、显示及保存图像 1.6使用Qt创建GUI应用 第2章操作像素 2.1引言 2.2存取像素值 2.3使用指针遍历图像 2.4使用迭代器遍历图像 2.5编写高效的图像遍历循环 2.6遍历图像和邻域操作 2.7进行简单的图像算术 2.8定义感兴趣区域 第3章基于类的图像处理 3.1引言 3.2在算法设计中使用策略 (Strategy) 模式 3.3使用控制器 (Controller) 实现模块间通信 3.4使用单件 (Singleton) 设计模式 3.5使用模型—视图—控制器 (Model—View—Controller) 架构设计应用程序 3.6颜色空间转换 第4章使用直方图统计像素 4.1引言 4.2计算图像的直方图 4.3使用查找表修改图像外观 4.4直方图均衡化 4.5反投影直方图以检测特定的图像内容 4.6使用均值漂移 (Mean Shift) 算法查找物体 4.7通过比较直方图检索相似图片 第5章基于形态学运算的图像变换 5.1引言 5.2使用形态学滤波对图像进行腐蚀、膨胀运算 5.3使用形态学滤波对图像进行开闭运算 5.4使用形态学滤波对图像进行边缘及角点检测 5.5使用分水岭算法对图像进行分割 5.6使用GrabCut算法提取前景物体 第6章图像滤波 6.1引言 6.2使用低通滤波器 6.3使用中值滤波器 6.4使用方向滤波器检测边缘 6.5计算图像的拉普拉斯变换 第7章提取直线、轮廓及连通区域 7.1引言 7.2使用Canny算子检测轮廓 7.3使用霍夫变换检测直线 7.4用直线拟合一组点 7.5提取连通区域的轮廓 7.6计算连通区域的形状描述符 第8章检测并匹配兴趣点 8.1引言 8.2检测IJHarris角点 8.3检测IJFAST特征 8.4检测尺度不变的SURF特征 8.5描述SuRF特征 第9章估算图像间的投影关系 9.1引言 9.2相机标定 9.3计算一对图像的基础矩阵 9.4使用随机采样一致算法 (RANSAC) 进行图像匹配 9.5计算两幅图之间的单应矩阵 第10章处理视频序列 10.1引言 10.2读取视频序列 10.3处理视频帧 10.4写入视频序列 10.5跟踪视频中的特征点 10.6提取视频中的前景物体

## 章节摘录

版权页：插图：作用原理 在之前的秘诀中，我们学习到能够从一组特征点的匹配中估算出与两幅图像相关的基础矩阵。

精确地说，这组匹配仅包含高质量的匹配。

然而，在实际应用中，很难保证通过比较特征点描述子得到的匹配集是完全相同的。

因此我们将使用RANSAC (RANdom SAmpling Consensus) 策略来进行基础矩阵的估算。

RANSAC算法的目的是从包含异常值的数据集中估算出给定的数学元素。

基本原理是随机地选取一些数据点，并且仅用它们来进行估算。

选择的数据点的个数应当是可用于进行估算的最小数。

对于基础矩阵而言，8个匹配是最小数（事实上可以是7，但是8个点的线性算法计算更迅速）。

一旦从这随机的8个匹配中算出基础矩阵，集合中所有剩下的匹配都将与矩阵对应的极性约束进行测试。

我们找到所有满足该约束的匹配，它们对应的特征非常靠近极线。

这些匹配组成了这个基础矩阵的支持集合。

RANSAC算法背后最主要的想法是支持集合越大，得到正确矩阵的可能性就越大。

显而易见的，如果一个（或多个）随机选择的匹配是错误的，那么得到的基础矩阵也是错误的，于是它的支持集合应当很小。

这个过程会重复数次，最后我们保留最大支持集合的矩阵作为最合适的。

因此，我们的目标是多次随机挑选8个匹配，最终能够得到8个足够好的匹配提供我们一个较大支持集合。

根据完整数据集中错误匹配的个数，挑选到8个正确数据的概率会不同。

然而我们值得挑选的次数越多，我们从中得到至少一个优质匹配集合的概率就越大。

更精确地，如果假设集合包含 $n\%$ 正确值，那么同时选中8个正确匹配的概率是 $8n$ 。

因此，包含至少一个错误匹配的概率是 $(1-8n)$ 。

如果我们挑选 $k$ 次，至少出现一次包含8个正确结果的概率是 $1-(1-8n)^k$ 。

这便是置信概率 $C$ ，我们希望它尽可能地高。

因此，当运行RANSAC算法时，我们需要确定 $k$ 的数量以得到给定的置信等级。

在`CV::findFundamentalMat`函数中使用RANSAC算法时，需要提供两个额外的参数。

第一个是置信等级，它决定迭代的次数。

第二个是归类为正确的点离极线的最大距离。

因此，该函数返回一个字符类型的`std::vector`，标志着对应的匹配被识别为`outlier (0)` 还是`inlier (1)`。

初始数据集中优质匹配的数量越大，RANSAC给出正确基础矩阵的概率也越高。

因此我们在调用`CV::findFundamentalMat`之前便对数据集使用了多个滤波器。

当然，你可以选择跳过其中的一些步骤。

这仅仅是如何在计算复杂性、最终匹配数目以及所需的置信等级之间进行平衡的问题。

## <<OpenCV2计算机视觉编程手册>>

### 编辑推荐

《OpenCV2计算机视觉编程手册》是一本循序渐进的计算机视觉指导手册，给予OpenCV2代码库中包含高级特性的C++接口。

《OpenCV2计算机视觉编程手册》介绍了OpenCV2中众多的视觉算法。

你将学会如何读、写、创建及操作图像，领略图像分析中常用的技术，并了解如何使用C++高效实现。

cvchina.info推荐，计算机视觉编程领域的最新力作，50个秘诀，让你轻松进阶。

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>