



快速解答常见问题

OpenCV 2 计算机视觉

(加) Robert Laganière 著
张 静 译

编程手册

cvchina.info 推荐

精通实时计算机视觉编程的 50 多个秘诀



科学出版社

OpenCV 2

计算机视觉编程手册

[加] Robert Laganière 著

张 静 译

科学出版社

图字：01-2012-0919号

内 容 简 介

本书以案例的形式介绍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和计算机视觉领域研究人员的参考手册。

图书在版编目（CIP）数据

OpenCV 2计算机视觉编程手册/（加）Robert Laganière著；张静译.—北京：科学出版社，2013.7

书名原文：OpenCV 2 Computer Vision Application Programming Cookbook
ISBN 978-7-03-037581-0

I.O… II.①R… ②张… III.图像处理软件-程序设计-技术手册
IV.TP391.41-62

中国版本图书馆CIP数据核字（2013）第111236号

责任编辑：喻永光 杨 凯 / 责任制作：魏 谨

责任印制：赵德静 / 封面制作：段淮沅

北京东方科龙图文有限公司 制作

<http://www.okbook.com.cn>

科 学 出 版 社 出版

北京东黄城根北街16号

邮政编码：100717

<http://www.sciencep.com>

北京源海印刷有限责任公司 印刷

科学出版社发行 各地新华书店经销

*

2013年7月第 一 版 开本：787×960 1/16

2013年7月第一次印刷 印张：16 1/4

印数：1—3 500 字数：289 000

定价：45.00元

（如有印装质量问题，我社负责调换）

译者序

我和OpenCV的故事

一切的起源是OpenCV_1.0.exe的安装包，那时我电脑中安装的还是 VC++6.0 绿色版，我以为计算机视觉只是图像处理的别名。当我运行samples/facedetect 时惊呆了，笔记本自带的摄像头居然实时地识别出我的脸。后来我意识到这句话说错了，其实是检测到了我的脸。人脸识别指的是计算机知道我是谁，这个功能在当时的OpenCV是不支持的。另外一个让我震惊的是samples/inpaint.exe，字面意思是图像修补，即智能地填补缺失的像素信息，那会儿我想这可以用来去马赛克。后来想明白了，信息是不会凭空出现的，如果你事先不知道马赛克背后可能是什么，那么无法去恢复它。

2008年上海双年展上，我看到一款摄像头互动的艺术作品《上海，我能邀你跳支舞吗？》，参展者在摄像头前通过身体来影响投影屏幕中的虚拟建筑。这是我从没想象过的人机交互方式，我意识到程序的操控设备可以脱离鼠标键盘，进入到现实的空间中。回来后我用OpenCV自制了一个用双手控制的虚拟打鼓游戏，录好视频后放在优酷上，启发了一些人。这便是最原始的体感，在硬件上只依赖一个摄像头。一年后的E3展上，Kinect以Project Natal的名称为世人所知。

提到摄像头互动，自然不能不说基于CCV的多点触摸，国内外不少公司都在凭借这款开源的软件接活。而CCV最核心的摄像头跟踪模块完全依赖OpenCV，所做的无非是阈值化加上寻找连通区域。因为打开了OpenCV这扇门，我接触到截然不同的编程世界。研究CCV的间隙我还学习了openFrameworks和Processing，今年我还有一本Processing 的翻译书籍要出版，这是另外一个故事。

之后从事了游戏行业，基本没机会用OpenCV，直到一个游戏项目中的字体贴图丢失了坐标信息，需要通过肉眼手动定位。这很不科学，于是我拾起OpenCV，阈值化加上寻找连通区域，输入一副图像，输出字体的包围盒坐标，同事纷纷觉得不可思议。计算机视觉的魅力便在此，从看似无关的像素中能够提取信息。这不是唯一一次在工作中使用OpenCV，之后我得到了一份增强现实、图像检索等领域的开发工作，可以全

职地进行OpenCV的开发。共事的有国内著名CV网站cvchina.info的站长张小军，本书的第2章也是由他翻译的，对于他在工作中的指点在此一并感谢。

OpenCV的用途很广，除了上述提到的，还有视频监控、双目视觉、机器学习、Kinect深度信息处理、照片美化、图像拼接、GPU加速等功能。但是OpenCV的书籍并不多，在本书面世前市场上只有一本讲述1.0接口的*Learning OpenCV*。本书的意义在于，讲述了2.0后出现的C++接口以及大量的新功能。如果你看过*Learning OpenCV*，想学习新接口使用，那么推荐你阅读这本书。如果你没有接触过OpenCV，想直接跳过C接口的函数，那么更加推荐你阅读本书。

最后，感谢陪伴我鼓励我的妻子M，如果没有你，我和OpenCV的故事到第二段就结束了，译者序也必定由他人撰写。

张 静

2013年4月25日 上海

前言

图像与视频在当今的数字世界可谓无处不在，随着运算能力强劲而又实惠的计算设备的问世，创建复杂的图像应用从未像今天这般容易。市面上有众多的软件和库可用于操作图像与视频，但是对于期望自己开发软件的人而言，OpenCV库是一款必备的工具。

OpenCV（Open Source Computer Vision）是一个开放源代码的图像及视频分析库，它包含500多个优化过的算法。自1999年问世以来，它已经被计算机视觉领域的学者和开发人员视为首选工具。OpenCV最初是由Intel的一个小组进行开发的，领头人是Gary Bradski，作为视觉领域研究的先锋，他推动了许多CPU密集的视觉应用。在发布一系列Beta版本后，1.0版本终于在2006年面世。第二次重要的版本发布是2009年的OpenCV 2，它带来了重要的变化，尤其是包括本书中使用的崭新的C++接口。写作本书时，最新的版本是2.2（2010年10月）。

本书覆盖了库中的许多特性，并且展示了如何使用它们完成特定的任务。我们的目标并非事无巨细地讲述OpenCV提供的函数与类，而是教你零基础进行视觉开发的必备知识。本书中还解释了图像分析中的基本概念，并且对计算机视觉中的一些重要算法进行了介绍。

本书对读者而言是接触图像处理与视频分析的一次良机，但这也只是开端。OpenCV一直处于改进与扩展功能的阶段，查询在线文档可以了解最新的进展：<http://opencv.org/>。

本书内容

第1章 接触图像：介绍OpenCV库，并展示如何使用微软Visual C++与Qt开发环境运行简单的应用。

第2章 操作像素：解释如何读取图像，将使用不同的方法扫描图像以进行逐像素操作。你也将学习到如何定义图像中的兴趣区域。

第3章 基于类的图像处理：展示多种面向对象的设计模式，将帮助你更好地构建

计算机视觉应用。

第4章 使用直方图统计像素：介绍如何计算图像的直方图，以及利用直方图进行图像修改。利用直方图可以实现图像分割、物体检测以及图像检索。

第5章 基于形态学运算的图像变换：探索形态学变换的概念，展示了不同的形态学操作子以及它们在检测边缘、角点与线段中的用途。

第6章 图像滤波：介绍频域分析及图像滤波的原理。展示了低通、高通滤波器在图像中的应用，并提出两个图像微分算子——梯度和拉普拉斯算子。

第7章 提取直线、轮廓及连通区域：专注于讲解对几何图像特征的检测，解释如何从图像中提取轮廓、线条以及连通区域。

第8章 检测并匹配兴趣点：介绍不同的兴趣点探测算法，还解释如何计算特征点的描述符，以及如何使用描述符来匹配图像。

第9章 估算图像间的投影关系：分析图像形成中涉及的不同关系，还探讨了同一个场景的两幅图像之间存在的投影关系。

第10章 处理视频序列：提供一个框架来读写视频并且处理每一帧，还展示了如何在相邻帧之间跟踪兴趣点，以及如何提取相机前移动的前景物体。

本书读者

如果你是一个C++程序员新手，并且想学习如何使用OpenCV库来构建计算机视觉应用，那么本书非常适合你。同时，它也适合希望学习计算机视觉编程的专业开发人员，可以作为大学计算机视觉课程的学生用书。对于图像处理和计算机视觉领域的研究生及研究人员而言，它也是一本极佳的参考手册。本书很好地综合了基础知识与进阶内容。当然，这要求读者掌握一定的C++技能。

本书约定

在这本书中，你会发现许多样式的文字，它们有着不同的含义。下面是一些样式的范例，以及它们的用途。

代码中的单词如下显示：“通过使用include指令我们能够引入其他的上下文。”

代码块的样式则是：

```
//获取迭代器
cv::Mat_<cv::Vec3b>::const_iterator it=
    image.begin<cv::Vec3b>();
```

当我们希望将读者的注意力集中到某个代码块的部分时，相关的代码行或者文字将设为粗体：

```
//转换为Lab色彩空间
cv::cvtColor (image, converted, CV_BGR2Lab) ;
//获取转换后图像的迭代器
cv::Mat_<cv::Vec3b>::iterator it=
```



提示和技巧以这样的样式显示。

读者反馈

我们始终欢迎读者的反馈意见。我们渴望知道你对于本书的看法，喜欢哪些内容或者不喜欢哪些内容。你的真实感受，对于我们开发读者切实需要的图书十分重要。

如果你有反馈意见，请发送电子邮件至 feedback@packtpub.com，并在邮件主题中注明你评论的书名。

如果你希望我们出版某方面的书，请在 www.packtpub.com 填写“SUGGEST A TITLE”表格，或发送邮件至 suggest@packtpub.com。

如果你对某个主题有经验或者有兴趣，愿意撰写或参与撰写一本书，请查看 www.packtpub.com/authors 页面中的“作者指南”。

客户支持

现在，你已经拥有一本Packt出版的书籍了，为了让你的付出得到最大的回报，请注意以下事项。

下载资源包

你可以从你在 <http://www.packtpub.com> 的账户中，下载已购买的所有Packt书籍的相关文件。如果是在别处购买的这本书，你可以访问 <http://www.packtpub.com/support>，然后填上邮箱，让服务器直接把这些文件通过电子邮件发给你。

勘 误

虽然我们会尽全力确保本书内容的准确性，但错误仍时有发生。如果你发现了本书中的错误（包括文字和代码错误），而且愿意告诉我们，我们将十分感激。这样做不仅可以减少其他读者的挫败感，也有助于本书再版时的改进。如果你发现任何错误，都可以访问<http://www.packtpub.com/support>，选择相应的书，以勘误表的形式报告具体错误。一旦得到验证，你的意见将被采纳，勘误表也会被上传到我们的网站，或者添加到已有的勘误表中。已有的勘误表可以在<http://www.packtpub.com/support>查阅。

举报盗版

互联网上涉及各种媒体的盗版行为始终存在。Packt非常重视版权的保护和许可。无论您在互联网上发现对我们的作品的任何形式的盗版，请告诉我们其链接或网址，以便我们采取相应的补救措施。

请将涉嫌包含盗版资料的链接发送至copyright@packtpub.com。

感谢你保护作者的权益及我们继续为你提供有价值的内容的能力。

疑问解答

有关于本书的任何问题，你都可以通过邮箱questions@packtpub.com联系我们，我们将尽最大的努力解决。

目 录

第 1 章 接触图像

1.1 引 言	1
1.2 安装OpenCV库	1
1.3 使用MS Visual C++创建OpenCV工程	4
1.4 使用Qt创建OpenCV项目	12
1.5 载入、显示及保存图像	18
1.6 使用Qt创建GUI应用	22

第 2 章 操作像素

2.1 引 言	29
2.2 存取像素值	30
2.3 使用指针遍历图像	33
2.4 使用迭代器遍历图像	40
2.5 编写高效的图像遍历循环	43
2.6 遍历图像和邻域操作	46
2.7 进行简单的图像算术	50
2.8 定义感兴趣区域	53

第 3 章 基于类的图像处理

3.1 引 言	57
3.2 在算法设计中使用策略 (Strategy) 模式	57
3.3 使用控制器 (Controller) 实现模块间通信	63
3.4 使用单件 (Singleton) 设计模式	67
3.5 使用模型-视图-控制器 (Model-View-Controller) 架构设计应用程序	70

3.6	颜色空间转换	73
第 4 章	使用直方图统计像素	
4.1	引 言	77
4.2	计算图像的直方图	77
4.3	使用查找表修改图像外观	84
4.4	直方图均衡化	88
4.5	反投影直方图以检测特定的图像内容	89
4.6	使用均值漂移 (Mean Shift) 算法查找物体	95
4.7	通过比较直方图检索相似图片	99
第 5 章	基于形态学运算的图像变换	
5.1	引 言	103
5.2	使用形态学滤波对图像进行腐蚀、膨胀运算	103
5.3	使用形态学滤波对图像进行开闭运算	107
5.4	使用形态学滤波对图像进行边缘及角点检测	110
5.5	使用分水岭算法对图像进行分割	116
5.6	使用GrabCut算法提取前景物体	121
第 6 章	图像滤波	
6.1	引 言	125
6.2	使用低通滤波器	126
6.3	使用中值滤波器	130
6.4	使用方向滤波器检测边缘	132
6.5	计算图像的拉普拉斯变换	138
第 7 章	提取直线、轮廓及连通区域	
7.1	引 言	143
7.2	使用Canny算子检测轮廓	143

7.3	使用霍夫变换检测直线	146
7.4	用直线拟合一组点	156
7.5	提取连通区域的轮廓	160
7.6	计算连通区域的形状描述符	164
第 8 章	检测并匹配兴趣点	
8.1	引 言	167
8.2	检测Harris角点	167
8.3	检测FAST特征	177
8.4	检测尺度不变的SURF特征	180
8.5	描述SURF特征	185
第 9 章	估算图像间的投影关系	
9.1	引 言	189
9.2	相机标定	191
9.3	计算一对图像的基础矩阵	198
9.4	使用随机采样一致算法 (RANSAC) 进行图像匹配	202
9.5	计算两幅图之间的单应矩阵	211
第 10 章	处理视频序列	
10.1	引 言	215
10.2	读取视频序列	215
10.3	处理视频帧	219
10.4	写入视频序列	228
10.5	跟踪视频中的特征点	233
10.6	提取视频中的前景物体	239

第 1 章

接触图像

本章主要探讨以下内容：

- ◆ 安装 OpenCV 库；
- ◆ 使用 MS Visual C++ 创建 OpenCV 工程；
- ◆ 使用 Qt 创建 OpenCV 项目；
- ◆ 载入、显示及保存图像；
- ◆ 使用 Qt 创建 GUI 应用。

1.1 引言

本章将讲授 OpenCV 的基础元素，并将带领你完成最基本的操作：读取、显示及保存图像。在着手开发前，需要先安装库。本章的第一则秘诀将解释这个简单的过程。

你还需要一个顺手的开发环境（IDE），我们推荐微软的 Visual Studio 及开源的 C++ 开发工具 Qt。两则独立的秘诀将分别介绍如何使用这两个工具建立项目，你也可以使用其他 C++IDE。事实上，在本书中展示的项目都不依赖特定的环境或操作系统，你可以自由选择。但是，需要确保编译后的 OpenCV 库与你使用的编译器与操作系统相兼容。如果你遇到怪异的结果，或是应用程序莫名地崩溃，那么很有可能是因为不兼容。

1.2 安装 OpenCV 库

OpenCV 是开源的计算机视觉代码库，它既可以用于学术上也可以用于商业应用。只要遵守 BSD 协议，便可自由地使用、发布及修改。本则秘诀将介绍如何进行库的安装。

准备工作

OpenCV 的官方网址是 <http://opencv.org/>，在这里你可以找到最新发布的版本、在线文档以及大量有价值的资源。

实现方法

从网站中可以找到对应平台（Linux/Unix/Mac 或 Windows）的下载页，下载后需要解压到一个文件夹中（如 OpenCV2.2）。解压完成后，你将看到一组文件夹，其中 doc 文件夹中包含的是文档，而 include 文件夹中是所有的头文件，modules 文件夹中包含所有的源文件（别忘了它是开源的），而 samples 文件夹中则是许多简短的学习用范例。

如果在 Windows 系统中使用 Visual Studio 进行开发，你可以选择下载可执行的安装程序。运行安装程序不仅会复制源代码库，还会安装所有预编译的二进制文件。在这种情况下，你可以直接开始使用 OpenCV，否则还需要一些额外的步骤。

为了在你选择的环境中进行 OpenCV 开发，需要使用合适的 C++ 编译器生成二进制的库文件。你需要使用 CMake 工具，它位于 <http://www.cmake.org/>，这是另外一个开源的软件工具，基于平台无关的配置文件来控制编译的过程。你可以使用命令行运行它，但是使用图形化界面更方便些。使用后者，你只需要指定包含 OpenCV 库的文件夹及包含二进制文件的文件夹，然后点击 Configure 按钮以选择编译器的类型（这里选择的是 Visual Studio 2010），并再一次点击 Configure，如图 1.1 所示。

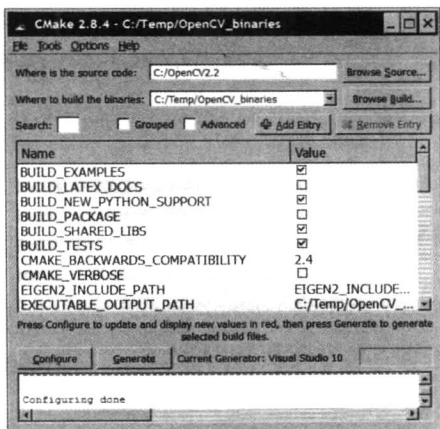


图 1.1

通过点击 **Generate** 按钮可以生成 `makefiles` 或是 `workspace` 文件。这些文件将允许你进行库的编译过程，这是安装的最后一步。

编译好库后，便可以进行开发了。如果你使用的是 `Visual Studio` 这种 IDE，那么你只需要打开顶层的 `Solution` 文件，这是 `CMake` 为你创建的。然后，开始 `Build Solution` 命令。在 `Unix` 环境中，你将使用 `makeutility` 命令来操作 `makefiles` 文件。

如果一切正常，那么你指定的文件夹中应该已经包含编译后的 `OpenCV` 库了。除了之前提到的文件夹，这个文件夹中还将包含一个 `bin` 文件夹。你可以把所有文件移动到合适的位置（如 `c:\OpenCV2.2`），并且将 `bin` 文件夹添加到系统路径中（在 `Windows` 系统中，通过打开控制面板中的系统（`System`）工具，在高级（`Advanced`）标签页中你将找到环境变量（`Environment Variables`）按钮）。

作用原理

自版本 2.2 开始，`OpenCV` 库便被划分为多个模块。这些模块编译成库文件后，位于 `lib` 文件夹中。

- ◆ `opencv_core` 模块，包含核心功能，尤其是底层数据结构和算法函数。
- ◆ `opencv_imgproc` 模块，包含图像处理函数。
- ◆ `opencv_highgui` 模块，包含读写图像及视频的函数，以及操作图形用户界面函数。
- ◆ `opencv_features2d` 模块，包含兴趣点检测子、描述子以及兴趣点匹配框架。
- ◆ `opencv_calib3d` 模块，包含相机标定、双目几何估算以及立体视觉函数。
- ◆ `opencv_video` 模块，包含运动估算、特征跟踪以及前景提取函数与类。
- ◆ `opencv_objdetect` 模块，包括物体检测函数，如脸部与行人检测。

库中还包含其他的工具模块，如机器学习（`opencv_ml`）、计算几何（`opencv_flann`），第三方代码（`opencv_contrib`）、废弃的代码（`opencv_legacy`）及 GPU 加速过的代码（`opencv_gpu`）。这些模块都有一个单独的头文件（位于 `include` 文件夹）。典型的 `OpenCV C++` 代码将包含所需的模块，本书推荐的声明方式如下：

```
#include<opencv2/core/core.hpp>
#include<opencv2/imgproc/imgproc.hpp>
#include<opencv2/highgui/highgui.hpp>
```

如果你看到 `OpenCV` 代码以如下方式开始：

```
#include "cv.h"
```

这是因为它使用了旧的代码方式，那时库还没有被划分为模块。

扩展阅读

你可以得到最新的代码，它们位于 OpenCV 的 GIT 服务器中：

```
git://code.opencv.org/opencv.git
```

从中你将会找到大量的范例，可以帮助你进行学习并且告诉你一些开发技巧。

1.3 使用MS Visual C++创建OpenCV工程

使用 Visual C++ 可以轻易地为 Windows 创建 OpenCV 应用。你能创建简单的控制台应用或是拥有图形用户界面的复杂应用。此处我们选择最简单的控制台应用。我们将使用 Visual Studio 2010，但是同样的步骤可以用于其他版本的微软 IDE，因为它们有着类似的菜单和选项。

如果你是第一次运行 Visual Studio，那么可以进行设置，将 C++ 视为默认开发环境。这样，当你启动时将处于 Visual C++ 模式。

假设你将 OpenCV 放置在 C:\OpenCV2.2 文件夹中，正如之前的秘诀所介绍的。

准备工作

使用 Visual Studio 进行开发时，很重要的一点是明白解决方案（Solution）与工程（Project）之间的区别。本质上说，解决方案是由多个工程组成的（每个工程是一个独立的软件模块，如一个程序和一个代码库）。这样做的好处是解决方案可以共享文件和代码库。通常，你为解决方案创建一个主文件夹，包括所有的工程文件夹。但是，你也可以将解决方案与工程放置于同一个文件夹内，这对于仅包含单个工程的解决方案来说是个惯例。当你熟悉 Visual C++ 并且创建更多复杂的应用之后，你应该使用包含多个工程的解决方案结构。

同时，当你编译和执行 Visual C++ 工程时，你可以在两种不同的配置下进行：Debug 及 Release。Debug 模式用于调试应用程序，这是个受保护的运行环境，它将告诉你程序是否含有内存泄露，在运行时也能对特定函数的结果进行检查。然而，它

生成的可执行文件运行较慢。因此，当你的应用经过测试准备投入使用时，你将在 Release 模式下进行编译，这将生成供最终用户使用的可执行文件。经常会发生 Debug 模式下运行正常而 Release 模式出错的情况，这时你需要进行更多测试以找到潜在的错误源。Debug 及 Release 模式并不仅在 Visual C++ 下才有，大部分 IDE 都支持这两种编译模式。

实现方法

现在我们将创建第一个工程。通过选择 File| New Project |Project... 菜单，你可以创建不同的工程类型。我们从最简单的 Win32 Console Application 开始，如图 1.2 所示。

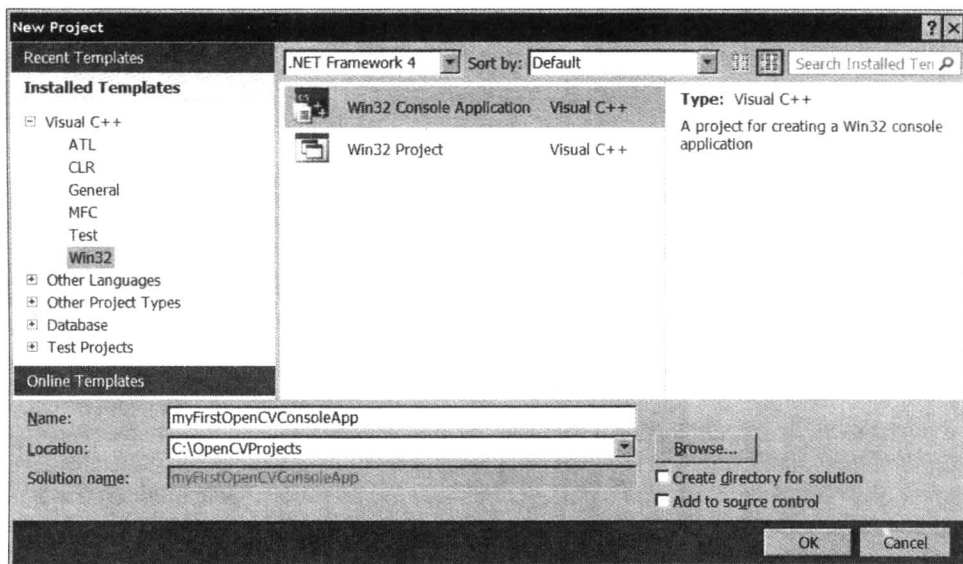


图 1.2

你需要指定项目创建的位置以及项目的名称，你也可以选择为解决方案创建一个文件夹（右下角）。如果选中，那么会创建一个额外的文件夹包含着解决方案。如果你简单地将它留着，那么扩展名为 .sln 的解决方案也会创建，但是它将与工程文件位于同一个文件夹。点击 OK 与 Next，进入 Win32 Application Wizard 的 Application Settings 窗口。正如图 1.3 所示，有一系列选项供选择，但是我们仅创建一个空的项目。