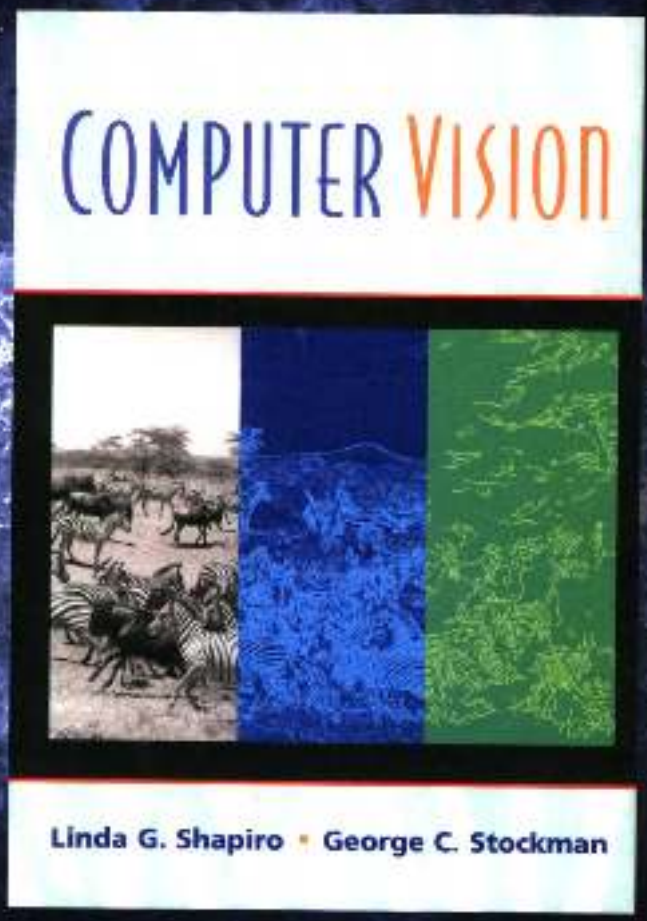
计 算 机 科 学 丛 书

计算机视觉

（美） Linda G. Shapiro George C. Stockman 著 赵清杰 钱芳 蔡利栋 译



Computer Vision

机械工业出版社
China Machine Press

长期以来，科学家与科幻作家一直梦想着人类能够制造出智能机器，而这种智能机器首先要能够对视觉信息进行理解。本书详细讨论了从图像自动抽取重要信息的理论和技术，把利用计算机视觉技术解决问题的重要研究内容汇集到一起。随着计算机技术的最新发展，计算机图像已经成为一种经济灵活的技术手段，并已渗透到各行各业。图像计算不再只属于科学研究领域，也属于艺术领域、社会科学领域，甚至成为人们的业余爱好。

本书适合作为计算机及相关专业的高年级本科生和研究生的教材，也适合相关技术人员参考。本书英文版被美国华盛顿大学等高等院校采用为教材。

本书特点：

- 除了传统内容外，增加了图像数据库、虚拟现实和增强现实方面的内容
- 介绍了两个运用计算机视觉技术的实际系统
- 应用面涉及工业、医学、地产、多媒体及计算机绘图
- 内含大量习题和编程项目，以及大量极具说服力的图片
- 书中提供大量相关网站，包括额外图像档案文件、图像处理代码和幻灯片等

作者简介

Linda G. Shapiro 是华盛顿大学计算机科学与工程学教授及电子工程学教授。她于1974年在艾奥瓦大学获得计算机科学博士学位，她曾在堪萨斯州立大学、维吉尼亚工学院、维吉尼亚州立大学任教，并在国际机器视觉组织负责智能系统方面的工作。Shapiro 教授曾经是 *Image Understanding* 杂志的主编，是 *Computer Vision and Image Understanding*以及*Pattern Recognition*杂志的编委。她与Robert M. Haralick一起合写了*Computer and Robot Vision*一书。1995年她当选为IEEE会士，2000年当选为模式识别国际协会的会士。

George C. Stockman 于1977年在马里兰大学获计算机科学博士学位，1982年至今是密歇根州立大学计算机科学与工程专业的教授，讲授编程、数据结构、计算机视觉和计算机图形学课程。Stockman教授参与了IEEE组织的多项活动，包括图像计算教学方面的讨论会。

ISBN 7-111-15972-1

9 787111 159728

华章图书

华章网站 http://www.hzbook.com

网上购书：www.china-pub.com

北京市西城区百万庄南街1号 100037

读者服务热线：(010)68995259，68995264

读者服务信箱：hzedu@hzbook.com

ISBN 7-111-15972-1/TP・4146

定价：55.00 元

计 算 机 科 学 丛 书

计算机视觉

(美) Linda G. Shapiro George C. Stockman 著 赵清杰 钱芳 蔡利栋 译



Computer Vision

机械工业出版社
China Machine Press

本书系统地介绍了计算机视觉方面的基础知识，详细讨论了从图像自动抽取重要信息的相关理论，内容包括最近出现的研究成果。本书取材新颖精练，重点突出，以解决实际问题为目的。前11章讨论的是2D情况；第12章到第15章从2D情况扩展到3D情况；第16章介绍了利用计算机视觉技术的实际应用系统。书中的大量实例及习题，贴近生活，面向应用，富有情趣。

本书适合作为高等院校计算机及相关专业的高年级本科生和研究生的教材，也可供相关技术人员参考。

Simplified Chinese edition copyright © 2005 by Pearson Education Asia Limited and China Machine Press.

Original English language title: *Computer Vision* by Linda G. Shapiro and George C. Stockman , Copyright 2001.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Prentice-Hall.

本书封面贴有Pearson Education（培生教育出版集团）激光防伪标签，无标签者不得销售。

版权所有，侵权必究。

本书法律顾问　北京市展达律师事务所

本书版权登记号：图字：01-2003-1996

图书在版编目（CIP）数据

计算机视觉/（美）夏皮罗（Shapiro, L. G.），（美）斯托克曼（Stockman, G. C.）著；赵清杰等译.-北京：机械工业出版社，2005.3

（计算机科学丛书）

书名原文：Computer Vision

ISBN 7-111-15972-1

Ⅰ.计…　Ⅱ.①夏…　②斯…　③赵…　Ⅲ.计算机视觉-高等学校-教材　Ⅳ.TP302.7

中国版本图书馆CIP数据核字（2005）第000945号

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：傅志红

北京诚信伟业印刷有限公司印刷·新华书店北京发行所发行

2005年3月第1版第1次印刷

787mm × 1092mm 1/16 · 28.5（彩插0.75印张）印张

印数：0 001-4000册

定价：55.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

本社购书热线：（010）68326294

出版者的话

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭橥了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短、从业人员较少的现状下，美国等发达国家在其计算机科学发展的几十年间积淀的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章图文信息有限公司较早意识到“出版要为教育服务”。自1998年开始，华章公司就将工作重点放在了遴选、移译国外优秀教材上。经过几年的不懈努力，我们与Prentice Hall，Addison-Wesley，McGraw-Hill，Morgan Kaufmann等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出Tanenbaum，Stroustrup，Kernighan，Jim Gray等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及庋藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专诚为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍，为进一步推广与发展打下了坚实的基础。

随着学科建设的初步完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都步入一个新的阶段。为此，华章公司将加大引进教材的力度，在“华章教育”的总规划之下出版三个系列的计算机教材：除“计算机科学丛书”之外，对影印版的教材，则单独开辟出“经典原版书库”；同时，引进全美通行的教学辅导书“Schaum's Outlines”系列组成“全美经典学习指导系列”。为了保证这三套丛书的权威性，同时也为了更好地为学校和老师们服务，华章公司聘请了中国科学院、北京大学、清华大学、国防科技大学、复旦大学、上海交通大学、南京大学、浙江大学、中国科技大学、哈尔滨工业大学、西安交通大学、中国人民大学、北京航空航天大学、北京邮电大学、中山大学、解放军理工大学、郑州大学、湖北工学院、中国国家信息安全测评认证中心等国内重点大学和科研机构在计算机的各个领域的著名学者组成“专家指导委员会”，为我们提供选题意见和出版监督。

这三套丛书是响应教育部提出的使用外版教材的号召，为国内高校的计算机及相关专业

的教学度身订造的。其中许多教材均已为M. I. T.，Stanford，U.C. Berkeley，C. M. U. 等世界名牌大学所采用。不仅涵盖了程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等国内大学计算机专业普遍开设的核心课程，而且各具特色——有的出自语言设计者之手、有的历经三十年而不衰、有的已被全世界的几百所高校采用。在这些圆熟通博的名师大作的指引之下，读者必将在计算机科学的宫殿中由登堂而入室。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证，但我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。教材的出版只是我们的后续服务的起点。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方法如下:

电子邮件: hzedu@hzbook.com
联系电话: (010) 68995264
联系地址: 北京市西城区百万庄南街1号
邮政编码: 100037

专家指导委员会

（按姓氏笔画顺序）

尤晋元	王　珊	冯博琴	史忠植	史美林
石教英	吕　建	孙玉芳	吴世忠	吴时霖
张立昂	李伟琴	李师贤	李建中	杨冬青
邵维忠	陆丽娜	陆鑫达	陈向群	周伯生
周立柱	周克定	周傲英	孟小峰	岳丽华
范　明	郑国梁	施伯乐	钟玉琢	唐世渭
袁崇义	高传善	梅　宏	程　旭	程时端
谢希仁	裘宗燕	戴　葵		

秘　书　组

武卫东　　温莉芳　　刘　江　　杨海玲

译 者 序

本书系统阐述了计算机视觉的相关理论和应用技术基础，内容广泛，深入浅出，列举了大量习题和应用实例，不仅适合作为高年级本科生和研究生的教材，也适合作为相关领域研究人员和工程技术人员的参考资料。

本书内容涉及计算机视觉的各个方面，很多内容参考了近期的研究成果，取材新颖精练，重点突出，并以解决实际问题为目的。书中列出了很多算法，都以函数或者过程的形式给出，读者只需用自己熟悉的编程语言稍加修改，即可实现这些算法。书中的大量实例及习题贴近生活，面向应用，富有情趣。另外，在每章后面列出了大量参考文献，这些文献不仅能够帮助读者巩固所学的内容，而且方便读者在自己感兴趣的方向上进行更深入的研究。

本书由赵清杰、钱芳、蔡利栋共同翻译。其中蔡利栋教授翻译了第1、2章的内容，钱芳博士负责第4、5、7、8、10章的翻译工作，赵清杰博士翻译了其余部分并负责全书的统稿工作。参与本书翻译的还有宋霏、王宗远等，在此对他们的工作表示感谢。

由于译者水平有限，加上时间仓促，译稿中难免有错误和遗漏，谨向读者和原作者表示歉意，并欢迎批评指正：zhaoqingjie@tsinghua.org.cn。

赵清杰

2004年9月于北京

前　言

本书系统地介绍了计算机视觉方面的基础知识，内容适合于从事视觉领域研究的广大读者。书中详细讨论了从图像自动抽取重要信息的理论知识，并列举了很多应用实例，为从事这方面学习和研究的学生及科研工作者提供帮助。该书不仅是专业技术人员的一本实用参考资料，更适合作为高年级本科生和研究生的教材。本书主要介绍基本概念与算法，对当前迅速发展的视觉应用领域也进行了论述。本书的独特之处在于，第8章的图像数据库以及第15章的虚拟现实和增强现实，这两部分是迅速发展的最新应用领域。第16章简单介绍了利用计算机视觉技术的实际应用系统。

随着计算机技术的最新发展，计算机图像已经成为一种经济灵活的技术手段，并渗透到各行各业。图像计算不再只属于科学研究领域，也属于艺术领域、社会科学领域，甚至成为人们的业余爱好。这本书适合有专业背景和正在进行专业学习的相关人员，包括对多媒体、艺术设计、地理信息系统和图像数据库感兴趣的读者，以及传统的自动化、图像科学、医学成像、远程感知和计算机绘图等领域的读者。

要使书的内容面面俱到是不可能的。微积分、物理学和常规计算等方面的内容，已有专门的相关教材。我们希望本书不仅可以作为教材，同时又能对一般读者有所帮助。本书所选内容新颖有趣，相信大多数读者都能够看懂。作为研究生或高年级本科生计算机视觉课程的教材使用时，应把参考文献作为课程的补充材料。每章后面都列出了适当数量的参考文献，但并没有包括全部文献。

前面各章首先介绍底层基本知识，并逐步过渡到数学模型部分。目的是为了在涉及图像特征之前，让大家先有一个直观性的了解。标注“*”的部分需要更多的数学知识或者难度更深，在专业性不强的课程中可以不讲这些内容。为了加强直观性理解，在前面的11章里，我们一直在讨论二维（2D）图像，到了后面几章才开始讨论三维（3D）计算机视觉。有经验的教师可以针对不同课程和教学风格，重新安排各章的讲解顺序。2D图像处理有很多用处，许多概念和算法在2D情况下讲解起来更容易理解。第4章介绍模式识别方面的基本知识，使学生在全面掌握图像特征和匹配之前，对完整的识别系统有所了解。学完第4章之后，读者会对2D图像处理应用有更深入的理解。第5、6和7章是有关灰度、颜色和纹理特征的内容。第8章介绍图像数据库方面的知识，这是一个较新的研究内容。一些同仁建议把这部分内容放在书的末尾，我们把它安排得稍微靠前，目的是为了强化前面几章中的有关概念，以及为学期中间的课程作业提供素材。第10和11章讲的是图像分割与匹配，主要针对的是2D情况，不涉及复杂的3D变换，这样可使基本概念描述起来更加简单。

关于3D特征，在第2章做了介绍，在第12章进行了详细讨论。第12章综述了从2D图像恢复3D世界的多方面内容，包括立体视觉的量化模型，由焦距变化恢复深度的薄透镜模型，以及分辨力的概念。第13章介绍3D计算机视觉变换，教学过程中发现这个问题对学生来说难度较大。关于齐次变换的内容安排在这一章内，而没有放到附录中。3D变换是对第11章中2D简单情况的推广。最小二乘拟合也从第11章的2D简单情况推广到第13章的3D情况。本章介绍了

P3P非线性优化方法，并用于进行摄像机标定，包括建立镜头的径向畸变模型。第14章讨论的是3D模型以及模型与3D数据的匹配，这部分难度更大。第15章讨论虚拟现实和增强现实技术，以及计算机视觉在其中扮演的重要角色。

编程语言问题

本书不依赖任何编程语言，而是使用了通用算法符号。用特定语言编写不仅没有必要，而且对许多读者来说也不合适。对会编程的学生来说实现这些算法并不困难，这一点在我们的学生身上已得到证明。在适当和可能的时候，相关例子会公布在WWW上，一方面是为了让学生能够快点儿进行实验，另一方面也使他们能够学习编写代码。

教师和学生可以利用软件工具和程序库，例如Khoros、NIH-Image、XView、gimp和MATLAB等软件工具，也可以从生产视觉硬件设备的公司购买现成的程序包。作者在书中没有用专用软件，因为多数读者使用不同的软件工具，另外工业专用软件具有复杂的数据结构和算法，用这种软件工具进行图像运算达不到预期的学习目的。在简单环境下掌握了算法的基本原理之后，读者在选用专用软件工具时就会得心应手。

如何使用本书

教师和学生可以根据课程的目的和兴趣有选择地学习书中的内容，也可以打乱书中的章节顺序。以下内容仅供参考:

• 第2章简介，第3章作为重点

在数据结构和算法课程中至少需要1~3讲。在第2章的背景知识基础上，第3章内容包括对2D图像阵列、深度优先搜索以及并查数据结构等的应用和编程练习。

• 第1、2、3章和第4、5、6章中的部分内容

大学生做课程设计时，这部分内容可选讲1 ~ 3周。要求他们写出简单的学期报告或者设计一个小项目，项目可以是建立2D零件识别系统，利用连通成分和特征向量原型匹配方面的知识。

• 第1~11章的大部分内容

作为地理学、自然资源或微生物学学生的选修课，可以不讲其中带“*”的选学部分。如果作为本科生的图像处理与分析课程，第1~11章的大部分内容都应该讲到。

• 书中的大部分内容

对于相关专业的高年级本科生或者研究生，要用一学期的时间学习计算机视觉这门课。而本书内容用一个学期是讲不完的，部分内容可以不讲或只做简单介绍，后面的习题也可以只做一部分。如果是半个学期的本科生课程，第1~4、6~12和14章内容应该介绍，这些内容可使学生对计算机视觉有比较深的了解。如果是半个学期的研究生课程，第 1~4章内容可做简单介绍，第6~14章应该重点讲解，第15章选讲主要内容。如果是研究生课程，应该增加参考文献中的有关内容。

感谢与我们有同样兴趣的同事、教师和学生，他们为本书做了大量的贡献，并把自己的研究成果拿出来与大家分享。许多人为这本书的出版提供了无私的帮助，他们不断鼓励我们，贡献自己的想法、图表和算法等，书中都做了引用说明。几位审稿人和同事提供的宝贵意见对本书的改进帮助很大。特别感谢Mohammad Ghavamzadeh、Nick Dutta、Kevin Bowyer、Adam Clark、Yu-Yu Chou、Habib Abi-Rached和Valentin Razmov，他们对书中的文字做了认真修改。我们对书中的任何错误负责，在以后的版本中将进行改正。

这本书撰写了四年。感谢Addison Wesley-Longman 的Paul Becker，他在这个过程中做了许多指导性工作。感谢Prentice Hall的Tom Robbins，因为他使本书得以顺利出版。感谢Cathy Davison 和Lorraine Evans，他们一直在对很多案例进行跟踪研究。感谢ICC的Rose Rummel-Eury 和Chanda Wakefield，他们认真修改了书中的符号和语言，并推进计划的顺利进行。编写这本书任务繁重，好在我们有一个训练有素和幽默的团队。

Linda Shapiro
shapiro@cs.washington.edu
George Stockman
stockman@cse.msu.edu

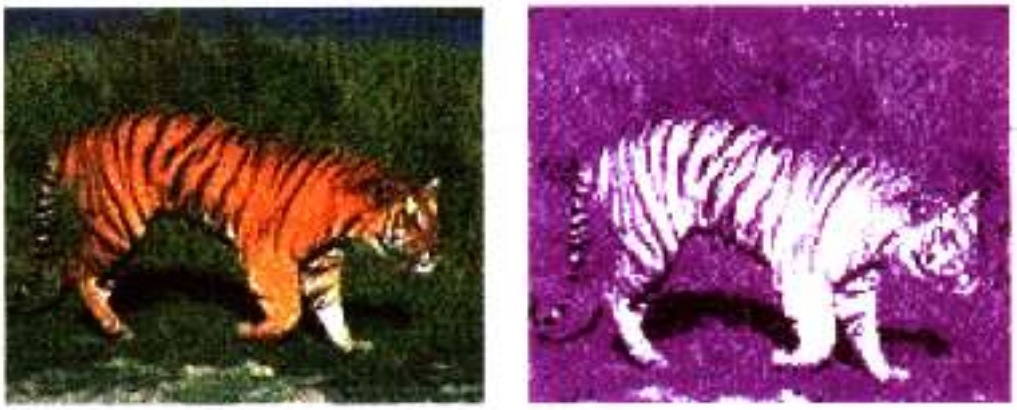


图 6-1

（左图）老虎在草地上的自然色图像。

（右图）由于颜色的改变，对老虎的但引变得不太可靠，这许是只能在地毯上的家猫？（原图经Corel Stock Photos许可）



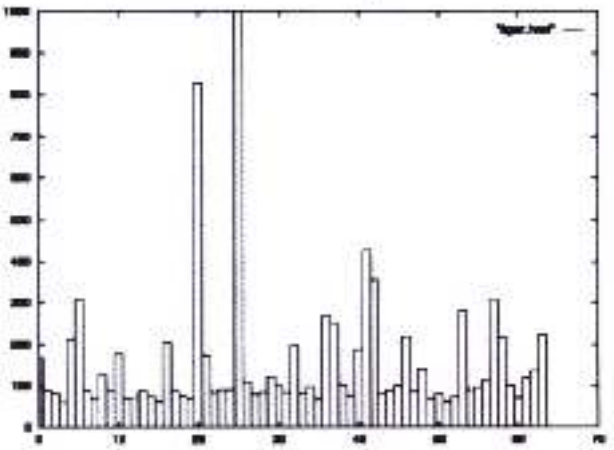
图 6-9

（左图）输入的KGB图像　（中图）饱和度8增加40%　（右图）饱和度5降低20%（Frank Rivera提供）



图6-10　从左边的彩色图像中分割出白色像素，白色像素的单个连通成分用第3章的颜色算法任意标记。（David Moore提供分析）







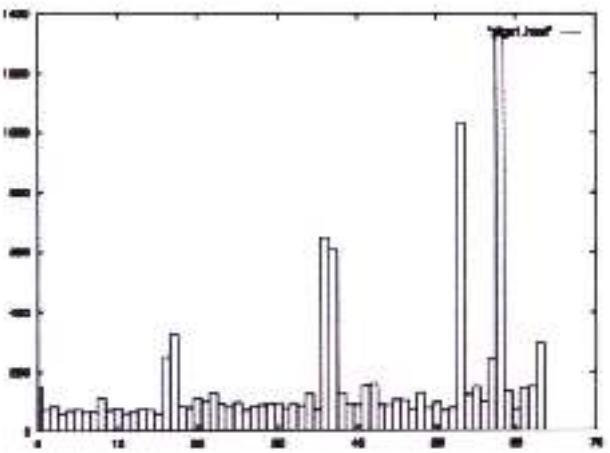


图6-11 彩色图像及其64箱等的直方图（直方图由A. Vailaya提供，图片经Corel Stock Photos许可）

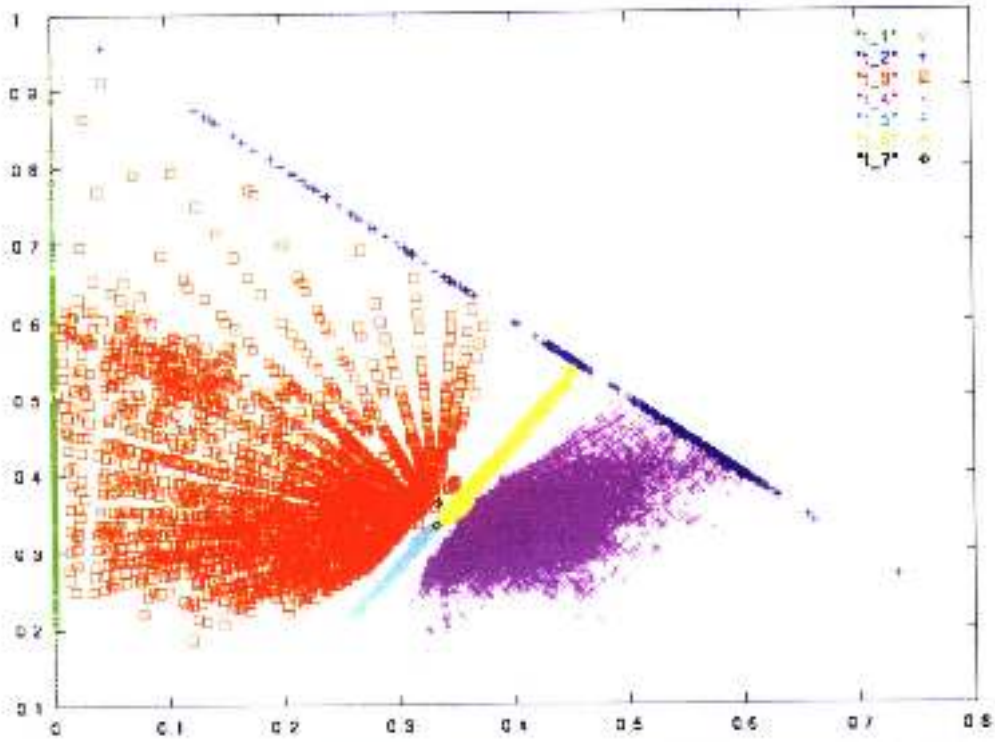


图6-12 通过训练得到的皮肤颜色类别，水平轴是$R_{c_{n,m}}$，垂直轴是$G_{c_{n,m}}$。t_4类是主要的人脸颜色，t_5和t_6是次要的人脸类，它们与人脸上的阴影和高频区域有关（V. Bakic提供）

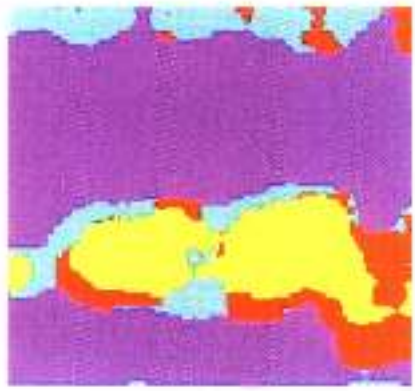
图6-13 人脸抽取实例（匹象由V. Bakic提供）

（左图）输入图像　（中图）标记匹象　（右图）抽取的人脸区域的边界





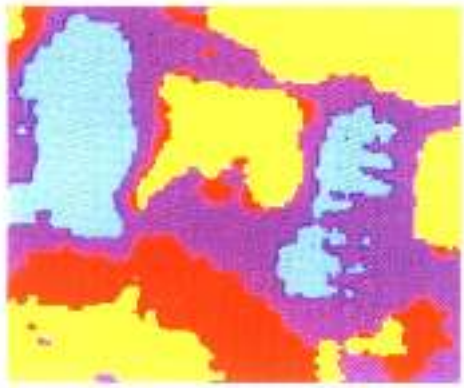
a) 原图



b) 分类成3个类别



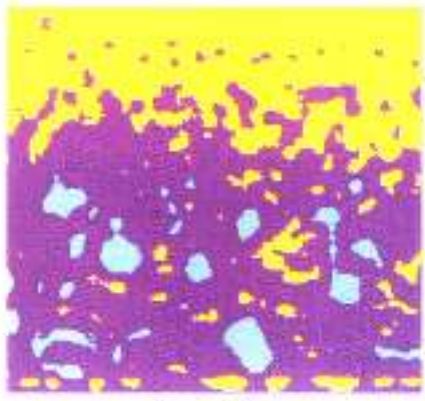
c) 原图



d) 分类成4个类别



e) 原图



f) 分类成5个类别

图7-8 利用Laws纹理能量测度分割图像（原图来自Corel Stock Photos和MIT媒体实验室VisTex数据库）



a）雷诺阿的绘画



b）紫水晶图像

图8-1 数字图像示例（皮埃尔·奥古斯特·雷诺阿的绘画，Beaulieu的风景，1893，经巴尔的摩艺术博物馆许可，Mildred Anna Williams 收藏，1949.9，紫水晶图像经Smithsonian学院许可，1992）

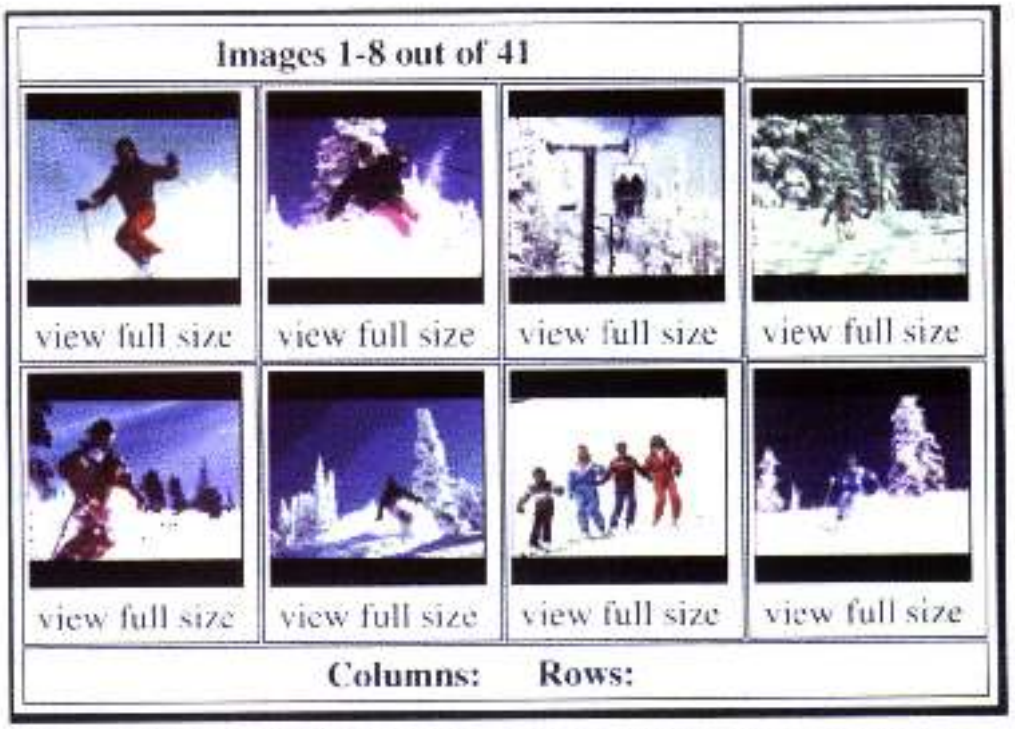


图8-3 基于颜色分布性能生成QBIC检索结果。查询图像是位于左上角的图像（Egames提供）

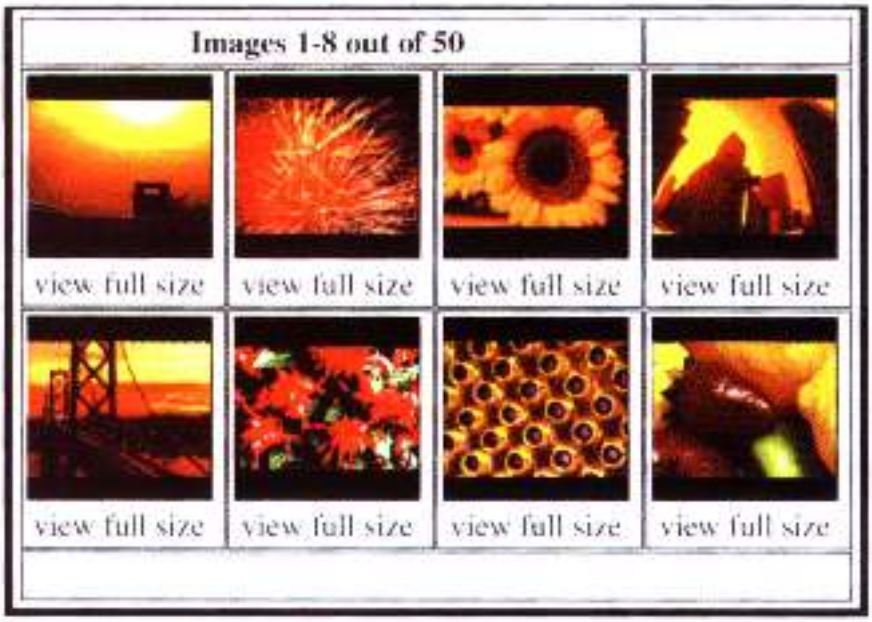


图8-4 基于颜色百分比的QBIC检索结果，查询定义为40%的红色、30%的黄色和10%的黑色（Egames提供）

图8-5 图像查询检索结果，其中查询图像是除色的图像（图像来自MIT媒体实验室的VisTex数据库http://vismod.www.media.mit.edu/vismod/imagery/VisionTexture/vistex.html）

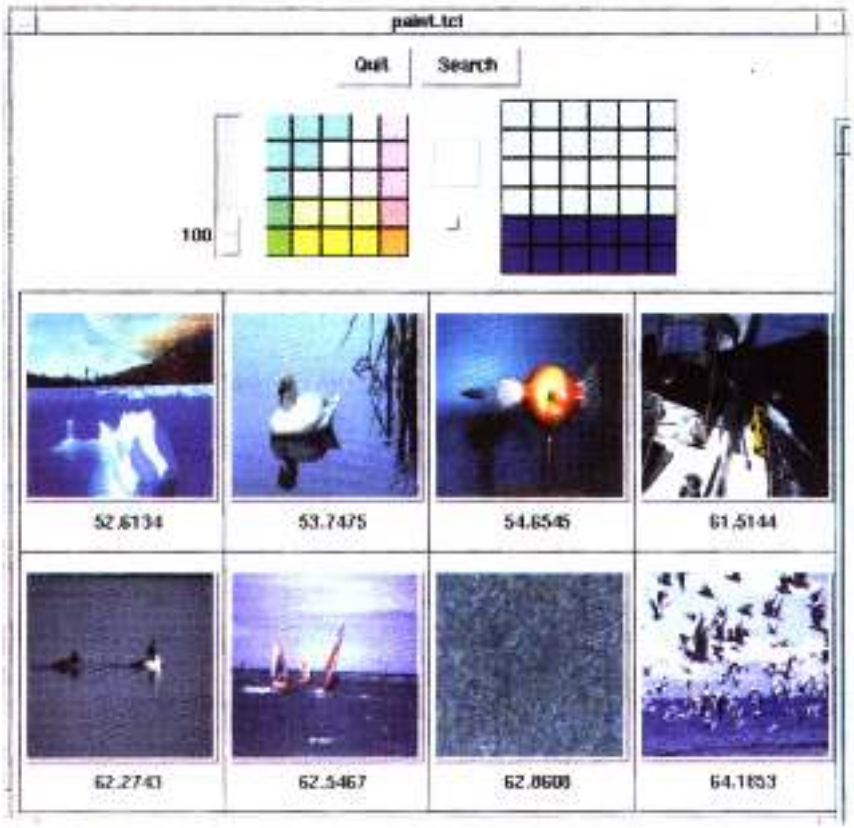


图8-6 基于纹理相似性的图像库检索结果（来自MIT媒体实验室的VisTex数据库：
http://vismod.www.media.mit.edu/vismod/imagery/VisionTexture
/vistex.html）

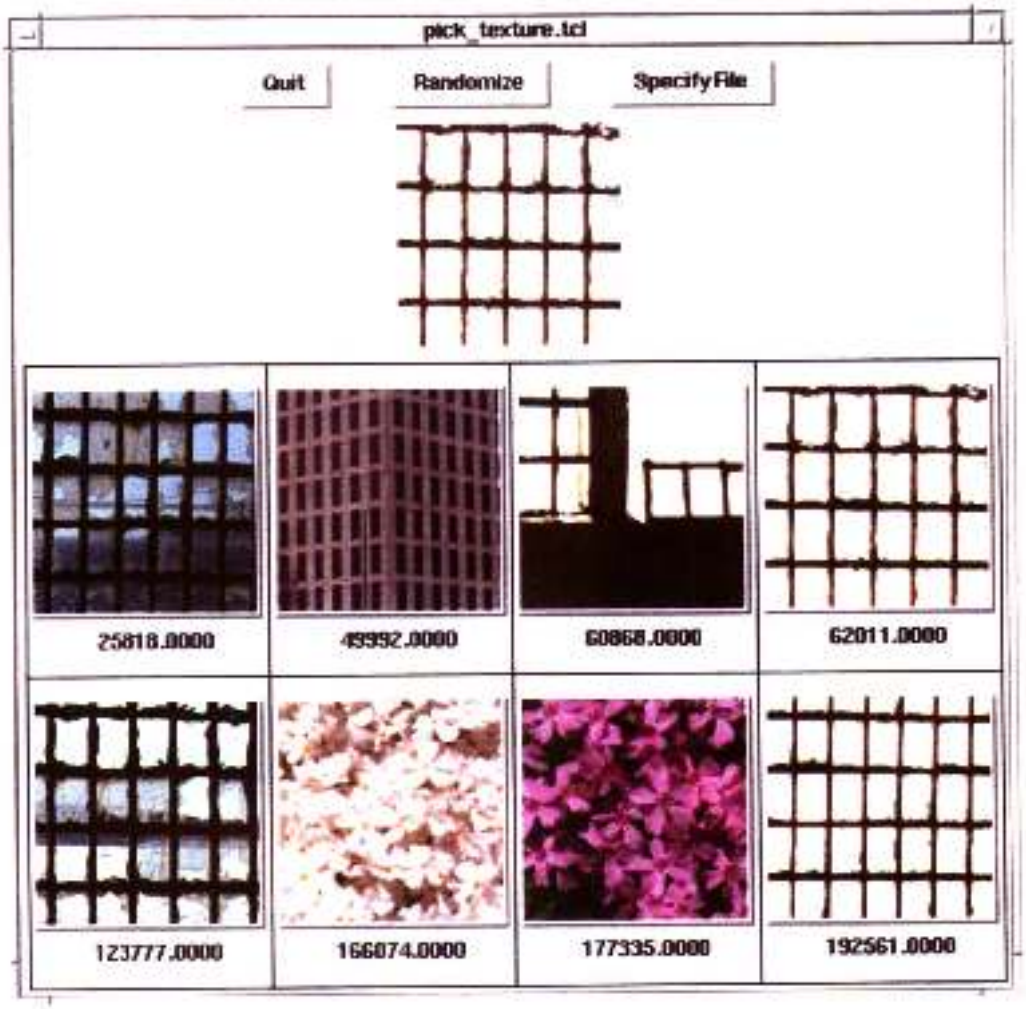
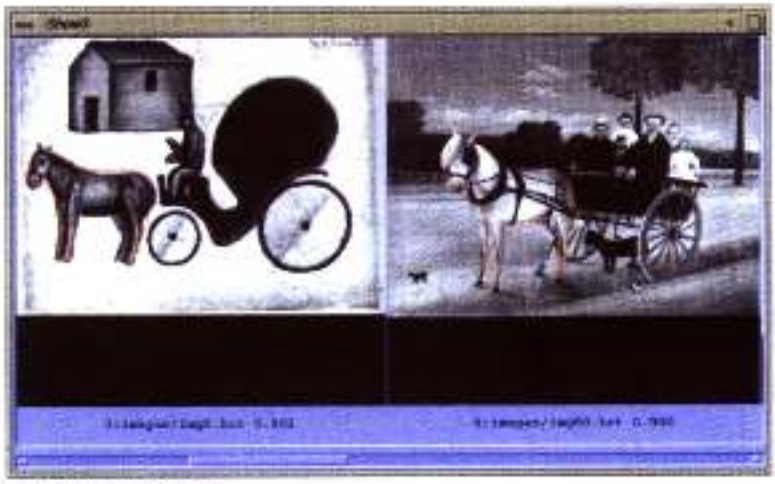


图8-7 弹性人型图像检索结果（Alberto Del Bimbo提供）



a) 用户提供的查询形状



b) 两幅检索出来的图像



c) 另外两幅检索出来的图像，包含两匹马



原图　　分割图　　符号表示

图8-9　从图像中抽取目标和空间关系并用于检索（原图像经Corel Stock Photos许可）

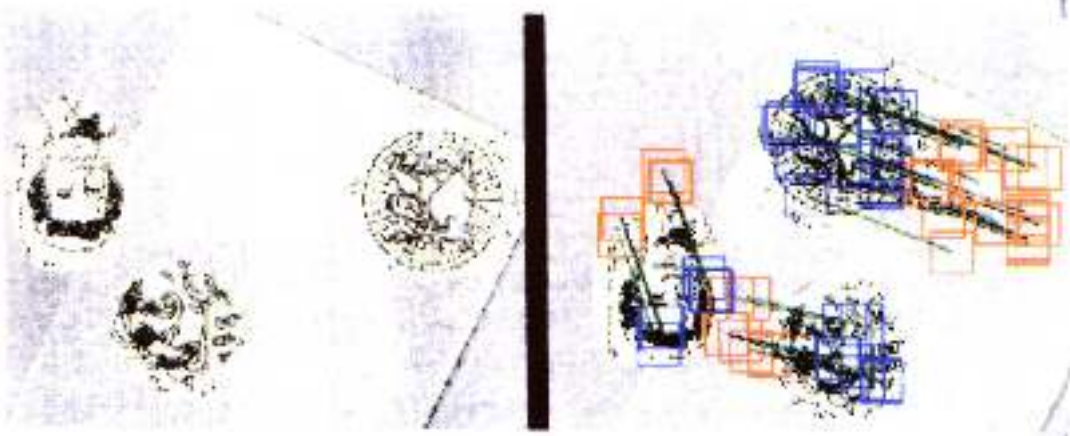


图9-6　算法9.3的应用结果。左边是有时刻的图像，右边是带有运动分析结果的后时刻的图像。红色方框表示原始邻域的位置，是对左图运用兴趣算子检测得出的。蓝色方框表示右图中与左图最佳匹配的邻域。三组绿线是表示运动向量，分别对应三个运动目标。最左边的目标向下偏右一点的方向运动，最下面的目标向右偏下一点的方向运动，最右边的目标向左偏上一点的方向运动（分析由Adam T. Clark提供）



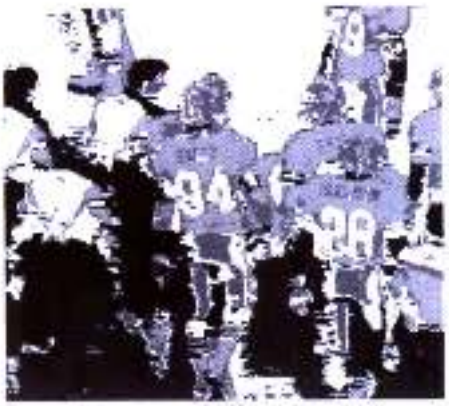


图 10-1

（左）橄榄球图像

（右）分割成区域的图像，每个区域是颜色相似的伪通像素集合





图 10-4

（左）极零点图像

（右）利用K均值聚类，得到K=6种不同灰度点聚类结果。6个聚类对应6种颜色：深绿色、绿色、深蓝色、白色、蓝色和黑色



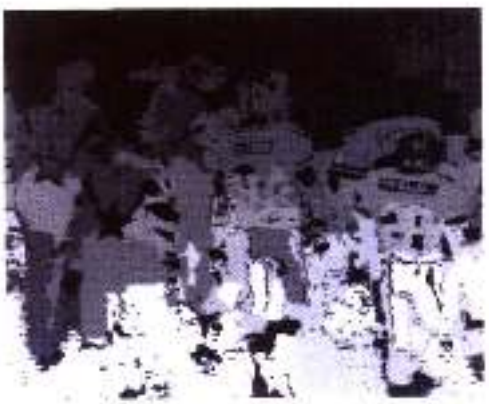


图 10-5

（左）橄榄球图像

（右）利用isodata聚类，得到K=5种不同灰度的聚类结果。5个聚类对应5种颜色：绿色、深绿色、白色、蓝色和黑色

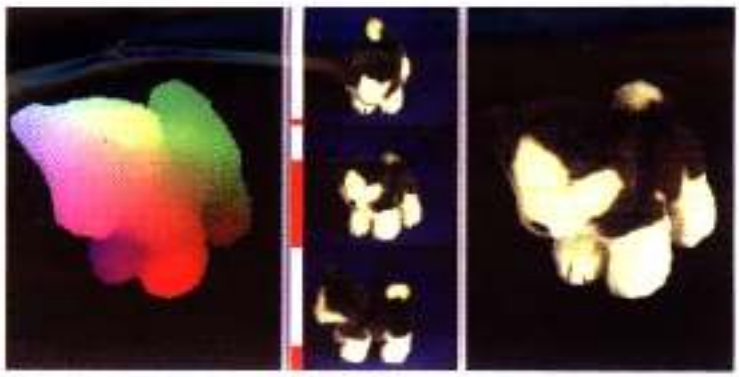


图 15-17

（左）小狗校准的深度图像

（中）对近视点的三幅真彩色视图像

（右）对视图像素进行加权得到的绘制图像（Kari Pulli 提供）

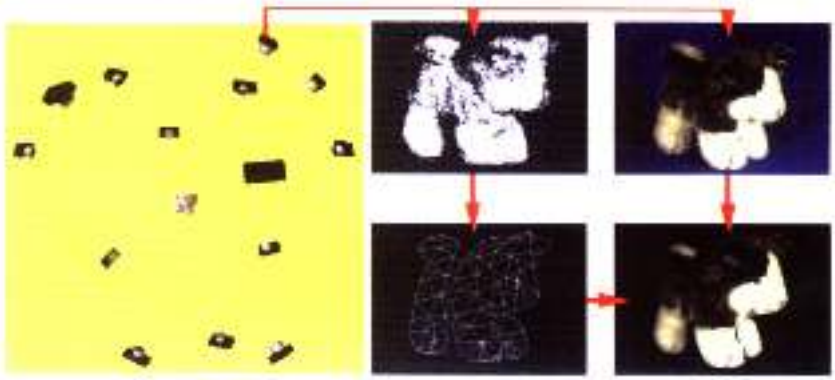


图15-18 由少量目标视图生成的配准深度图像和彩色图像，可用来生成高质量的绘制图像，而不需构造目标的全三维模型（Kari Pulli提供）

（左）可靠的视点

（中上）某视点对应的深度图像

（右上）同一视点对应的彩色图像

（中下）根据深度数据建立的网格模型

（右下）把彩色数据放进映射到网格模型得到的绘制图像

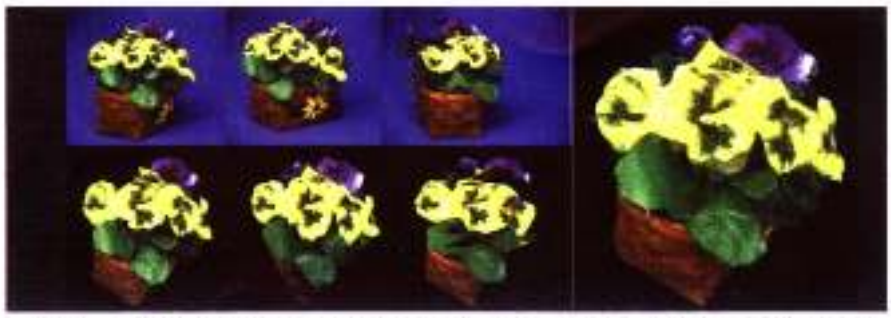


图15-19 由于目标的部件太薄，用同样的技术建立全三维模型几乎是不可能的（Kari Pulli提供）

（左上）三幅不同的目标彩色图像

（左下）把三幅原始图像的像素映射到新的视点，产生不同视点的三幅新图像

（右）最后的绘制图像，三幅新图像的加权结果

目　录

出版者的话
专家指导委员会
译者序
前言

第1章　绪论 ……1
1.1　机器视觉 ……1
1.2　应用问题 ……2
1.2.1　数字图像 ……2
1.2.2　查询图像数据库 ……3
1.2.3　检查交叉支撑杆上的螺孔 ……3
1.2.4　诊断人脑内部 ……5
1.2.5　处理扫描的文本页面 ……6
1.2.6　解释积雪覆盖 ……6
1.2.7　理解零件场景 ……7
1.3　图像运算 ……8
1.3.1　邻域运算 ……8
1.3.2　整幅图像增强 ……9
1.3.3　多幅图像运算 ……9
1.3.4　图像特征计算 ……10
1.3.5　抽取非图像表示 ……10
1.4　面临的问题 ……11
1.5　计算机和应用软件 ……11
1.6　相关领域 ……12
1.7　内容安排 ……12
1.8　参考文献 ……12
1.9　附加习题 ……14
第2章　图像生成与图像表示 ……17
2.1　光线感测 ……17
2.2　成像设备 ……18
2.2.1　CCD摄像机 ……18
2.2.2　图像形成 ……19
2.2.3　视频摄像机 ……20
2.2.4　人眼 ……21
2.3　数字图像中的问题* ……21
2.3.1　几何畸变 ……21
2.3.2　散射 ……21
2.3.3　光晕 ……21
2.3.4　CCD差异 ……22
2.3.5　削波与逆变 ……22
2.3.6　彩色畸变 ……22
2.3.7　量化效应 ……22
2.4　图像函数与数字图像 ……22
2.4.1　图像类型 ……22
2.4.2　图像量化与空间度量 ……24
2.5　数字图像格式* ……27
2.5.1　图像文件头 ……28
2.5.2　图像数据 ……28
2.5.3　数据压缩 ……28
2.5.4　常用图像格式 ……28
2.5.5　游程编码二值图像 ……28
2.5.6　PGM格式 ……28
2.5.7　GIF格式 ……29
2.5.8　TIFF格式 ……30
2.5.9　JPEG格式 ……30
2.5.10　PostScript格式 ……30
2.5.11　MPEG格式 ……30
2.5.12　图像格式比较 ……31
2.6　成像影响因素 ……31
2.7　从二维图像到三维结构 ……32
2.8　5种参考坐标系 ……33
2.8.1　像素坐标系I ……33
2.8.2　物体坐标系O ……34
2.8.3　摄像机坐标系C ……34
2.8.4　实际图像坐标系F ……34
2.8.5　世界坐标系W ……34
2.9　其他类型的传感器* ……34
2.9.1　测微密度计 ……34
2.9.2　彩色图像和多谱图像 ……35
2.9.3　X射线 ……35

2.9.4 磁共振成像 ……36
2.9.5 距离扫描仪和深度图像 ……36
2.10 参考文献 ……38
第3章 二值图像分析 ……39
3.1 像素与邻域 ……39
3.2 图像模板运算 ……40
3.3 目标计数 ……41
3.4 连通成分标记 ……42
3.4.1 递归标记算法 ……43
3.4.2 逐行标记算法 ……45
3.5 二值图像形态学 ……49
3.5.1 结构元 ……49
3.5.2 基本运算 ……49
3.5.3 二值形态学的应用 ……51
3.5.4 条件膨胀 ……54
3.6 区域特征 ……54
3.7 区域邻接图 ……61
3.8 灰度级图像阈值化 ……62
3.8.1 直方图阈值选择 ……62
3.8.2 自动阈值处理：Otsu方法* ……63
3.9 参考文献 ……66
第4章 模式识别 ……69
4.1 模式识别问题 ……69
4.2 分类模型 ……70
4.2.1 类别 ……70
4.2.2 传感器/变换器 ……70
4.2.3 特征抽取算子 ……70
4.2.4 分类器 ……70
4.2.5 分类系统的建立 ……70
4.2.6 系统错误估计 ……71
4.2.7 误报和漏报 ……72
4.3 查准率与查全率 ……72
4.4 特征表示 ……73
4.5 特征向量表示 ……74
4.6 分类器的实现 ……75
4.6.1 最近均值分类 ……75
4.6.2 最近邻分类 ……76
4.7 结构方法 ……77
4.8 混淆矩阵 ……79
4.9 决策树 ……79
4.10 贝叶斯决策 ……83
4.11 多维数据决策 ……87
4.12 机器学习 ……88
4.13 人工神经网络* ……88
4.13.1 感知器模型 ……88
4.13.2 多层前向网络 ……91
4.14 参考文献 ……94
第5章 图像滤波与增强 ……95
5.1 图像处理 ……95
5.1.1 改善图像质量 ……95
5.1.2 检测低层特征 ……95
5.2 灰度级映射 ……96
5.3 去除小图像区域 ……99
5.3.1 去除盐椒噪声 ……99
5.3.2 去除小成分 ……100
5.4 图像平滑 ……100
5.5 中值滤波 ……102
5.6 差分模板边缘检测 ……104
5.6.1 1D信号差分 ……104
5.6.2 2D图像差分算子 ……107
5.7 高斯滤波与LOG边缘检测 ……111
5.7.1 LOG边缘检测 ……112
5.7.2 人类视觉的边缘检测 ……114
5.7.3 马尔－海尔德斯理论 ……115
5.8 Canny边缘检测 ……116
5.9 匹配滤波模板* ……117
5.9.1 向量空间 ……117
5.9.2 利用正交基 ……119
5.9.3 柯西－施瓦茨不等式 ……120
5.9.4 m × n图像的向量空间 ……120
5.9.5 2 × 2邻域的Robert基 ……120
5.9.6 3 × 3邻域的Frei-Chen基 ……121
5.10 卷积和交叉相关* ……124
5.10.1 模板运算定义 ……124
5.10.2 卷积运算 ……125
5.10.3 并行计算 ……128
5.11 正弦波空间频率分析* ……128
5.11.1 傅里叶基 ……128
5.11.2 2D图像函数 ……131
5.11.3 离散傅里叶变换 ……132

5.11.4 带通滤波器 ……134
5.11.5 傅里叶变换讨论 ……135
5.11.6 卷积定理* ……135
5.12 总结和讨论 ……136
5.13 参考文献 ……137
第6章 颜色与明暗分析 ……139
6.1 颜色物理学 ……139
6.1.1 感测被照射物体 ……140
6.1.2 其他因素 ……141
6.1.3 感受器的敏感性 ……141
6.2 RGB三基色 ……142
6.3 其他基色系统 ……143
6.3.1 CMY减色系统 ……143
6.3.2 HSI系统 ……144
6.3.3 电视信号的YIQ与YUV系统 ……146
6.3.4 基于颜色的分类 ……147
6.4 颜色直方图 ……147
6.5 颜色分割 ……149
6.6 明暗分析 ……150
6.6.1 来自单一光源的照射 ……151
6.6.2 漫反射 ……151
6.6.3 镜面反射 ……152
6.6.4 随距离增大而变暗 ……153
6.6.5 复杂因素 ……154
6.6.6 Phong明暗模型* ……154
6.6.7 基于明暗的人类感知 ……155
6.7 相关话题* ……155
6.7.1 颜色应用 ……155
6.7.2 人类的色感机制 ……155
6.7.3 多谱图像 ……156
6.7.4 主题图像 ……156
6.8 参考文献 ……156
第7章 纹理分析 ……159
7.1 纹理、纹理素和统计 ……159
7.2 基于纹理素的描述 ……160
7.3 定量纹理测度 ……161
7.3.1 边缘密度和方向 ……161
7.3.2 局部二值分解 ……162
7.3.3 共生矩阵和特征 ……162
7.3.4 Laws纹理能量测度 ……164
7.3.5 自相关和功率谱 ……166
7.4 纹理分割 ……166
7.5 参考文献 ……167
第8章 基于内容的图像检索 ……169
8.1 图像数据库实例 ……169
8.2 图像数据库查询 ……170
8.3 示例查询 ……171
8.4 图像距离度量 ……171
8.4.1 颜色相似性度量 ……172
8.4.2 纹理相似性度量 ……174
8.4.3 形状相似性度量 ……175
8.4.4 目标检测及空间关系度量 ……179
8.5 数据库组织 ……182
8.5.1 标准索引 ……182
8.5.2 空间索引 ……184
8.5.3 基于内容的多距离测度图像索引 ……184
8.6 参考文献 ……185
第9章 二维运动分析 ……187
9.1 运动现象及应用 ……187
9.2 图像相减 ……188
9.3 计算运动向量 ……189
9.3.1 Decathlete游戏 ……190
9.3.2 点对应 ……191
9.3.3 MPEG视频压缩 ……194
9.3.4 图像流计算* ……195
9.3.5 图像流方程* ……195
9.3.6 利用传播约束求解图像流* ……197
9.4 计算运动点路径 ……197
9.5 检测视频中的显著变化 ……202
9.5.1 视频序列分割 ……203
9.5.2 忽略摄影特效 ……205
9.5.3 存储视频子序列 ……205
9.6 参考文献 ……205
第10章 图像分割 ……207
10.1 区域分割 ……207
10.1.1 聚类方法 ……208
10.1.2 区域增长 ……214
10.2 区域表示 ……215
10.2.1 覆盖图 ……215
10.2.2 标记图像 ……216

10.2.3 边界编码 ……216
10.2.4 四叉树 ……217
10.2.5 特征表 ……217
10.3 轮廓分割 ……218
10.3.1 区域边界跟踪 ……218
10.3.2 Canny边缘检测和连接 ……220
10.3.3 相邻连贯的边缘生成曲线 ……223
10.3.4 用霍夫变换检测直线和圆弧 ……225
10.4 线段拟合模型 ……231
10.5 识别更高层结构 ……235
10.5.1 条带检测 ……235
10.5.2 角点检测 ……236
10.6 运动一致性分割 ……237
10.6.1 时空边界 ……237
10.6.2 运动轨迹聚类 ……237
10.7 参考文献 ……239
第11章 2D匹配 ……241
11.1 2D数据配准 ……241
11.2 点的表示 ……242
11.2.1 参考坐标系 ……242
11.2.2 齐次坐标 ……243
11.3 仿射映射函数 ……243
11.3.1 缩放 ……243
11.3.2 旋转 ……244
11.3.3 正交和标准正交变换* ……245
11.3.4 平移 ……245
11.3.5 旋转、缩放和平移 ……245
11.3.6 仿射变形实例 ……246
11.3.7 目标识别与定位实例 ……247
11.3.8 一般仿射变换* ……249
11.4 最佳2D仿射变换* ……250
11.5 仿射映射法2D目标识别 ……251
11.5.1 局部特征焦点法 ……252
11.5.2 位姿聚类 ……254
11.5.3 几何散列 ……256
11.6 相关匹配法2D目标识别 ……259
11.6.1 解释树 ……260
11.6.2 离散松弛 ……262
11.6.3 连续松弛* ……264
11.6.4 相关距离匹配 ……266
11.6.5 相关索引 ……269
11.7 非线性变形 ……269
11.7.1 径向畸变矫正 ……271
11.7.2 多项式映射 ……272
11.8 总结 ……272
11.9 参考文献 ……272
第12章 2D图像中的3D信息 ……275
12.1 本征图像 ……275
12.2 线条图标记 ……278
12.3 2D图像中的3D线索 ……283
12.4 其他3D现象 ……286
12.4.1 从X恢复形状 ……286
12.4.2 消隐点 ……289
12.4.3 根据焦距变化求深度 ……289
12.4.4 运动现象 ……290
12.4.5 边界和虚拟线 ……290
12.4.6 非偶然对齐 ……290
12.5 透视成像模型 ……291
12.6 通过立体视觉求深度 ……293
12.7 薄透镜方程* ……297
12.8 总结性讨论 ……300
12.9 参考文献 ……300
第13章 3D感知与目标位姿计算 ……303
13.1 一般体视结构 ……303
13.2 3D仿射变换 ……304
13.2.1 坐标系 ……305
13.2.2 平移 ……306
13.2.3 缩放 ……306
13.2.4 旋转 ……306
13.2.5 任意旋转 ……308
13.2.6 基于变换的比对 ……309
13.3 摄像机模型 ……311
13.3.1 透视变换矩阵 ……311
13.3.2 正投影与弱透视投影 ……314
13.3.3 基于多摄像机的3D点计算 ……315
13.4 最佳仿射标定矩阵 ……317
13.4.1 标定物 ……317
13.4.2 最小二乘问题 ……317
13.4.3 仿射方法讨论 ……321
13.5 使用结构光 ……322

13.6 简单的位姿估计过程 ……323
13.7 改进的摄像机标定法* ……327
13.7.1 摄像机内部参数 ……327
13.7.2 摄像机外部参数 ……328
13.7.3 标定举例 ……331
13.8 位姿估计* ……334
13.8.1 2D-3D点对应求位姿 ……334
13.8.2 约束线性最优化 ……335
13.8.3 计算变换Tr＝{R,T} ……336
13.8.4 位姿验证和位姿最优化 ……337
13.9 3D目标重建 ……338
13.9.1 数据获取 ……338
13.9.2 视图配准 ……340
13.9.3 表面重建 ……341
13.9.4 空间切割算法 ……341
13.10 从明暗恢复形状 ……343
13.10.1 光度立体 ……345
13.10.2 结合空间约束 ……346
13.11 从运动恢复结构 ……346
13.12 参考文献 ……348
第14章 3D模型和匹配 ……351
14.1 模型表示 ……351
14.1.1 3D网格模型 ……351
14.1.2 表面－边－顶点模型 ……352
14.1.3 广义圆柱体模型 ……353
14.1.4 八叉树 ……354
14.1.5 超二次曲面模型 ……355
14.2 实际3D模型与视类模型 ……356
14.3 物理学模型和可变形模型 ……357
14.3.1 蛇形活动轮廓模型 ……357
14.3.2 3D气球模型 ……360
14.3.3 建立心脏跳动模型 ……361
14.4 3D目标识别范例 ……361
14.4.1 几何模型比对匹配 ……362
14.4.2 关系模型匹配 ……367
14.4.3 功能模型匹配 ……372
14.4.4 基于外观的识别 ……374
14.5 参考文献 ……379
第15章 虚拟现实 ……383
15.1 虚拟现实系统的特征 ……383
15.2 虚拟现实的应用 ……384
15.2.1 建筑漫游 ……384
15.2.2 飞行仿真 ……384
15.2.3 解剖组织的交互式分割 ……384
15.3 增强现实 ……385
15.4 遥操作 ……386
15.5 虚拟现实设备 ……388
15.5.1 头戴式显示器 ……389
15.5.2 虚拟灵巧手术 ……390
15.5.3 立体显示设备 ……390
15.6 虚拟现实感知设备 ……391
15.6.1 视觉 ……391
15.6.2 听觉 ……391
15.6.3 位姿 ……391
15.6.4 触觉 ……391
15.6.5 运动觉 ……391
15.7 简单3D模型绘制 ……392
15.8 实际图像和合成图像融合 ……393
15.9 人机交互与心理问题 ……395
15.10 参考文献 ……395
第16章 案例研究 ……397
16.1 Veggie Vision系统 ……397
16.1.1 应用场合和要求 ……397
16.1.2 系统设计 ……398
16.1.3 识别过程 ……398
16.1.4 详细分析 ……399
16.1.5 性能分析 ……400
16.2 基于虹膜的身份识别 ……401
16.2.1 对识别系统的要求 ……401
16.2.2 系统设计 ……402
16.2.3 系统性能 ……404
16.3 参考文献 ……405
索引 ……407

第1章 绪 论

计算机视觉的研究内容非常广泛，本书对计算机视觉所涉及的方方面面都做了介绍。毫无疑问，人类能够制造出具有视觉功能的机器。例如，进行瑕疵检测的机器，每天要检测上百万根电灯灯丝和很多织物；自动柜员机（ATM）已经能够通过对人眼的扫描实现身份识别；利用摄像信息，计算机能够驾驶汽车。这一章介绍利用计算机视觉提供解决方案的几个重要的问题领域。读过本章之后，大家对一些应用问题和计算机视觉的方法就会有比较全面的了解[⊖]。

定义1 计算机视觉的研究目标是，根据感测到的图像对实际物体和场景做出有意义的判定。

为了对实际物体做出判定，总是需要根据图像来构造它的某个描述或模型。因此专家们会说*计算机视觉的目标是根据图像来构造出对场景的描述*。尽管我们研究的内容是面向实际问题的，但也要讨论原理性问题。本章提到的并且将在后面章节中进行讨论的问题包括：

感测：传感器是如何获得外部世界图像的？图像是如何对外部特征（如材料、形状、照明以及空间关系）进行编码的？ 1

信息编码：为了理解三维世界，如何由图像得出相关信息，包括物体的几何特征、纹理特征、运动特征和身份特征？

表示：在计算机中如何表示物体的部件、属性和关系？

算法：用什么方法进行图像信息处理，以及建立对世界和其中目标的描述？

这些问题以及其他一些问题都将在后续章节中进行研究。下面介绍几种应用，以及据此提出的一些重要的问题。

1.1 机器视觉

科学家与科幻作家一直梦想着人类能够制造出智能机器，而这种智能机器首先要能够对可视世界进行理解。人脑中有很多组织参与视觉信息处理。人类能够轻而易举地处理许多视觉问题，可是视觉认知作为一个过程，大部分人却知之甚少。Alan Turing，现代数字计算机与人工智能两个领域的奠基人之一，相信数字计算机可具备理解场景的智慧和能力。这样的远 2
大目标已经证明难以实现，人类的工程技术还不能与我们丰富的想象力相匹配。但在某些研究领域已经产生了令人惊奇的进展。虽然本书的主题是建造实用系统而非人工智能，但我们将不时思考更深层次的问题，只要有可能我们就向大家介绍最新的研究进展。举例来说，考虑下面可能在随后几年之内实现的情景：你家门口的摄像机摄取图像并输入到你家的计算机中去，某些人物对你来说很重要，你用这些人脸对计算机进行了识别训练。你往家庭信息中心挂电话时，计算机不仅向你报告所记录的电话信息，而且报告说你妹妹Elenor和报童Chad可

⊖ 在本书中，我们认为“机器视觉”和“计算机视觉”这两个术语是一样的。不过对于工业应用，我们常用“机器视觉”，而一般情况下常用“计算机视觉”。

能到家拜访你。在本书中的多个地方，我们都会讨论到类似这样的具有前沿性的研究思想。

1.2 应用问题

计算机在图像分析中的应用实际上是无止境的。这里只包括几个方面的应用，但对于我们的研究动机和研究方向的确定将起到很好的作用。

1.2.1 数字图像

一幅数字图像可以表示一帧动画，一页文本，一张人脸，一幅加德满都市的地图或者导购清单中的一件物品。数字图像包含固定的像素(pixel)行数与列数，像素是图像元素(picture element)的缩写。像素就像小方块，其数值范围通常在0到255之间，像素值表示图像上各点的亮度。0表示最暗，255表示最亮，或者反过来255表示最暗，0表示最亮，这与编码方案有关。图1-1的左上图是一张人脸数字图像，高257行、宽172列。中上图是一幅8×8的子图像，取自左上图中的右眼部位。下部的64个数，表示子图像中各像素的亮度，子图右上角的像素值低于100，表示眼中的黑色瞳孔区域，而子图中较高的像素值表示眼白部分。彩色图像的每个像素会有3个数值，分别表示红、蓝、绿。数字图像通常用显示器显示，一般是带数字图像存储器的电视屏幕，一幅500×500的彩色图像大致相当于某一时刻电视显示的画面。激发发光材料的一个小点就显示一个像素，彩色显示则需要激发不同材料的3个邻点。高分辨率的计算机显示器大致有1200×1000个像素。对数字图像更详细的讨论在第2章进行，而数字图像的编码和颜色解释将在第6章讨论。



	0	1	2	3	4	5	6	7
0	130	146	133	95	71	71	62	78
1	130	146	133	92	62	71	62	71
2	139	146	146	120	62	55	55	55
3	139	139	139	146	117	112	117	110
4	139	139	139	139	139	139	139	139
5	146	142	139	139	139	143	125	139
6	156	159	159	159	159	146	159	159
7	168	159	156	159	159	159	139	159

图1-1 左上图是人脸图像，中上图是右眼区域的8×8像素子图像，右上图是计算机程序检测到的眼睛位置，下面是8×8子图像的亮度值（图像由Vera Bakic提供）

1.2.2 查询图像数据库

海量数字存储、高带宽传送和多媒体个人计算机促进了图像数据库的发展。有效使用现有的众多图像需要采用合适的检索方法。标准数据库技术适用于加注文本关键字的图像，而基于内容（content-based）的检索方法是当前研究的一个热点问题。假定一个新公司要设计一个新徽标并进行保护，艺术家已经设计出几种方案供公司选择。如果徽标与某个现有公司的徽标太相似是不能用的，所以要对现有的徽标数据库进行检索。这个过程类似于专利检索，是由人工完成的。这时机器视觉方法就可以派上大用场（参见图1-2）。有许多与此类似的问题。假设建筑师或艺术史学家寻找具有特殊入口的建筑物，希望只提供一张图片，也许就是取自数据库的图片，要求系统能够输出其他相似的图片。在第8章中，你将看到如何用几何、颜色和纹理特征进行图像数据库查询。假如广告代理商想搜索幼儿享受美味的图片，理解其中的语义对人类而言非常简单，但对机器视觉来说也许是个难度很高的问题。表征“幼儿”、“享受”、“美味”需要综合应用颜色、纹理和几何特征。顺便说一句，现在已经有人设计出判断某幅彩图中是否含有裸体人物的算法。对于那些想对子女从网上下载的图片进行审查的父母来说，这个计算机算法是有用处的。图像数据库检索方法在第8章进行讨论。



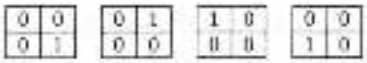
图1-2 示例查询。左图是查询图像，右图是从图像数据库系统中检索出的最相似的两幅图像（东京美术社提供）

1.2.3 检查交叉支撑杆上的螺孔

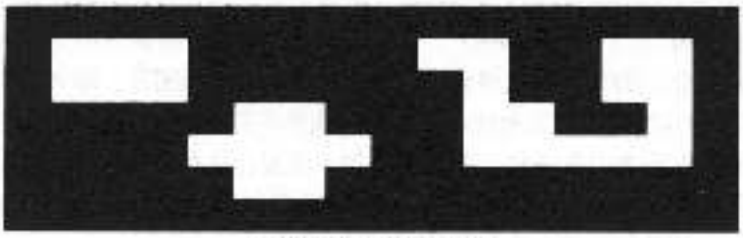
70年代后期，密尔沃基的一位工程师为卡车公司设计了一套机器视觉系统，成功地计算出卡车交叉支撑杆上的螺孔数目。卡车公司要求所有的交叉支撑杆在装运前必须经过检查，因为在未装配完的卡车上如果少一个螺孔，将造成代价不菲的损失：要么迫使装配线停止，重新钻孔；要么出现更糟糕的情况，即工人为了使生产线正常运转可能不安装必要的螺栓。为获得卡车交叉支撑杆的数字图像，把光源放在传送带下方，数字摄像机则安装于上方。当交叉支撑杆通过摄像机视场时，摄像机拍摄它的图像。在图像上，对应交叉支撑杆钢铁部分的像素是深色，像素值为1；对应孔区的像素是亮色，像素值为0，表示螺孔已钻。孔数可以通过外角（external corner）数减去内角（internal corner）数然后除以4计算。图1-3中，有三个像素值为0的亮孔，背景的像素值为1。外角由2×2的相邻像素形成，包含三个1值像素；内角也由2×2的相邻像素形成，包含一个0值像素。图1-3中显示的是对7行16列图像的处理情况，并给出了算法的框架。孔计数只是数字图像处理中简单但实用的例子之一。（如下面的习题1.1所示，仅当孔是4-连通（4-connected）而且是简单连通（simply connected）时，也就是孔内没有背景像素时，孔计数算法才是正确的。第3章进一步讨论了这些概念，更详细的讨论请参考Rosenfeld所编的教材。



a) 2×2外角模式



b) 2×2内角模式



c) 暗背景下的三个总孔区

	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	e	i
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
1	1	0	0	0	1	1	1	1	1	0	0	1	1	0	0	1		
2	1	0	0	0	1	1	1	1	1	1	0	1	1	0	0	1		
3	1	1	1	1	1	0	0	1	1	1	0	0	1	1	0	1		
4	1	1	1	1	0	0	0	0	1	1	0	0	0	0	0	1		
5	1	1	1	1	1	0	0	1	1	1	1	1	1	1	1	1		
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		

d) 7×16的二值输入图像

	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	e	i
0	e			e					e		e		e		e		6	0
1									e	i							1	1
2	e			e	e		e				i	e	e	i			6	2
3				e	i		i	e				i		i			2	4
4				e	i		i	e		e					e		4	2
5					e		e										2	0
6																	0	0

e) "e" 表示外角模式，"i" 表示内角模式

图1-3 对二值图像进行孔计数。外角模式e的个数21个减去内角模式i的个数9，然后除以4得到孔的个数为3。为什么？

5

算法1.1 二值图像孔计数的算法框架

M是**R**行**C**列的二值图像。

像素值"1"对应物体的材料区域，光线不能通过；

像素值"0"对应缺材料的孔区，光线能够穿过。

```
每个0值区域必须是4-连通的，而且图像的边界像素值必须为1。
E是外角（三个1和一个0）的个数;
I是内角（三个0和一个1）的个数。

    integer procedure Count_Holes (M)
    {
      examine entire image, 2 rows at a time;
      count external corners E;
      count internal corners I;
      return (number_of_holes = (E−I)/4);
    }
```

习题1.1 孔计数

考虑下列三幅图，大小分别为4 × 5，4 × 4和4 × 5。

1	1	1	1	1
1	0	1	0	1
1	0	1	0	1
1	1	1	1	1

1	1	1	1
1	1	0	1
1	0	1	1
1	1	1	1

1	1	1	1	1
1	0	1	0	1
1	0	0	0	1
1	1	1	1	1

在扫描角模式时，对上述三图采用算法1.1进行实验，它们分别有12、9和12个2 × 2邻域。e、i、n分别表示外角、内角和非内外角，每个2 × 2邻域与e、i、n中的一个匹配。(a) 对于三幅图中的每一幅，e、i、n模式各有多少个？(b) 孔计数公式是否对这三幅图都适用？

1.2.4 诊断人脑内部

磁共振成像（MRI）设备能感测到三维目标内部的组织。图1-4是人头的剖面图，亮区与头部组织的运动有关。这实际上是一张关于头部血液流动的图片。人们可以看见重要的血管，其中的星星状结构表示人眼区域。医生通过MRI图像检查肿瘤或血流问题，例如反常的血管收缩和扩张。图1-4中的右图是对左图进行二值处理的结果，大于等于208的像素值设为亮(255)，低于208的像素值设为暗（0）。相对于背景，大多数亮区像素正确地突显出血管，但是无论是亮区还是暗区，其中都有不少着色不正确的像素。医学图像分析常常要用到机器视觉技术，尽管常常是为了辅助数据表示和度量而非诊断本身。如果我们能够看到思想突然浮现在大脑中那岂不很妙？哦，原来MRI能感测与思考过程有关的器官活动。目前这是一个非常令人振奋的研究领域。

6

习题1.2 每个孔有多少个像素？

进一步考虑计算卡车交叉支撑杆上孔数的应用实例。假设交叉支撑杆尺寸是50英寸长、10英寸宽，成像后形成大约100行500列像素的数字图像。如果交叉支撑杆上有一个特殊螺孔，直径是1/2英寸，你认为图像上孔的半径和面积是多少？以像素为单位。

习题1.3 硬币成像

该问题与上一个问题有关。取一些坐标纸（最好是0.25平方英寸）和一个0.25美元的硬币。把硬币随机地放在坐标纸上，勾出它的圆周线，这样共做5次。对每个位置，以像素为单位估

计硬币图像的面积。(a) 判断某个像素是否属于硬币图像（不计小数），(b) 对每个被圆周穿过的像素，估计有多大面积属于0.25美元硬币，精确到0.1个像素。做完这些估算后，分别计
7 算硬币图像面积的均值和标准差。



图1-4 左图是核磁共振图像，其中亮区与血流有关。右图是对左图进行二值处理后的图像，所有大于或等于208的像素值设为255，低于208者设为0（图像由密歇根州放射科的James Siebert提供）

1.2.5 处理扫描的文本页面

把纸质文档转化为适合于信息系统的数字形式是一个常见的问题。例如，把一本旧书到登在因特网上，或者将盖图转化为几何文件以便于用数控机床制造零件。

图1-5中的中文和英文表达同样的意思。中文字写在纸上并被扫描成482 × 405的图像。对图形编码并表示为postscript格式的数字文件，大小为68 464字节。英文文本则存储在一个115字节的文件中，每字节存放一个ASCII字符。这在文档处理方面应用广泛。从扫描或传真文件的点阵中识别字符，就属于这样的应用。如今这项技术已经非常成熟，但要求字符与标准字体一致。对信息进行语义解释是更难的问题，它可用于大型数据库检索之中。



图1-5 左图是中文字符，右图是英文对照。机器有可能自动地对它们进行互译吗？（英文诗作者为George Stockman，由John Weng翻译成中文）

1.2.6 解释积雪覆盖

卫星有规律地扫过地球表面的大部分，并把数字图像传送到地面。对这些图像进行处理，抽取出各种各样的信息，例如河流分水岭上的积雪量，对于调节大坝控制洪水、水供应或者野生动物居住是非常重要的。通过统计图像中代表雪的像素个数，可以估计出积雪量。卫星

图像中的一个像素可能与地面10m × 10m的区域范围对应，但是据报导一些卫星能看到更小的范围。必须经常把卫星图像和地图或其他图像进行比较，以确定哪些像素位于特定区域或分水岭上。这种工作常常是用户与图像处理软件以交互的方式完成的。关于这方面的内容将在第11章进行更多的讨论。图像匹配也在第11章讲解。图1-6是在一次太空飞行时拍摄的照片，此次飞行由位于德克萨斯州休斯顿的约翰逊太空中心控制。照片显示出华盛顿州的Wenatchee镇，Wenatchee河在那儿汇入哥伦比亚河。 8

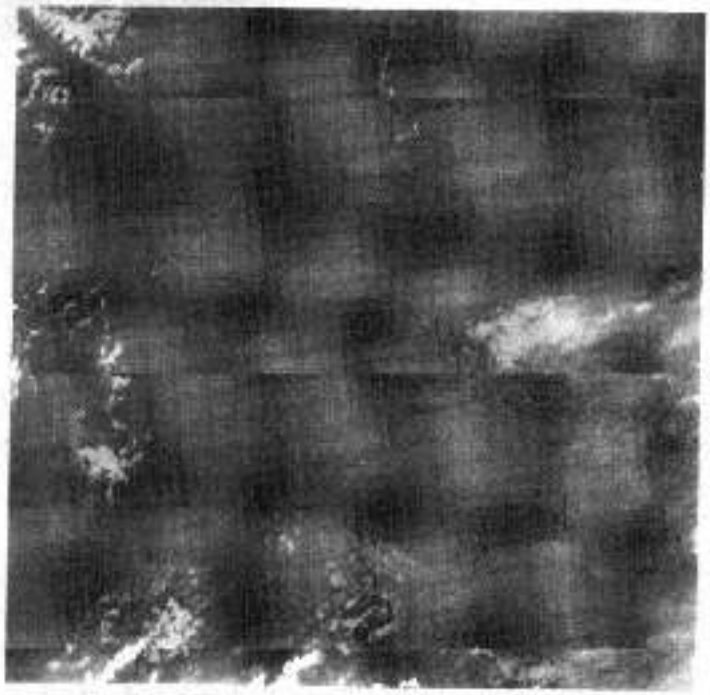


图1-6 华盛顿州的Wenatchee河与哥伦比亚河（约翰逊太空中心提供）

众所周知，计算机能够处理大量的数据。卫星扫描地球产生大量数据，这些数据在许多方面都要用到，例如关于某地区水文地理的计算机模拟程序，就需要输入积雪像素的个数和位置。（该地区的温度信息也要输入到程序中去。）另一个应用是调查农作物的种植情况并对收成进行预算。再有一个应用则是为了税收目的对建筑物进行清点，这常常是利用在飞机上拍出的图片由人工完成的。

1.2.7 理解零件场景

在制造过程的许多环节中，通过传送带或箱子搬运零件。零件必须分别地用机器放置，包装或检查等。如果操作枯燥或者危险，就可以借助视觉引导机器人。图1-7显示的是机器人工作区中的三个零件。机器人视觉系统通过识别边缘和孔从而识别出零件，并确定零件在工作区中的位置。对于每个推测出的零件及其位置，借助计算机辅助设计（CAD）制作三维模型。视觉系统随后对感测到的图像数据与按照模型及其空间位置生成的计算机图形进行比较。忽略不好的匹配，而用好的匹配对推测结果进一步修改完善。图1-7中的亮线表示图像与目标模型间的三个精确匹配结果。最后，机器人的眼-脑告诉机器人手臂如何捡起零件并放到某个地方。三维视觉的问题和技术在第13章和14章中介绍。 9

习题1.4 其他应用领域

举出其他可应用机器视觉解决问题的领域。如果你脑中还没有特定的应用领域，现在就选一个。会感测到什么样的场景？图像会是什么样的？会产生什么输出？

习题1.5 问题的来龙去脉

问题可以通过不同的方式解决，解决问题的人不应该过早地陷入某种解决途径。考虑在不同情况下识别车辆的问题：（a）进入一个停车场或保安区，（b）通过一个收费卡，（c）车速超限。几个研究组正在开发或已经开发出读取车牌的机器视觉方案。提出其他能代替机器视觉的方案，并与机器视觉方案相比较，它的经济成本与社会成本如何？

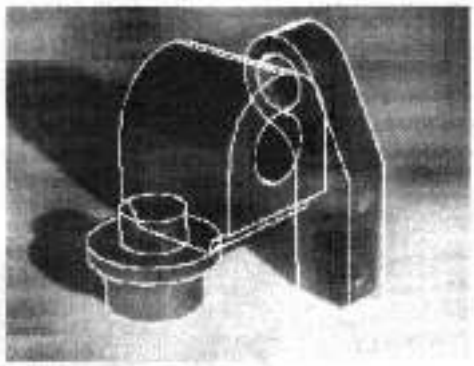


图1-7 检测器或装配机器人对存储的三维模型与感测到的二维图像进行匹配（Mauro Costa提供）

1.3 图像运算

本书包括很多图像运算，依照它们的结构、等级或目的，划分为不同的类别。有些运算只是为了方便人们观赏而改善图像，另一些则是为后继的自动处理提取信息。有些运算产生新的输出图像，而另一些则输出非图像描述。下面介绍几类重要的图像运算。

1.3.1 邻域运算

像素的值可以根据它们与少量相邻像素（比如说相邻行或列中的邻点）的关系而改变。二值图像中孤立的1或0值经常要改变，以便与邻点一样。这一运算是为了消除数字化过程中可能带来的噪声，或者只是对图像内容进行简化。例如，忽略湖面上的微小岛屿或纸面上的瑕疵。另一个常见运算是把边界像素（border pixel）变为背景像素（background pixel）。如图1-8所示，细菌的图像有着模糊的边缘而且经常连在一起。通过把边缘像素由黑色改为白色，细菌图像虽然小了一些却有了更清晰的边缘，而且分开了一些原来连在一起的细菌。这些运算将在第3章中讨论。



图1-8 （原始图像由Frank Dazza提供）

左图是细菌的二值图像（在原始的显微图像中，由于荧光染色剂的缘故，细菌是蓝色的）右图是把周围是白色邻点的黑色像素改为白色，产生出较清晰图像

习题1.6 找出残留在图1-8右图中的缺陷，描述能改善图像的简单邻域运算。

1.3.2 整幅图像增强

有些运算统一处理整幅图像，图像可能太暗，例如它的最大亮度值是120，将所有亮度值被放大2倍可以改进显示结果。如果每个像素的值用其邻域的9个像素的平均值代替，就可以去除噪声和不必要的细节。另一方面，把像素值替换为它与邻点的反差，则可以增强细节。图1-9显示对一幅图像的所有像素进行简单反差计算的结果。可以看到多数物体的边界都检测出来了。只需要对输入图像的各点在3×3局部邻域上计算对比度，就可以产生输出图像。第5章将介绍几种属于这一类的算子。一幅图像也许来自鱼眼镜头，而我们希望得到畸变较小的输出图像，这时就需要把像素值移到更靠近图像中心的位置上。这样的运算称为图像变形(image warping)，在第11章中进行介绍。 11

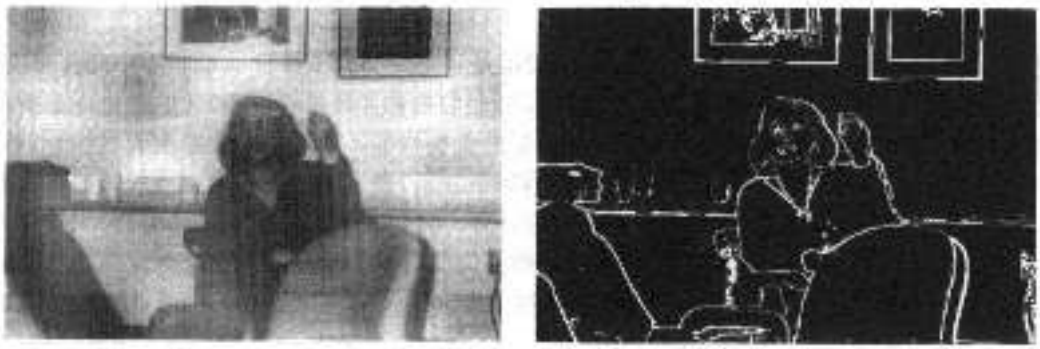


图1-9 右图是对左图进行反差计算的结果，根据反差前10%的像素取亮值，其余90%的像素取暗值，在每个像素的3×3邻域计算反差

1.3.3 多幅图像运算

两幅图像相加或者相减可以得到一幅新图像，一般用图像减法检测图像随时间的变化。图1-10显示一个运动部件的两幅图像，以及第一幅图像中的像素值减去第二幅中对应的像素值后得到的差图。通过图像减法得到运动物体的边界，但并不完整。(因为没有用到负的像素值，所以输出图像中不包含全部变化。)在另一个应用中，用当前的城市航测图像减去五年前拍的图像，可以更容易地看到城市的发展情况。图像相加也是有用的。图1-11显示托马斯·杰弗逊的一幅相片，被加到路易斯安娜州的人拱门图像上。在这种情况下，要把图像融合得更好还要做更多的工作。 12



图 1-10

左图和中图是运动部件的两幅图像
右图是表示部件边界的差图



图 1-11

左图是路易斯安娜州大拱门的图像
中图是杰弗逊头像
右图是两者相加后的结果

1.3.4 图像特征计算

我们已经看到孔计数的例子。更一般的情况是，与应交叉支撑杆质检问题中的孔对应的0值区域是我们关心的目标图像区域，这种小目标经常被称为团儿（blob）（原意也许是指水样中的微生物）。平均面积、周长、方向等是目标的重要特征，要对每个被检测的目标分别输出这些重要特征，第3章将讨论这种处理过程。第6章和第7章定量讨论图像区域的颜色和纹理特征，第4章介绍如何根据这些特征对目标进行分类，例如提取出的区域是微生物A还是B的图像？图1-12显示某著名算法应用于图1-8的细菌图像的结果，给出了从图像中识别出来的各区域的特征，包括区域面积和位置，面积为几百个像素的区域表示孤立的细菌，大的区域则表示几个相连的细菌。

13

目标	面积	边界框	中心
1	247	[(20 , 26), (32 , 56)]	(26.1, 42.0)
2	4	[(25 , 22), (25 , 24)]	(25.5, 23.0)
3	116	[(35 , 73), (54 , 86)]	(44.1, 79.4)
4	4	[(37 , 69), (38 , 70)]	(37.5, 69.5)
5	15	[(46 , 86), (50 , 89)]	(47.6, 87.4)
6	586	[(49 , 122), (95 , 140)]	(71.7, 134.8)
7	300	[(54 , 91), (77 , 112)]	(65.6, 101.9)
8	592	[(57 , 138), (108 , 163)]	(83.6, 150.8)
9	562	[(57 , 158), (104 , 183)]	(81.2, 171.4)
10	5946	[(74 , 195), (221 , 313)]	(138.0, 256.5)
11	427	[(204 , 115), (229 , 151)]	(217.1, 132.3)
12	797	[(242 , 42), (286 , 97)]	(264.9, 71.8)
13	450	[(248 , 170), (278 , 204)]	(262.7, 188.1)
14	327	[(270 , 182), (291 , 215)]	(279.9, 200.3)
15	364	[(293 , 195), (311 , 224)]	(300.8, 206.7)
16	145	[(304 , 179), (316 , 193)]	(310.4, 186.4)

目标总面积 = 10784 像素

图1-12 从图1-8右边的细菌图像中自动识别出来的区域成分。单个细菌区域包含几百个像素，较大的区域由几个相连的细菌组成。微小目标2、4和5则是噪声

1.3.5 抽取非图像表示

高层运算通常要抽取出非图像表示，也就是说数据结构不像一幅图像。（前面说过抽取这类描述经常被定义为计算机视觉的目标。）图1-12显示的是从细菌图像抽取的非图像描述。除了已经提到的例子，考虑在显微镜下统计涂片上A类和B类微生物数目的报告，以及根据视频

计算城市的两交叉路口之间的交通流量。另一个重要应用是，输入的是一篇扫描的杂志文章，而输出的是图形超文本结构，包含要识别的ASCII文本和原始图像部分。最后，在图1-7所示的应用例子中，机器视觉系统将输出对三个零件的检测结果，每个结果对应零件编号、表示零件位置的三个参数和表示零件方向的三个参数。然后将场景描述输入到运动规划系统，通过运动规划系统决定如何对这三个零件进行操作。

1.4 面临的问题

到目前为止，已经列举了不少计算机视觉的应用，但实际中的应用往往是非常困难的，真正实现起来要受到环境的制约，这将影响系统的灵活性，例如成像前对场景的光照要小心控制，或者需要用机械把目标分开或者归位。因为外界环境对图像的影响很大，使抽取目标本质特征或者不变特征（invariant feature）的最佳算法面临挑战。意想不到的光照变化或者其他物体的出现对目标外观影响很多，如图1-7和1-9中的阴影。另外决定目标结构时经常要对图像像素的各种信息进行集成。如图1-9中柜台上玻璃杯上边界的亮度同墙是一样的，因此玻璃杯上边界与墙之间的边界在像素级看不出来。为了把每个玻璃杯作为独立目标进行识别，对较宽的区域上的像素要进行分类和组织。人类在这方面很擅长，但对机器视觉来说，灵活的分类处理很困难。遮挡问题妨碍对3D物体的识别。如果图1-9中的人和椅子都没有露出腿部，视觉系统能识别出人和椅子吗？在更高层次上，什么样的狗（dog）模型才能使机器能够根据图像识别出不同个体呢？这些困难还有其他一些困难，在本书中都要进行讨论。

问题1.7

下列物体存在哪些不变特征，使它们无论在雨中还是阳光下，无论是单独存在还是伴随他物存在，无论是从正面还是从侧面，你都能够识别出它们？（a）你的网球鞋。（b）你家的正门。（c）你妈妈。（d）你最喜爱的汽车。

14

1.5 计算机和应用软件

在对定量信息的准确计算方面，计算机的能力是神奇的。图像运算已经发展了30多年，最初的研究大都在装备大型机的实验室，或者在装备专用机的生产车间里进行。近年来，大容量廉价的存储器和高速通用处理器的快速发展，使多媒体个人电脑用户也能够进行图像运算。图像爱好者在餐厅就可以工作。

人们以不同的方式进行图像运算。最省事的做法是找到一个现成的程序完成要进行的图像运算。有的程序是公开免费的，有的则必须购买。很多免费图像可从万维网上得到。如果想自己生成输入图像，可以购买一台平台扫描仪或一部数字摄像机，价格是几百美元。包含图像处理子程序的软件库也能够得到，用户编写应用程序调用软件库中的子程序，对自己的图像数据进行所需的运算。大多数销售机器视觉输入设备的公司也提供图像运算库，甚至提供美妙的图形用户接口（GUI）驱动程序。有的图像运算，用通用处理器计算需要许多秒甚至几分钟的时间，而使用特殊硬件可以加速图像运算。许多早期的并行机价值几百万美元，设计时以图像处理为首要任务，而今天多数关键的运算用几块价值几千美元的板卡就能完成。一般特殊硬件只在高生产率或有实时性要求时才需要。以图像和图像运算为要素的特殊编程语言已经开发出来，但这些语言有时与控制工业机器人的运算相结合。现在图像处理能够用通用语言编程实现，如C语言。通用计算机通过邮购或到本地商店可以很方便地买到。这些对

于机器视觉来说，都是很有利的条件。从各个方向向挑战性的问题进攻的时候到了！请读者们加入吧。

1.6 相关领域

计算机视觉同许多其他学科关系密切，我们无法在本书中深入研究所有这些关系。首先，区分图像处理（image processing）和图像理解（image understanding）很重要。图像处理主要关心的是图像到图像的变换；而图像理解关心的是基于图像的判定并显性地构造场景描述。图像处理经常用于支持图像理解，因此本书将在某种程度上进行论述。与图像处理有关的书籍中，所用到的图像模型一般是两个空间参数x和y的连续函数$f(x, y)$，而本书中所用的图像模
15 型主要是整型亮度值的二维离散阵列**I**[**r**, **c**]。在本书中，我们不区分术语计算机视觉、机器视觉和图像理解，但是，专家们肯定会争辩它们的细微差别。

人类感知的心理学因存在两个理由而显得非常重要：首先，为满足人类需要的图像制作者必须注意到客户的特点；其次，对人类在图像理解上巨大能力的研究可以指导我们开发新的算法。本书也讨论了一些人类感知和认知方面的内容，主要是为了解决现存问题。光物理学，包括光学和颜色科学，对我们的研究是很重要的。我们将讨论必要的基本知识。但是，想成为照明、感知或镜头方面专家的读者需要阅读相关的文献。本书从头到尾使用了各种数学模型，为了熟练掌握，读者必须清楚函数、概率、微积分和解析几何的概念。图像处理的有关概念经常会加强对数学概念的理解。最后，任何关于计算机视觉的书必定同计算机图形学密切相关。两个领域都涉及物体如何被观察和如何被建模，主要差别在于方向——计算机视觉是根据图像对目标进行描述和识别，而计算机图形学是根据目标描述生成图像。最近，这两个领域出现了明显的集成趋势：计算机图形学用来显示计算机视觉的结果，而计算机视觉用来建立物体模型。通常使用数字图像作为计算机图形产品的输入。

1.7 内容安排

前面几节非正式地介绍了书中的不少概念，并指明讨论这些概念的章节。读者现在应该对机器视觉涉及的领域及几种视觉算法有所了解。后面紧随着的几章主要描述2D机器视觉，其中图像分析以像素、行、交点、颜色和纹理等术语为基础。可以肯定地说，从3D场景获取2D图像的知识是存在的，图像像素与自然要素之间的关系是明显的，只是尺度上不同而已。例如一名放射线专家，能够很容易从一幅图像看出血管是否狭窄，而不用知道太多的传感器知识，或者知道像素表示身体的什么部位。机器视觉程序也能做到这一点。同样，文字识别算法实质上与被扫描的真实字体大小毫不相干。从第2章到第11章讨论的都是2D特征，比第12章到第16章的内容更一般、更简单。在第13章到第15章中，目标的3D特征和成像视点是讨论的重点。对单幅图像没法进行分析，需要把多幅图像、或者把图像与模型联系起来，或者把传感器的视线与机器人的视线联系起来。在第13章到第15章中，分析的是3D场景，而不是2D图像，最重要的分析工具是3D解析几何。和计算机图形学一样，无论是在模型抽象上还是
16 在计算量上，从2D到3D都要迈上很大的台阶。

1.8 参考文献

计算视觉方面的文献具有很强的、与应用领域相关的专业性。例如，Fleck等人（1996）

的论文，讲述如何检测色情图片，以便在儿童的计算机中对这些图片进行屏蔽。孔计数算法的讨论参考了Kopydlowski（1983）的工作，其中对卡车交叉支撑杆的质检用到了该算法。再如卫星传感器的设计与医用仪器的设计差别很大，制造系统也有自己的特色。特殊领域的参考文献，如Nagy（1972）和Hord（1982）是关于遥感的，Glasby与Horgan（1995）是关于生物学的，Ollus（1987）和QCAV（1999）是关于工业应用的，ASAE（1983）是关于农业应用的。有关彩色CCD摄像机早期发展的几篇论文之一是Dillon等人于1978年发表的论文。几个应用领域共存的问题、方法和理论自然是教科书的主要内容，这也是本书的主要内容。第一本使用计算机处理图像的教科书可能是Rosenfeld（1969）编写的，主要内容是图像处理，而不涉及高层的模型。Ballard与Brown（1982）所编教材算是第一本计算机视觉（Computer Vision）的教科书，内容集中在基于高层模型的图像分析方面。Levine（1985）编写的教材值得注意，它包含有关人类视觉系统的重要内容。Haralick与Shapiro（1992）的两卷集是算法及其数学基础的最新资源。Jain、Kasturi和Schunk（1995）的著作，主要从工程的角度介绍机器视觉的最新进展。

1. ASAE. 1983. Robotics and intelligent machines in agriculture. *Proc. Ist Int. Conf. Robotics and Intelligent Machines in Agriculture* (2–4 Oct. 1983), American Society of Agricultural Engineers, Tampa: FL, St. Joseph, MI.
2. Ballard, D. H., and C. M. Brown. 1982. *Comput. Vision.* Prentice-Hall, Englewood Cliffs, NJ.
3. Dillon, P., D. Lewis, and F. Kaspar. 1978. Color imaging system using a single CCD area array. *IEEE Trans. Electron Devices,* ED-25(2):102–107.
4. Fleck, M., D. Forsyth, and C. Pregler. 1996. Finding naked people [in images]. *Proc. European Conf. Comput. Vision.* Springer-Verlag, New York, 593–602.
5. Glasby, C. A., and G. W. Horgan. 1995. *Image Analysis for the Biological Sciences.* John Wiley & Sons, Chichester, England.
6. Haralick, R., and L. Shapiro. 1992/3. *Computer and Robot Vision, Volumes I and II.* Addison-Wesley, New York.
7. Hord, R. M. 1982. *Digital Image Processing of Remotely Sensed Data.* Academic Press, New York.
8. Igarashi, T., ed. 1983. *World Trademarks and Logotypes.* Graphic-sha, Tokyo.
9. ——— ed. 1987. *World Trademarks and Logotypes II: A Collection of International Symbols and Their Applications.* Graphic-sha, Tokyo.
10. Jain, R., R. Kasturi, and B. Schunk. 1995. *Machine Vision.* McGraw-Hill, New York.
11. Kopydlowski, D. 1983. 100% inspection of crossbars using machine vision. Publication MS83-210, Society of Manufacturing Engineers, Dearborn, MI.
12. Levine, M. D. 1985. *Vision in Man and Machine.* McGraw-Hill, New York.
13. Nagy, G., 1972, Digital image processing activites in remote sensing for Earth resources. *Proc. IEEE,* v. 60(10):1177–1200.
14. Ollus, M., ed. 1987. Digital image processing in industrial applications. *Proc. Ist IFAC Workshop,* Espoo, Finland (10–12 June 1986), Pergamon Press, Oxford.
15. Pratt, W. 1991. *Digital Image Processing,* 2nd ed. John Wiley, New York.
16. QCAV. 1999. Quality control by artificial vision. *Proc. 5th Int. Conf. Quality Control by Artificial Vision* (18–21 May 1999), Trois-Rivieres, Canada.
17. Rosenfeld, A. 1969. *Picture Processing by Computer.* Academic Press, New York.

17

1.9 附加习题

对下面几个问题的回答要求写出短篇报告，有的问题要求进行定量分析，其中多数问题在后面的内容中会进行更详细地讨论。书中凡是标注星号（*）的问题表示有一定的难度，需要仔细研究、推导或者编程。

习题1.8 商品销售问题

食品店收款员正在结算你所购买的货物。条形码技术使得处理某些商品变得比较容易，例如汤品罐头，只需让条码阅读器对准条码直到听见"嘟"的一声。但这个系统不适用于散装物品，收款员必须停下来单独处理。怎么办呢？如果把一部摄像机安装在台秤与条码阅读器的上方或内部，告诉收银机当前处理的是什么货物，这个办法你想到没有？利用本书介绍的方法，机器视觉系统就能够区分出绿菠菜与绿甘蓝、富士苹果与麦金托什苹果。请描述这个机器视觉系统是如何结合现有的收款技术，来帮助收款员计算你的账单。

习题1.9 细菌数量

参考图1-8中的细菌图像和图1-12中的自动特征计算实例。在这个例子中，能不能统计出细菌的数量，使精度保证在5%之内？请加以解释。

习题1.10 从视频到三维模型

假设你有一套巴黎圣母院的视频资料，视频是由一个人在教堂内外边走边拍得到的，因
18 此里面包括多个视点。你能只用视频制作出教堂的3D模型吗？（如果没有信心，就假设自己是名建筑师。）如果不能，为什么？如果能，在只有二维图像的情况下如何构造三维模型？

习题1.11 计算反差

类似图1-9所示，思考在每个3×3邻域上计算反差的方法，假定9个像素值是0到255间的亮度值；而输出像素值是0到255间的某个值，表示反差大小。（图1-9的右图实际上只用了两个像素值0和255，你可用整个范围的值。）

习题1.12 人脸解释

（a）确定杂志广告中的人物性别和大致年龄对你来说容易吗？（b）心理学家告诉我们，人类具有看到人脸就马上确定其年龄、性别和敌意程度的能力。假设人类确实有这样的能力，如果你认为这种能力是基于图像特征的，那么是什么特征？如果你认为用不到图像特征，那么解释人类是根据什么做出这种结论的？

习题1.13 一画抵千言吗？

考虑下列短文，短文出自William Faulkner的《声音与愤怒》（The Sound and the Fury）（Vintage Books Edition,1987版，©1984，Jill Faulkner Summers著，p.195）。你认为一部机器能从论及的场景视频中提取出这样的描述吗？

我荣誉出河流的雾气，在那黄昏之后
我看见夕阳静静躺在溪面上，像月光般柔的铺面
超过它们，光芒如下清白的天空
很低倾斜，轻如蝶迷失的翅膀

习题1.14 孔计数的正确性*

这一问题应作为提高习题，需要进行较深入的思考并阅读本章内容之外的知识。(a) 二值图像中的2 × 2邻域，有多少种可能的模式？把它们全部列出来。(b) 哪些模式不是4-连通的？边界点定义为2 × 2邻域的中心格点，邻域中包含0值和1值像素。(c) 证明通过统计沿边界的e和o模式的数目能够得到单个孔，并且当只有一个孔时，公式$n = (e - o)/4$是正确的。(d) 证明没有两个孔能共有一个边界点。(e) 证明当有任意个孔时，公式仍然正确。 19

习题1.15 二值图像合适吗？

拍摄场景图像，并转换成二值图像，比如红血球图像，其中图像上对应目标物的区域像素值为0，对应背景或非目标的区域像素值为1。考虑这种情况对于下列场景是否能够实现。你认为为什么能够生成或者不能够生成这样的二值图像？

1. 一张打过字的纸，通过页面扫描生成输入图像，总目标是识别出其中的多数字符，并生成ASCII文件，这样就可以用文字处理器对文本进行编辑。

2. 输入人头部的X光图片，0值区域表示肿瘤，1值区域为背景。

3. 输入表现美国弗吉尼亚州Richmond春天的卫星图像，通过调整传感器或一些简单的计算机算法，生成二值图像，其中0值区表示杜鹃花丛，1值区为背景。

4. 通过统计阀杆暗区的像素数，检测汽车发动机阀杆的宽度。我们每天要制造几十万个阀门，可以对环境进行很好的控制，设备的价格也是合理的。 20

第2章　图像生成与图像表示

人类关于外部世界的大量信息是通过视觉获得的。物体表面的反射光或者通过物体的透射光，在人类双眼的视网膜上形成图像。根据这一对图像就能得出三维环境的结构信息。成像的要素是：（a）物体所在的场景，（b）光照条件，（c）对反射光或透射光的感测。

本章的主要目的是说明传感器如何产生2D或3D场景的数字图像。对于自然界中的物体，其反射光或者透射光可通过不同的成像设备进行检测。2D数字图像是经物体反射或者传播的光强阵列。用机器或计算机程序对该图像进行处理，从而对场景做出判定。通常2D图像是3D场景的一种投影，这种表示方法在机器视觉和本书中常常用到。在本章末，讨论3D环境结构和2D图像结构之间的关系。

标注星号（*）的内容主要涉及一些技术性细节，不太感兴趣的读者可以跳过不看。

2.1　光线感测

许多科学史可以根据测量和产生电磁辐射的设备发展史来述说，如无线电波、X射线、微波等。人眼感受器中的化学物质，能感测的光波范围大约从400nm的紫色到800nm的红色。蛇类和CCD传感器（参见图2-2）能够感测到大于800nm的红外波长。有的装置能检测波长很短的X射线，也有的能检测无线电长波。不同波长的辐射光有不同的性质，如X射线能够穿透人类骨骼，而波长较长的红外光甚至不能穿透云层。

图2-1是普通摄影的简单模型。被单光源（太阳或镁光灯）照射的面元，向摄像机方向反射光线，摄像机胶片上的化学物质发生感光反应。更详细的内容在第6章中讨论。物体表面的光反射和光生成机制，产生可见范围的光波。本书会涉及很多电磁辐射的性质，但通常只进行定性分析，详细的定量分析请参考物理学或光学方面的书籍。应用领域的工程师需要了解一些感光材料、光辐射和传感器方面的知识。

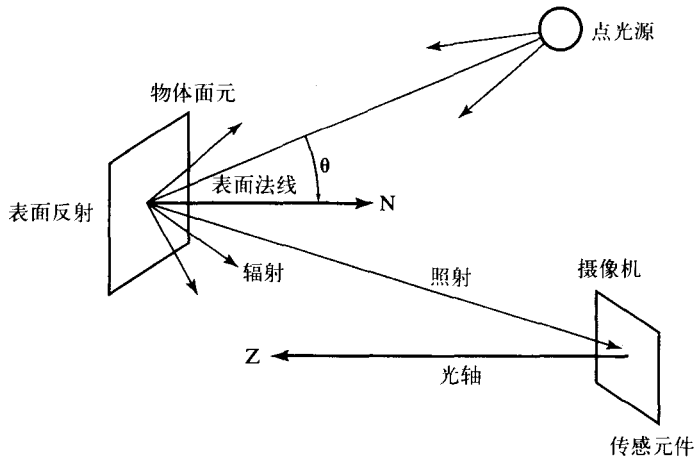


图2-1　对单光源照射的反射

2.2 成像设备

产生数字图像的设备有很多种，它们的检测原理和机电设计是不同的。本章介绍几种不同的传感器，最常用的在本节讨论，其他的则作为选读内容在2.9节介绍。我们重点放在传感器的主要功能和概念方面，把技术性信息作为课外阅读内容。

2.2.1 CCD摄像机

图2-2显示了用电荷耦合器件（CCD）技术制作的摄像机，这是机器视觉系统最灵活、最
22 通用的输入装置。CCD摄像机非常像家庭用的35mm胶片相机，只不过在成像平面上使用转化光能为电荷的微小固态感光元，代替了能进行光学反应的化学胶片。每个感光元把接收到的光能转换为电荷。首先把所有的感光元清零，然后根据光照强度感光元产生累积电荷。可以用快门来控制感光时间，也可以不用。成像平面就像数字存储器，能通过计算机逐行读出所存的信息。图中显示了一台简单的黑白摄像机情况。

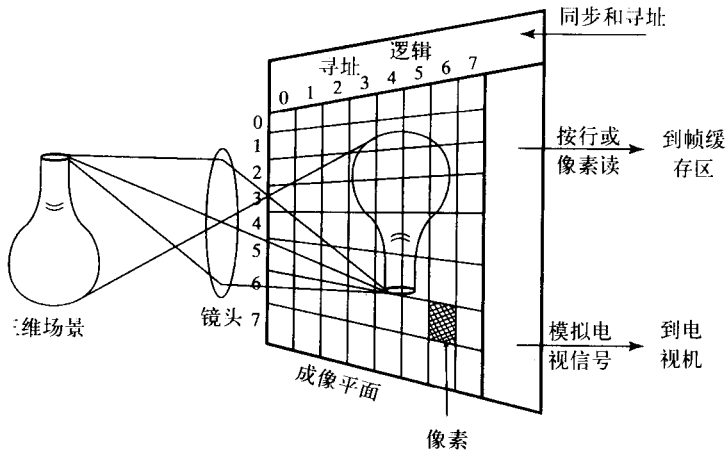


图2-2 CCD摄像机拍摄一个花瓶。离散的感光元转换光能为电荷，输入到计算机时，电荷对应一个比较小的数

如果数字图像是500行和500列，灰度值占一个字节，则产生250 000字节的存储阵列。CCD摄像机一般与称为帧捕捉卡（frame grabber）的计算机板卡相连，帧捕捉卡具有图像存储器，也许还能对摄像头进行控制。新的设计支持直接数字通信（如采用IEEE 1394标准）。数字摄像机自身带有能存放几十帧图像的内存，有的还带有软盘。任何时候都可以把这些图像输入到计算机中进行处理。图2-3是一个计算机系统示意图，同时具有摄像机输入和图像输出。这是工业视觉或者医学成像的典型系统，也是典型的多媒体计算机系统，配有为电视会议摄像的廉价的摄像头。帧缓存区（frame buffer）作为高速图像存储器在此起着中心作用。一幅图像经模数转换后，其数字形式存储在帧缓存区内，于是就可以进行图像显示，以及使用各种计算机算法进行处理。帧缓存区实际上可存储好几幅图像或者它们的衍生图像。

处理数字图像的计算机程序把像素值表示为**I**[r, c]或**I**[r][c]，其中**I**是数组名，**r**和**c**分别是
23 行号和列号。本书在算法中采用这样的表示方式。有的摄像机可以通过设置产生二值图像（binary image），像素值0代表暗、1代表亮，或者1代表暗、0代表亮。通过简单计算也可以产

生二值图像，即把低于某阈值（threshold）t的所有像素值取为0，把大于等于阈值t的像素值取为1。第1章给出了一个例子，对磁共振图像进行阈值化，以对比高、低血流量。

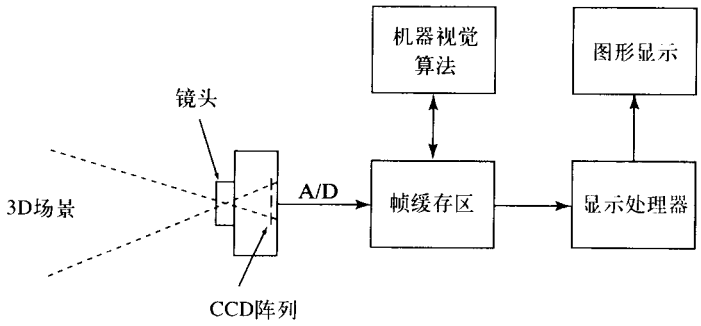
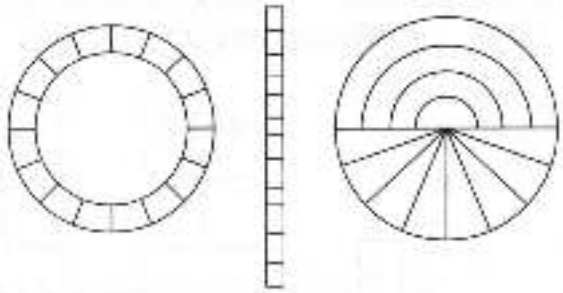


图2-3 帧缓存区在图像处理中的中心作用

2.2.2 图像形成

成像的几何原理可以概括为3D场景中每个点通过投影中心（center of projection）或镜头中心（lens center）投影到成像平面上。图像点的光强与三维表面点辐射出来的光强有关，我们后面会看到这种关系非常复杂。这种投影模型在物理上是合适的，因为利用带小孔而无镜头的摄像头盒，能够实际做出针孔（pin-hole）摄像头来。CCD摄像机采用的镜头通常与家用35mm胶片相机的镜头一样，具有两个凸面的单镜头，见图2-2所示。实际上多数镜头是由两个以上的折射面复合而成的，有两点很重要：首先，镜头是光线采集器。来自3D点的光线，经过3D点到镜头的整个锥体空间，然后会聚到图像上的一点。图2-2中，三道光线从花瓶的顶部投射出来，它们确定了镜头采集光线的锥体空间。对于其他的场景点也存在类似的锥体空间。由于镜头几何缺陷、不同颜色光弯曲不同及其他影响因素，锥体空间实际上在成像平面上产生一个有限而模糊的斑，称为模糊圈（circle of confusion）；其次，CCD传感器阵列由物理上分散的感光元而不是非无限小的点构成。于是每个感光元接收到3D表面上多个相邻点发出的光线。这两个效应使图像变得模糊，影响了图像的清晰度和可被感测的最小场景细节的尺寸。

CCD阵列制作在芯片上，典型的芯片尺寸约为1cm × 1cm。如果阵列有640 × 640个像素或512 × 512个像素，则每个像素的实际宽度约为0.001英寸。如图2-4所示，还有其他把CCD感光元分布在图像平面上（或图像线上）的实用方法。线状阵列可用在只需要测量物体宽度的 24
情况，或者用摄像机成像和检测连绵布匹的场合。线状阵列的一行，可以有1000到5000个像素。这样一个阵列能用在推扫方式，线状传感器横着移过被扫描的材料，就像用手持扫描仪或高精度机械扫描仪如平台扫描仪扫描一样。目前许多平台扫描仪仅用几百美元就能买到，通过扫描彩色图片或印刷媒体得到数字图像。柱状镜头一般用来把真实世界中的一条直线聚焦到线状CCD阵列上去。圆形阵列可方便地用于检查诸如钟表或速度表的模拟刻度盘。把目标在摄像头前放好，圆形阵列扫描得到指针的图像。图2-4c是令人感兴趣的ROSA分块，对所有落进扇状或环状区域的光能，提供一个硬件集成解决方案。它原来的设计是为了量化一幅图像的能谱，但也可能有其他简单的用途。芯片制造技术为实现客户设计的其他方案提供了机会。



a) 圆形 b) 线式 c) ROSA

图2-4 其他实用的几何阵列

习题2.1 观察CCD摄像机的结构

如果你有一部CCD摄像机，并获准对它的构造进行研究，卸下镜头并观察摄像机的结构。它有快门来隔断所有的光线吗？它有光圈来改变光线通过的锥体空间大小吗？有办法改变焦距吗？焦距就是镜头与CCD之间的距离。检查CCD阵列，主动感知区域有多大？你能看到单个的感光元吗？需要放大镜吗？

习题2.2

假定要用CCD摄像机读出模拟钟表上的数字，CCD摄像机正对着钟表。钟表图像的中心位于256×256数字图像的中心，时针的宽度是分针的2倍，但长度是分针的0.7倍。为了确定
25 钟表指针在图像上的位置，需要以圆周方式扫描数字图像的像素。(a) 在半径为R、圆心在图像中心I[128,128]的圆周上，对像素I[r,c]给出计算r(t)，c(t) 的公式，其中t是到I[r,c]的光线与水平轴之间的夹角。(b) 对t进行控制，以便生成数字圆周的唯一像素序列。这有问题吗？(*c) 阅读计算机图形学方面的课外书，写报告说明产生数字圆周的实际方法。

2.2.3 视频摄像机

供人类消遣的视频摄像机，以每秒30帧的速度记录图像序列。除了每幅图像或每帧图像含有空间特征外，图像序列能够表达目标随时间的运动情况。采用每秒60个半帧的场频，主要是为了让人眼感觉不出帧与帧之间的切换。前半帧扫描奇数行，后半帧扫描偶数行，连续交替。声音信号也作了编码。供机器使用的摄像机，能够以任何速率记录图像，而不需要采用半帧技术。

图像序列的各帧之间有分离标记。为了减少数据量经常用到一些图像压缩技术。制定的模拟电视标准，可以满足多种需求。最有意义的是同一信号不仅能用彩色电视播放，也能用黑白电视播放，并且还能携带声音或文字信息。具体内容请感兴趣的读者阅读2.5节的电视和MPEG编码标准。这里继续把数字视频作为三维数字图像序列。

机器视觉中的CCD摄像机技术，常常受到为人类消遣而制定的显示标准的影响。首先，视频序列中奇、偶帧的交错，可以让人眼感到画面流畅，却为机器视觉带来了不必要的麻烦；其次，许多CCD阵列中像素的宽高比为4:3，这是因为大多数为人设计的显示器的尺寸比例是4:3。正方形像素和统一的尺度参数更有利于机器视觉。巨大的消费市场使摄像装置为人类消遣而设计，机器视觉的研发者不得不适应这种现状，或者为制造有限数量的摄像装置付

出更多的代价。

2.2.4 人眼

人眼大致相当于球形摄像机，靠近外面的是焦距为20mm的晶状体，在视网膜（retina）上形成图像。视网膜位于晶状体的对面，附着在球面内侧（参见图2-5）。通过调节瞳孔（pupil）的大小，虹膜（iris）对穿过晶状体的光线多少进行控制。每只眼睛有上亿个感受器细胞，这比一般CCD阵列中的感光元要多得多。此外感受细胞非均匀地分布在视网膜上。靠近视网膜中心的一个区域称为中央凹（fovea），排列着密集的彩色感受器，称为锥状体（cone）。离开中心越远，锥状体越少，而黑-白感受器即杆状体（rod）越多。对于3D表面上的一个点，在中央凹上成像，人眼感受到的是对应三原色的三个分离的光强。因为来自该点的光线落在三种不同类型的锥状体上，而每个类型的锥状体具有特殊的色素，该色素对某个波长范围内的光线具有敏感性。人的眼-脑系统最令人惊奇的一点是，能够平稳感受到不间断而且稳定的三维世界，即使眼球在不断地转动也如此。人类特殊的视觉感知系统需要眼球不断地快速运动。人脑的相当一部分功能是进行视觉信息处理。人类视觉系统的其他特征将在本书必要的地方进行讨论，特别是有关颜色感知的更详细内容在第6章进行讨论。

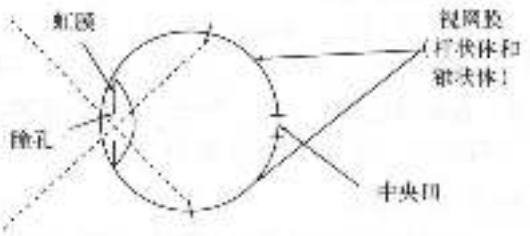


图2-5 人眼摄像机示意图（更详细的内容参见Levine1985年的著作）

习题2.3

假定人的眼球直径是1英寸，1亿个杆状体和锥状体分布在其内侧面的1/π面积上，被单个感受器所覆盖的平均面积有多大？（要记住中央凹中的感受器分布比这平均值要稠密得多，而外围的感受器分布则稀疏得多。）

2.3 数字图像中的问题 *

存在几方面因素会对感测过程产生影响，下面列出的是比较重要的几个方面。我们前面所讲的理想化情况只是对真实物理系统的一种近似。这些因素对图像造成的总体效果是，图像的几何形状和亮度两方面都发生畸变。第11章介绍纠正畸变的一些方法，但更常见的是不考虑这些影响而直接做出决策。

2.3.1 几何畸变

图像处理过程中造成几何畸变（geometric distortion）的影响因素有几个方面，如有缺陷的镜头使来自场景面元的光束不沿预期的光路弯曲，焦距小的镜头常常发生桶状畸变，如图2-6的右图所示，场景外围的直线发生远离图像中心的弯曲。

2.3.2 散射

辐射光通过介质时会发生弯曲或散射（scattering）现象，航测图像和卫星图像特别容易发生这种情况，水蒸气和温度梯度使大气层具有类似透镜的特性。

2.3.3 光晕

由于检测元件是离散的，如CCD感光元，它们相互间并不能做到完全绝缘，一个感光元上的电荷会泄漏到相邻的感光元中，如图2-6的中图所示，这种电荷泄漏反映在图像平面上，

结果一个很亮的区域向外展开，生成一朵比它实际尺寸要大的明亮“花朵”，因此称这种现象为光晕（blooming）。

2.3.4 CCD差异

由于制造上的问题，不同的感光元对于同样的光强会产生不同的响应。为了精确地测量光强，应该用均匀的光照进行标定，确定出针对每个像素的比例矩阵s[r, c]和平移矩阵t[r, c]，使光强修正为$\mathbf{I}_2[\mathbf{r}, \mathbf{c}] = \mathbf{s}[\mathbf{r}, \mathbf{c}]\mathbf{I}_1[\mathbf{r}, \mathbf{c}] + \mathbf{t}[\mathbf{r}, \mathbf{c}]$。在极端情况下，CCD阵列中可能有一些失灵感光元（dead cell），它们对光照不发生响应。这种缺陷能够通过检查检测出来，软件的补救措施是把失灵感光元的响应用相邻感光元响应的平均值代替。

2.3.5 削波与逆变

模/数转换时，非常高的光强会被限制到一个最大值，否则其高位数就会丢失，结果使数值逆变成低强度的编码。在灰度图像中，逆变的结果表现为明亮的区域内带有较暗的核心点；在彩色图像中，则产生明显的颜色变化。图2-6的左图反映了逆变现象，亮线的交点处有一些暗点，比两条亮线都要暗。

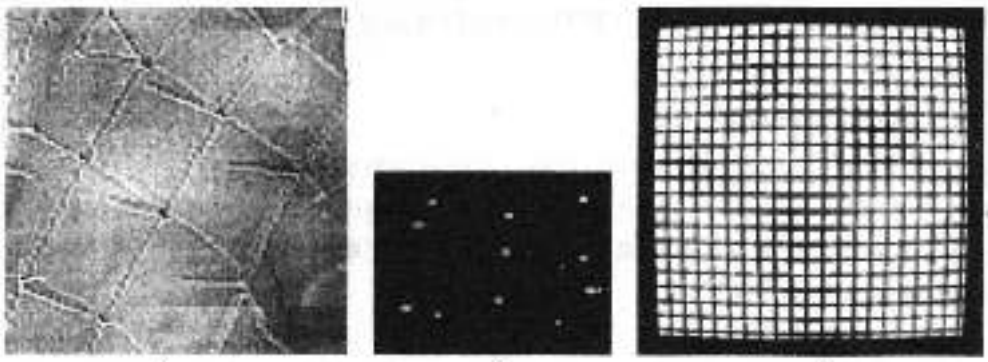
28

2.3.6 彩色畸变

不同波长的光线通过透镜时，产生不同程度的弯曲（透镜的折射率（index of refraction）与波长有关），结果来自同一场景点的不同波长的光能，在检测器上可能形成几个分开的像素。例如，场景外围黑白分明的边界，会在图像上形成几个像素宽的亮度变化斜坡（ramp）。

2.3.7 量化效应

数字化处理过程中，光强是从场景的离散区域中采集的，光强值又被映射为离散的灰度值，所以进行混合和舍入时量化效应比较明显。下一节更详细地讨论这些问题。



a) b) c)

图2-6 各种畸变的图像

a）模/数转换期间，发生亮线交叉点上的灰度级削波

b）亮点周围的像素光强因光晕而增大

c）焦距很小时经常出现的桶状畸变

2.4 图像函数与数字图像

现在讨论一些概念和符号，这对图像运算的理论与编程都很重要。

2.4.1 图像类型

在图像计算中，要了解模拟图像（analog image）和数字图像（digital image）两个概念。

图像函数是一个数学模型，经常用于分析图像，我们一般把图像看成是双变量的函数。这样，分析图像就可以用所有的函数分析方法。数字图像只是具有离散值的二维矩形阵列。图像空间位置和强度值都被量化成离散的数值，这样图像就能够存储在2D计算机存储器中。一般像素强度用8位（1字节）数表示，取值范围为0到255。256级一般是可从传感器获取的全部精度，通常足以满足消费者需要。以字节为单位也方便计算机的存储与运算。例如，一幅图像在C程序中可被说明为**charI[512][512]**。彩色图像的每个像素需要三个8位数值来表示。在一些医学应用中，采用10位编码方法，允许有1024个不同的强度值，这已经接近于人类分辨的极限了。

为了理解重要的概念和建立一套全书通用的表示方式，下面对几个概念进行定义。首先从理想的光学系统产生理想的模拟图像开始，假设精度是无限的。在离散位置上对模拟图像采样，并把各位置处的图像强度用离散数值表示，于是形成数字图像。所有实际图像要受到物理过程的影响，位置和强度的精度都有一定的限制。

定义2　模拟图像是指二维图像**F(x, y)**，其空间参数x和y具有无限精度，在每个空间点（**x, y**）的光强也具有无限精度。

定义3　数字图像是指二维图像**I(r, c)**，用离散的二维光强阵列表示，光强的精度是有限的。

29

把图像的数学模型看成是两个实际空间参数的函数，在描述图像和定义图像运算时都非常有用。图2-7d显示，如何在各图像点[**x, y**]处，通过对连续图像进行采样得到图像像素。如果在X方向，w距离内有M个采样点，则像素间的x间距Δx为w/M。图2-7给出了像素中心点与强度阵列中的某个元素之间的关系。

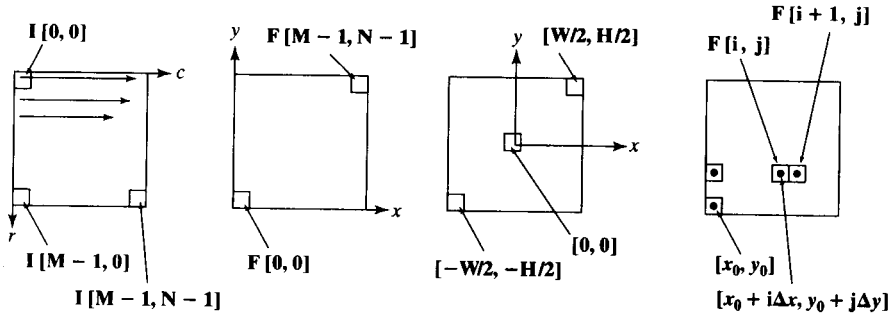


图2-7　不同的图像坐标系

a）与显示屏一致的光栅坐标系，行列坐标原点[0, 0]位于左上角

b）笛卡尔坐标系，原点[0, 0]位于左下角

c）笛卡尔坐标系，原点[0, 0]位于图像中心

d）像素中心[x, y]与阵列元素I[i, j]所在的面积元素之间的关系

定义4　图像函数f(x, y) 是图像的一种数学表示方法，它是两个空间变量x和y的函数。x和y是实数，确定图像上的一点。**f(x, y)** 通常也是实数，确定图像在点（**x, y**）处的强度。

定义5　灰度图像是单色数字图像**I[r, c]**，其中每个像素只有一个强度值。

定义6 多谱图像是二维图像M[x, y]，在每个空间点或像素位置存在一个强度值向量。如果是一幅彩色图像，则该向量有三个元素。

定义7 二值图像是指所有像素值要么为0要么为1的数字图像。

定义8 标记图像L[r, c]是数字图像，其中的像素值是有限的字符标记。像素的字符值表示对该像素作某个判定的结果。相关的概念有**主题图像**和**伪彩色图像**。

30 讨论图像中的像素、用计算机进行图像运算、用数学公式描述图像、或相对于设备坐标讨论图像都要用到坐标系。本书内外常用的几种坐标系如图2-7所示。遗憾的是，不同的计算机工具所用的坐标系也不同，用户必须习惯使用这些坐标系。还好我们所讲的概念并不受坐标系的约束。在本书讨论概念时，一般使用与数学课本一致的笛卡尔坐标系，而图像处理算法则通常使用光栅坐标系。

2.4.2 图像量化与空间度量

如图2-2所示，数字图像的每个像素表示实际图像中某个基本区域的采样结果。如果把该像素从图像平面反投影到场景中的实物上，那么场景元素的大小就是传感器的*标称分辨率*（nominal resolution）。例如一张10in.[⊖]见方的纸片，对应500 × 500的数字图像，则传感器的标称分辨率就是0.02in.。如果场景的深度变化比较大，这个概念就没有意义，因为标称分辨率随着深度和表面方向而变化。成像传感器的*视场*（field of view, FOV）是对传感器能看到的场景范围的度量。传感器的*分辨率*（resolution）则与它进行空间测量或细微特征检测的精度有关。（如果使用得当再加上模型信息，一幅500 × 500的像素图像作出的测量精度可达1/5000，这个精度称为*亚像素分辨率*（subpixel resolution）。）

定义9 CCD传感器的**标称分辨率**指图像平面上的一个像素所对应的场景元素的大小。

定义10 分辨率是指传感器的测量精度，但定义方式多种多样。如果在实际三维空间定义，则可能就是标称分辨率，如“这台扫描仪的分辨率是地面上的1m”，或者是感测图像中每毫米距离能分开或区分出来的线耦数。一个完全不同的概念是有效的像素数，如“这部摄像机的分辨率是640 × 480像素”。后面的定义有个好处，它提到视场能被分成多少部分，而这与精密测量和覆盖场景区域的能力有关系。如果测量精度小于标称分辨率，则称为**亚像素分辨率**。

图2-8是同一个人脸的四幅图像，主要是为了强调分辨率的影响。用64 × 64的分辨率我们可以识别出熟悉的人脸，用32 × 32的分辨率也许也能识别出来，但是16 × 16就不够用了。在利用计算机视觉解决问题时，采用的分辨率要合适。分辨率太低会影响识别效果或者测量不准，分辨率太高则会使算法过慢而且浪费内存空间。

定义11 传感器的**视场**是它能感知到的场景的大小，例如10in. × 10in.。由于这个数字会
31 随着深度而变，因此采用**角视场**（angular field of view）或许更有意义，如55° × 40°。

由于图像中的一个像素度量的是实际场景中的一个区域而不是一个点，所以像素值经常是不同目标的混合结果。例如卫星图像中每个像素对应地面上10m × 10m的一个点。那么像素值可能是水、土壤和植被组合的结果。在生成二值图像时问题就显得严重了。考虑前面一张纸成像的例子，纸上每英寸分布10个字母。许多像素将重叠一条字符边界，因此得到背景的

⊖ 1in. = 0.0254m

高强度光线与字符的低强度光线的混和结果，最后的结果是介于背景和字符之间的某个值，可能被置为0或者1。不管是哪个值，它都不是完全正确的。

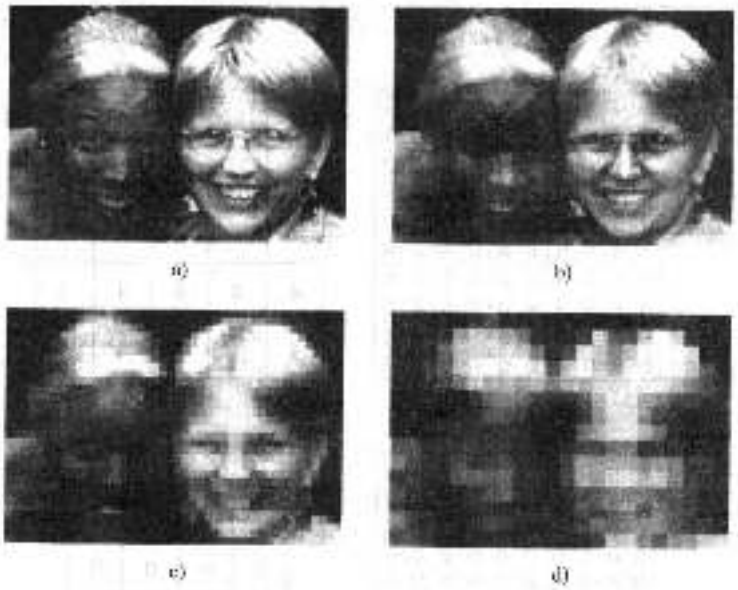


图2-8 两个人脸的四幅数字图像（图片由Frank Biocca提供）

a）127 × 176

b）126 × 176，对a中每一个2 × 2邻域做平均，对均值复制四次生成一个2 × 2的平均块，从而生成图像

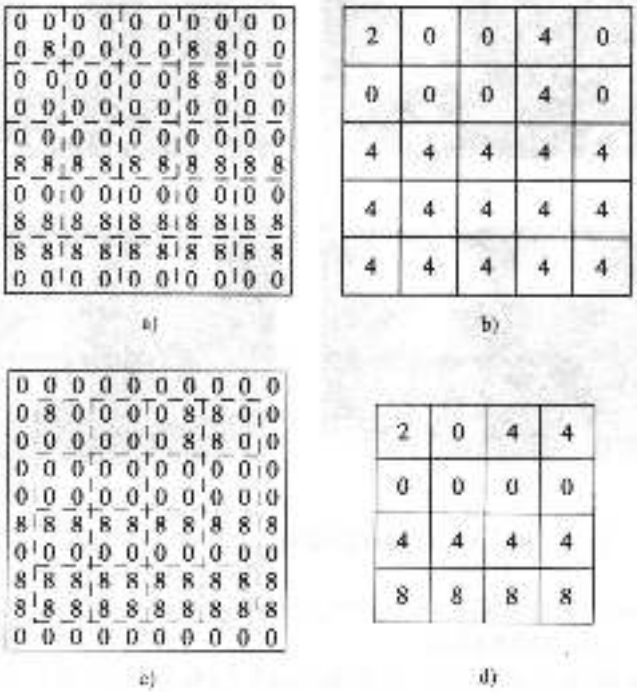
c）124 × 176，用同样的方式由图像b生成

d）120 × 176，用同样的方式由图像c生成。有效的标称分辨率分别是（127 × 176）、（63 × 88）、（31 × 44）、（15 × 22）（斜视观察呈块状的图像，这种方式可用来让分明的区域边界变得模糊）

图2-9是量化问题的一个具体例子。图中的左边是一个10 × 10的2D阵列，其中的黑色背景亮度值为0，白色砖块亮度值为8。砖块构成的模式包括两个亮点和两条宽度不同的亮线。如果场景的图像落到5 × 5的CCD阵列上，每2 × 2的方块邻域精确地落到CCD的一个感光元件上，结果就产生图2-9b所示的数字图像。左上角的CCD感光元感知到的强度为2 = (0 + 0 + 0 + 8)/4，是四个方块的平均强度。右上角的四个亮块落到两个CCD感光元件上，每个感光元集成两个亮块和两个暗块的强度。强度为8的单行亮块经CCD变换后，成为图像上强度为4的一行像素。 32
强度为8的双行亮块经CCD变换后，成为图像上强度为4的两行像素。而场景中的两条线在图像中混合在一起。如果取t = 3对图像进行阈值化，那么含一个亮块的亮度模式将在图像中消失，而其他三个特征区域将融合成一个区域！如果摄像机在水平和垂直两个方向上都平移一块砖的位移，则会产生图2-9d的结果。由四块砖组成的亮区形状在d中的变换方式与b中不同，场景中的两行亮线在d中形成亮度斜坡而不像b中是灰度一致的区域。另外d中有三个目标区域而b中只有两个。图2-9表明，大小近似一个像素的场景特征，其图像是不稳定的。

图2-9表明空间量化效应（spatial quantization effects）对检测精度和检测能力有较大的影响。较小的特征可能被丢失或融合，即使在检测较大的特征时，也存在不能恰当表示其空间范围的可能。在砖块例子中，注意观察四块砖组成的亮区，成像后对应的不是强度为4的垂直CCD感光元对，就是强度为4的水平CCD感光元对。当通过阈值化产生二值图像时，由于混合 33
像素（mixed pixel）的舍入，可以预测到边界误差可高达0.5个像素。这暗示两条边界之间的测

最误差可能会达到1个像素。此外，如果要检测二值图像中的某些特征，该特征的图像至少要有两个像素那么大，而且包括两目标之间的间隙在内。下面考虑传真中的句号，它的图像直径是一个像素，且严格地落在四个CCD感光元会合点的正中。这四个像素中的每个像素参加混和时，属于背景的部分要多于属于字符的部分，当形成二值图像时，句号就有可能丢失！



a)

2	0	0	4	0
0	0	0	4	0
4	4	4	4	4
4	4	4	4	4
4	4	4	4	4

b)

c)

2	0	4	4
0	0	0	0
4	4	4	4
8	8	8	8

d)

图2-9 量化问题的例子

a) 10×10的块阵列，亮度值取0或8

b) 块阵列的5×5强度图像，其中每个像素对应2×2方块邻域的平均亮度

c) 摄像机向下、向右移动一块后感知到的图像。注意量化的亮度值不仅取决于实际像素的大小，而且和在阵列上的位置有关

d) 摄像机移动后得到的强度图像，成像方式与b中一样。为了解释实际场景中的特征，不论是用b还是用d都存在问题

定义12 **混合像素**是一类图像像素，其强度表示对真实世界多个目标类型的混合采样结果。

习题2.4 面积的变化

白纸上有一黑色矩形，成像时该矩形对应图像上5.9×8.1的像素范围内。生成二值图像时，根据像素中目标或者背景成分的多少决定该像素值是0或者是1。变换时矩形的双边可以与CCD的行和列平行。二值图像中的像素最小面积是多少？最大面积又是多少？

习题2.5 细小特征的丢失

考虑印刷电路板上的两根明亮的平行导线，每根导线在图像平面上的宽度是0.8个像素。会像上个习题一样，在二值图像中出现一根消失而另一根存在的情况吗？请加以解释。

在第13章中，对光学薄透镜方程进行了讨论，并研究了它与摄像机分辨率、图像模糊和景深的关系。有兴趣的读者可学习有关章节的内容。讨论完感知特点、分辨率和混合像素的概念之后，就具备了足够的背景知识，可以开始某个二维机器视觉应用系统的研究工作。例如用显微镜发现一定的目标，检查一块PC机电路板，或者识别一个背光三维目标的阴影。必须设计好成像环境，使得看到的特征在图像中有适当的大小。假定考虑了从场景到图像的尺度变化。在图像中没有保留明显的三维特征，那么就用第3章到第10章所讲的二维方法分析图像。

习题2.6 检测纸币的面值

设计专收$1、$5、$10和$20面值的自动售货机传感器。你只需建立一种表达方式提供给识别器，不用设计识别算法，也不用考虑识别假钞。（在回答之前要进行一些采样。）假定在 34
钞票进入机器时，必须用线性CCD阵列进行数字化处理。(a) 应该使用什么样的镜头和照明？(b) 线性阵列中需要有多少像素？请加以解释。

2.5 数字图像格式*

数字图像在通信、数据库和机器视觉中广为应用，并且已经开发了标准格式以便不同的硬件和软件能共享数据。图2-10说明了这种情况。遗憾的是仍然有几十种不同的图像格式在使用。本节对几种重要的图像格式进行简单讨论。原始图像（raw image）只是字节流，图像像素按一行一行的顺序编码，这种顺序称为光栅顺序（raster order）。图像行与行之间允许用换行符进行分隔。图像的类型、大小、生成时间和创建方法等信息并不是原始图像的一部分。这些信息可以手写在磁带的标签上或者研究记录本上。这是不妥当的。（在作者参加的一个项目中，录像前先录下条形码，计算机程序随后处理该条形码，就得到实验处理的全部非图像信息。）最近开发的标准图像格式包含着一个文件头，文件头中记录着标记数据和解码所必需的非图像信息。

有的图像格式最初是由公司规定的，这些公司主要进行图像处理和图形工具的开发工作。有时能得到公开文档和转换软件，但多数情况下得不到。下面的内容是实用资料，可以帮助读者从事于计算机图像处理。虽然细节内容随着技术的进步变化很快，但本节介绍的基本概念则不会变。 35

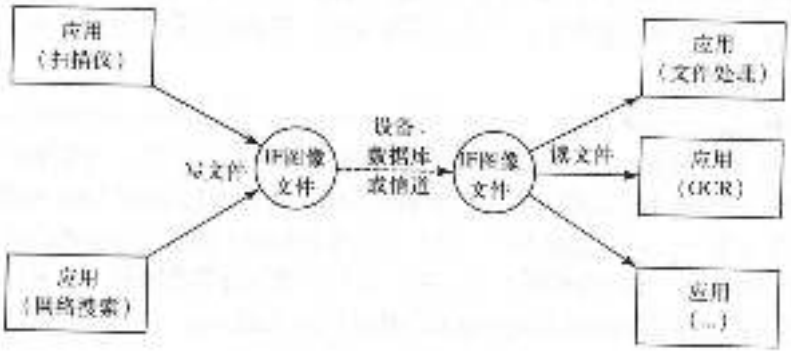


图2-10 建立、使用或转换图像数据的设备或应用程序有很多，标准格式的图像文件（IF）可以方便地用于不同的设备和程序

2.5.1 图像文件头

文件头（file header）是图像的自我说明，通过文件头图像处理工具能同它们一道工作。文件头应该包含图像的维数、类型、创建日期和某类标题。它也可以包含用于解释像素值的颜色表或编码表。很不错的特征是*历史段*（history section），其中包含如何建立和处理图像的信息，但这个特征不容易得到。

2.5.2 图像数据

有的图像格式只能处理有限类型的图像，如二值图像和灰度图像。至今仍幸存的格式通过不断发展，能够包括更多的图像类型和特征。文件格式不同，对像素大小与图像大小的限制一般也不同。有的格式可处理帧序列。*多媒体*（multimedia）格式正在发展，并同时包括图像数据、文本、图形和音乐等。

2.5.3 数据压缩

许多格式提供了对图像数据的*压缩*（compression），不是对所有像素值直接进行编码。图像压缩能使图像数据减少到原来的30%甚至3%，这取决于需要的图像质量和所用的压缩方法。压缩可以是*无损*（lossless）的或*有损*（lossy）的。使用无损压缩，能完全恢复出原始图像；使用有损压缩，不能完全恢复出原始图像，有时能观察出图像质量有损失，但并不是总能观察到。为了实现压缩，图像文件必须包括一些关于压缩方法和参数的抬头信息。许多数字图像和符号数字信息不同，丢失或改变几位数字图像数据，不管是对于人还是对于机器，关系都不是很大。这种情况与其他计算机文件不同，如在员工档案中更改一个字位就有可能改变薪水字段达8 192美元，或者会把公寓地址从A变成B。图像压缩是个振奋人心的研究领域，涉及范围从信号处理到目标识别。在本书的几个地方会讨论到图像压缩，但不做系统性的讨论。

定义13 如果解压缩的方法能够精确地恢复原始图像表示的每一位，则所用的图像压缩方法是**无损**的；否则该压缩方法是**有损**的。

2.5.4 常用图像格式

本书的许多图像都具有多种格式。有些图像由同事提供，或取自图像数据库，格式包括GIF、JPG、PS；有的图片是通过扫描照片得到的，其原始图像格式是GIF或TIFF。用图像工具xv做了简单的图像处理，较为复杂的图像运算则用hips工具或专门的C 或C++程序处理。下面简单介绍最常用的图像格式。*图像/图形文件格式还在发展之中*，趋势是具有更强的包容性。
36 读者要明白，下面讨论的内容也在发展之中，应通过参阅最新的文献了解最新变化。

2.5.5 游程编码二值图像

对于二值图像或标记图像来说，*游程编码*（run-coding）是一种有效的编码方法。它不仅能够减少存储空间而且能够加速图像运算，例如加快集合运算的速度。当图像的行像素存在大量冗余时，游程编码就非常有效。对于二值图像的每一行，我们可以记录下0的数目，接着是1的数目，如此交替下去直到完成全行。图2-11中的游程码A就是这种编码的例子。图中的游程码B是更紧凑的只有1-游程的编码方式，据此我们仍能恢复初始的行。本书中的一些算法就采用这种编码方式。游程编码常常是标准文件格式中的压缩方法。

2.5.6 PGM格式

存储和交换图像数据的简单文件格式之一是可转移式点阵图系列（PBM/PGM、PPM）。图像头和像素信息以ASCII方式编码。图2-12所示的图像文件，表示8 × 16、最大灰度值为192的图

像。下部是绘制出的两幅图，每个都是对原始文本进行转换后输出的图像。左下方的图像通过复制像素的方式得到较大的32×64图像，右下方图像则先使用有损压缩转换成JPEG格式。PGM文件的第一项是魔值（Magic Value），即本例中的“P2”，指明图像信息如何编码的（本例中的ASCII灰度级）。大型图片可以利用二值而不是ASCII像素编码。（二值码的魔值是“P4”）。

```
                  000000000011111111112222222222333333333344444444
Column c      :   012345678901234567890123456789012345678901234567

Image Row r   :   00000000111111000000000011100000001111111111000000

Run-code A    :   8(0)5(1)12(0)3(1)7(0)9(1)5(0)

Run-code B    :   (8,12) (25,27) (35,43)
```

图2-11 游程编码对连续0值或1值的运行长度进行编码，在一定范围内生成有效的压缩图像

```
P2
# sample small picture 8 rows of 16 columns, max gray value of 192
# making an image of the word "Hi".
  16 8   192

 64 64  64  64  64  64  64  64  64 64 64  64  64 64 64 64
 64 64 128 128  64  64  64 128 128 64 64 192 192 64 64 64
 64 64 128 128  64  64  64 128 128 64 64 192 192 64 64 64
 64 64 128 128 128 128 128 128 128 64 64  64  64 64 64 64
 64 64 128 128 128 128 128 128 128 64 64 128 128 64 64 64
 64 64 128 128  64  64  64 128 128 64 64 128 128 64 64 64
 64 64 128 128  64  64  64 128 128 64 64 128 128 64 64 64
 64 64  64  64  64  64  64  64  64 64 64  64  64 64 64 64
```



图2-12 表示图像中单词“Hi”的文本（ASCII）文件。背景的灰度级是64，“H”以及“i”的下半部灰度级是128，“i”的圆点的灰度级是192。左下方是一幅打印图画，使用图像格式转换工具对上述文本文件转换后得到。右下方是使用有损压缩算法后得到的图像

习题2.7 创建一幅PPM图画

类似图2-12所示的“P2”单色编码文件，利用魔值“P3”和每个像素的三个强度值（R,G, B），彩色图像可以编码成PBM格式。用编辑器创建一个文件bullseye.ppm，对不同颜色的三个同心圆区域进行编码。对于每个像素，三个颜色值前后紧挨着，而不是像在其他格式中分别对三幅单色图像进行编码。应用图像工具或网络浏览器显示你的图片。 37

2.5.7 GIF格式

图形交换格式（GIF）由CompuServe公司开发并用来对万维网上或当前数据库中的海量图像进行编码。使用GIF文件格式相对容易，但不能应用于高精度色彩，因为只用了8位二进制数对颜色编码。256个颜色值对于计算机显示图像来说绰绰有余，也可以使用更节省空间的

16色编码。可以采用Lempel-Ziv-Welch（LZW）无损压缩方法。

2.5.8 TIFF格式

由Aldus公司开发的TIFF或TIF格式是非常通用和复杂的，它用于所有流行的平台，常常是扫描仪使用的格式。标记图像文件格式（Tag Image File Format）支持多种图像，图像的每个像素的颜色编码可以是1到24位的二进制数，可以用有损或者无损压缩方法。

2.5.9 JPEG格式

JPEG（JFIF/JFI/JPG）是更近期的标准，来自联合摄影专家组（Joint Photographic Experts Group），其主要目的是提供高质量彩色静止图像的实用压缩。JPEG是面向数据流的编码方法，而且允许对实时硬件进行编码和解码。尽管每个文件只有一幅图像，而图像大小
38 可达64K × 64K像素，每个像素可用24位二进制数表示。文件头能包含一幅相当于64K未压缩字节的缩略图。JPEG的一个主要优点是独立于颜色编码系统，颜色系统的详细内容在第6章中给出。为了实现高比例压缩，采用灵活但复杂的有损编码方案，常常能以20:1压缩一幅高质量图像而没有明显的图像失真。当图像存在大片颜色不变的区域，以及细节区域中的高频变化对用户不重要时，采用这种方法进行压缩的效果就很好。（JPEG有一个很少用的无损压缩选项，可通过使用预测编码实现2:1压缩。）压缩方法采用离散余弦变换（discrete cosine transformation），随后是赫夫曼编码（Huffman coding）。离散余弦变换将在第5章中讨论。赫夫曼编码在本书不讨论。JPEG不是为视频压缩设计的。

习题2.8

在计算机系统上找到一个图像浏览工具。（可能只要点击图像文件图标就可做到。）用一幅人脸图像和一幅风景画。原始图像应是高质量的，如800 × 600彩色像素，来自平台扫描仪或数码摄像机。把图像变换成不同的格式如GIF、TIFF、JPEG等。记录已编码的图像文件的字节数，并注意观察图像的质量，同时考虑图像全部和图像细节。

习题2.9 JPEG研究*

（a）研究对8 × 8图像块的JPEG压缩方案。（b）以无损压缩方式（除了可能的舍入误差外）实现和测试DCT压缩方法。（c）利用现成的图像工具进行有损压缩。（d）利用来自有损压缩的64个系数，重新产生一幅8 × 8的图像，并与原始的8 × 8图像进行像素值比较。

2.5.10 PostScript格式

BDF/PDL/EPS格式系列利用可打印的ASCII字符存储图像数据，并经常同X11图形显示器和打印机一起使用。PDL是一种页面描述语言，而EPS是封装的postscript（源于Adobe）格式，这种文件格式常常用于插入到较大文档中的图形或图像。像素值用7位ASCII码进行编码，因此这些文件能用文本编辑器检查和更改。可以做到每英寸75到3000点的灰度级或颜色，较新的版本包括了JPEG压缩技术。PDL文件头包含了图像所在页面的图像边框。本书中的大多数图像都是EPS格式。

2.5.11 MPEG格式

MPEG（MPG/MPEG-1/MPEG-2）是用于视频、音频、文本和图形的面向流的编码方式。MPEG代表运动图像专家组（Motion Picture Experts Group），是成员来自工业界和政府的国
39 际小组。当前MPEG系列的标准正随计算机和通讯技术快速发展。MPEG-1主要是针对多媒体

系统设计的，它提供0.25Mbits/s的压缩音频数据率，以及1.25Mbits/s的压缩视频数据率。这些速率适合于处理个人计算机的多媒体信息，但对于高质量电视来说太慢了。MPEG-2标准能提供15Mbits/s的数据率来适应高清晰度电视。MPEG压缩方案利用了与JPEG中一样的空间冗余，同时也利用了时间冗余。实用的压缩比一般是25:1，甚至可能达到200:1。时间冗余（temporal redundancy）本质上意味着从一帧到下一帧期间图像上的许多区域变化不大，而编码方案可以只对变化部分进行编码，甚至可以根据视频序列中的前后帧进行帧的预测。（MPEG的未来版本将具有识别物体的代码和生成目标图像的程序代码。）媒体质量在编码时刻就确定了。运动JPEG是一种混合编码方案，它对视频单帧用JPEG压缩技术，而不利用时间冗余。运动JPEG简化了编码和解码过程，但压缩效果不是很好，所以存储和传输时的效果比不上MPEG。第9章中介绍MPEG运动向量用于视频压缩。

2.5.12 图像格式比较

表2-1根据存储量的大小对一些常用的图像格式做比较。其中左边一列用的是8 × 16的小型灰度图片“Hi”，而右边一列用的是347 × 489的彩色图像。对于同一幅图像，用不同的格式转换顺序，可能产生大小不同的图像。例如从扫描仪输出的“Cars”TIF文件是509 253个字节，转换成256色的GIF文件则需要138 267个字节，再转换成TIF文件需要171 430个字节。最后的TIF文件中的彩色代码具有较少字位，但是在阴极射线管（CRT）上看起来都差不多。大小只占三分之一的JPEG文件显示效果也一样。从空间的角度来说，有损JPEG显然是最佳压缩方法，但代价是增加了解码的复杂性，为了满足实时性需要用硬件来实现。 40

表2-1 同一图像不同编码格式下的文件大小（以字节为单位）。图2-12所示的是8 × 16灰度“Hi”图像和图2-13所示的是347 × 489彩色“Cars”图像

图像文件格式	“Hi”的字节数	“Cars”的字节数
PGM	595	509 123
GIF	192	138 267
TIF	918	171 430
PS	1 591	345 387
HIPS	700	160 783
JPG（无损）	684	49 160
JPG（有损）	619	29 500

2.6 成像影响因素

睁开双眼，敞开心灵，到室外去散步，我们会发现自然景观是多么丰富，这是艺术家们早就明白的一点。室外丰富的景观增加了我们的见识，却给机器视觉带来了问题。（参见图2-13。）图像点的亮度或颜色以复杂的方式受到材料、几何位置和光照的影响。不仅材料类型是重要的，而且目标与传感器、光源、其他目标之间的相对方向也是重要的影响因素。例如，存在镜面反射、阴影、互反射等现象，材料也可能是透明的。在识别表面或者识别目标时，与依赖多个像素而不只是一个像素的形状特征或纹理特征相比，颜色特征相对不太重要。对于我们几乎不能控制的环境如交通监控，令人感兴趣但实现起来却很困难。

即使是精心设计的工业环境或电视播放室，问题仍然存在。在第6章我们会看到，点光源照射金属圆筒，圆筒表面的反射光强度可在100 000到1的范围内变化，而多数传感器不能适应这样大的动态范围。太阳光或者人造光会加热表面，使它们随时间而产生不同的辐射，红

外线的增加会使CCD图像变亮，飞机起飞后会在跑道上留下影子。受控的单色激光能够对成像过程起帮助作用，但是它也可能被某些表面完全吸收，或者被其他表面的二次反射所支配。



图2-13 人类感知的多种深度线索的复杂场景

在许多自动化应用中，可通过工程途径解决问题，把不相关的光线滤掉。举个例子，如果使用只允许红外光线通过的滤波器，那么深红色樱桃上的擦伤就能更清楚地看到。在稳定的照明下运动目标会导致图像模糊，利用闪光灯（strobe light）进行短时间照明，用高灵敏度
41 探测器拍摄图像，其中的目标物就相当于是静止的。结构光（structured light）的使用使表面测量和检查变得容易，例如用红和绿交替的精细条纹光对涡轮机叶片进行照明，表面有缺陷的地方在二维图像上就表现为明显的光线间断。在本书的某些地方会提到这些方法。

2.7 从二维图像到三维结构

人类视觉系统综合不同的线索特征对三维世界的结构进行感知。我们在此仅仅做出定性说明，认知心理学家J.J. Gibson对这些线索给出了定量模型。80年代，计算机视觉研究者以极大的精力投身于对这些模型的实现和实验上。书中在几处对一些定量模型进行讨论。

成像过程记录了三维世界结构和二维图像结构之间的复杂关系。透视投影的模型见图2-2，并参考图2-13。穿插（interposition）也许是最重要的深度线索。近处的目标部分遮挡远处的目标，识别遮挡能得到相对深度。看起来位于墙内洞的人显然比墙更靠近传感器。位于汽车后面的人离得要比这辆车更远。相对尺寸也是重要的线索。20m远的汽车图像比10m远的汽车图像要小得多，即使远处的汽车体型较大也是这样。远处的汽车对于我们不仅显得小而且动作缓慢，经验已经教会我们如何把大小和速度与距离联系起来。当我们沿铁轨漫步时，两条铁轨在远处似乎相交于一点（消隐点，vanishing point），尽管我们知道在三维空间中它们一定是平行的。一扇朝里开的门在我们的视网膜上成像为梯形，而不是我们知道的矩形。门上离得远的那条边显得比离得近的那条边要短。这是透视投影中的缩短（foreshortening）效应，并且传递门的三维朝向信息。一个相关的线索特征是纹理梯度。表面纹理随观察距离和表面方向而变化。在公园里，凑近可以看到一片草叶或者枫叶，离得远时就只能看到绿色了。后退时观察表面，视图中的纹理发生变化，这种图像纹理的变化称为纹理梯度（texture gradient）。在第12章中将对提到的问题进行更多的讨论。

习题2.10 像艺术家一样观察

有意识地在两种不同的环境中观察，并且叙述上面讨论过的线索特征。比如在繁忙的咖啡馆，从几层楼的高处观察城市街道的一角，或者在树林中的某个地方。

2.8 5种参考坐标系

对三维场景定性或定量分析都离不开参考坐标系（reference frame）。三维场景分析中一般要用到5种坐标系，三维场景如用机器人和传感器控制工作间中的操作，或者为人机交互提供一个虚拟的三维环境。这些坐标系不仅对机器人学很重要，而且对心理学家以及理解人类空间感知也很重要。这5种坐标系的图示参见图2-14。实际上图中有6个坐标系，因为在场景中有两个不同的物体，一个方形物和一个锥形物，每个物体都有自己的参考坐标系。在所有这些坐标系中，除了图像坐标是像素阵列的整数下标外，其他坐标都是沿连续轴的实数。对于这个例子，你可以想像是模拟一个球场情况，其中摄像机是电视摄像机，在拍摄全场棒球比赛，场景中的目标是球员、球拿、球和球棒等等。 42

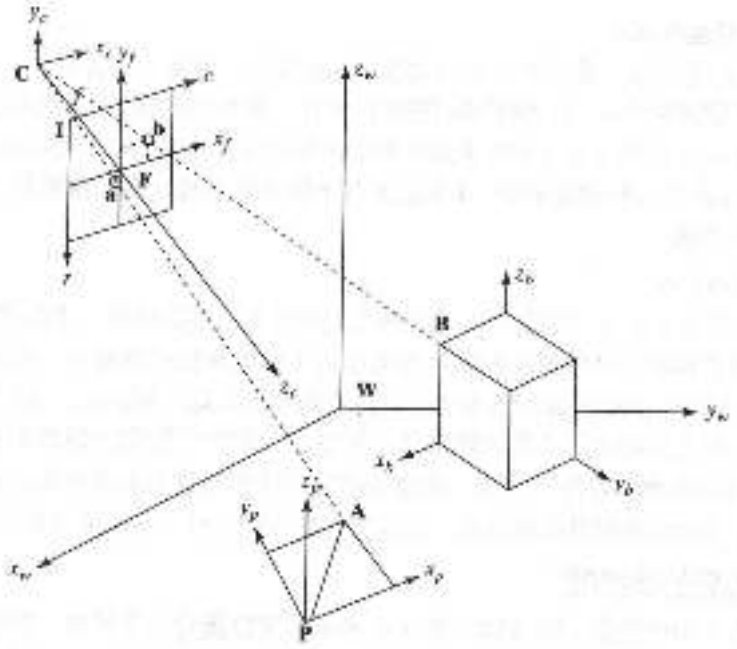


图2-14 三维场景分析使用的5种坐标系：世界坐标系W，物体坐标系O（锥形物O_p或方形物O_b），摄像机坐标系C，实际图像坐标系F和像素图像坐标系I

2.8.1 像素坐标系I

像素阵列中的每个点都具有整数的像素坐标。图2-14中，锥顶A对应的像点是像素$\mathbf{a} = [a_r, a_c]$，其中a_r和a_c分别是行数和列数，都是整数。场景中的许多事物只通过分析行和列像素图像就能确定。例如搬运机器人或其他搬运机械，总是搬运大概位于摄像头前面的箱子（或一箱洗涤剂），只利用行和列像素构成的阵列图像就可以检测到前表面上的标记。在模拟棒球比赛中，只用图像就能确定击打手是否在用一根黑色球棒。不过如果只用图像I而没有任何其他信息，就不能确定在三维空间中哪个目标实际上更大一些，或者是否有目标发生碰撞。 43

2.8.2 物体坐标系O

在计算机图形学和计算机视觉中，理想的物体建模都是用物体坐标系表示的。图2-14中显示了两个物体坐标系，一个表示方形物$\mathbf{O}_b$，一个表示锥形物$\mathbf{O}_p$。三维角点**B**相对物体坐标系的坐标是$[x_b, 0, z_b]$。不管这个方形物相对世界或工作区坐标系**W**的姿态如何变化，这些坐标依然不变。检查目标时要用到物体坐标系，例如检查一个特殊的孔是否与其他孔或角有合适的相对位置。

2.8.3 摄像机坐标系C

当以观察者（摄像机）为中心时，常常要用到摄像机坐标系**C**。例如观察一个目标是否刚好在传感器的正前方，是否正在离开等。如果一个球的图像在你的视网膜中不断变大，则球有可能要击中你。对于有视觉的机器人或者人来说，既是目标又是传感器，因此物体坐标系和传感器坐标系几乎相同，但不是严格相同。（看上去好像不会撞到门，但你却撞到了，发生过这种事吧？）计算机图形学系统允许用户选择不同的摄像机视点观察三维场景。例如把摄像机对准第一垒比赛，可以更好地进行观察。）

2.8.4 实际图像坐标系F

摄像机坐标是实数，其单位常常与世界坐标系相同，即英寸或者毫米，包括深度坐标z_c在内。三维点投影到位于$[x_f, y_f, f]$的实际图像平面上，其中f是焦距，x_f和y_f不是图像阵列中像素的下标，而与像素大小以及与光轴像点的相对位置有关。在图2-14中实际图像中的点**a**在坐标系**F**中的横坐标和纵坐标都是负的。**F**坐标系包含的图像函数把实际图像数字化，形成像素阵列**I**表示的数字图像。

2.8.5 世界坐标系W

通过坐标系**W**来建立三维空间中的物体之间的关系。例如确定一名跑垒者是否远离球垒，或者跑垒者是否与第二垒的球手相撞。在机器人工作室或虚拟环境中，执行器和传感器常用世界坐标进行通信。例如图像传感器告诉机器人在哪里捡起一根螺栓，并把它插入哪个螺孔。

这些坐标系之间的几何关系和数学关系很重要，书中后面的内容将要用到。在接下来的
44 几章中，我们只处理像素阵列图像，假设像素阵列图像与真实世界存在直接对应。对透视变换的代数运算及缩放效果熟悉的读者，可以直接进入第12章学习透视成像模型。

2.9 其他类型的传感器*

我们再谈几种传感器。读者第一次阅读本书时可以跳过这节内容，除非某种传感器对你当前的研究很重要。传感器技术正在迅速发展，我们不仅希望生产出新的传感器，而且希望现有传感器的性能更加完善。

2.9.1 测微密度计

让一束光线穿过幻灯片或胶片，由对面的单感光元传感器记录在[r, c]位置处材料的光密度。通过机械平台精确地移动幻灯片或胶片，直到扫描完整个矩形区域为止。对于CCD阵列传感器，由于各感光元制造上的差异，会对光密度有所影响。在这一点上，单感光元传感器要优于CCD阵列。单感光元传感器的另一个优点是，能够扫描到更多的行和列，但这种仪器速度缓慢，无法用于自动化场合。

读者通过了解下面的扫描技术发展史，会从中发现一些有意思的地方。70年代在Azriel Rosenfeld的实验室中，许多图片按下面的方式输入到计算机中：把黑白图片贴在一个钢筒上。

一般一次扫描9×9英寸的图片或拼贴画，圆筒装在一台标准的车床上，车床带动图片区域上的所有点旋转，面前是小的发光二极管LED，由传感器测量从每个点反射过来的光线。圆筒每转一圈产生3600个行像素点，这些像素点作为一个数据块存储在磁带上。磁带的记录速度与车床同步！最终的磁带文件有3600×3600个像素，通常包含着许多次实验的数据结果，这些数据随后通过软件进行分离。

2.9.2 彩色图像和多谱图像

人眼利用不同的感受细胞感知不同波段的光线，可以称它为多谱（multispectral）传感器。有的彩色CCD摄像机，在CCD阵列的正前方安装有折射薄膜。折射薄膜把单束白光分成四束光，落到CCD阵列的四个相邻的感光元上。由此产生的数字图像可以看成是四幅交错的彩色图像的集合，每一幅都对应着经折射分离出的一种波长，光谱信息上的增益以空间分辨率上的损失为代价。另一种设计是，一色轮在光路中同步旋转，在一个时间间隔内只让红光通过，然后是蓝光，随后是绿光。（色轮是一个圆盘形的透明薄膜，每种颜色所占的扇区大小相等。）色轮旋转一周期间，读取CCD阵列三次，可获取三幅分离的图像。这一设计中，感光速度以颜色灵敏度下降为代价。如果物体是快速运动的，那么在获取三幅分离的图像期间，物体上的一点实际上成像到图像平面的不同像素位置上。

有的卫星利用瞄准感知（sensing through a straw or boresight）技术，通过视轴（boresight）观察地球上的一点，以便在同一时刻收集从该点发出的辐射光，而其他位置的辐射光则被屏蔽，参见图2-15。辐射光束通过棱镜，被分离成不同的波长，落到CCD线性阵列上。CCD线性阵列同时对几个波段光的强度进行采样和数字化。（波长较短的光穿过棱镜比波长较长的光弯曲得更多。）图2-15显示，五个不同波段的光谱产生一个像素，该像素是含有五个强度值$[b_1, b_2, b_3, b_4, b_5]$的向量像素。通过移动视轴或者使用一个扫描镜，得到给定行的所有列，这样就产生一幅2D图像。卫星在围绕地球的轨道上运动，产生图像的不同行。正如你想到的，得到的图像会由于运动的存在而发生畸变，所有扫描点的集合形成地球上的一个梯形区域，利用11章中的变形方法可以把矩形（rectangular）数字图像变换成这种形状。地球上的一点，对应一个强度谱（spectrum of intensity values）而不是一个强度值，这样就可以把地面类型分为水、森林或沥青路等。

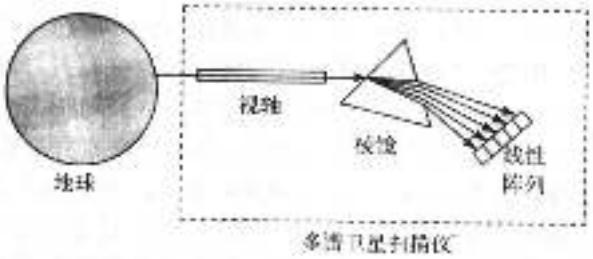


图2-15 卫星上的视轴多谱扫描仪。来自单个面元的辐射光，根据波长被折射成不同的成分

2.9.3 X射线

X射线设备产生X射线穿透某种材料，经常是透视人体组织，有时也可能是透视焊接好的管道和苹果酱瓶子。在发射器的对面，传感器记载图像点上的能量，其工作方式与测微密度计相同。如果图像点上记载的能量较低，表明沿发射器发射线方向上的物质密度较大。很容易想像到，一张2DX射线胶片被穿过人体的X射线曝光。三维感知可以通过CT扫描仪（CAT）实现，投影X射线沿着不同的方向穿过人体，得到不同位置的密度数据，然后在数学上构造出3D密度立体。图2-16中的右图是计算机绘制的2D图像，在这之前先用CT扫描一只狗得到3D

高密度体素。可以看出，对这些体素的绘制效果，就好像是从特定视点看到的不透明反光表面。诊断专家能够从任何视点检查被测件头的结构。

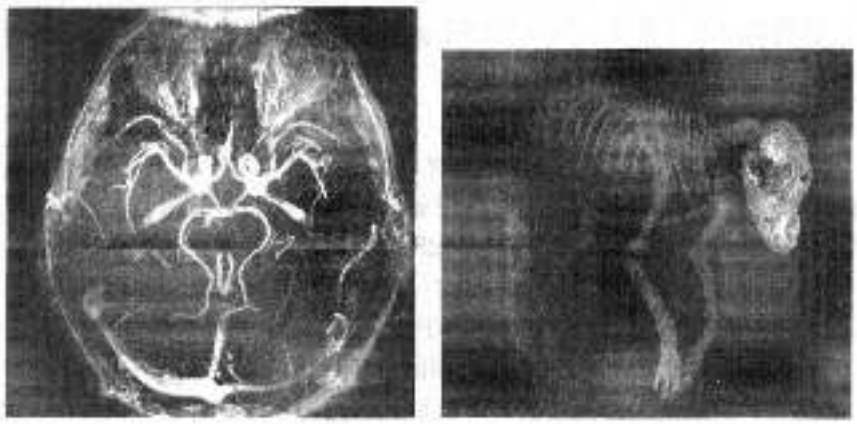


图 2-16

左图是对人头MRA切片上的最亮像素进行投影形成的最大强度投影（MIP）（由MSU放射科提供）

右图是计算机生成的图像，被光照射的表面显示出CT扫描的高密度体素（数据由Theresa Demarco提供）

习题2.11

46 想想自己牙齿的X光片，亮区和暗区各表示什么部位？是正常牙齿还是蛀洞？为什么？

2.9.4 磁共振成像

磁共振成像（MRI）能够产生组织（通常是人体组织）的三维图像。生成的数据是三维阵列**I**[s, r, c]，其中s表示身体的切片，r和c与前面一样。每个小的体积元素即体素（voxel）代表直径大约2mm的样本，该处的强度与组织的化学性能有关。磁共振血管造影术（MRA）产生的强度与体素上组织（血流）的速度有关。这样的扫描仪价值上百万美元，扫描一次要花费一千美元，但是诊断效果非常好。MRI扫描能够检查水果和蔬菜的内部缺陷，将来设备便宜的话可以用来做这个事情。图2-16中的左图是从三维MRA数据中抽取的数字图像。通过选择所有切片s中的最亮体素**I**[s, r, c]，生成最大强度投影（maximum intensity projection）即**MIP**[r, c]，在任何观测方向作投影，计算机算法都能生成MIP图像，一般作出诊断需要满满一墙这种打印的二维图像，但是现在有了真正的三维显示仪，放射线学者正在学习使用它们。

2.9.5 距离扫描仪和深度图像

有的设备可以感知到二维面元的深度或者距离，而不仅仅是辐射强度。在深度图像中，能直接得出物体表面的各种形状；而在强度图像中，只能通过麻烦而又易于出错的分析才能推出表面的形状。图2-17是LIDAR装置，发射一束调幅的激光到三维表面上的一点，并且接收反射回来的信号。通过比较发送和接收信号的相位变化（延迟），LIDAR能够根据激光束的
47 调制周期测出距离。由于歧义性，这个办法只对一个周期的距离有效，距离为$d + n\lambda/2$的点产生的响应与距离为d的点一样，其中λ是调制周期。此外，通过比较接收的强度和发送的强度，LIDAR也能估算出该表面点对这个波长激光的反射率。因此，LIDAR产生两幅配准了的图像：深度图像和强度图像。由于需要间歇时间（dwell time）来计算每点的相位变化，LIDAR要比

CCD摄像机慢。由于需要机械部件来控制激光束，LIDAR也很昂贵。对于采矿机器人和太阳系天体探索机器人来说，这个花费是合理的。

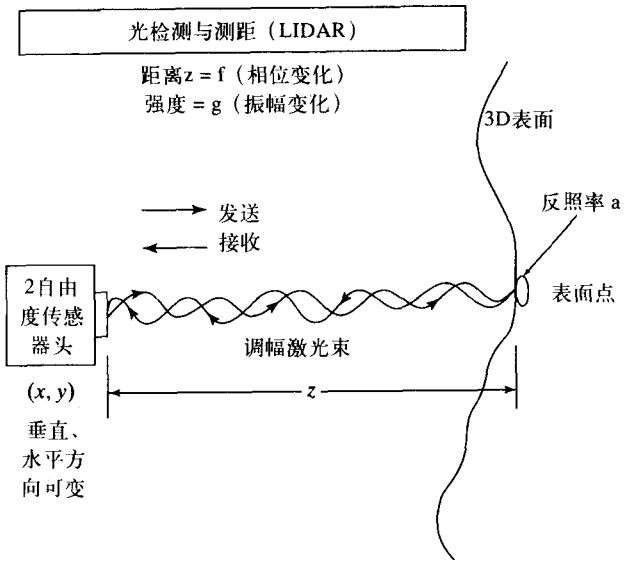


图2-17 LIDAR传感器能够产生兼有距离和强度的像素

已有5000年历史的三角测量方法，稍加变形可用来测量三维表面，如图2-18所示，光平面照射物体表面，表面产生的反射光线进入摄像机镜头。图像上的亮点[x_c, y_c]与物体上的3D点[x_w, y_w, z_w]对应。

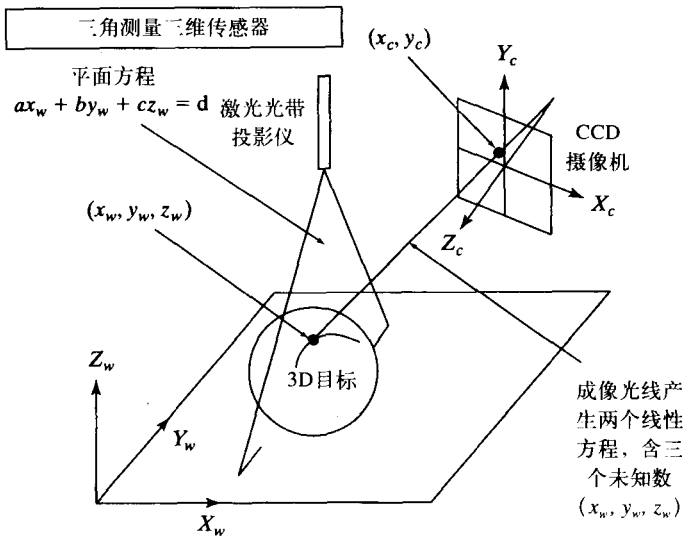


图2-18 条纹光传感器通过三角测量产生三维点的坐标

因此，测量装置知道光平面，以及从摄像机中心穿过图像平面进入三维空间的光线。从几何关系图上我们可以直观地看到，成像光线与光平面交于一点。通过几何分析可得到坐标x_w, y_w, z_w。由光平面能得出一个方程，含三个未知数；由成像光线能得出二个方程，也含这三

个未知数。求解这三个线性联立方程就得到三维表面点的位置。第13章给出的标定（calibration）方法，通过在工作台上做几次测量，就能推出必要的方程。

如果照射的是单束光而不是光平面，则上述讨论更加简单。还有其他很多三角测量方法，
48 并根据具体应用来选择传感器。要扫描整个场景必须用光平面，或者用一束光扫过场景。可以用扫描镜来实现这个光能，或者用传送带系统带动目标通过光面。在文献中可以发现很多创造性设计。具有多个光平面的机器，在汽车制造时用来校正车轮以及检测车门适配情况。当观察处于特定位姿的特定目标时，图像分析可能只是用来证实一条特殊的图像条纹是否接近理想的位置。传感器拍摄的图像流，用来在线调节生产操作以进行质量控制，以及离线进行报告分析。

2.10 参考文献

有关设计成像装置方面的资料可参考Schalkoff（1989）所编教材。电荷耦合器件的指南和技术说明可通过搜索引擎在网上找到：如威斯康星大学网址`www.mrsec.wisc.edu/edetc/ccd.html`上提供的自学材料。Dillon等人（1978）的文章是彩色CCD摄像机方面较早的几篇文章之一。光学现象的讨论与建模可在Hecht和Zajac（1976）的书中找到。

引出计算机视觉技术的许多基本内容，在心理学家J.J. Gibson（1950）的著作中能够找到。Levine（1985）所编教材中，从工程的角度说明了动物视觉系统和人类视觉系统的特点。Nalwa（1993）的著作，一开始讨论了人类视觉系统的能力和缺陷，并对成像和透视变换做
49 了很好的直观性描述。Margaret Livingstone（1988）给出了面向艺术欣赏的人类感知的一个流行处理。Haralick与Shapiro（1992）第二卷中，包含关于透视变换的数学知识。关于图像文件格式的应用细节和综述，请参考Murray与VanRyper（1994）的百科全书，其中包括一张从几处收集来的常用软件工具CD盘。

1. Dillon, P., D. Lewis, and F. Kaspar. 1978. Color imaging system using a single CCD area array. *IEEE Trans. Electron Devices,* ED-25(2):102–107.
2. Gibson, J. J. 1950. *The Perception of the Visual World.* Houghton-Mifflin, Boston.
3. Hecht, E., and A. Zajac. 1974. *Optics.* Addison-Wesley, Reading, MA.
4. Haralick, R., and L. Shapiro. 1992. *Computer and Robot Vision, Volumes I and II.* Addison-Wesley, Reading, MA.
5. Levine, M. D. 1985. *Vision in Man and Machine.* McGraw-Hill, New York.
6. Livingstone, M. 1988. Art, illusion and the visual system. *Sci. Am.* (Jan. 1988) 78–85.
7. Murray, J., and W. VanRyper. 1994. *Encyclopedia of Graphics File Formats.* O'Reilly and Associates, Inc., 103 Morris St., Suite A, Sebastopol, CA 95472.
8. Nalwa, V. 1993. *A Guided Tour of Computer Vision.* Addison-Wesley, Reading, MA.
9. Schalkoff, R. J. 1989. *Digital Image Processing and Computer Vision.* John Wiley &
50 Sons, New York.

第3章　二值图像分析

在许多实际应用中，如文档分析或工业机器视觉系统，执行任务需要的算法是以二值图像为基础的。这些算法的适用范围非常广泛，从简单的目标计数到复杂的目标识别、定位及检查等。在分析灰度图像与彩色图像之前，先对二值图像分析有所了解，将有助于深入理解整个图像分析过程。

本章介绍二值机器视觉的基本算法。首先通过简单的目标计数算法让大家明白：有时只用简单的视觉算法，就可以满足实际任务的需要。接下来讨论连通成分标记运算，即对每个连通的像素集合赋以独有的标记，这一步是后面大多数处理步骤的基础。然后介绍一系列细化和粗化算子，数学形态运算可以对成分进行连接或分离，可以闭合孔和计算图像中的兴趣特征。如果几个不同的成分被分离开，每个成分的重要特征就可以算出来，从而能够进行更高级的识别与跟踪等任务。本章将对一些基本特征进行定义，并讨论计算这些特征的算法的精度。最后研究通过自动阈值处理，把灰度图像或彩色图像转化为有效二值图像这一问题。

3.1　像素与邻域

对一幅灰度图像或者彩色图像I进行处理，把其中感兴趣的像素分离出来作为前景（foreground）像素，而把不感兴趣的其余部分作为背景（background）像素，就可以得到一幅二值图像B。分离运算有简有繁，如简单的阈值运算能够分离出属于某个灰度范围或者某个颜色子空间的像素，也可以使用更复杂的分类算法。阈值运算将在本章末进行讨论，而高级的分类选择运算分布在本书的各个部分。作为本章的开始，我们约定所讨论的问题都是在二值图像B的基础上进行的。图3-1借助四幅手写字符的二值图像，对有关概念进行说明。

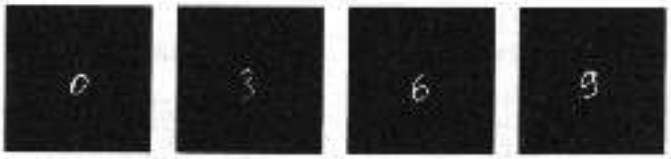


图3-1　手写字符的二值图像

二值图像B中的像素值要么是0，要么是1，其中1表示前景像素的值，0表示背景像素的值。B[r, c]表示位于图像阵列中第r行、第c列的像素的值。一幅M × N的图像具有M行和N列，行的编号从0到M − 1，列的编号从0到N − 1。这样B[0, 0]表示图像左上角的像素值，B[M − 1, N − 1]表示图像右下角的像素值。

在许多算法中，当对某个像素进行运算时，不仅要用到该像素的值，也要用到它邻近像素的值。关于邻点的定义最常见的有两种，即4-邻点（4-neighbor）和8-邻点（8-neighbor）。像素[r, c]的4-邻域N_4[r, c]包括4个像素，即[r − 1, c]、[r + 1, c]、[r, c − 1]和[r, c + 1]。这4个像素常称为北邻点、南邻点、西邻点和东邻点。像素[r, c]的8-邻域N_8[r, c]共包括8个像素，除了前面的4个像素，再加上对角线上的4个像素[r−1, c−1]、[r−1, c + 1]、[r + 1, c−1]和[r + 1, c + 1]。这4个像素点常称为西北邻点、东北邻点、西南邻点和东南邻点。图3-2注明了这些概念。

在各种算法中，邻域（neighborhood）可以是4-邻域也可以是8-邻域（或者其他定义）。一般来说，如果像素[r', c']位于像素[r, c]的某个邻域内，我们就说像素[r', c']是像素[r, c]的邻点。

	N	
W	*	E
	S	

a）4-邻域N_4

NW	N	NE
W	*	E
SW	S	SE

b）8-邻域N_8

图3-2 常用的两种像素邻域

3.2 图像模板运算

图像处理中的一个基本概念是图像模板（mask），这个概念来自图像处理中的卷积运算，但通常可用于图像分析的各个方面，模板是一组像素位置及其对应值的集合，这些对应值称为权（weight）。图3-3是三个不同的模板，前两个模板a与b是方形模板，一个具有相等的权值，即所有的权值均为1，另一个具有不等的权值。第三个模板c是一个长方形模板，各位置的权值相等。

1	1	1
1	1	1
1	1	1

a)

1	2	1
2	4	2
1	2	1

b)

1
1
1
1
1

c)

图3-3 三个不同的模板

每个模板都有一个原点（origin），一般是模板上的一个位置点。对称模板（如图3-3中的a和b）的原点常常就是它的中心像素。对于不对称模板，根据使用的目的不同可以选择任何像素作为原点。比如模板c中可以选最上面的像素作为原点。

对一幅输入图像进行模板运算后，将产生与输入图像大小一样的输出图像。把模板放在输入图像上，让模板的原点分别与输入图像的每个像素点重合，模板下面的每一个输入图像的像素值乘以模板上对应的权值，然后把结果相加产生一个输出值，这个值在输出图像上的位置与输入图像中正处理的像素位置对应。图3-4显示用图3-3中的模板b对一幅灰度图进行模板运算的情况。

40	40	80	80	80
40	40	80	80	80
40	40	80	80	80
40	40	80	80	80
40	40	80	80	80

a）原始灰度图像

1	2	1
2	4	2
1	2	1

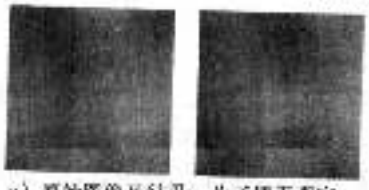
b）3×3模板

640	800	1120	1280	1280
640	800	1120	1280	1280
640	800	1120	1280	1280
640	800	1120	1280	1280
640	800	1120	1280	1280

c）模板运算结果

40	50	70	80	80
40	50	70	80	80
40	50	70	80	80
40	50	70	80	80
40	50	70	80	80

d）除以各权之和（即16）后的规范化结果



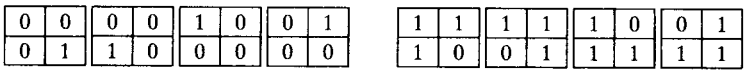
e）原始图像及结果，为了便于观察，放大到120×120

图3-4 灰度图像的加权模板运算

原始灰度图像如图3-4a所示。注意当模板中心位于图像边界上时，模板上的一些像素会处在输入图像的外面。为了使输出图像与输入图像大小一致，我们在输入图像的边界外添加虚拟的行和列。在下面的例子中，我们分别在图像的上下左右添加了两行和两列。这些虚拟行列中的像素值可任意设置为0或其他常数。这里是采用最靠近它们的行列上的像素值。因此最上面一行的像素值是40、40、80、80、80，最左边一列的像素值都是40，最右边一列的像素值都是80，最下面一行的像素值是40、40、80、80、80。应用模板b产生的输出图像c是对输入图像a进行平滑处理后的结果。可以看出结果中所有的像素值都比原图的像素值大得多。为了规范化，把结果中的每个像素值除以模板中的各权之和。本例中各权之和为16，规范化处理后得到图像d。把原始灰度图像与结果灰度图像显示在e中，为便于观察，图像放大为120 × 120。由于 53
放大作用，一个像素在结果图像中对应为宽24像素的带状，因此平滑效果表现在带级而不是像素级。

3.3 目标计数

在第1章的应用实例中，我们知道对交叉支撑杆上面螺栓孔的数量进行统计是非常必要的。对图像前景中目标个数的统计，与统计螺栓孔数量的问题差不多，可以用同样的算法，只是把两组模板**E**和**I**的角色互换一下。统计前景中目标的数量时，外角模式是具有三个0值像素和一个1值像素的2 × 2模板。内角模式是具有三个1值像素和一个0值像素的2 × 2模板。图3-5显示了这两组模板。注意该算法要求每个目标是4-连通的1值像素的集合，并且内部没有0值孔。 54



a) **E**: 外角　　b) **I**: 内角

图3-5 2 × 2模板，用来统计二值图像中前景目标的个数。
1表示前景像素的值，0表示背景像素的值

用这些模板对二值图像进行处理，可形象化地认为是把模板放在图像上，使模板的左上角像素与图像中被考虑的像素重合。这时模板确定了图像像素的一个邻域，该邻域由被考虑的像素、其右边的像素和下边的两个像素组成。如果图像上对应位置的四个像素值正好与模板上的像素值一致，则模板的角类型就是图像上该点像素所对应的角类型。函数*external_match*(**L**, **P**)依次采用四个外角模板进行计算，如果包含左上角像素[**L**, **P**]的子图像与某一个外角模板相匹配，则函数返回值为真，否则函数返回值为假。同样地，如果包含左上角像素[**L**, **P**]的子图像与某一个内角模板相匹配，则函数*internal_match*(**L**, **P**) 返回值为真，否则返回值为假。除了二值图像**B**中的最后一行和最后一列，目标计数函数*count_objects*(**B**) 对其他位置的每个像素都循环计算一次，并返回图像中目标物体的个数。图像**B**中最后一行和最后一列的像素，不能用这样的2 × 2模板。

算法约定 算法3.1是统计目标数量的伪代码程序。全书所有的程序都使用这种句法结构。注意我们把所有的例程都称为*过程*（procedure），通过*return*语句返回值（类似C语言）的过程称为函数。为了保证程序尽量简短，采用功能函数，如*external_match*和*internal_match*。类似这样的功能函数非常直观明了，其代码在书中就省略了。我们也省略了与语言相关的类型声明，但书中会对类型要求进行详述，并在注释行对重要的变量进行解释。最后，程序使用 55
全局常量，以避免参数传递所带来的麻烦。

在目标计数程序中，常数MaxRow是图像最后一行的编号，而MaxCol是图像最后一列的编号。第一行和第一列的编号都是0，这是C语言中数组的缺省设置。

算法3.1 计算二值图像B中的前景目标的数量

目标是4-连通的而且是单连通。
E是外角的数目。
I是内角的数目。

```
procedure count_objects(B);
|
E :=0;
I :=0;
for L :=0 to MaxRow-1
   for P :=0 to MaxCol-1
      {
      if external_match(L, P) then E := E + 1;
      if internal_match(L, P) then I := I + 1;
      };
return((E - I)/4);
|
```

习题3.1 计数效率问题

过程*count_objects*对图像中的每个像素都计算一遍，最多需要计算多少次？如何编写*external_match*与*internal_match*的代码，使程序的效率尽可能地高？

习题3.2 驾车问题

用坐标纸表示像素阵列，把一些相连方块涂黑（一开始范围要小一点儿）。涂黑的方块对应前景像素，未涂的方块对应背景像素。把方块看成是城市街区，你正以顺时针方向绕黑色区域驾车行驶。你右转弯时对应的是E角还是I角？左转弯时情况又如何？驾车转过完整的一周，左转弯的数目与右转弯的数目之间有关系吗？如果有，是什么关系？行驶完一周，你通过或者接触先前经过的交叉点吗？这种情况可能吗？为什么？在回答之前，先考虑只有两个黑色方块的情况，它们沿对角线方向接触，有一个交叉点。你的左右计数规则还起作用吗？目标计数的公式还起作用吗？

3.4 连通成分标记

假设**B**是一幅二值图像，而且$\mathbf{B}[r, c] = \mathbf{B}[r', c'] = v$，其中$v = 0$或者$v = 1$。如果存在一个像素序列$[r, c] = [r_0, c_0], [r_1, c_1], \ldots, [r_n, c_n] = [r', c']$，其中$\mathbf{B}[r_i, c_i] = v$，$i = 0, \ldots, n$，并且对任何$i = 1, \ldots, n$，$[r_i, c_i]$与$[r_{i-1}, c_{i-1}]$都是相邻的，则像素$[r, c]$与像素$[r', c']$通过值$v$连在一起。像素序列$[r_0, c_0], \ldots, [r_n, c_n]$就形成了从$[r, c]$到$[r', c']$的连接路径（path），一个值为$v$的连通成分，即值为$v$的像素集合**C**，集合中的每一对像素都通过值$v$相连接。图3-6a是一幅二值图像，内有值为1的五个连通成分，实际上这些成分是8-邻域连通或者是4-邻域连通。

定义14 连通成分标记（connected components labeling）对二值图像B做标记，生成标号图像**LB**，标号图像中每个像素的值就是像素所在连通成分的标号。

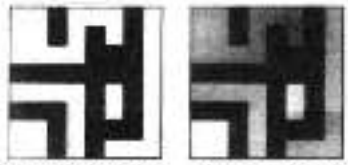
标号是专门命名一个实体所用的符号。虽然可以用字符标记，但正整数用起来更加方便，因此常常用正整数标记连通成分。图3-6b显示的是连通成分标记，是对图3-6a的二值图像进行标记的结果。

1	1	0	1	1	1	0	1
1	1	0	1	0	1	0	1
1	1	1	1	0	0	0	1
0	0	0	0	0	0	0	1
1	1	1	1	0	1	0	1
0	0	0	1	0	1	0	1
1	1	0	1	0	0	0	1
1	1	0	1	0	1	1	1

a）二值图像

1	1	0	1	1	1	0	2
1	1	0	1	0	1	0	2
1	1	1	1	0	0	0	2
0	0	0	0	0	0	0	2
3	3	3	3	0	4	0	2
0	0	0	3	0	4	0	2
5	5	0	3	0	0	0	2
5	5	0	3	0	2	2	2

b）连通成分标记



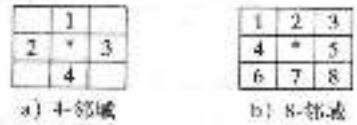
c）二值图像与标号图像，为便于观察进行了放大

图3-6 内含五个值为1的连通成分的二值图像

连通成分标记有多种不同的算法。一些算法假设内存能够载入整幅图像，使用简单的递归算法每次处理一个成分，可对整幅图像进行扫描。有的算法针对较大的图像，由于内存有限不能载入整幅图像，算法每次只处理图像中的两行。还有其他一些算法适合在大型并行机上使用，采用并行传播策略。本章我们讨论两种不同的算法：递归搜索算法和逐行算法。逐行算法用特殊的并查数据结构来跟踪成分。

3.4.1 递归标记算法

假设**B**是**MaxRow**+1行，**MaxCol**+1列的二值图像。我们希望找到像素值为1的连通成分，并输出标号图像**LB**，在标号图像中每个像素的值就是连通成分的标号。参考Tanimoto所著的《Artificial Intelligence》，算法策略是：首先把二值图像的像素值取负，使原来值为1的像素变成值为-1。这样就可以把未处理的像素（值为-1）与成分标记1分开。由函数*negate*实现这一功能，输入的是二值图像**B**，输出的是取负后的图像，这个图像最后成为标号图像**LB**。寻找连通成分的过程变成了寻找**LB**中值为-1的像素的过程，把找到的像素赋以一个新的标号，并调用过程*search*去寻找值为-1的邻点，并对这些邻点递归地重复执行这个过程。效用函数*neighbors*(**L**, **P**) 中的**L**和**P**确定像素的位置，该函数返回所有邻接像素的位置，可以是4-邻域，也可以是8-邻域，只返回二值图像中合法的邻点位置。函数返回邻点的顺序与扫描顺序一致，如图3-7所示。递归连通成分标记算法包括六个程序块，其中*negate*，*print*和*neighbors*需要读者编写代码。



a）4-邻域　　b）8-邻域

图3-7 像素邻点扫描顺序

图3-8以二值图3-6的第一个成分（左上角区域）为例，显示递归连通标记算法的运行过程。

Step 1.

−1	−1	0	−1	−1	−1
−1	−1	0	−1	0	0
−1	−1	−1	−1	0	0

Step 2.

1	**−1**	0	−1	−1	−1
−1	−1	0	−1	0	0
−1	−1	−1	−1	0	0

Step 3.

1	1	0	−1	−1	−1
−1	−1	0	−1	0	0
−1	−1	−1	−1	0	0

Step 4.

1	1	0	−1	−1	−1
1	**−1**	0	−1	0	0
−1	−1	−1	−1	0	0

Step 5.

1	1	0	−1	−1	−1
1	1	0	−1	0	0
−1	−1	−1	−1	0	0

图3-8 显示递归标记算法的前5步，对二值图3-6中的第一个成分进行搜索。显示的图像是（部分）标记了的图像LB。图中的粗体像素是搜索程序正处理的像素。按图3-7所示的邻域搜索顺序，在每一步，选择粗体像素的邻点中首次遇到的未处理的邻点（即值为−1），作为下一步要处理的像素

算法3.2 计算二值图像中的连通成分

B是原始二值图像。

LB是连通成分标号图像。

```
procedure recursive_connected_components(B, LB);
{
LB := negate(B);
label :=0;
find_components(LB, label);
print(LB);
}
procedure find_components(LB, label);
{
for L :=0 to MaxRow
   for P :=0 to MaxCol
      if LB[L, P] == −1 then
         {
         label := label + 1;
         search(LB, label, L, P);
         }
}
procedure search(LB, label, L, P);
{
LB[L, P] := label;
Nset := neighbors(L, P);
for each[L′, P′] in Nset
   {
   if LB[L′, P′] == −1
   then search(LB, label, L′, P′);
   }
}
```

3.4.2 逐行标记算法

经典算法是由Rosenfeld和Pfaltz于1966年提出的，称之为经典算法是因为它以经典的图连通成分算法为基础。这种算法需要扫描图像两次：一次是记录等价对并赋予一个临时标号，第二次是用等价类的标号代替每个临时标号。在这两次之间，记录的等价对集合以二元关系进行存储，对等价集合进行处理从而确定二元关系的等价类。从那时起，*并查*（union-find）算法，即随着找到等价对而动态地构造等价类的算法，被广泛应用于计算机科学中。并查数据结构能够有效地构造和操作用树结构表示的等价类，增加这一数据结构使经典算法的性能得到了提高。

1. 并查结构

并查数据结构的目的是为了把不相交的集合储存在一起，以及为了有效地实现合并（union，即把两个集合合并为一个）运算及查找（find，确定特殊元素所在的集合）运算。每个集合存储成树形结构，树的节点代表一个标号，并指向它的父节点。实现这个结构只需要一个向量数组**PARENT**，其下标就是标号，元素的值是父节点的标号。父节点的值为零意味着这个节点是树的根节点。图3-9是两组标号{1, 2, 3, 4, 8}和{5, 6, 7}的树形结构。标号3是父节点，并作为第一个集合的标号。标号7是另一个父节点，并作为第二个集合的标号。数组**PARENT**中的元素值告诉我们节点3与节点7没有父节点，标号2是标号1的父节点，标号3是标号2、4和8的父节点，等等。注意数组中没有标号为0的元素，因为0表示背景像素，而且数组中元素的值为0意味着这个节点没有父节点。

PARENT

1	2	3	4	5	6	7	8
2	3	0	3	7	7	0	3

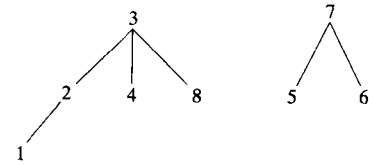


图3-9 两组标号的并查数据结构。第一组包括标号{1, 2, 3, 4, 8}，第二组包括标号{5, 6, 7}。对每一个整数标号**i**，**PARENT[i]**的值是**i**的父节点的标号；如果**i**是一个根节点，没有父节点，则**PARENT[i]**的值是0

*find*过程所带的参数是标号**X**和父数组**PARENT**。该过程只是沿树向上跟踪父指针，查找标号**X**所在树的根节点的标号。*Union*过程所带的参数是标号**X**、标号**Y**和父数组**PARENT**。该过程对结构进行修改（如果有必要），合并含**X**的集合和含**Y**的集合。从标号**X**和标号**Y**开始，沿树向上跟踪父指针，直到找到两集合的根节点为止。如果两个根节点不一样，则把其中一个标号作为另一个标号的父节点。下面的合并程序中把**X**作为**Y**的父节点。根据集合的规模，把较小的集合附加到较大集合的根部也是可以的，而且可以保持树的深度向下伸展。

算法3.3 查找集合中的父节点标号

X是集合的标号。

PARENT是包含并查数据结构的数组。

```
procedure find(X, PARENT);
{
j := X;
while PARENT[j]<>0
  j := PARENT[j];
return(j);
}
```

算法3.4 合并两个集合

X是第一个集合的标号。

Y是第二个集合的标号。

PARENT是包含并查数据结构的数组。

```
procedure union(X, Y, PARENT);
{
j := X;
k := Y;
while PARENT[j] <> 0
   j :=PARENT[j];
while PARENT[k] <> 0
   k :=PARENT[k];
if j<>k then PARENT[k] :=j;
}
```

2. 具有并查结构的经典连通成分标记算法

并查数据结构使经典的连通成分标记算法更加高效。算法的第一次扫描执行标号传播，把像素标号传播到右下方向的邻点。当出现两个不同的标号传播到同一个像素的情况时，就传播较小的标号，每当发现这样的等价对就进入并查结构。第一次扫描结束后，已完全确定每个等价类，而且每个等价类有惟一标号，也就是并查结构中树的根节点。第二次扫描图像时，进行变换，把等价类的标号赋给每个像素。

程序用到两个附加的功能函数：*prior_neighbors* 和*labels*。*prior_neighbors*函数返回上边及左边的1值像素的集合，程序代码可以针对4-邻域（返回北、西邻点）或者针对8-邻域（返回西北、北、东北和西邻点）。*labels*函数返回赋给已知像素集合的当前标号集合。

图3-10以二值图像图3-6为例，显示具有并查结构的经典算法的应用。图3-10a显示第一次扫描图像后各像素对应的标号。图3-10b是等价类的并查数据结构，显示出第一次扫描后确定的等价类是{{1, 2}, {3, 7}, 4, 5, 6}。图3-10c表示第二次扫描之后的图像标号。连通成分代表
61 图像中的区域，其形状及亮度特征可以计算出来。我们将在3.5节讨论这些特征。

算法3.5 经典连通成分数据结构的初始化

```
procedure initialize();
\\初始化全局变量label和数组PARENT。
{
\\初始化label。
labelc := 0;
\\初始化并查结构。
for i := 1 to MaxLab
PARENT[i] := 0;
}
```

1	1	0	2	2	2	0	3
1	1	0	2	0	2	0	3
1	1	1	1	0	0	0	3
0	0	0	0	0	0	0	3
4	4	4	4	0	5	0	3
0	0	0	4	0	5	0	3
6	6	0	4	0	0	0	3
6	6	0	4	0	7	7	3

a）第一次扫描后的结果

PARENT

1	2	3	4	5	6	7
0	1	0	0	0	0	3

b）等价类的并查结构

1	1	0	1	1	1	0	3
1	1	0	1	0	1	0	3
1	1	1	1	0	0	0	3
0	0	0	0	0	0	0	3
4	4	4	4	0	5	0	3
0	0	0	4	0	5	0	3
6	6	0	4	0	0	0	3
6	6	0	4	0	3	3	3

c）第二次扫描后的结果

图3-10 对应二值图像图3-6，使用具有并查数据结构的经典标记算法

算法3.6 计算具有并查结构的二值图像的连通成分

B是原始二值图像。

LB是连通成分标号图像。

```
procedure classical_with_union-find(B, LB);
{
\\初始化结构。
initialize();
\\第一次: 为图像的每一行L赋初始标号。
for L := 0 to MaxRow
  {
  \\ L行的所有值初始化为0。
  for P := 0 to MaxCol
    LB[L, P] := 0;
  \\处理L行。
  for P := 0 to MaxCol
    if B[L, P] == 1 then
      {
      A := prior_neighbors(L, P);
      if isempty(A)
      then {M := label; label := label+1;};
      else M :=min(labels(A));
      LB[L,P] :=M;
      for X in labels(A) and X<>M
      union(M, X, PARENT);
      }
      }
\\第二次: 用等价类的标号代替第一次的标号。
```

```
    for L := 0 to MaxRow
      for P := 0 to MaxCol
        if B[L, P] == 1
        then LB[L, P] := find(LB[L, P], PARENT);
    };
```

3. 游程编码连通成分标记

第2章已经讲过，二值图像的游程编码（Run-Length Encoding）是像素值连续为1的水平游程的列表，对每一个游程，必须记录它起始像素的位置，它的长度或者结束像素的位置。参考图3-11所示的游程数据结构的例子。图像中的每一个游程，用它的起始和结束像素位置进行编码。（**ROW, START_COL**）是起始像素的位置，（**ROW, END_COL**）是结束像素的位置，**LABEL**字段中存储着本次游程所属的连通成分的标号，开始时**LABEL**字段被初始化为0，第一次扫描时赋以**LABEL**字段临时值，第二次扫描结束，**LABEL**字段中包含最终的、永久的游程标号。然后通过这种结构输出标号结果，结果显示在输出图像的
62 对应像素位置上。

	0	1	2	3	4
0	1	1	0	1	1
1	1	1	0	0	1
2	1	1	1	0	1
3	0	0	0	0	0
4	0	1	1	1	1

a)

	ROW_START	ROW_END
0	1	2
1	3	4
2	5	6
3	0	0
4	7	7

b)

	ROW	START_COL	END_COL	LABEL
1	0	0	1	0
2	0	3	4	0
3	1	0	1	0
4	1	4	4	0
5	2	0	2	0
6	2	4	4	0
7	4	1	4	0

c)

图3-11 二值图像a及其游程编码b和c。像素值为1的游程通过所在的行（**ROW**）以及起始点、终止点所在的列（**START_COL**和**END_COL**）进行编码。另外对图像的每一行，**ROW_START**指向本行的第一个游程，**ROW_END**指向本行的最后一个游程。字段**LABEL**最初的值是0，最后存储的是游程的成分标号

习题3.3 标记算法比较

假设一幅二值图像具有一个前景区域，是1000 × 1000的正方形区域。递归算法需要访问（读或写）每个像素多少次？经典算法需要访问每个像素多少次？

习题3.4 再标记

由于等价标号被合并到一个等价类，一些在第一次得到的初始标号在第二次中就会丢失，

结果使最终的标号数字序列存在许多间隔，而不是连续的。编写再标记程序，把结果转换成从1到图像成分个数的连续序列。

习题3.5 游程编码

设计并实现逐行标记算法，要求利用二值图像的游程编码而不是图像本身，用结构的LABEL字段储存游程的标号。

3.5 二值图像形态学

形态学（morphology）这一名词涉及到形状与结构，在计算机视觉中可用来计算区域的形状。数学形态学（mathematical morphology）的运算最初是集合的运算，二维图像点的集合可通过形态运算进行处理。本节对二值形态运算进行定义，并说明如何用这些算法处理经连通成分标记后的区域。

3.5.1 结构元

二值形态运算的对象是二值图像B和结构元（structuring element）S，结构元一般是一幅很小的二值图像，结构元代表一种形状，其大小和结构可以是任意的，并能通过二值图像表示出来。有一些通用的结构元，如一定维数的长方形[BOX(l, w)]，或者一定直径的圆形区域[DISK(d)]。有的图像处理软件包中提供基本的结构元库。图3-12 显示的是一些通用的结构元和几个非标准的结构元。

63

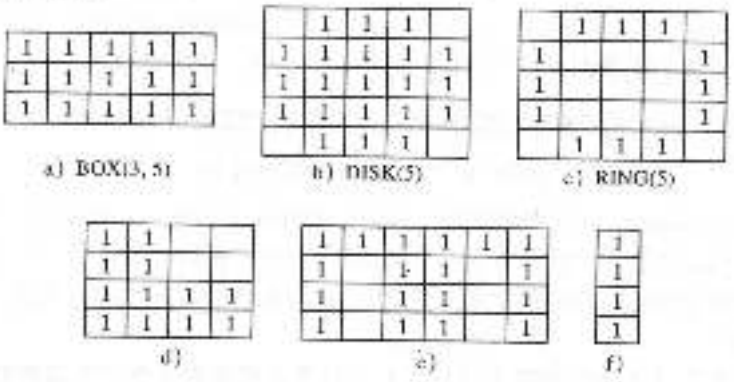


图3-12 结构元示例（空白处的值为0）

结构元能够充当二值图像的探针，结构元上的某个像素作为结构元的原点（origin），对称的结构元一般选中心像素作为它的原点，但原则上可以选任何像素作为原点。把原点作为参考点，在图像上面任意移动结构元，就可以通过结构元的形状使一个区域变大，或者检查这个形状是否能填入一个区域。例如，我们想检查一个孔的大小，就可以通过试探一个小的圆盘是否能够完全填入某个区域，而稍大的圆盘则不能填入。

64

3.5.2 基本运算

基本的二值形态运算有四种：膨胀（dilation）、腐蚀（erosion）、闭合（closing）与开启（opening）。顾名思义，膨胀运算使区域扩大，而腐蚀运算使区域变小。闭运算可以填充区域内的小孔和消除沿边界的缺口，开运算可以去掉区域边界处由里向外的毛刺。数学定义如下：

65

定义15 平移：像素集X通过位置向量z进行的平移$\mathbf{X}_z$，定义如下：

$$X_t = \{x + t \mid x \in X\} \tag{3-1}$$

这样对于二值图像中值为1的像素集，其平移是按规定的量整体移动。平移量t用有序数对$(\Delta r, \Delta c)$确定，其中Δr是行方向的移动量，Δc是列方向的移动量。

定义16 膨胀：用结构元S对二值图像B进行的膨胀运算表示为$B \oplus S$，定义如下：

$$B \oplus S = \bigcup_{b \in B} S_b \tag{3-2}$$

这种合并运算可以认为是一种邻域算子。用结构元S扫过整幅图像，输出图像的像素值初始化为0，一旦结构元的原点每次遇到二值图像中值为1的像素时，结构元整体形状就与输出图像进行逻辑"或"运算。图3-13a是二值图像，图3-13b是3×3方形结构元，图3-13c是经图13-13b方形结构元膨胀后的结果。

为了理解数学定义，考虑二值图像B中的第一个值为1的像素，它的坐标是[1,0]，表示位于图像的第1行、第0列。平移$S_{(1,0)}$的意思是，将结构元S的原点（即中心）与二值图像上的点[1,0]重合，然后把结构元的每一点与输出图像对应点进行逻辑"或"运算。"或"运算的结果是，输出图像（其像素初值为0）在实际点[0, 0]、[0, 1]、[1, 0]、[1, 1]、[2, 0]、[2, 1]处的像素值为1，在点[0, −1]、[1, −1]、[2, −1]处的像素值也为1，但这几个位置实际是不存在的，所以要忽略掉。对于图像B的下一个像素[1, 1]，平移$S_{(1,1)}$就是将结构元S的原点与二值图像上的点[1, 1]重合，再把结构元的每点与图像中的对应点进行"或"运算，输出图像在位置[0, 0]、[0, 1]、[0, 2]、[1, 0]、[1, 1]、[1, 2]、[2, 0]、[2, 1]、[2, 2]处的像素值为1。这个过程继续进行，直到对输入图像的每个像素都进行了逻辑"或"运算为止，最后结果显示在图3-13c中。

定义17 腐蚀：用结构元S对二值图像B进行的腐蚀运算表示为$B \ominus S$，定义如下：

$$B \ominus S = \{b \mid b + s \in B \ \forall s \in S\} \tag{3-3}$$

腐蚀运算也要用结构元扫过整幅图像。针对二值图像上的每一个像素点，如果结构元上每一个值为1的像素都覆盖着二值图像上一个值为1的像素，则将二值图像上与结构元原点对应的像素与输出图像对应点进行逻辑"或"运算。3-13d是经3×3方形结构元对二值图像3-13a进行腐蚀运算的结果。

膨胀与腐蚀是最基本的数学形态运算，对它们进行组合就产生另外两种常用的运算：闭运算和开运算。

定义18 闭合：用结构元S对二值图像B进行的闭运算表示为$B \bullet S$，定义如下：

$$B \bullet S = (B \oplus S) \ominus S \tag{3-4}$$

定义19 开启：用结构元S对二值图像B进行的开运算表示为$B \circ S$，定义如下：

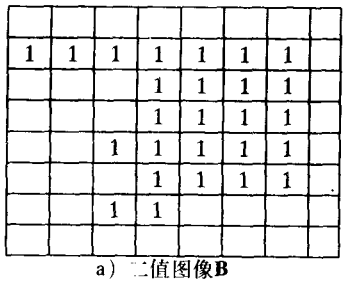
$$B \circ S = (B \ominus S) \oplus S \tag{3-5}$$

图3-13e是经3×3方形结构元对二值图像3-13a进行闭运算的结果，3-13f是用同样结构元对二值图像进行开运算的结果。

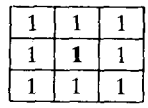
习题3.6 使用基本的二值形态运算

摄像机拍了一幅图像I，图中有1分、1角和25美分的硬币各一枚，互相不接触，背景为白色。用阈值运算建立二值图像B，硬币区域像素值为1，背景像素值为0。已知硬币的直径是

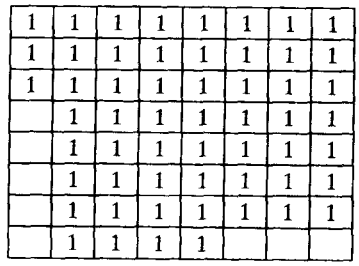
D_P、D_D和D_Q。用数学形态运算（膨胀、腐蚀、开启和闭合）以及逻辑运算AND、OR、NOT和MINUS（求差），看看如何产生三幅二值图像P、D和Q。P中应该只有1分的硬币（像素值为1），D中应该只有1角的硬币，Q中应该只有25美分的硬币。



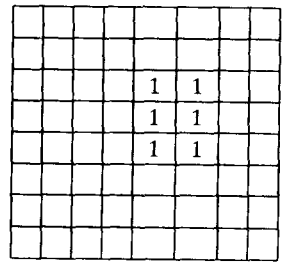
a）二值图像**B**



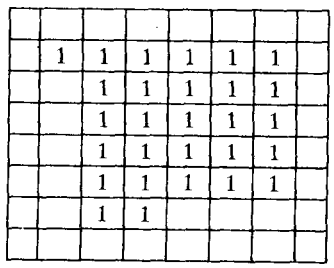
b）结构元**S**



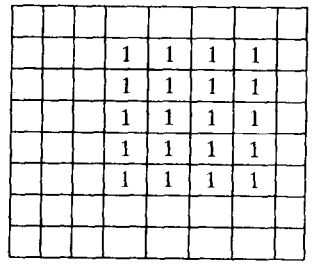
c）膨胀运算**B⊕S**



d）腐蚀运算**B⊖S**



e）闭运算**B•S**



f）开运算**B∘S**

图3-13 基本的二值形态运算。前景像素值为1，背景像素值为0（图中空白处）

3.5.3 二值形态学的应用

经阈值化或者其他方法预处理后的图像，如果连通成分内部有小孔，或者应当分开的一对成分被前景像素构成的细小区域连接，这时就可通过闭运算和开运算解决问题。图3-14a是一幅512 × 512的16位灰度医学图像；图3-14b是阈值处理后的结果，所用的阈值为1070；图3-14c是形态运算结果，通过开运算把不同的组织分开，又通过闭运算去掉小孔。开运算中用的结构元是DISK(13)，闭运算中用的结构元是DISK(2)。

在工业机器视觉中，也可用形态运算完成特殊的检查任务。Sternberg于1985年通过形态运算检查手表的齿轮是否有缺损或断齿现象。图3-15a是手表齿轮的二值图像。齿轮主体上有四个圆孔，边缘是轮齿，每个轮齿在图像中清晰可见。为了对手表齿轮图像进行处理，Sternberg 定义了几个专用的结构元，其形状和大小从齿轮的物理特性得出。在手表齿轮检查算法中，使用的结构元如下:



a）医学图像G　　b）阈值处理后的图像B　　c）形态运算结果

图3-14 形态运算在医学图像中的应用。a中显示的是512×512的16位医学图像。经阈值（阈值为1070）处理后产生了二值图像b，通过结构元DISK（13）进行开运算，再通过结构元DISK（2）进行闭运算，产生出结果c

- **hole_ring**：像素环，其直径比手表齿轮上四个圆孔的直径稍微大一点，它正好包围这些圆孔，可用来标出圆孔中心位置的几个像素。
- **hole_mask**：八边形结构元，比手表齿轮上的圆孔稍微大一点。
- **gear_body**：圆盘形结构元，大小等于齿轮去掉轮齿所剩余的部分。
- **sampling_ring_spacer**：圆盘形结构元，可把齿轮体稍微向外扩大一点。
- **sampling_ring_width**：圆盘形结构元，可把齿轮体向外扩大到齿尖部分。
- **tip_spacing**：圆盘形结构元，直径等于齿尖轮廓的直径。
- **defect_cue**：圆盘形结构元，用于扩大瑕疵以便于观察。

图3-15显示的是轮齿瑕疵检查过程。图3-15a是要检查的原始二值图像。图3-15b是用**hole_ring**结构元对原图进行腐蚀后的结果。在每个圆孔的中心生成几个像素的聚集点，其聚集像素的值为1。只有能被**hole_ring**结构元完全覆盖的目标区域，才能产生这样的图像。图3-15c是用结构元**hole_mask**对该图像进行膨胀运算后的效果，即四个八边形代替了原来位置上的四个圆孔。图3-15d是将4个八边形与原始二值图像进行逻辑“或”运算的结果，齿轮上的四个孔被填满。

下一步是生成取样圆环，用来检查轮齿。通过结构元**gear_body**对图3-15d进行开运算除去轮齿，然后用结构元**sampling_ring_spacer**进行膨胀使之达到齿根部，再通过结构元**sampling_ring_width**膨胀图像到齿尖部，最后一次的结果减去第二次的结果就得到一个圆环，圆环正好覆盖轮齿部分。采样环见图3-15e。

一旦有了采样环，把它与原始图像进行逻辑“与”运算，生成只有轮齿的图像，如图3-15f。这时已经能够看到轮齿之间有间隙，但还没有标记出来。通过结构元**tip_spacing**对轮齿图像进行膨胀，生成实线圆环图像即图3-15g。其中在轮齿有缺损的地方，实线环上有缺口。用采样环减去这个实线环得到只有缺口的图像，再用结构元**defect_cue**进行膨胀处理，在屏上显示出用户可以观察到的足够大的斑点。

习题3.7 结构元选择

Sternberg用环形结构元检测齿轮上圆孔的中心，如果你的系统只有圆盘形和方形结构元，你怎么来检测这些圆孔的中心？

习题3.8 形态运算的应用

假设对一幅卫星图像进行阈值处理，有水的区域像素值为1。但是，河流上面的桥对应的

是像素值为0的细线。它们把河流分成不同的区域。(a) 如何把表示桥的像素区域恢复成水区域？ (b) 作为独立的目标物体，如何检测这些细线形的桥？

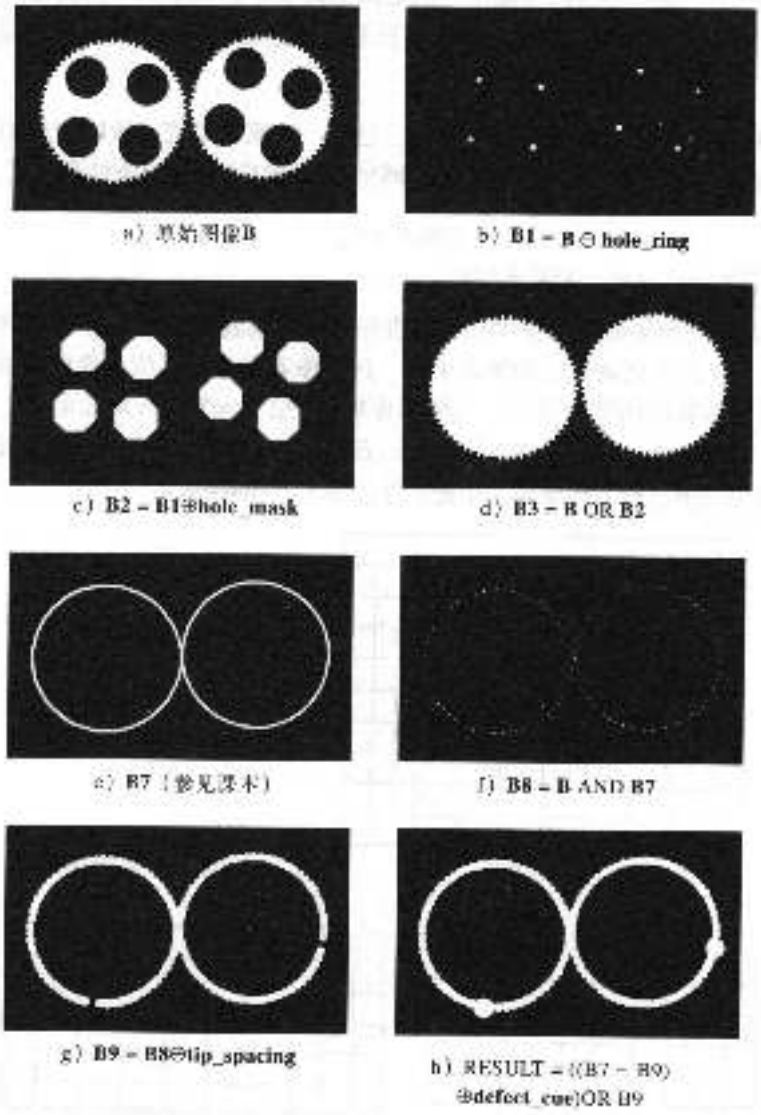


图3-15 齿轮检查过程（经Academic Press允许，由Stanley R. Sternberg授权）

二值图像形态运算，也可用来抽取目标的基元特征，这些基元特征可用于目标识别，例如，二维物体的角在形状识别时是很好的基元特征。如果带尖角的目标用圆盘形结构元进行开运算，这些角会被切掉，如图3-16所示。如果用原始二值图像减去开运算的结果，将只有角被保留下来，并用于结构识别算法中。形状匹配系统中，可以用形态特征检测方法快速检测用于目标识别的基元特征。



图3-16 用形态运算抽取形状基元特征

3.5.4 条件膨胀

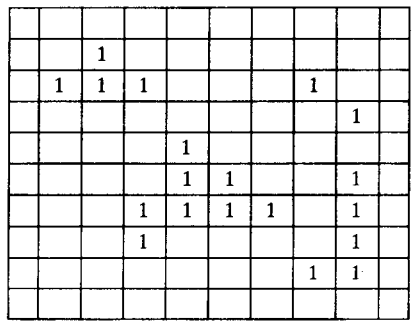
通过二值形态运算，可以确定二值图像中满足一定形状和大小约束的组成成分。我们能
71 够推导出一个结构元，通过它去掉图像中不满足约束的成分，只保留一些满足约束的值为1的像素。但实际上我们想要的是整体成分，而不只是腐蚀之后保留下来的几个像素。条件膨胀运算可以解决这个问题。

定义20 条件膨胀（conditional dilation）：已知一幅原始二值图像**B**，处理过的二值图像**C**，及结构元**S**，设$\mathbf{C}_0=\mathbf{C}$，$\mathbf{C}_n=(\mathbf{C}_{n-1}\oplus\mathbf{S})\cap\mathbf{B}$。**S**对**C**关于**B**的条件膨胀定义为：

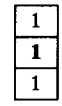
$$C\oplus|_B S = C_m \tag{3-6}$$

其中下标m是满足$C_m = C_{m-1}$的最小下标。

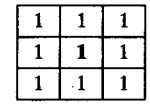
这个定义是为了对有限数字图像中的点集进行分离。也就是说，用结构元**S**对集合$C=C_0$进行多次膨胀，每一次都得到减少了的像素子集，这些像素在原始二值图像中的值为1。图3-17
72 显示的是条件膨胀运算的结果。图中，二值图像**B**通过结构元**V**进行腐蚀运算，选择出包含3个像素长的垂直边的成分。选出成分中的两个，表示在结果图像**C**中。为了看到这两个的整体成分，**D**对**C**关于原图**B**进行条件膨胀，于是产生了图3-17e的结果。



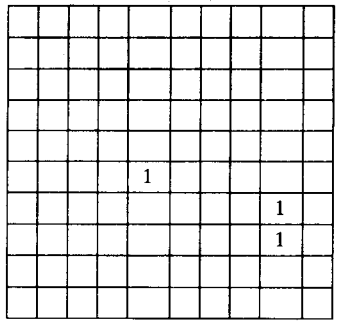
a）二值图像**B**



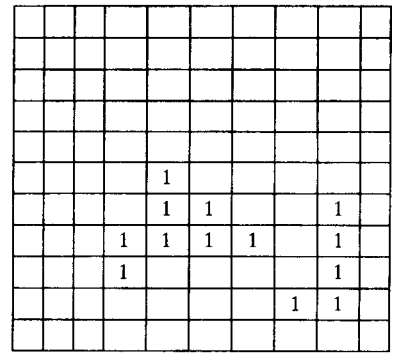
b）结构元**V**



c）结构元**D**



d）$\mathbf{C}=\mathbf{B}\ominus\mathbf{V}$



e）$\mathbf{C}\oplus|_{\mathbf{B}}\mathbf{D}$

图3-17 条件膨胀运算

3.6 区域特征

一旦区分出了区域，区域特征就变成执行决策任务如识别、检查等高级过程的输入。许多图像处理包中具有计算区域特征集的算子。特征一般包括几何特征（如区域面积、中心和

端点）、形状特征（如环形、长条形的测度值）、亮度特征（如平均灰度）以及各种纹理统计特征等。本节定义最常用的几何和形状特征，并解释如何用它们进行决策。第7章图像纹理部分包含了灰度特征。

在下面的讨论中，我们把区域中的像素集表示为**R**。最简单的几何特征是区域面积**A**和区域的中心$(\bar{r}, \bar{c})$，假设像素形状是正方形的，这些特征定义如下：

面积：

$$A = \sum_{(r,c)\in R} 1 \tag{3-7}$$

这意味着面积只是区域**R**中的像素的个数。

中心：

$$\bar{r} = \frac{1}{A} \sum_{(r,c)\in R} r \tag{3-8}$$

$$\bar{c} = \frac{1}{A} \sum_{(r,c)\in R} c \tag{3-9}$$

中心$(\bar{r}, \bar{c})$即区域**R**中像素的平均位置。注意即使每个像素坐标$[r, c]\in R$是一对整数，$(\bar{r}, \bar{c})$一般也不是一对整数。对于中心位置来说，精度取1/10像素是比较合适的。

习题3.9 区域特征应用

前面提到的齿轮例子，只用到形态运算和逻辑运算。这些运算只有在专用机器上才能快速执行。假设我们要寻找齿与齿之间超过正常值的大缝隙，由于在通用机上形态运算的执行速度很慢，那么怎样做才能使执行检测时用到的形态运算最少？

区域周界（perimeter）**P**的长度是另一个全局性特征。内部无孔区域的周界，简单定义为它内部边界像素的集合。一个区域的像素如果具有该区域外的邻点，则这个像素是一个边界像素。如果基于8-连通来判断区域内像素是否与区域外的像素连接，周界像素的集合就是4-连通的。如果基于4-连通性来判断区域内像素是否与区域外的像素连接，周界像素的集合就是8-连通的。这样就出现关于区域**R**周界的两种定义方式：4-连通周界P_4和8-连通周界P_8。 73

周界：

$$P_4 = \{(r,c)\in R \mid N_8(r,c) - R \neq \phi\}$$
$$P_8 = \{(r,c)\in R \mid N_4(r,c) - R \neq \phi\}$$

习题3.10 根据周界产生区域

只知道区域的周界，试设计算法，生成无孔区域的二值图像。

习题3.11 根据周界计算面积

只知道区域的周界，试设计算法，计算无孔区域的面积。如果不重构二值图像，有可能计算出区域的面积吗？

为了算出周界**P**的长度$|P|$，**P**中的像素必须按顺序排成一个序列$P=<(r_0, c_0), \cdots, (r_{L-1}, c_{L-1})>$，序列中前后两个像素是相邻的，包括第一个像素和最后一个像素在内。那么周长（perimeter length）$|P|$定义为：

周长：

$$|P| = |\{k \mid (r_{k+1}, c_{k+1}) \in N_4(r_k, c_k)\}| + \sqrt{2}|\{k \mid (r_{k+1}, c_{k+1}) \in N_8(r_k, c_k) - N_4(r_k, c_k)\}| \tag{3-10}$$

其中计算$k+1$的模是K，即像素序列的长度。这样在周界的竖直和水平方向的两相邻像素使总数加1，而对角线上的两相邻像素使总数加1.4左右。

有了区域面积A和周界P，区域圆度的一般度量方法是用周长的平方除以面积。

圆度（1）：

$$C_1 = \frac{|P|^2}{A} \tag{3-11}$$

对于数字形状，$|P|^2/A$的最小值不适合数字化圆形，虽然它适合连续的平面形状。对于数字八边形或菱形，不管是按4-连通边界像素数计算还是按边界长度计算，如果是竖直或水平方向移动，结果要加1；如果是沿对角线方向移动，结果要加$\sqrt{2}$。为了解决这个问题，Haralick于1974年提出了另一种圆度的度量方法。

圆度（2）：

74

$$C_2 = \frac{\mu_R}{\sigma_R} \tag{3-12}$$

其中μ_R和σ_R分别为形状的中心到边界距离的均值和标准差，计算公式如下：

平均径向距离：

$$\mu_R = \frac{1}{K}\sum_{k=0}^{K-1} \|(r_k, c_k) - (\bar{r}, \bar{c})\| \tag{3-13}$$

径向距离的标准差：

$$\sigma_R = \left(\frac{1}{K}\sum_{k=0}^{K-1} [\|(r_k, c_k) - (\bar{r}, \bar{c})\| - \mu_R]^2\right)^{1/2} \tag{3-14}$$

其中像素集(r_k, c_k)，$k = 0, \cdots, K-1$位于区域的周界P上。当数字化形状变得更圆时，圆度C_2单调上升，无论是数字形状还是连续形状，结果都是一样的。

图3-18给出区域的一些在简单标记图像上的基本特征，标记图像有三个区域：一个椭圆，

75 一个矩形和一个3×3的正方形。

习题3-12 运用特征

假设你有一些二维形状，如三角形、矩形、八边形、圆形和椭圆形，请设计识别这些形状的策略。可以利用数学形态运算和目前学过的特征。

边界框和极点

常常需要粗略地知道一个区域位于一幅图像的什么位置。这时要用到区域的边界框（bounding box）这个概念。边界框是一个矩形，由水平和竖直四条边把整个区域围起来，并与区域的最上、最下、最左和最右点相接，如图3-19所示。一个区域可以有多至8个不同的极点：右边最上、上边最右、下边最右、右边最下、左边最下、下边最左和上边最左点。每个极点都有一个极坐标值，要么是行坐标值，要么是列坐标值。每个极点都在区域的边界框上。

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
2	2	2	2	0	0	0	0	0	1	1	1	1	1	1	0
2	2	2	2	0	0	0	0	1	1	1	1	1	1	1	1
2	2	2	2	0	0	0	0	1	1	1	1	1	1	1	1
2	2	2	2	0	0	0	0	1	1	1	1	1	1	1	1
2	2	2	2	0	0	0	0	0	1	1	1	1	1	1	0
2	2	2	2	0	0	0	0	0	0	1	1	1	1	0	0
2	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0
2	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0
2	2	2	2	0	0	3	3	3	0	0	0	0	0	0	0
2	2	2	2	0	0	3	3	3	0	0	0	0	0	0	0
2	2	2	2	0	0	3	3	3	0	0	0	0	0	0	0
2	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0

a）连通成分标号图

区域标记	区域面积	区域中心所在的行	区域中心所在的列	区域周长	区域圆度1	区域圆度2	半径平均值	半径方差
1	44	6	11.5	21.2	10.2	15.4	3.33	0.05
2	48	9	1.5	28	16.3	2.5	3.80	2.28
3	9	13	7	8	7.1	5.8	1.2	0.04

b）三个区域的特征

图3-18 区域的基本特征

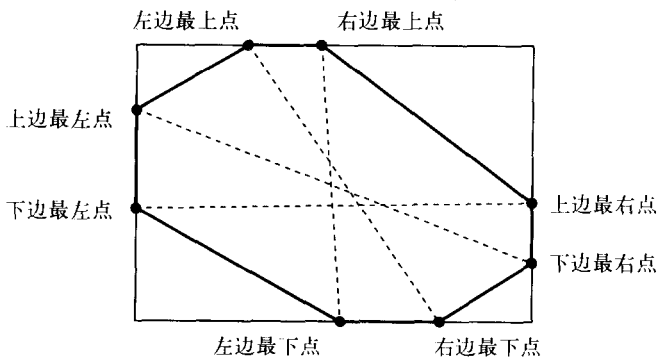


图3-19 区域的8个极点以及包围该区域的正常取向边界框。虚线把两个相对的极点连起来，形成形状的极点轴线

极点以相对位置成对出现：左边最上点对右边最下点，右边最上点对左边最下点，上边最右点对下边最左点，下边最右点对上边最左点。每对极点确定一条轴线。可用的轴线特征包括轴线的长度与方向。由于极点是空间数字化或量化处理的结果，用标准欧几里德距离公式计算出的结果稍微偏低一点（例如，计算水平方向两紧邻像素之间的长度。从左边像素的左边到右边像素的右边之间是两个像素的长度，但两像素中心之间的距离只是1个像素。）解决的办法是在欧几里德距离公式上加上一个微小增量。增量的大小取决于轴线的方向角θ，具体如下: 76

$$Q(\theta)=\begin{cases}\frac{1}{|\cos\theta|} & : \ |\theta|<45^\circ\\ \frac{1}{|\sin\theta|} & : \ |\theta|>45^\circ\end{cases} \tag{3-15}$$

加上这个增量后，极点（r_1, c_1）到极点（r_2, c_2）之间的轴线长度如下：

极轴长度：

$$D=\sqrt{(r_2-r_1)^2+(c_2-c_1)^2}+Q(\theta) \tag{3-16}$$

空间矩通常用于描述一个区域的形状。区域的二阶空间矩（spatial moment）有三个，表示为μ_{rr}, μ_{rc}和μ_{cc}, 分别定义如下：

二阶行矩：

$$\mu_{rr}=\frac{1}{A}\sum_{(r,c)\in R}(r-\bar{r})^2 \tag{3-17}$$

二阶混合矩：

$$\mu_{rc}=\frac{1}{A}\sum_{(r,c)\in R}(r-\bar{r})(c-\bar{c}) \tag{3-18}$$

二阶列矩：

$$\mu_{cc}=\frac{1}{A}\sum_{(r,c)\in R}(c-\bar{c})^2 \tag{3-19}$$

μ_{rr}表示偏离行均值的行变差，μ_{cc}表示偏离列均值的列变差，μ_{rc}表示偏离中心的行列变差。它们不随二维形状的平移和尺度变化而变化，因此常用于描述简单的形状。

形状区域情况下二阶空间矩的数值和含义，与二维概论分布协方差矩阵的数值和含义类似。如果区域是一个椭圆，就可以看出二阶空间矩的代数含义。

设区域R是一个椭圆，其中心位于原点，则R可被表达为：

$$R=\{(r,\ c)\mid dr^2+2erc+fc^2\leqslant 1\} \tag{3-20}$$

77 椭圆方程的系数d、e、f与二阶矩μ_{rr}、μ_{rc}和μ_{cc}之间存在一定的关系，关系如下：

$$\begin{pmatrix}d & e\\ e & f\end{pmatrix}=\frac{1}{4(\mu_{rr}\mu_{cc}-\mu_{rc}^2)}\begin{pmatrix}\mu_{cc} & -\mu_{rc}\\ -\mu_{rc} & \mu_{rr}\end{pmatrix} \tag{3-21}$$

由于系数d、e、f确定了椭圆主次轴的长度及其方向，这种关系意味着二阶矩μ_{rr}、μ_{rc}和μ_{cc}也确定椭圆主次轴的长度及其方向。椭圆常常是圆形目标的成像结果，也是对其他长条形目标的粗略近似。

椭圆两轴的长度与方向*　为了根据二阶矩计算椭圆主次轴的长度及其方向，我们必须考虑下面4种情况（注意下面的方向角，是从纵轴沿逆时针转动的方向）：

1. $\mu_{rc}=0$, $\mu_{rr}>\mu_{cc}$

主轴方向角为-90°，主轴长度为$4\mu_{rr}^{1/2}$；次轴方向角为0°，次轴长度为$4\mu_{cc}^{1/2}$。

2. $\mu_{rc}=0$, $\mu_{rr}\leqslant\mu_{cc}$

主轴方向角为0°，主轴长度为$4\mu_{cc}^{1/2}$；次轴方向角为-90°，次轴长度为$4\mu_{rr}^{1/2}$。

3. $\mu_{rc}\neq 0$, $\mu_{rr}\leqslant\mu_{cc}$

主轴方向角为：

$$\tan^{-1}\left\{\frac{-2\mu_{rc}}{\mu_{rr}-\mu_{cc}+\left[(\mu_{rr}-\mu_{cc})^2+4\mu_{rc}^2\right]^{1/2}}\right\}$$

其长度为:

$$\left\{8\left(\mu_{rr}+\mu_{cc}+\left[(\mu_{rr}-\mu_{cc})^2+4\mu_{rc}^2\right)^{1/2}\right]\right\}^{1/2}$$

次轴方向角为90°，其长度为:

$$\left[8\left\{\mu_{rr}+\mu_{cc}-\left[(\mu_{rr}-\mu_{cc})^2+4\mu_{rc}^2\right]^{1/2}\right\}\right]^{1/2}$$

4. $\mu_{rc}\neq 0, \mu_{rr}>\mu_{cc}$

主轴方向角为:

$$\tan^{-1}\frac{\left[\left\{\mu_{cc}+\mu_{rr}+\left[(\mu_{cc}-\mu_{rr})^2+4\mu_{rc}^2\right]^{1/2}\right\}\right]^{1/2}}{-2\mu_{rc}}$$

78

其长度为:

$$\left[8\left\{\mu_{rr}+\mu_{cc}+\left[(\mu_{rr}-\mu_{cc})^2+4\mu_{rc}^2\right]^{1/2}\right\}\right]^{1/2}$$

次轴方向角为90°，其长度为:

$$\left[8\left\{\mu_{rr}+\mu_{cc}-\left[(\mu_{rr}-\mu_{cc})^2+4\mu_{rc}^2\right]^{1/2}\right\}\right]^{1/2}$$

最佳轴* 一些图像区域（目标）具有自然轴，例如一只铅笔、一把锤子或字符“I”、“/”、“-”等。最佳轴（Best Axis）是最小的二阶矩所对应的轴线。模仿机械学中的术语，称它为最小惯性轴，也就是说像素绕该轴旋转时需要的能量最小。对于一个圆盘，所有轴线具有相等的最小（和最大）惯性。众所周知，最小惯性轴一定通过像素集合（像素具有单位质量）的中心$(\bar{r},\bar{c})$，这点可以保证。首先计算像素点集关于任意轴线的二阶矩，然后寻找使二阶矩最小的轴线。计算出关于这些轴线的二阶矩，也许能提供一组比较好的特征用于目标识别，下一章我们会看到这一点。例如，字符“I”关于过中心垂直轴线的二阶矩是很小的，但字符“/”和“-”关于这个轴线的二阶矩则不小。

图3-20显示一些像素点以及与横（行）轴成α角的轴线。该轴线的垂线与横轴夹角为$\beta=\alpha+90$。为了计算点集关于该轴线的二阶矩，需要求所有像素点到该轴线的距离d的平方和，再除以像素数进行规范化处理，就得到不随像素数变化而显著变化的特征，这些像素确定了目标区域的形状。注意，由于我们是求d^2的和，如果α、β加上或减去π，将不改变二阶矩的大小。公式（3-22）给出了二阶矩的计算公式，“$\circ$”表示向量的标量积，即向量$\bar{V}$投影到方向为β的单位向量上，产生的投影长度为d。任何轴线可以用$\bar{r}$、$\bar{c}$和α三个参数确定。

79

二阶轴矩*:

$$\begin{aligned}\mu_{\bar{r},\bar{c},\alpha} &= \frac{1}{A}\sum_{(r,c)\in R} d^2\\ &= \frac{1}{A}\sum_{(r,c)\in R}(\bar{V}\circ(\cos\beta,\sin\beta))^2\\ &= \frac{1}{A}\sum_{(r,c)\in R}((r-\bar{r})\cos\beta+(c-\bar{c})\sin\beta)^2\end{aligned}$$

(3-22)

其中$\beta = \alpha + \pi/2$。

80 利用上面的公式（3-22）能够算出几个矩，从而得到点集的形状信息。例如利用关于垂直轴线、水平轴线和对角轴线的矩，可对标准字体的字符分类。最小（最大）惯性是点集的不变特征，并随点集进行平移和旋转。使$\mu_{r,c}$最小可得到最小惯性轴。假设最佳轴必须通过中心，那么只需将公式（3-22）对α求导，就可以得到最佳值α。

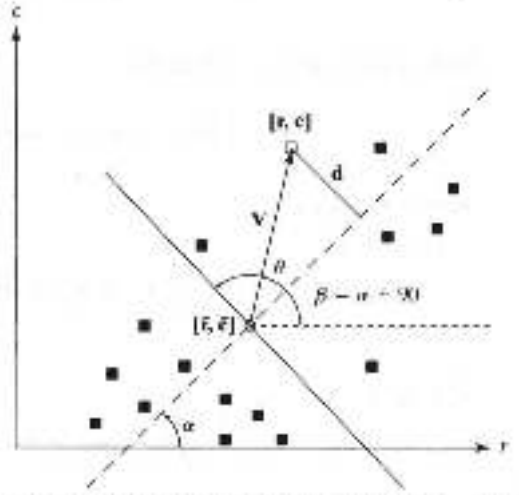


图3-20 计算每个像素到轴线距离的平方和，得到二阶轴矩

习题3.13 编程计算点集的特征

编写程序模块，或C++类，对二维点包进行管理，并提供下面的功能。包与集的不同之处在于包允许对点进行复制。

- 构造一个二维点[r,c]的空包；
- 把点[r,c]加进包里；
- 计算当前点包的中心；
- 计算当前点包的行矩和列矩；
- 计算边界框；
- 计算最佳轴和最差轴，及关于最佳轴和最差轴的二阶矩。

习题3.14 编程计算图像特征

在前面的习题中已经编写了特征抽取模块，对这个模块进行改进，计算关于过中心点的水平轴线、垂直轴线和对角轴线的二阶矩。这样对任何点包将得到5个不同的二阶矩。建立大小为20 × 20的二值图像集，测试数据从数字0到9，或利用已有的数据，编写程序扫描某个数字图像并计算5个矩。研究是否能用这5个矩识别出输入的数字。

具有最小二阶矩的轴*：

$$\begin{aligned}\tan 2\hat{\alpha} &= \frac{2\sum(r-\bar{r})(c-\bar{c})}{\sum(r-\bar{r})(r-\bar{r}) - \sum(c-\bar{c})(c-\bar{c})} \\ &= \frac{\frac{1}{A}2\sum(r-\bar{r})(c-\bar{c})}{\frac{1}{A}\sum(r-\bar{r})(r-\bar{r}) - \frac{1}{A}\sum(c-\bar{c})(c-\bar{c})} \\ &= \frac{2\mu_{rc}}{\mu_{rr}-\mu_{cc}}\end{aligned} \tag{3-23}$$

α有极小值和极大值两种极值，二者相差90°。在上面关于椭圆长短轴的讨论中，我们已经知道了区分这两个轴线的方法。实际上根据这些矩的含义，我们能够通过上面的公式计算近似点集的椭圆。对于高度对称的形状如正方形、圆等，公式（3-23）中的分母将为0，这时就要用椭圆分析方法。

习题3.15 计算惯性极值

对公式（3-22）求导，看看如何得到公式（3-23）所示的最佳轴（最差轴）。

习题3.16 证明最佳轴通过中心

证明最小惯性轴一定通过中心。请参考本章末的文献，或其他关于统计回归、机械学的参考资料，或者你自己证明。

3.7 区域邻接图

除了单个区域的特征，不同区域之间的关系在图像分析中也是有用的。一种最简单但最有用的关系是区域邻接（region adjacency）。如果一个区域的像素与另一个区域的像素相邻，则称这两个区域是邻接的。在二值图像中，只有两种区域：前景区域和背景区域。所有的前景区域是和背景区域邻接的，而前景区域之间互不邻接。如果背景是一个单连通区域，则不需要计算下去。如果前景区域内部有孔，而这些孔属于背景区域，这时利用连通成分标记算法对前景像素进行标记，生成标记图像，其中每个前景区域具有一个数字标号，而所有背景区域的标号为0。也可以对背景进行连通成分标记，赋给每个背景区域一个标号，对比较大、从图像左上角开始的区域，可进行特殊标记如标记为0，其他的背景区域就是前景区域中的孔。有了前景标号图和背景标号图，就能确定哪些背景区域与每个前景区域邻接，或者确定哪些前景区域与背景邻接。记录区域邻接的结构图称为区域邻接图（region adjacency graph）。可用它来记录二值图像中前景区域与背景区域的邻接关系，以及在图像分割时记录所有的邻接关系。 81

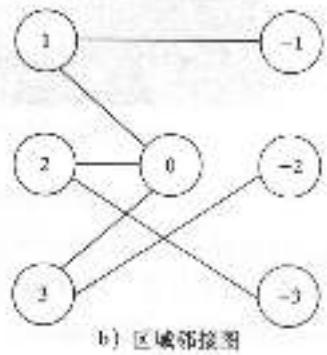
定义21 区域邻接图用节点表示图像区域，如果两个区域是邻接的，则用一条边缘线连接两节点。

图3-21是二值图像前景和背景区域邻接图的例子。前景区域按惯例用正整数标记。从图像左上角开始的大背景区域标记为0，孔区域用负整数标记。

构造区域邻接图的算法是比较简单的，处理图像，着眼于当前行及上面一行，检测具有不同标号的水平邻接区和垂直邻接区；对于8-邻接，还要检测对角邻接区。如果检测到新的邻接区，在要构造的区域邻接图数据结构中添加新的边。算法的效率受到两个问题的影响。第一个是空间问题，一幅图像可能有几万个标号，要在内存中同时保持整个数据结构存在，这样的算法是不可行的，或者至少在内存页面环境中是不可行的。第二个是执行时间问题，当逐个像素扫描图像时，会反复检测到同样的邻域（即同样的两个区域标号），我们希望把邻接信息加入数据结构的频度越少越好，习题3.17中要谈到这些问题。

0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	2	2	0
0	1	-1	-1	-1	1	0	2	2	0
0	1	1	1	1	1	0	2	2	0
0	0	0	0	0	0	0	2	2	0
0	3	3	3	0	2	2	2	2	0
0	3	-2	3	0	2	-3	-3	2	0
0	3	-2	3	0	2	-3	-3	2	0
0	3	3	3	0	2	2	2	2	0
0	0	0	0	0	0	0	0	0	0

a）前景与背景区域的标号图



b）区域邻接图

图3-21 标号图和区域邻接图

习题3.17 构造RAG的有效性

设计数据结构，在构造区域邻接图时记录邻接信息，对于任意一幅标号图，写出构造区域邻接图的算法，并且要使数据结构的参数个数最少，讨论怎样在永久存储器上（磁盘）保
82 存最终的RAG，以及怎样处理由于RAG太大而在构造期间受内存限制的问题。

3.8 灰度级图像阈值化

灰度级图像通过阈值运算，可以转化为二值图像。阈值运算把感兴趣的目标像素作为前景像素，其余部分作为背景像素。如果图像的灰度值分布已知，可选择合适的灰度值作为阈值，并据此把图像像素分成组。最简单的情况是选用单阈值t，灰度值大于等于t的所有像素作为前景像素，把其余像素作为背景像素。这种阈值运算称为上阈值化（threshold above）。此外还有其他多种算法，如下阈值化（threshold below），即把灰度值小于等于t的所有像素作为前景像素；内阈值化（threshold inside），即确定一个较小的阈值和一个较大的阈值，把灰度值界于二者之间的像素作为前景像素；外阈值化（threshold outside），与内阈值化相反，把灰度值界于小阈值与大阈值之外的像素作为前景像素。这些都是简单的阈值运算，它们最主要的问题是如何选择合适的阈值。

3.8.1 直方图阈值选择

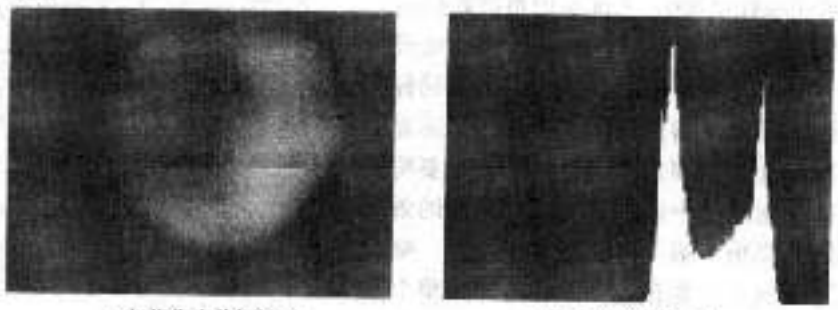
阈值可以通过软件由用户以交互的方式进行选择，但对于自动图像分析与处理，希望阈
83 值计算能够自动进行。选择阈值的基本方法是利用灰度图像的直方图（histogram）。

定义22 直方图。灰度图像**I**的直方图**h**定义为：

$$h(m) = |\{(r, c) | I(r, c) = m\}|$$

其中m的取值范围是整个灰度级值。

图3-22显示带伤痕的樱桃及其直方图。直方图有两个明显的模式，表示樱桃上坏了的部分和没坏的部分。



a）带伤痕的樱桃图　　b）樱桃图的直方图

图3-22 带伤痕的樱桃的直方图，显示两个模式（Patchrawat Uthaisombut授权）

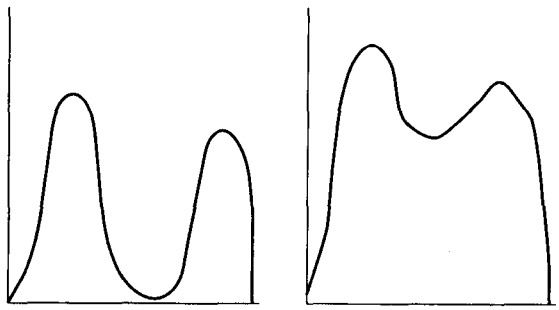
直方图计算可用数组数据结构和简单的程序实现。设**H**是向量数组，维数从0到**MaxVal**，其中0是最小灰度级值，**MaxVal**是最大灰度级值。设**I**是二维图像数组，行号从0到**MaxRow**，
84 列号从0到**MaxCol**，这和前面的一节一样。计算直方图的程序代码如下：

算法3.7 计算灰度图像I的直方图H

```
procedure histogram(I, H);
{
\\初始化直方图数组的各元素为0。
for i :=0 to MaxVal
   H[i] :=0;
\\计算累加值。
for L :=0 to MaxRow
   for P :=0 to MaxCol
      {
      grayval :=I[r,c];
      H[grayval] : = H[grayval] + 1;
      };
}
```

直方图计算程序中，假设每个可能的图像灰度级对应直方图的一个箱格。当存在很多可能的灰度级时，为了显示直方图，希望将几个灰度级组合起来，对应一个箱格。对前面的程序稍加修改，就可以算出该箱格中的像素个数，它是灰度级的函数。如果用箱格大小表示每个箱格包含的灰度级数，那么对灰度值与箱格大小之比grayval/binsize取整后就得到正确的箱格下标。

已知直方图，可编写程序自动检测直方图函数的波峰和波谷。最简单的情况是找到一个阈值把图像分成黑白两种像素。如果黑白两种像素的分布是分开的，则图像直方图将是双模式的，其中一种模式对应黑色像素，另一种模式对应白色像素。由于像素的分布几乎没有重叠，可以在两模式之间的谷底上方便地选择阈值，如图3-23a所示；如果黑白像素的分布有很多重叠，选择阈值就比较困难，因为随着两种分布融合到一起，谷底就开始消失，如图3-23b所示。



a）直方图两种区分明显的模式　　b）直方图的重叠模式，很难找到合适的阈值

图3-23 两幅图像直方图

3.8.2 自动阈值处理：Otsu方法*

科研人员已经提出几种自动确定阈值的方法，这里主要讨论Otsu方法。阈值运算把所有像素分成两组，Otsu方法通过使两组像素的组内方差最小来确定阈值。首先定义直方图函数为一个概率函数P，其中$P(0)$, …, $P(I)$表示灰度值0,…, I的直方图概率，$P(i) = |\{(r, c)|Image(r, c) = i\}|/|R \times C|$，其中$R \times C$是图像的空间区域。如果直方图是双模式的，通过直方图求阈值也就是确定一个最好的阈值t，利用这个阈值把直方图的两种模式分开。根据阈值t，可以确定灰度值小于或等于t的像素集的方差，以及灰度值大于t的像素集的方差。Otsu关于最佳阈值的定义是使组内方差的加权和最小的阈值，其中权是指各组概率。

滑雪学校的情形促使我们采用组内方差标准。事先对学员的能力进行测试，测试结果的直方图是双模式的。由于存在滑雪高手和新手，面向高手的课程对其他人来说进度太快，面向新手的课程又让高手厌烦。为了解决这个矛盾，教练决定根据测试得分把学员分成组间互斥、组内均衡的两组。问题是用什么评分标准来分组。理想情况是，各组自身的直方图曲线应是单模式的钟形曲线，一组的曲线具有较低的均值，另一组的曲线具有较高的均值。这指的是各组自身是均衡的，但不同于另一组。

组内均衡性的测度是方差。均衡性较高的组有较低的方差，均衡性较低的组有较高的方差。选择分割标准的方法之一是，确定合适的得分分界线使组内方差的加权和最小，这个标准强调高的组内均衡性；第二种方法是选择合适的得分分界线使两组均值之间的平方差最大。这个差与组间方差有关。这两个分割标准会产生同样的得分分界线，因为组内方差和组间方差之和是一个常数。

设 σ_w^2 是小组内各方差的加权和，即组内方差；$\sigma_1^2(t)$是值小于或等于t的小组的方差，$\sigma_2^2(t)$是值大于t的小组的方差；$q_1(t)$ 是值小于或等于t的小组的概率，$q_2(t)$ 是值大于t的小组的概率；$\mu_1(t)$ 是第一组的均值，$\mu_2(t)$ 是第二组的均值。则组内方差 σ_w^2 定义为：

$$\sigma_W^2(t) = q_1(t)\,\sigma_1^2(t) + q_2(t)\,\sigma_2^2(t) \tag{3-24}$$

其中

$$\begin{aligned} q_1(t) &= \sum_{i=1}^{t} P(i) \\ q_2(t) &= \sum_{i=t+1}^{I} P(i) \\ \mu_1(t) &= \sum_{i=1}^{t} i\ P(i)/q_1(t) \\ \mu_2(t) &= \sum_{i=t+1}^{I} i\ P(i)/q_2(t) \end{aligned} \tag{3-25}$$

86

$$\begin{aligned} \sigma_1^2(t) &= \sum_{i=1}^{t} [i - \mu_1(t)]^2\ P(i)/q_1(t) \\ \sigma_2^2(t) &= \sum_{i=t+1}^{I} [i - \mu_2(t)]^2\ P(i)/q_2(t) \end{aligned} \tag{3-26}$$

利用简单的顺序搜索搜索所有可能的t值，确定使 $\sigma_w^2(t)$ 最小的最佳阈值t。在许多情况下，可以简化到在两个模式之间搜索。而模式识别实际上就是识别两模式之间分界值。

组内方差 $\sigma_w^2(t)$ 与总方差 σ^2 之间有一定的关系，它不依赖于阈值。总方差定义为：

$$\sigma^2 = \sum_{i=1}^{I} (i - \mu)^2 P(i)$$

其中

$$\mu = \sum_{i=1}^{I} i\ P(i)$$

总方差与组内方差之间的关系可以简化最佳阈值的计算。通过重写 σ^2，我们有

$$\begin{aligned}\sigma^2 &= \sum_{i=1}^{t}[i-\mu_1(t)+\mu_1(t)-\mu]^2\ P(i) + \sum_{i=t+1}^{I}[i-\mu_2(t)+\mu_2(t)-\mu]^2\ P(i)\\ &= \sum_{i=1}^{t}\{[i-\mu_1(t)]^2 + 2[i-\mu_1(t)][\mu_1(t)-\mu] + [\mu_1(t)-\mu]^2\}P(i)\\ &\quad + \sum_{i=t+1}^{I}\{[i-\mu_2(t)]^2 + 2[i-\mu_2(t)][\mu_2(t)-\mu] + [\mu_2(t)-\mu]^2\}P(i)\end{aligned}$$

但是

$$\sum_{i=1}^{t}[i-\mu_1(t)][\mu_1(t)-\mu]P(i) = 0$$

$$\sum_{i=t+1}^{I}[i-\mu_2(t)][\mu_2(t)-\mu)]P(i) = 0$$

87

由于

$$\begin{aligned}q_1(t) &= \sum_{i=1}^{t}P(i) \text{ 以及 } q_2(t) = \sum_{i=t+1}^{I}P(i)\\ \sigma^2 &= \sum_{i=1}^{t}[i-\mu_1(t)]^2\ P(i) + [\mu_1(t)-\mu]^2\ q_1(t)\\ &\quad + \sum_{i=t+1}^{I}[i-\mu_2(t)]^2\ P(i) + [\mu_2(t)-\mu]^2\ q_2(t)\\ &= \left[q_1(t)\ \sigma_1^2(t) + q_2(t)\ \sigma_2^2(t)\right]\\ &\quad + \{q_1(t)\ [\mu_1(t)-\mu]^2 + q_2(t)\ [\mu_2(t)-\mu]^2\}\end{aligned} \tag{3-28}$$

第一个括号项是组内方差 σ_w^2，是两组方差的加权和。第二个括号项称为组间方差 σ_B^2，是每组均值和总均值之间的距离平方的加权和。组间方差可进一步简化。注意总均值μ可写为:

$$\mu = q_1(t)\ \mu_1(t) + q_2(t)\ \mu_2(t) \tag{3-29}$$

把公式（3-29）代入公式（3-28）消去μ，用$1-q_1(t)$代替$q_2(t)$，简化后得到:

$$\sigma^2 = \sigma_W^2(t) + q_1(t)[1-q_1(t)]\ [\mu_1(t)-\mu_2(t)]^2$$

因为总方差 σ^2 不依赖t，使 $\sigma_w^2(t)$最小化的t将是使组间方差 $\sigma_B^2(t)$ 最大的t。

$$\sigma_B^2(t) = q_1(t)\ [1-q_1(t)]\ [\mu_1(t)-\mu_2(t)]^2 \tag{3-30}$$

为了求使 $\sigma_B^2(t)$ 最大的 t，要知道由公式（3-25）到公式（3-27）确定的参数大小，而这不需要对每个 t 都算一遍。为 t 计算的值与为 $t+1$ 计算的值之间有一定的关系。从公式（3-25）可直接推出这个迭代关系

$$q_1(t+1) = q_1(t) + P(t+1) \tag{3-31}$$

初始值取为 $q_1(1) = P(1)$。

从公式（3-26）可推出迭代关系

$$\mu_1(t+1) = \frac{q_1(t)\,\mu_1(t) + (t+1)P(t+1)}{q_1(t+1)} \tag{3-32}$$

初始值取为 $\mu_1(0) = 0$。最后从公式（3-29），我们有

88

$$\mu_2(t+1) = \frac{\mu - q_1(t+1)\,\mu_1(t+1)}{1 - q_1(t+1)} \tag{3-33}$$

只有当图像整体灰度分布满足假设的条件下，自动寻找阈值的算法才能很好地工作。Otsu的自动阈值寻找器假设灰度值呈双模式分布。如果图像近似满足这个约束，算法将能很好地工作。如果图像根本不是双模式，算出的结果是无用的。图3-24显示的是用Otsu算子处理a图中积木玩具的灰度图像。图像的灰度范围是0~255，算子返回的阈值是93。小于和大于阈值的像素分别显示在b和c中。图像中只有非常黑的区域才被分割出来。



a) 原图　b) 小于阈值93的像素　c) 大于阈值93的像素

图3-24　灰度图像以及由Otsu自动阈值算子得到的小于和大于阈值93的像素（显示为白色）（原图由John Illingworth和Atta Etamad提供）

如果图像的灰度值与在图像中的位置密切相关，例如左上角较亮，右下角较暗，那么用局部阈值代替全局阈值也许更加合适。这个思想有时称为*动态*（dynamic）阈值化。在一些应用中，目标的近似形状和尺寸事先可以知道，这时称为*基于知识*（knowledge-based）的阈值化。该方法对区域结果进行评价，并进行最佳阈值选择。最后，有的图像不能阈值化，必须用别的方法来查找目标。

习题3.18　自动确定阈值

编写程序实现Otsu自动阈值确定方法。试着在几种不同类型的扫描图像上运行编写的程序。

3.9　参考文献

关于连通成分的标记运算还有其他一些不同的算法。每种算法都是针对某个目的开发的。Tanimoto（1990）假设整幅图像能够载入内存，采用简单的递归算法，算法每次处理一个成分，可对整幅图像进行扫描。还有一些算法是针对较大图像的，这些图像受内存限制不能一次全部载入。算法每次只处理一幅图像的两行像素。Rosenfeld和Pfaltz（1966）提出两阶段算法，算法用了全局性等价表，有时被称为经典（classical）连通成分算法。Lumia、Shapiro和Zuniga（1983）提出另一种两阶段算法，采用局部等价表以避免内存不足问题。Danielsson和Tanimoto（1983）为大型并行机设计的算法，采用并行传播策略。记录等价对的任何算法都可用并查数据结构（Tarjan, 1975）来有效执行集合合并运算。

89

Serra（1982）第一次提出数学形态运算的系统化理论。Sternberg（1985）为快速运算设计了并行流水线结构，并用于医学成像和工业机器视觉。他也把二值形态运算扩展到灰度形态运算（1986），这已成为标准的图像滤波算法。Haralick、Sternberg和Zhuang（1987）发表了关于二值形态运算和灰度形态运算的导论性论文，体现了他们在计算机视觉领域的价值。Shapiro、MacDonald和 Sternberg（1987）的研究表明形态特征检测可用于目标识别。

在一些论文中论述了自动阈值化。本书描述的方法参考的是Otsu（1979）。其他方法由Kittler和Illingworth（1986）以及Cho、Haralick和Yi（1989）提出。Sahoo等人（1988）对阈值技术进行了综述。

1. Tanimoto, S. L. 1990. *The Elements of Artificial Intelligence Using Common LISP.* W. H. Freeman and Company, New York.
2. Rosenfeld, A., and J. L. Pfaltz. 1966. Sequential operations in digital picture processing. *J. Assoc. Comput. Machinery,* v. 13:471–494.
3. Lumia, R., G. Shapiro, and O. Zuniga. 1983. A new connected components algorithm for virtual memory computers. *Comput. Vision, Graphics, and Image Proc.,* v. 22: 287–300.
4. Danielsson, P.-E., and S. L. Tanimoto. 1983. Time complexity for serial and parallel propagation in images. In *Architecture and Algorithms for Digital Image Processing,* A. Oosterlinck and P.-E. Danielsson, eds. *Proc. SPIE,* v. 435:60–67.
5. Tarjan, R. E. 1975. Efficiency of a good but not linear set union algorithm. *J. Assoc. Comput. Machinery,* v. 22:215–225.
6. Serra, J. 1982. *Image Analysis and Mathematical Morphology.* Academic Press, New York.
7. Sternberg, S. R. 1985. An overview of image algebra and related architectures. *Integrated Technology for Parallel Image Processing.* Academic Press, London, 79–100.
8. Sternberg, S. R. 1986. Grayscale morphology. *Comput. Vision, Graphics, and Image Proc.,* v. 35:333–355.
9. Haralick, R. M., S. R. Sternberg, and X. Zhuang. 1987. Image analysis using mathematical morphology. *IEEE Trans. Pattern Analysis and Machine Intelligence,* v. PMI-9:523–550.
10. Shapiro, L. G., R. S. MacDonald, and S. R. Sternberg. 1987. Ordered structural shape matching with primitive extraction by mathematical morphology. *Pattern Recog.,* v. 20(1)75–90.
11. Haralick, R. M. 1974. A measure of circularity of digital figures. *IEEE Trans. Syst., Man, and Cybern.,* v. SMC-4:394–396.
12. Otsu, N. 1979. A threshold selection method from gray-level histograms. *IEEE Trans. Syst., Man and Cybern.,* v. SMC-9:62–66.
13. Kittler, J., and J. Illingworth. 1986. Minimum error thresholding. *Pattern Recog.,* v. 19:41–47.
14. Cho, S., R. M. Haralick, and S. Yi. 1989. Improvement of Kittler and Illingworth's minimum error thresholding. *Pattern Recog.,* v. 22:609–617.
15. Sahoo, P. K., and others. 1988. A survey of thresholding techniques. *Comput. Vision, Graphics, and Image Proc.,* v. 41:233–260.

90

91

第4章 模式识别

本章概述了目标识别的方法，这些方法主要是对图像中的目标进行识别，也适用于其他类型的数据，其基本方法是将待识别的样本表示成向量。在所列举的几个例子中，中心问题是字符识别。本章还向读者介绍一些简单方法，根据这些方法，机器能够通过向示例学习从而识别出目标。在学习完前四章之后，读者就能够理解完整的机器视觉系统设计过程，并且能够针对一些简单而实用的问题，编写出完整的算法并进行实验。

4.1 模式识别问题

在许多应用问题中，都需要对图像内容作出判定或者对图像中包含的目标进行分类。例如，用户输入到笔记本电脑的可能是手写字符。在这种情况下，有m = 128个ASCII字符，每个手写字符将被归类为这m类中的一类，参见图4-1。确定一个目标的类别，就是说判断它是“A”还是“8”，是基于它的光学图像特征或者是压痕特征，压痕的表示与图像类似。分类过程实际上也可能失败，原因是字迹过于潦草，或是人们发明的新字符。一般为了包括这种情况，在设计系统时加入一个“拒绝”类别。属于拒绝类别的图像数据在后面的更高一层要再进行一次检测，结果或者成为一个新类别，或者就以原始形式保存下来。

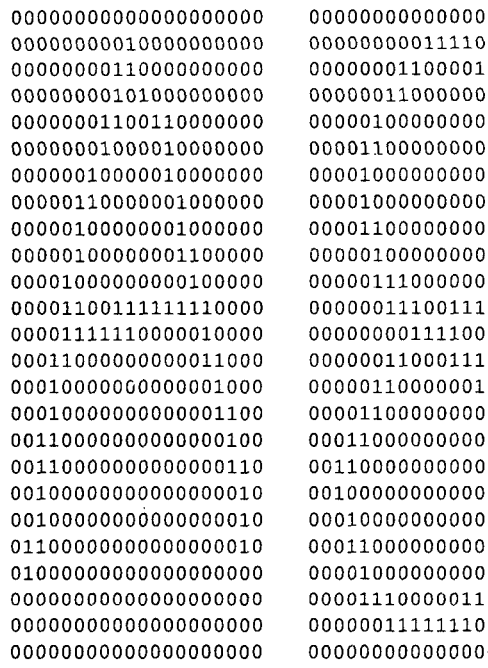


图4-1 “A”和“8”的二值图像

银行自动柜员机（ATM）借助摄像头来验证当前的用户是否是合法用户。这时要将当前用户的面部图像与已经存储的图像做比较，或者是与当前帐户有关联的、存在计算机网络或银行卡上的图像做比较。

定义23 将一个目标实例与一个目标原型或类别定义进行匹配的过程称为**验证**。

第1章的习题中介绍过另一种应用，食物识别系统对放在收银机台秤上的水果和蔬菜进行分类。类别是所有可识别商品类型的集合，如爱达荷州的苹果、富士苹果、绿甘蓝、绿菠菜和蘑菇等。每类都有自己的名字及每磅的价钱。[⊖]

识别的一个定义是再认识。识别系统必须记忆要识别的目标。这种记忆可能是天生的，如青蛙眼中的飞虫模型；也可能是从大量实例中学到的，像学校老师教字母表那样；或者是

⊖ 这样的系统称为Veggie Vision，已由IBM开发出来，参见第16章。

编入程序的具体图像特征，像母亲教孩子区分消防车与公共汽车那样。模式识别和模式学习是认知心理学、模式识别和计算机视觉的深层研究主题和兴趣所在。本章从实用的角度，介
93 绍具有成功应用背景的方法，并在本章末列出了大量的理论参考文献。

4.2 分类模型

我们对分类模型的组成部分进行总结，这种划分是从实用的角度而不是从理论的角度进行的。这样做便于在设计模式识别系统时，采取分块开发硬件和软件模块的方法。

4.2.1 类别

有m个已知的目标类别，所谓已知是指或者有关于类别的描述，或者有属于每个类别的样本集合。例如，对于字符识别，或者是对每个字符有其外形描述，或者是对每个字符都有对应的样本集合。一般还包括一个特殊的拒绝类别，这是为那些不属于任何已知类别的目标而设计的。

定义24 一个理想**类别**是一些具有重要共同属性的目标的集合。在实际中，某目标所属类别用**类别标号**来标识。**分类**就是根据目标的属性表示赋予目标类别标号的过程。**分类器**是一种设备或算法，它输入的是目标的表示，输出的是类别标号。

定义25 **拒绝类别**是为无法归入任何已知类别的目标设置的通用类别。

4.2.2 传感器/变换器

为了能用计算机处理，必须有某种设备能够感测实际的目标，并输出目标表示（通常是数字信息）。最一般的做法是从现有的成品传感器中选择一种。例如，为了对超市的蔬菜进行分类，首先用一般的彩色摄像机，从它摄取的图像中抽取颜色、形状和纹理特征。为了识别压出的字符，采用压力敏感阵列进行测量。

由于本书是关于机器视觉的，我们最感兴趣的是产生2D阵列感知数据的传感器。然而模式识别本身更加通用，如用于识别语音电话号码，以及识别写在纸上的电话号码。

4.2.3 特征抽取算子

特征抽取算子从传感器得到的数据中抽取分类的相关信息。一般特征抽取由软件完成。软件的输入与传感器的硬件输出相适应，经过中间的研究与开发，最后输出分类结果。第3章
94 中定义了许多图像特征。

4.2.4 分类器

分类器利用从目标数据中抽取的特征，赋予目标m个指定类别中的一个类别，m个类别为$C_1, C_2, \cdots, C_{m1}, C_m = C_r$，其中$C_r$表示拒绝类别。

图4-2所示为一个分类系统的框图。输入是一个d维的特征向量$\mathbf{x}$，表示待分类的目标。对每一个可能的类别，系统都用一个方框表示，它包含该类别的相关知识$\mathbf{K}$和处理能力。m个类别的计算结果被传递给最终的决策阶段，在该阶段决定目标的类别。一般说来，这个框图足以表示将讨论的三种分类方法：（a）最近均值分类，（b）最大后验概率分类，（c）前向人工神经网络分类。

4.2.5 分类系统的建立

系统的每个部分都有多种实现方法。图像传感器在第2章中介绍过。第3章讨论了如何根据目标的二值图像计算多种不同的特征。颜色和纹理特征计算将在第6和第7章讨论。这里再次用到字符识别这个例子。30 × 20的窗口中的字符有600个像素点，特征抽取需要处理这600个像素点，并输出10到30个特征，这些特征是分类决策的依据。这个例子将在下面继续讨论。

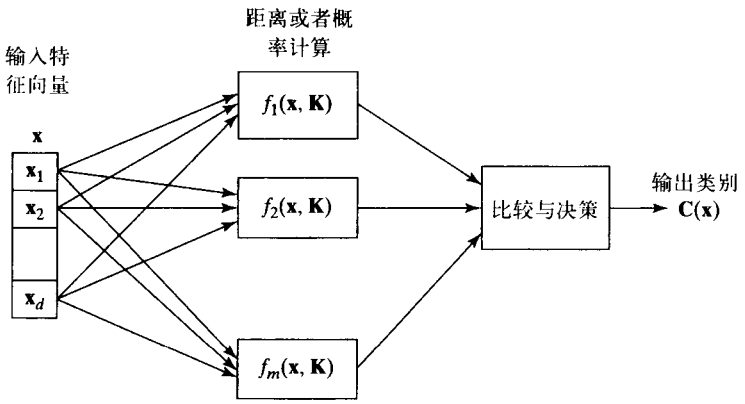


图4-2 分类系统框图。判别函数$f(\boldsymbol{x}, K)$利用训练中得到的知识K对输入特征向量$\boldsymbol{x}$进行计算，并把结果传递给最终的决策阶段，决策阶段输出所属类别

特征抽取算子的另一个常用名字是预处理器（preprocessor）。在传感器和分类器之间，要进行滤波和去噪，这些也是预处理的一部分。第3章已经涉及了一些去噪运算，更多的将在第5章讨论。特征抽取与分类的划分界限多少带有随意性，这种划分更多地是从工程的角度，而 95
不是从应用问题本身的内在属性上进行的。事实上，我们将看到神经网络可以从输入图像直接地一步得到分类结果。

4.2.6 系统错误估计

分类系统的**错误率**（error rate）是衡量系统设计好坏的一个指标。其他的指标还有速度和成本等。速度指每单位时间可被分类的目标数量，成本指硬件和软件的开发成本。性能由错误率和拒绝率决定。如果将所有的输入样本都分成拒绝类别，虽然错误率为0但这毫无意义。

定义26 如果一个输入样本的真实类别为C_j，而分类器将其划分为类别C_i，$i \neq j$，且$C_i \neq C_r$（拒绝类别），那么称分类器产生了一个**分类错误**（classification error）。

定义27 分类系统的**经验错误率**（empirical error rate），指在独立测试数据集上产生的错分样本个数与总的分类样本个数之比。

定义28 分类系统的**经验拒绝率**（empirical reject rate），指在独立测试数据集上产生的拒绝样本个数与总的分类样本个数之比。

定义29 **独立测试数据**（Independent Test Data）指在设计特征抽取和分类算法时未被使用过的样本，且这些样本的类别是已知的，包括来自拒绝类别的样本。

在实际应用中，可用上面的定义测试分类系统的性能。对于系统将处理的样本而言，必须保证用来设计系统的样本和用来测试系统的样本，是具有代表性的样本，并且测试样本必须与设计样本独立。有时假设数据服从一定的理论分布。在这个假设前提下，可以对系统性能进行预测，从理论上算出系统的误差概率，而不是通过测试得到经验错误率。这个概念将在下面讨论。

掌上电脑的手写字符识别模块，对输入字符的正确识别率可达95%。如果用户要对一篇输入文档进行编辑，5%的错误率是可以接受的。有趣的是，这种系统实际上在不断训练用户，同时用户也在训练系统，结果使性能逐渐提高。例如用户认真学习写“8”，目的是为了不与

“6”混淆。然而对于读取存款单上手写数据的银行系统，5%的错误率可能是无法容忍的。

4.2.7 误报和漏报

有些问题是特殊的*二类问题*（two-class problem），两个类别可能是如下情况：（a）好的
96 与坏的；（b）目标出现在图像中与目标没有出现；（c）患有疾病D的人与没患疾病D的人。这里误差具有特殊的意义并且是不对称的。情况（c）最能说明这个问题。如果系统不正确地判断某人患有疾病D，则这个错误称为*误报*或*假阳*（false alarm或false positive）；相反，如果系统不正确地判断病人未患有疾病D，则这个错误称为*漏报*或*假阴*（false dismissal或false negative）。在误报的情况下，可能意味着这个人将经受更多的检查，或者服用并不需要的药品，在漏报的情况下，病人错过了诊断，将得不到治疗，可能导致严重的后果。由于这两种错误的代价显著不同，具有倾向性的决策是有意义的，即为了使漏报最小化，不惜增加误报的次数。如果是为了挑出坏樱桃，情况（a）则不会产生很大的问题。误报可能造成好樱桃被做成馅饼，而不是进入更能体现其价值的生产车间。情况（b）中的误报可能意味着，当场景中事实上并无动静的时候我们打开灯浪费了能源，或者当高速公路上并没有汽车通过时我们错误地算了一辆。情况（b）中的漏报也具有有趣的结果。图4-3是一条典型的*受试者操作曲线*（ROC，receiver operating curve），它反映了误报率与检测率的关系。为了增加正确识别目标的百分比，常常要将本该拒绝的目标错误确定为接受。

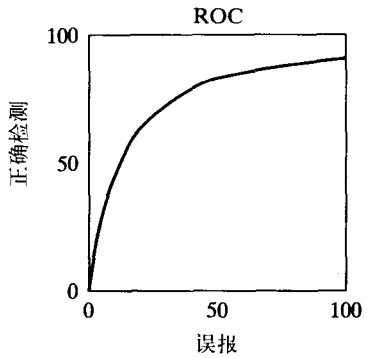


图4-3 受试者操作曲线（即ROC）显示了正确检测率与误报率的关系。总的来说，为了以较高百分比检测出已知目标，则系统误报率会随之升高。采用保守的检测一般能得到较低的误报率。若要全部目标几乎都被正确检测，将导致大百分比的未知目标被错误地分在已知类别中

4.3 查准率与查全率

在文档检索（DR）或图像检索中，检索目标是，根据用户提供的查询特征，检索出感兴趣的C_1类的目标和少量不感兴趣的C_2类的目标。例如，用户想检索出日落或马的图像。衡量这个系统的指标是*查准率*（precision）和*查全率*（recall）。

定义30 文档检索系统的**查准率**，是检索出的相关文档数（确实属于C_1类）与检索
97 出的文档总数（确实属于C_1类的文档数加上实际是C_2类的误报文档数）之比。

定义31 文档检索系统的**查全率**，是检索出的相关文档数与数据库中总的相关文档数之比，即分子是检索出的确实属于C_1的文档数，分母是检索出的属于C_1的文档数与漏报的文档数之和。

例如，假设一个图像数据库包含200幅用户感兴趣的日落图像，用户希望能与查询图像匹配。假设系统检索出200个相关图像中的150幅以及另外100幅用户不感兴趣的图像。这次检索（分类）的查准率是150/250 = 60%，而查全率是150/200 = 75%。如果系统将数据库中的所有图像返回，则查全率是100%，但查准率将非常低。另一方面，如果分类是为了获得低误报率的话，查准率将偏高而查全率将偏低。图像数据库检索将在第8章详细讨论。

4.4 特征表示

无论是理论上还是实践中，一个关键的问题都是在识别过程中对目标怎样表示或者编码？或者说，什么特征对于识别而言是重要的？让我们回到手写字符识别的应用问题上。假设单个字符可以通过连通成分算法分离出来，或者要求把它们写在指定的方框中，那么采用第3章的方法就可以算出下面的特征：

- 以黑色像素数表示的字符面积
- 字符边界框的高度和宽度
- 字符内孔的个数
- 字符的笔画数
- 像素集合的中心（质心）
- 通过像素的最佳轴的方向，即最小惯性轴的方向
- 像素关于最小惯性轴和最大惯性轴的二阶矩

利用常识推理，我们可以根据以上特征值列出一个字符属性表。研究每个字符的许多样例可以使表格更准确。然后进行简短的决策过程对字符进行分类，至少是一组用来比较的字符原型。

表4-1表示10个不同字符的8种特征，现在假设特征计算没有误差。如算法4.1，可用一系列的决策过程对这10类样本进行分类。这个分类结构称为决策树（decision tree）。表中的决策可以很容易由计算机程序实现，特征值也通过计算机程序读取。在决策过程的每一点，都有小部分特征分支到决策过程的其他点。在当前的例子中，每个决策点只用到了一个特征。分支过程表示当接连考虑更多的特征时，可能性集合变小。 98

采用这个例子是因为它比较直观。如果认为迄今所描述的决策过程，与实际手写字符识别系统的有效决策过程是差不多的，想法则不免简单。例如，可靠地定义和计算笔画数就非常困难，在第10章我们将看到一种有效的算法。此外在抽取特征前，需要采用第3章和第5章的方法去除数据中的干扰。在受控工业环境中，可以建立这样简单的分类过程，然后根据样例图像调整定量参数。我们应该对类内特征差异和类间特征重叠进行预测。处理这些差异和重叠的方法将在下面进行研究。

算法4.1 根据三个特征对10个字符进行分类的简单决策过程

输入：特征向量[#holes, #strokes, moment of inertia]

输出：字符类别

```
case of #holes
  0: character is 1, W, X, *, -, or /
    case of moment about axis of least inertia
      low: character is 1, -, or /
        case of best axis direction
          0: character is -
          60: character is /
          90: character is 1
```

```
large: character is W or X or *
    case of #strokes
      0: character is *
      2: character is X
      4: character is W
1: character is A or O
  case of #strokes
    0: character is o
    3: character is A
2: character is B or 8
  case of #strokes
    0: character is 8
    1: character is B
```

99

表 4-1 字符样本集的特征举例

（类别）字符	面积	高度	宽度	孔个数	笔画数	中心	最佳轴	最小惯性
'A'	medium	high	3/4	1	3	1/2,2/3	90	medium
'B'	medium	high	3/4	2	1	1/3,1/2	90	large
'8'	medium	high	2/3	2	0	1/2,1/2	90	medium
'0'	medium	high	2/3	1	0	1/2,1/2	90	large
'1'	low	high	1/4	0	1	1/2,1/2	90	low
'W'	high	high	1	0	4	1/2,2/3	90	large
'X'	high	high	3/4	0	2	1/2,1/2	?	large
'*'	medium	low	1/2	0	0	1/2,1/2	?	large
'-'	low	low	2/3	0	1	1/2,1/2	0	low
'/'	low	high	2/3	0	1	1/2,1/2	60	low

4.5 特征向量表示

比较目标的相似度可以基于它们的向量表示。假设每个目标可以通过d个量度表示，特征向量的第i个坐标对每个目标A都有同样的意义。例如，第一个坐标可能是目标的面积，第二个是在第3章中定义的行矩μ_{rr}，第三个是伸长度等等。量度是实数或者浮点数是很方便的。两个目标特征向量间的相似度或接近度，可用公式（4-1）定义的欧几里得距离描述，如图4-4所示，这在下一节将进行讨论。有时被测向量和存储的类别原型之间的欧几里得距离可以提供实用的分类函数。

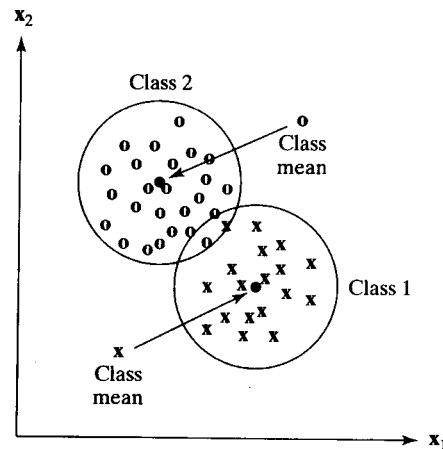


图4-4 两个紧凑类别。用最近均值分类方法，将得到较低的错误率

定义32 两个d维特征向量$\mathbf{x}_1$和$\mathbf{x}_2$的**欧几里得距离**定义为:

$$\|\mathbf{x}_1 - \mathbf{x}_2\| = \sqrt{\sum_{i=1,d} (\mathbf{x}_1[i] - \mathbf{x}_2[i])^2} \tag{4-1}$$

100

4.6 分类器的实现

我们回到经典的范例，将待分类的未知目标表示成原子特征向量。一个识别系统可以基于特征向量以不同的方式设计，这些特征向量可以通过样例学习得到或者由模型预测得到。利用一个训练样本数据库来调查这两种不同的方法。假设有m个类别的目标，不包括拒绝类别，对类别i有n_i个样本向量。在算法4.1的字符识别的例子中，有$m = 10$类的字符，对每一类可能有$n_i = 100$个样本，在这个例子中特征向量的维数是$d = 8$。

4.6.1 最近均值分类

一个简单的分类算法是用类别均值向量，即中心（centroid），来概括每个类别的样本数据，$\mathbf{x}_i = 1/n_i \sum_{j \in \omega_i} \mathbf{x}_{ij}$，其中$\mathbf{x}_{ij}$是来自类$i$的第$j$个样本特征向量。一个未知目标，其特征向量是$\mathbf{x}$，如果它到类别$i$的均值向量要比到其他类的均值向量更近，那么就把它分成类别i。如果$\mathbf{x}$与任何样本均值都不够接近，就将其归为拒绝类。这种分类方法简单快速，当每个类别的样本向量是紧密的且远离其他类别时，这种分类方法也是有效的。简单的二类问题如图4-4所示，其中特征向量维数$d = 2$。类别1的样本向量用x表示，类别2的样本向量用o表示。尽管我们期望当样本结构能很好地表示未来感测的目标结构时，错误率将非常低，但由于每类都有到两类中心等距的样本，错误率将不会是零。现在对图4-2中的功能框进行具体解释：第i个功能框计算未知输入$\mathbf{x}$与第i类训练样本的均值向量间的距离。训练样本构成了类别的知识K。

101

习题4.1 硬币分类

对美国硬币分别进行10采样（1美分、5美分、1角、25美分、50美分、1美元），利用千分尺分别测量60个样本的直径和厚度，精确到0.01in，然后像图4-4那样画出这6个类别的散点图。（测量厚度的地方要保持一致，要么在硬币的中心，要么在硬币的边缘）估计最近均值分类器的错误率。

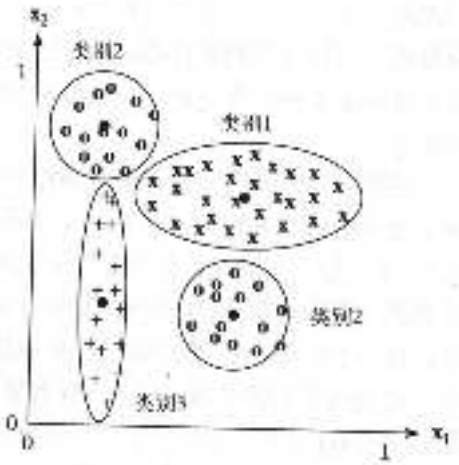


图4-5 具有复杂结构的二个类别。基于最近均值的分类将得到很差的结果

当样本的结构复杂时，分类的难度会增加，图4-5显示的是样本类别可分的情况，但样本结构却使最近均值分类效果不佳，这有多种原因。首先，类别2(o)是多模式的（multimodal），其样本聚集成两个分开的区域，这样总均值落在两个模式的中间，从而无法很好地表示该类别。类别1(x)中有几个样本离类别2的均值比离类别1的均值更加接近。研究这些样本，可以发现类别2的两种模式，并把类别2分成两个子类，用两个均值表示。对于2D情况来说，如图4-5所示的散点图表示起来比较简单。

但当维数d比2大许多时，分析样本的结构就变得非常困难。第二个问题与类别1和类别3的狭长聚类区域有关。显然，类别3中坐标x_2值大的那些样本，离类别2的均值比离类别3的均值更近。同样，即使类别2分成两个模式，类别1中坐标x_1小的那些样本离类别2的子类的均值仍然很近。这个问题可以通过修改距离计算来解决，计算距离时需要考虑样本沿不同维度的散差。

可以对未知特征向量$\mathbf{x}$到类别均值向量$\mathbf{x}_c$的距离进行修改，利用类别c沿维度i的散差，或
102 者称为标准差σ_i（standard deviation）。标准差是方差的平方根。

定义33 从向量$\mathbf{x}$到类别均值向量$\mathbf{x}_c$的**尺度化欧几里得距离**为:

$$\|\mathbf{x}-\mathbf{x}_c\| = \sqrt{\sum_{i=1,d}((\mathbf{x}[i]-\mathbf{x}_c[i])/\sigma_i)^2} \tag{4-2}$$

由于沿不同维度的单位不同，欧氏距离总是需要进行尺度变换。例如对车辆进行分类，特征$\mathbf{x}[1]$ = 长度，以英尺(ft.)⊖为单位；$\mathbf{x}[2]$ = 重量，以磅(lb)⊖为单位。如果不进行尺度变换，欧氏距离将由大数值的磅控制，车辆长度特征在分类中的作用就体现不出来。

在图4-5所示的例子中，类别2的两个模式各自都有均值向量，根据类别对特征x_1和x_2分别进行尺度变换，将得到很好的分类结果。但是多数情况并不这样简单。如果样本分布的椭圆不像图4-5那样与坐标轴平行的话，为了正确计算未知样本到类别均值的距离则需要进行坐标变换。在下面的贝叶斯分类方法中将讨论这个问题。如果样本集在d维空间中结构弯曲，则分类问题更加困难。

4.6.2 最近邻分类

认为未知特征向量$\mathbf{x}$的类别与其最近的样本类别一致，这种方法虽然灵活但计算开销大，这就是**最近邻规则**。即使当类别在d维空间中具有复杂的结构以及当类别有重叠时，最近邻分类也是有效的。对特征向量在空间中的分布模型不需要做任何假设，算法利用的仅仅是已知的训练样本。一种笨方法（算法4.2）是，计算从$\mathbf{x}$到数据库中所有样本的距离，并记住最小的距离。这种方法的优点是新标记的样本可以在任何时候加入数据库。可采用一定的数据结构去除不必要的距离计算。树状或网格状数据集就是这样的两个例子，在本章参考文献中有所描述。

更好的分类决策是检查数据库中的最近的k个特征向量。当$k>1$时，可以对d维空间中的向量分布进行更好地采样。对类别重叠的区域，这尤其有用。已经表明当样本数量趋于无穷大时，即便$k=1$错误率也不超过最优错误率的两倍。理论上，当$k>1$时效果更好；但是有效地利用一个大值k取决于在空间的每个邻域都存在着大量的样本，不需要搜索距离$\mathbf{x}$过远的样本。在一个$k=3$的二类问题中，如果某类别中的3个最近的样本有2个最接近$\mathbf{x}$，则将$\mathbf{x}$分到该类。如果这样的类别数大于2，则有更多的可能组合，决策也更复杂。在下面的算法4.2中，
103 如果多数最近k个样本不属于一个类别，则将输入向量归为拒绝类别。算法假设训练样本集中没有用数据结构。没有数据结构，当样本数n和k增加时，算法速度则变慢。利用有效的样本数据结构的算法在本章结尾的参考文献中可以找到。

⊖ 1ft. = 0.3048m

⊖ 1 lb = 0.4536kg

算法4.2 计算x的k个最近邻并返回多数样本的类别

S是n个已标记类别样本s_i的集合，其中s_i.x是特征向量，s_i.c是它的整型类别标号。
x是待分类的未知输入特征向量。
A是一个数组，可以存储以距离d排序的k个样本。
返回值是在范围[1, m]内的类别标号。

```
procedure K_Nearest_Neighbors(x, S)
{
make A empty;
for all samples s_i in S
{
   d = Euclidean distance between s_i and x;
   if A has less than k elements then insert (d, s_i) into A;
   else if d is less than max A
      then {
            remove the max from A;
            insert (d,s_i) in A;
            }
};
assert A has k samples from S closest to x;
if a majority of the labels s_i.c from A are class c_0
   then classify x into class c_o ;
   else classify x into the reject class;
return(class_of_x);
}
```

4.7 结构方法

只有目标的简单数字特征或符号特征，对于识别来说有时是不够的。例如，考察图4-6所示的两个字符。它们具有相同的边界框，同样数目的孔和笔画，同样的中心，行方向和列方向具有相同的二阶矩，主轴方向相差在0.1弧度内。每个字符都有两个湾（bay），湾是指背景侵入到字符的部分。每个湾都有一个盖（lid），即一条使湾合上的虚拟线段。这两个字符最显著的区分特征是关系：两个湾之间的空间关系。左边的字符，上部湾的盖在下部湾的盖的右边；右边的字符，上部湾的盖在下部湾的盖的左边。这意味着基本特征之间的关系可以作为高层特征使用，并且对于识别来说可能更加有效。*结构模式识别*（structural pattern recognition）就是从这个前提发展出来的。 104

统计模式识别通常用特征向量表示实体，其分量一般是原子值，比如数字和布尔值（真或假）。这些值可度量实体的一些全局特征，如面积或空间矩。字符例子又前进了一步，因为对于每个字符都度量了孔的个数和笔画数目。这暗示着存在寻找和计算孔的算法以及一些可将字符分割成笔画的分割算法。

0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	1	0
0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1	1	0
0	1	0	0	0	1	1	1	0	0
0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	0	0
0	1	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0
0	0	0	1	1	1	1	1	1	0
0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0	1	0
0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0

图4-6 两个全局特征相同但结构不同的字符

在结构模式识别中，一个实体可以由它的基本部件、部件属性、部件间的关系以及全局特征来表示。图4-7显示三个具有类似结构的A字符。每个都可分解成4个主要的笔画：两个水平的和两个竖直（或倾斜）的。每个字符的顶部都有一个孔，或称“湖”，其下有一湾；湖和湾由一个水平的笔画分开。

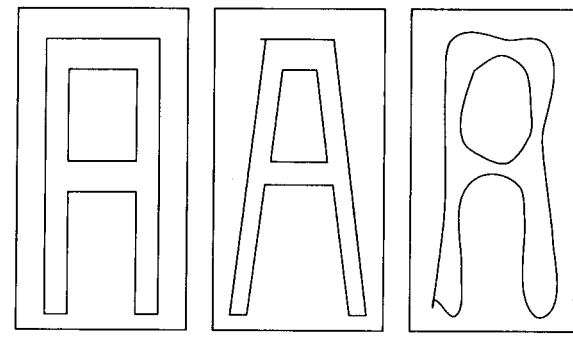


图4-7 三个具有相似结构的A

当基元之间的关系是二值关系时，一个实体的结构描述就可看成图的结构。字符识别中下列的笔画、湾和湖之间的关系是有用的：

105

- CON：定义两个笔画的连接
- ADJ：定义一个笔画的区域与一个湖或一个湾的区域邻接
- ABOVE：定义一个孔（湖或者湾）在另一个之上

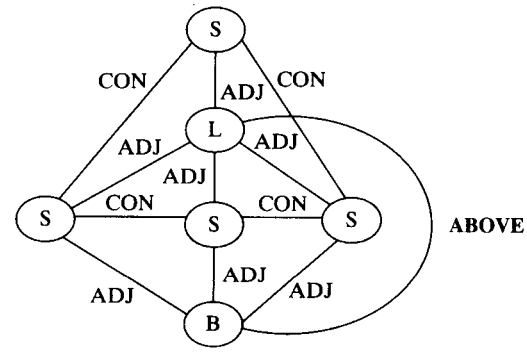


图4-8 字符“A”的图结构表示。“S”、“L”、“B”分别代表边、湖和湾

图4-8说明利用这三种二值关系，对字符“A”结构描述的图表示方法。更高层的关系，如三元甚至四元关系，如果能够进行定义，则用来提供更强的约束关系。例如在湖、湖下的水平笔画和笔画下的湾三者之间就存在着一种约束关系。

结构模式识别通常依靠图匹配的算法来实现，这部分内容包含在第11章中。两个基元之间的关系本身也可看作一个原子特征，可以作为特征向量中的分量，在统计决策过程中使用。一种简单的方法是，仅仅计算两个特殊特征类型（例如，一个湾在水平笔画的下面）之间的某种特定关系在一个模式中出现的次数。计数的整型值则成为识别整个模式的一个特征。

结构方法对于识别包含许多子模式的复杂模式是很有用的。在更高层次的场景理解中它也具有优势，尤其是当有多个目标出现时。一般地说，结构模式识别和本章其他方法涵盖了计算机视觉的大部分内容。本书其余章节将提供更多从2D或3D的目标和场景中抽取特征或部件的方法。

4.8 混淆矩阵

定义34 混淆矩阵通常用来反映分类实验的结果。图4-9给出了一个例子。第i行第j列的元素记录实际类别是i的目标被分成类别j的次数。 106

		模式识别系统输出类别j										
		'0'	'1'	'2'	'3'	'4'	'5'	'6'	'7'	'8'	'9'	'R'
真实的目标类别i	'0'	97	0	0	0	0	0	1	0	0	1	1
	'1'	0	98	0	0	1	0	0	1	0	0	0
	'2'	0	0	96	1	0	1	0	1	0	0	1
	'3'	0	0	2	95	0	1	0	0	1	0	1
	'4'	0	0	0	0	98	0	0	0	0	2	0
	'5'	0	0	0	1	0	97	0	0	0	0	2
	'6'	1	0	0	0	0	1	98	0	0	0	0
	'7'	0	0	1	0	0	0	0	98	0	0	1
	'8'	0	0	0	1	0	0	1	0	96	1	1
	'9'	1	0	0	0	3	0	0	0	1	95	0

图4-9 数字识别的假定混淆矩阵。“R”代表拒绝类别

混淆矩阵的对角线上，即$i = j$处，表示正确分类的次数。完美的分类结果是所有非对角线的元素都是0。非对角线的元素太大，说明类别之间混淆程度太高，就要重新考虑特征抽取过程以及分类过程。如果完成整个测试，混淆矩阵则表示工作系统中预期的错误种类和错误率。在图 4-9所示的例子中，系统的1000个输入向量中有7个被拒绝。标记为类别9的三个输入被不正确地分成了类别4，同时标记为类别4的两个输入被不正确地分成了类别9。总计，有25个输入向量被错误分类。假设测试数据与用来训练分类系统的数据无关，可以得到经验拒绝率是7/1000 = 0.007，总的错误率是25/1000 = 0.025。对类别9的错误率是5/100 = 0.05。

4.9 决策树

当模式识别的任务变得复杂，包含许多不同的可能特征时，将一个完全未知的特征向量与许多不同模式的特征向量比较太耗时间。在医学诊断中这样做甚至是不可能的，因为医学诊断中特征测量通常意味着高成本的、困难的实验室测试。决策树的利用使得特征抽取和分类过程交织在一起。决策树是一个紧凑结构，它每次利用一个（可能是多个）特征将搜索空间分成各种可能的模式。算法4.1的简单决策过程，实现的控制流程如图 4-10所示的决策树。树的节点代表特征向量的不同特征。每个分支节点对该特征的每个可能值有一个子节点。决策过程根据未知特征向量的特定特征值选择子节点。一个子节点可能定义另一个被测试的特征，也可能是一个叶子节点，叶子节点包含由根到叶子整个路径得到的分类结果信息。 107

定义35 二叉决策树是一个二叉树结构，每个节点都关联着一个决策函数。对未知特征向量应用决策函数来决定下一个被访问的节点是当前节点的左子节点还是右子节点。

最简单的情况是采用数值特征值，节点的决策函数仅仅是将未知特征向量的一个特定特征值与阈值比较，如果特征值小于阈值，则选择左子节点，否则选择右子节点。在这种情况下，树的每个分支节点仅需要存储要用的特征和阈值，每个叶子节点存储模式类别的名称。如果决策树的决策过程到达某个叶子节点，则未知特征向量就被分到该模式类别。图4-11表示了这种类型的决策树，它的构造就是为了将所示的训练数据正确分类。

图4-11中树的构造是通过观察数据选择合适的特征和阈值人工完成的。这里的训练数据只是一个简单例子，实际数据可能有更多的特征和更多的样本。对于像医学诊断这样的实际应用，具有几百个特征和成千个训练样本是常见的。在这种情况下就需要决策树能够自动构造。此外，对任意给定的训练样本集，能将它们分类的决策树可能有不止一种，因此根据某种标准选择特征得到最好的决策树是很重要的。最好的决策树具有简单、层数少和测试少的优点。

108 考察图4-12中的训练数据和两种可能的决策树。两棵树都能将训练数据分成两个类别：类别I 和II。左边的树非常简单，它仅用一次比较就可对特征向量做出分类，而右边的树则要复杂些，需要更多的比较。

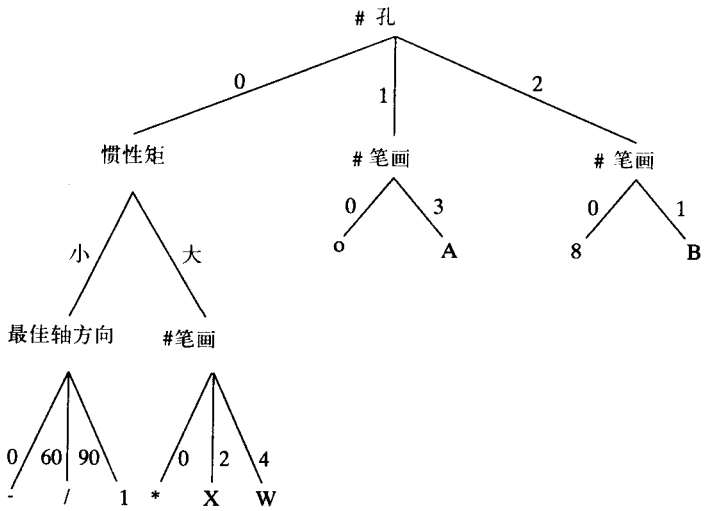
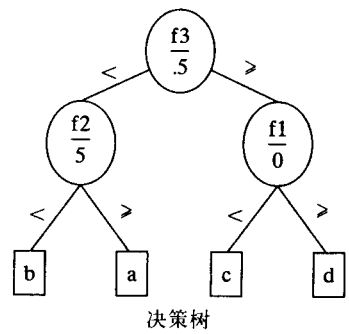


图4-10 实现算法4.1分类过程的决策树

f1	f2	f3	CLASS
3	6	0	a
−5	9	1	c
4	5	1	d
7	4	0	b
−1	10	0	a
2	6	1	d
−2	2	1	c
−1	3	0	b

训练数据



决策树

图4-11 基于节点特征和阈值构造的二叉树

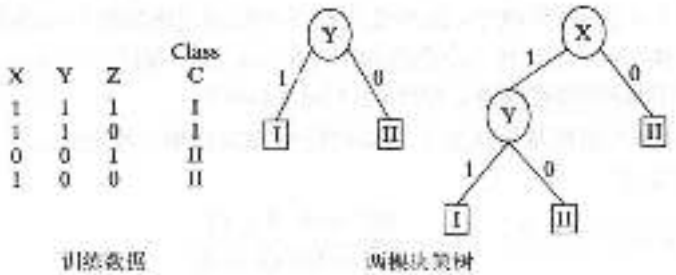


图4-12 两棵不同的决策树，都能对给定的训练样本分类

决策树的自动构造 从训练数据构造最优决策树有许多不同的方法，每种方法都有自己的最优化定义。(深入了解请参见Haralick and Shapiro, vol. I, Chapter 4。) 一种简单有效的方法来自信息论。信息论最基本的概念是熵 (entropy)。

定义36 事件集$x = \{x_1, x_2, \cdots, x_n\}$的**熵**定义为:

$$H(x) = -\sum_{i=1}^{n} p_i \log_2 p_i \tag{4-3}$$

其中每个x_i表示一个事件，p_i是事件x_i的概率。

熵可以解释为信息源的平均不确定性。Quinlan (1986) 曾利用一种基于熵的测度，称为信息增益来估计特征并生成最优决策树。 109

对可能事件集进行熵计算的例子

考察三种可能事件以及它们的概率

$$X = \{(x_1, 3/4), (x_2, 1/8), (x_3, 1/8)\}$$

熵的计算如下:

$$\begin{aligned} H(x) &= -[(3/4)\log_2(3/4) + (1/8)\log_2(1/8) + (1/8)\log_2(1/8)] \\ &= -[(3/4)(-0.415) + (1/8)(-3) + (1/8)(-3)] \\ &= 1.06 \end{aligned}$$

类似地，四个等概率事件的熵值是2.0

$$X = \{(x_1, 1/4), (x_2, 1/4), (x_3, 1/4), (x_4, 1/4)\}$$

$$H(x) = -[4 \cdot ((1/4)(-2))] = 2$$

习题4.2

(a) 计算两个等概率事件的熵。(b) 计算四个可能事件的熵，它们的概率分别是{1/8, 3/4, 1/16, 1/16}。

信息论使得我们能度量一个事件的信息内容。对于每个特征事件，类别事件的信息内容对我们的问题尤其有用。信息内容$I(C; F)$可由下式定义，其中类别变量C可能的取值是$\{c_1, c_2, \cdots, c_m\}$，特征变量$F$可能的取值是$\{f_1, f_2, \cdots, f_d\}$

$$I(C; F) = \sum_{i=1}^{m} \sum_{j=1}^{d} P(C = c_i, F = f_j) \log_2 \frac{P(C = c_i, F = f_j)}{P(C = c_i)P(F = f_j)} \tag{4-4}$$

其中$P(C=c_i)$是类别C具有值c_i的概率，$P(F=f_j)$是特征F具有值f_j的概率，$P(C=c_i;F=f_j)$是类别$C=c_i$和变量$F=f_j$的联合概率。这些先验概率可以对训练数据中的事件频率进行估计得到。例如，由于类别I在四个训练样本中出现两次（见图4-12），则$P(C=I)=2/4=0.5$。由于四个训练样本中的三个的特征值X都是1，则$P(X=1)=3/4=0.75$。

我们可以利用信息内容度量来决定在决策树的根部选择哪个特征最佳。对三个特征X、Y
110 和Z分别计算$I(C,F)$如下：

$$
\begin{aligned}
I(C,X) &= P(C=I,X=1)\log_2\frac{P(C=I,X=1)}{P(C=I)P(X=1)} \\
&\quad + P(C=I,X=0)\log_2\frac{P(C=I,X=0)}{P(C=I)P(X=0)} \\
&\quad + P(C=II,X=1)\log_2\frac{P(C=II,X=1)}{P(C=II)P(X=1)} \\
&\quad + P(C=II,X=0)\log_2\frac{P(C=II,X=0)}{P(C=II)P(X=0)} \\
&= 0.5\log_2\frac{0.5}{0.5\times0.75} + 0 + 0.25\log_2\frac{0.25}{0.5\times0.25} + 0.25\log_2\frac{0.25}{0.5\times0.75} \\
&= 0.311 \\
I(C,Y) &= 0.5\log_2\frac{0.5}{0.5\times0.5} + 0 + 0.5\log_2\frac{0.5}{0.5\times0.5} + 0 \\
&= 1.0 \\
I(C,Z) &= 0.25\log_2\frac{0.25}{0.5\times0.5} + 0.25\log_2\frac{0.25}{0.5\times0.5} + 0.25\log_2\frac{0.25}{0.5\times0.5} + 0.25\log_2\frac{0.25}{0.5\times0.5} \\
&= 0.0
\end{aligned}
$$

特征Y的信息内容是1.0，在确定类别时信息量最大，因此应该被选作第一个特征，在决策树的根结点上测试。在这个简单的例子中，两个类别完全可分，决策树用一个分支节点就完成了分类。更一般的情况下，在树的每个分支节点，当选用的特征不能完全将训练样本集分到合适的类别时，样本集根据在这个节点的决策被划分成子集。对于仍包含多个类别的样本子集，在相应的子节点递归调用决策树的构造算法。

这里描述的算法与图4-10的决策树相同，这是一棵通用的树，在每个节点上被测特征的每个可能的取值都有分支。为了适应如图4-11所示的阈值类型的二叉树，对每个可能的阈值必须考虑每对特征-阈值对的信息内容。看起来可能的集合有无穷多，但对训练样本中出现的每个特征都只有有限的几种取值，这个有限集合就是需要考虑的全部。

上面的例子非常简单，针对几十个甚至几百个特征，自动构造实用的决策树是完全可能
111 的。再次考虑字符识别问题，但这次是对于更困难的手写字符。这类字符的特征是4.6节讨论过的*湖*、*湾*和*盖*。湖是字符中的孔（标记为0的区域，完全被标记为1的字符像素所包围的）。湾是背景侵入字符的部分（标记为0的区域，部分被标记为1的字符像素包围）。盖是可用来闭合湾的线段。图4-13a表示的手写字符6，b中是它的湾特征和湖特征，c中是它的盖特征。第3章描述的数学形态运算可用来抽取这些基本特征。从这些基本特征，可计算出下面的数值特征：

- **湖数**：抽取出的湖的个数
- **湾数**：抽取出的湾的个数
- **盖数**：抽取出的盖的个数
- **湾在湾之上**：布尔特征，如果有任何一个湾完全位于另一个湾之上，则该值为真
- **盖在湾右侧**：布尔特征，如果存在一个盖完全位于一个湾右边，则该值为真
- **湾在湖之上**：布尔特征，如果存在一个湾完全位于一个湖之上，则该值为真
- **盖在图像底部**：布尔特征，如果任何一个盖的最低点在整个字符最低点的像素集合之内，则该值为真

当训练样本充足时，可利用这些特征构造一棵能对手写数字分类的决策树。图4-14显示数字0－9的训练数据样本集。



a) b) c)

图 4-13

a) 手写字体"6"的图像
b) 上部是用形态图像处理抽取的湾，下部是湖
c) 进一步用形态处理得到的湾的盖

习题4.3 构造决策树

训练数据如图4-14所示，写一个程序，它利用信息内容来构造决策树，对10个数字进行分类。基于全部40个样本构造的树，对训练数据的分类效果如何？如果基于后20个样本构造树，用前20个样本进行测试，情况会怎样？

习题4.4

(a) 怎样利用第3章的形态图像处理方法抽取一个湖？

(b) 如果已经识别出一个湾，怎样抽取它的盖？

4.10 贝叶斯决策

考虑如何用概率分布的知识，进行分类决策使期望的错误率最小。假设从暗红色樱桃的红外图像中得出测量值x，由该值判断樱桃的好坏。设好樱桃为类别ω_1，坏樱桃为类别ω_2。另外假设已从大量的好樱桃和坏樱桃中研究了许多的表面元素，因此我们已有分布函数的知识，如图4-15所示。右边的曲线，$p(x|\omega_1)$表示好樱桃表面样本测量值x的分布。左边的曲线，$p(x|\omega_2)$表示坏樱桃表面样本测量值x的分布。对数据进行规范化，使得在每条曲线下的面积是1.0，两条都表示概率分布。(坏的组织含有水，比好的组织吸收更多的红外辐射，因此反射系数很可能更低些。水分含量不同，表面颜色的暗度不同，导致了两类分布的重叠，即一些深暗的好樱桃和一些明亮的坏樱桃具有类似的反射。)

湖数	湾数	盖数	湾位于湾之上	盖位于湾右侧	湾位于湖之上	盖位于图像底部	类
1	0	0	F	F	F	F	0
1	0	0	F	F	F	F	0
0	0	0	F	F	F	F	1
0	2	2	F	T	F	T	2
0	2	2	T	F	F	F	3
1	1	1	F	F	F	T	4
1	1	1	F	T	T	F	6
0	2	2	T	T	F	F	2
0	2	2	T	T	F	T	4
0	1	1	F	F	F	F	7
0	0	0	F	F	F	F	1
1	0	0	F	F	F	F	0
1	1	1	F	F	F	F	9
0	2	2	T	T	F	F	2
1	1	1	F	F	F	F	9
0	1	1	F	F	F	F	1
0	2	2	T	F	F	T	4
0	2	2	T	T	F	F	5
1	1	1	F	T	T	F	6
0	2	2	T	F	F	F	3
0	1	1	F	F	F	F	1
1	0	0	F	F	F	F	0
0	2	2	T	T	F	F	5
1	1	1	F	T	T	F	6
0	1	1	F	F	F	T	7
2	0	0	F	F	F	F	8
1	1	1	F	F	F	F	9
1	0	0	F	F	F	F	0
2	0	0	F	F	F	F	8
1	1	1	F	T	T	F	6
0	2	2	F	T	F	F	7
1	1	1	F	F	F	F	9
1	0	0	F	F	F	F	0
0	2	2	T	F	F	F	3
0	2	2	T	T	F	F	5
0	2	2	T	T	F	F	2
0	2	2	T	T	F	F	2
0	1	1	F	F	F	F	1
0	1	1	F	F	F	F	7
0	2	2	T	T	F	F	5
0	2	2	T	F	F	T	4
0	2	2	T	F	F	F	3

图4-14 手写字符的训练数据

如果好樱桃与坏樱桃出现的概率相等，并且分类错误的代价是相同的，那么可以做这样的决策：当$x > t$时，x属于类别ω_1；否则属于类别ω_2。对于这样的决策规则，t右边的阴影部分表示（两倍的）漏报率，这是把坏樱桃接受为好樱桃的高测量值x的概率。面积之所以是漏报率的两倍是因为已经假设每类的先验概率是0.5，这样每种密度应该缩小使得曲线下的总面积是0.5。t左边的阴影部分表示（两倍的）误报率，这表示好樱桃由于特征$x < t$被分成坏樱桃的概率。由于假设在好樱桃和坏樱桃系统的输入中出现的概率是相等的，所以每条曲线实际上仅仅表示总概率的一半，所有显示的面积是它们实际大小的两倍。总误差是曲线下阴影部分的面积总和。重要的是，将决策阈值t向左或向右移动都将导致阴影面积扩大即错误率增加。

112～114

上面的例子仅考虑了一种特殊情况，即两个类别的概率相同，错误的代价也是相同的。现在将方法扩展到可以覆盖m个类别的情况，这m个类别都有各自不同的先验概率。为简单起

见，我们仍保留所有错误的代价相同的假设。利用贝叶斯决策，可以将目标分到它最可能属于的类别。

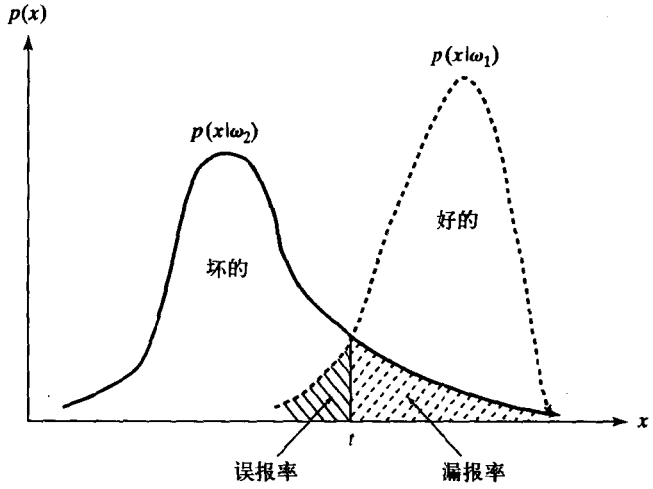


图4-15 亮度测量值x的条件分布，关于x是取自好樱桃还是坏樱桃

定义37 贝叶斯分类器基于观测的特征，将目标划分为它最可能属于的类别。

为了计算观测值x的概论，需要知道下面的分布：

类别条件分布：每个类别ω_i的条件概率$p(x|\omega_i)$ (4-5)

先验概率：每个类别ω_i的先验概率$P(\omega_i)$ (4-6)

无条件分布：$p(x)$ (4-7)

如果所有类别ω_i之间都是不相交的，给定每类的先验概率和每类x的分布，可以应用贝叶斯规则计算每类的后验概率

$$P(\omega_i \mid x) = \frac{p(x \mid \omega_i)P(\omega_i)}{p(x)} = \frac{p(x \mid \omega_i)P(\omega_i)}{\sum_{j=1,m} p(x \mid \omega_j)P(\omega_j)} \tag{4-8}$$

回到图4-2的分类器框图，在每个类别的计算方框内，令$f_i(x, K) = P(\omega_i|x)$，这由公式（4-8）的贝叶斯规则可以计算为$p(x|\omega_i)P(\omega_i)/p(x)$。由于$p(x)$对于所有类别的计算都是相同的，可以忽略它，分类决策定为选择最大的$p(x|\omega_i)P(\omega_i)$。为设计贝叶斯分类器，必须具备知识**K**，在这里是每个类别的先验概率$P(\omega_i)$以及类别条件分布$p(x|\omega_i)$。这些知识可以帮助设计最优决策。建立这些先验概率的知识通常非常困难。例如，如何知道进入分类器的樱桃是坏樱桃的概率？如果这个概率随着天气和采摘成员变化，那么获取条件变化所需要的信息要耗费太多的采样工作。 115

分布参数模型

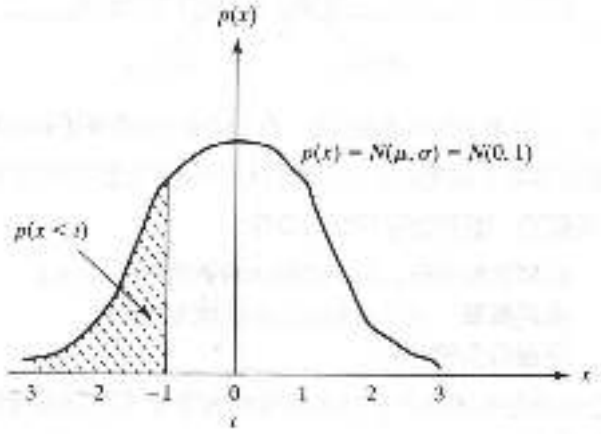
在实际中，必须以某种方式实现$p(x|\omega_i)$的计算。一种经验方法是量化x的范围，在每个时间间隔记录x在该范围内出现的频率，将结果存储在一个数组或直方图中。根据这些数据可以拟合出一条光滑的样条函数，对所有实数都可生成有效的概率函数。注意需要将结果规范化使得x的所有值之和是1.0。如果观察到x的分布服从某种已知的参数模型，就可以利用少数几个可表征的参数来表示分布。泊松分布、指数分布以及正态（或高斯）分布都是经常使用的模型。正态分布是著名的“钟形曲线”，大学课程中，经常用它来评定分数等级。

定义38 正态分布由均值μ和标准差σ确定，定义如下：

$$p(x) = N(\mu, \sigma)(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right] \tag{4-9}$$

χ^2检验可以用来确定采样数据是否能用正态分布（或其他分布）模型。这可参考统计学方面的文献（例如，文献中Hogg和Craig所编教材）。从样本数据中可以很容易算出均值和标准差，从而得到正态分布模型。由于正态分布简单并具有其他一些方便的数学特征，多数实现都利用正态模型，即便它对实际数据只是一个大概的近似。

利用参数模型（parametric model）来模拟类别样本的分布，比如正态分布，那么图4-2实现的贝叶斯决策中的概率比较就可以采用简单的公式。一旦对每个类别i都已知分布$p(x|\omega_i)$，可利用图4-16对x设置阈值以便区分类别。而且，概率模型可以直接用来估计错误概率，因为现在可以用公式表示图4-15所示的错误区域。



t	p(x<t)	t	p(x<t)	t	p(x<t)
-3.0	0.0014	-2.0	0.0227	-1.0	0.1587
-2.9	0.0019	-1.9	0.0287	-0.9	0.1841
-2.8	0.0026	-1.8	0.0359	-0.8	0.2119
-2.7	0.0035	-1.7	0.0446	-0.7	0.2420
-2.6	0.0047	-1.6	0.0548	-0.6	0.2743
-2.5	0.0062	-1.5	0.0668	0.5	0.3085
-2.4	0.0082	-1.4	0.0808	-0.4	0.3446
-2.3	0.0107	-1.3	0.0968	-0.3	0.3821
-2.2	0.0139	-1.2	0.1151	-0.2	0.4207
-2.1	0.0179	-1.1	0.1357	-0.1	0.4602
0.0	0.5000				

利用对称性来扩展表中从0.0到3.0的数值，例如

```
p ( -2.0 < x < 1.0 ) = p ( -2.0 < x < 0.0 ) + p ( 0.0 < x < 1.0 )
= [ p ( x < 0.0 ) - p ( x < -2.0 ) ] + p ( -1.0 < x < 0.0 )
  [ 0.5000 - 0.0227 ] + 0.1587 = 0.6360.
```

图4-16 正态分布，其中均值$\mu=0$，标准差$\sigma=1$

习题4.5

如何估计以下情况的先验概率？（a）超市中一个顾客会买菠菜；（b）在ATM机前的人是假冒的；（c）一个刚摘下的暗红色的樱桃是坏的；（d）四十多岁的人患有胃癌？

习题4.6 硬币分类B

习题4.1曾要求测量美国硬币的直径和厚度，利用1美分、5美分和1角这些硬币的数据，(a) 设特征x是硬币的厚度，计算这三类硬币的均值和标准差。是否存在阈值t_1和t_2可以分开这三个类别并使得总的错误率小于5%？给出答案并进行解释。(b) 设特征x是硬币的直径，重复 (a) 的计算。 116

4.11 多维数据决策

在当今处理的许多实际问题中，维数$d = 10$或者更多是很常见的。如前所述，最近邻分类过程适用于具有任意维数d的特征向量。对于多维特征向量x也可用参数概率模型，关于涉及的数学处理方法，读者可查阅相关文献。这里我们简要讨论多维结构的概念，读者可借助参考文献进行相关问题的深入研究。

考虑二维空间中的两类样本，每类的形状都像一棵树，两棵树在一起成长。类别1的数据外形像一棵枫树，可用一个大的球面近似。类别2的数据外形像一棵松树，比枫树高并且细，它可用一个椭球近似，椭球的主轴比其他两个次轴大许多。类别1的样本对应于枫树的叶子，类别2的样本对应于松树的针。另外，假设松树穿过枫树的树冠成长并超过它。将一个未知的 117
3D特征向量x分类的问题要求与3D空间中已知的样本结构相联系。如果x在枫树的树冠内，又不接近松树的树干，那么x很可能是枫树（类别1）。另一方面，如果x在枫树的树冠外或者接近松树的树干，那么x则很可能是松树（类别2）。在空间中有些位置模棱两可，这是由于两类样本存在重叠。最重要的一点是对d维空间中样本结构的理解不仅有助于做出有根有据的决策，而且有助于理解发生的错误。空间结构可以用大型样本数据库表示，其中用数据结构概括样本的子集，或者用样本子集的参数几何模型表示空间结构。

第二个3D例子也同样具有启发意义。假设类别1的样本结构为弹簧状，或螺旋状，类别2的样本结构为铅笔状，或杆状，位于螺旋的轴的位置。（或者想像两个弹簧缠绕在一起，因为它们可能被放在同一个硬件仓库的储藏箱里。）这两个类别高度结构化，事实上是一维的，一旦已知它们的结构则能够很容易地分开。最近均值分类器在这里不起作用，因为均值是相同的。沿各维度作尺度变换也无能为力，因为样本仍然缠绕在一起。最近邻分类器虽然可以，但是需要存储大量的样本。一种现实的替换方法是用许多杆的连接来逼近螺旋的数据。杆可以简单地用一个圆柱体表示，分类可以通过简单的几何计算检查未知的x是否位于任何圆柱体之内来进行。另一个更好的替换的方法是对螺旋形采用一个公式描述，其参数为它的轴、半径和攀升率。

在将这些想法付诸实施时，我们注意到一些重要的观点。首先，捕捉样本数据的内在结构和维数非常重要，结构可用几何或统计模型表示，模型允许通过简单计算进行决策，而不是搜索一个巨大的无结构的样本数据库；其次，数据的本质结构与度量空间的轴不一定一致。比如，松树或螺旋形的轴不一定是沿坐标轴x[1]、x[2]或者x[3]的。发现结构或者坐标变换的方法将在参考文献中给出。

习题4.7 阈值化错误率

为了把目标从背景中分割出来，要对图像进行阈值化处理。本题研究面积计算的潜在错误。提出以下假设：

- 图像具有512 × 512个像素点，图像中的目标恰好覆盖3932个像素点。（不存在混合像素点；目标的边界与像素的边界严格对应。不存在由于焦距引起的邻域像素的模糊现象。）
- 由于表面变化，图像中目标像素的亮度服从分布N（80, 5）（这表示均值为80标准差为5的正态分布）。
- 类似地，背景亮度分布为N（50, 10）。
- 任何像素点的灰度值与其邻域像素的灰度值无关。

1. 如果图像阈值取为70，当**I** [**r**, **c**] >= 70，则**LABEL**[**r**, **c**] = 1；否则**LABEL**[**r**, **c**] = 0，那么预期被标记为目标的像素点的个数是多少？
2. 图像中哪些被标记为背景的像素事实上是目标？（这些是漏报的）
3. 图像中哪些被标记为目标的像素事实上是背景？（这些是误报的）
4. 计算目标的面积时，只统计被标记图像中值为“1”的像素点的个数，那么预期的错误百分比是多少？
5. *假设对标记图像去除盐椒（salt and pepper）噪声，方法是如果某像素的4邻点都具有与其不同的值，则用邻域像素点的值代替该像素的值，通过这种方法可以创建一幅新的图像。在这个新标记的图像中统计值为“1”的像素点个数作为目标的面积，那么预
118 期的错误百分比是多少？

4.12 机器学习

我们来总结一下要点：本章讨论的方法提供了一种机器学习的基本类型，称为*监督学习*（supervised learning）。第16章中的物品分类就是一个很好的应用实例。我们已经假定对需要区分的所有类别均可获得有标记的样本；换句话说，教师知道数据的结构以及期望的输出。也可采用*无监督学习*或*聚类*的方法。在无监督学习中，机器还需要决定类别的结构，即类别是什么样、有多少类别等。读者可借助参考文献来研究这个问题。

采用最近邻分类时，所有的数据样本都被记忆在内存中，需要识别未知目标时则要访问内存。机器的识别行为完全由训练数据决定。在采用参数模型时，类别模型的参数从训练数据中学习得到，用来建立可能目标的整个空间模型。下面一节是选学内容，介绍监督学习技术，通过设计判别函数来模拟有机体的神经元。目前机器学习是研发的热点领域，读者最好查阅更多的文献进行更深入研究。

4.13 人工神经网络*

有机体神经元具有很强的学习能力，为了在机器学习中运用这种能力，人们进行了大量
119 的研究工作。图4-17是神经元的简单模型。虽然这个模型仅仅是对生物学神经元的一种近似，但是它已经成为非常重要的计算模型。这些模拟神经元组成的网络，即*人工神经网络*或ANN，已经证明在许多机器视觉问题上非常有用，特别是因为它们的学习能力。人工神经网络能够学习多维空间中样本的复杂结构，与最近邻分类方法相比需要较少的内存，它还可实现海量的并行计算。这里对人工神经网络只做简单介绍，要了解更多关于这个广阔且发展以迅速的领域，请参考相关文献。

4.13.1 感知器模型

如图4-17所示，神经元（AN）通过树突与其他神经元或传感细胞相连，接收d个输入**x**[**j**]。细胞体将每个输入乘以增益因子**w**[**j**]，并将结果加起来。神经元的输出y沿着轴突送出，轴突

有很多分支与树突联接，为神经元网络中的其他神经元提供输入。将输入的加权和送入细胞，细胞的输入输出模型可以是阶跃函数，即把输入的加权和与阈值t比较，如果加权和超过阈值，则输出$y=1$；反之，输出$y=0$。这个二值输出函数如图4-17的左下角所示。为得到介于0和1之间的光滑输出，可以采用图中右下角的sigmoid函数。参数β是在$x=t$处的斜率或增益，它对输入乘以一个系数，从而算出$x=t$附近的输出值。为了便于表示和编程，神经元阈值t的负值 120
存储成w[0]，它对应的输入x[0]设为1.0，如公式（4-10）所示。神经元通过调整输入向量x的权值w[j]进行学习。

$$y=g\left(\sum_{j=0,d} w[j]x[j]\right) \tag{4-10}$$

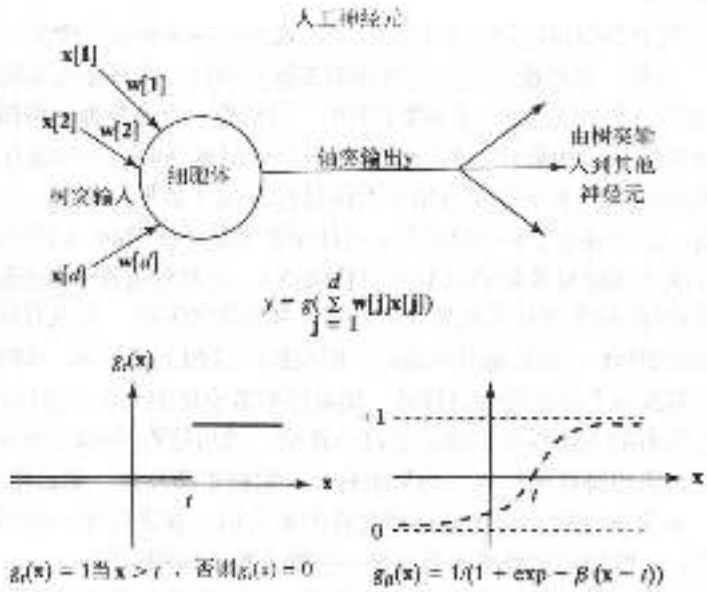


图4-17 神经元的简单模型以及两种输出条件函数

习题4.8 神经元模拟

（a）研究神经元的行为。它的两个输入是x[1]和x[2]，权值w[1]=0.8，w[2]=0.3，阈值t=1.0，输出函数采用阶跃函数$G(x)$，x[1]和x[2]的值分别取为0，1，2和3，共有16种可能的输入组合。绘出输出结果，当输出是1时，则画“1”；当输出是0时，则画“0”。

（b）绘制另一个图，这次采用光滑的S型函数，$\beta=4$。问题中的其他元素都相同，注意现在的输出将是实数，而不简单的是0或1。

习题4.9 用神经元实现AND、OR和NOT门运算

1. 设计具有“或”门功能的神经元。令x[1]和x[2]是仅具有布尔值0或1的两个输入。当这两个输入的值都为1时，神经元的输出就是1；当两个输入的值都为0时，神经元的输出是0。如前所述，x[0]=1，阈值是−w[0]。确定神经元的所有连接权值。在2D坐标系中画出4种输入组合并说明由AN实现的决策边界。

2. 设计具有“与”门功能的神经元，重复上面的问题。仅当两个输入的值均为1时，“与”门的输出才为1。

3. 说明单个神经元是如何实现“非”门功能的。输入的值为0，其输出则为1；若输入的值为1，其输出则为0。

简单人工神经元的计算能力，无论是在理论上还是在实际中都非常有趣。从习题4.9知道，一个神经元可模拟AND、OR和NOT门运算，其重要性在于任何布尔函数都可以通过几个神经元的级联实现。从习题4.10知道，单个神经元甚至不能实现简单的异或功能，许多其他重要的功能也无法由单个的神经元实现。Minski和 Pappert于1987年发表的文章使一段时期内关于神经元的研究陷于低潮。几年后出现了多层人工神经网络的成功实例，这样的ANN更为复杂，不受计算能力的限制。人工神经网络计算能力的理论问题留给读者进一步研究，下面看单个神经元的简单训练算法。

假设2D样本的两个类别可以用一条直线分开，比如图 4-4所示的情况，只要把重叠区域
121 中的“X”和“O”去除。显然能够得到分割线的参数并构造一个神经元来完成分类决策。对于3D样本，我们可用一个分割平面。在d维空间中，则需要一个超平面，但概念上和构造上都是类似的。令人称奇的是，如果对二类问题存在一个分割超平面，那么就存在简单的学习算法能够从两个类别的训练样本中找到该超平面的计算公式（参见算法4.3.）关于算法收敛到分割超平面的证明，已经超出了本书的范围，具体请参考Duda和Hart（1973）的文献。

感知器学习算法开始时权值是随机的（包括阈值）。在对标记样本x的迭代学习中，权值得到调整。对于类别1的样本当权值向量（感知器）使输出为正时，则从权值向量中减去$gain*x$。类似地对类别2的样本，如果输出为负时，则权值向量加上$gain*x$。策略是根据对当前样本中的学习，将分割线向着合适的方向移动。增益控制着变化的大小。通过过程*training_pass*实现这些调整。当所有的训练样本都被处理过一轮后，调用过程*check_samples*来计算有多少个样本被具有当前权值的神经元错分。如果所有样本都被正确分类，那么算法就找到一个解，算法退出。否则，如果训练的次数仍未达到允许的最大值，就进行下一轮的训练，这次的增益是前一轮的一半。一般算法实现中还有其他一些细节性的控制方法。

图4-18是实现感知器学习算法程序的输出结果。通过构造，所有类别1的样本在直线$y = 1-x$下面，而所有类别2的样本在该直线上方。两类样本之间存在一段间隙。算法非常快速地找到直线$-1 + 5/4x_1 + 5/4x_2 = 0$来分割这两个类别。如输出所示，每个类别1的样本都产生一个负响应，每个类别2的样本都产生一个正响应。

虽然基本学习算法很简单，但也存在一些难点。（1）样本以什么样的次序排列可以加快学习速度？理论表明，为保证收敛性，每个样本可能要出现任意多次。有的算法对某个给定的样本重复训练直至它被正确分类，然后再继续下一个样本。（2）所用的增益因子影响收敛性。例子程序中在学习完所有的训练样本后将增益因子减半。（3）为使将来样本的分类能有更好的性能，要求算法在两类之间搜索一条最佳直线，而不是任意的分割线。（4）当训练耗费了很长时间时，怎么知道这是不是因为样本本身不可分？（5）怎样修改学习算法，使得当样本线性不可分时，能找到使分类错误最小的一条直线？这些问题留给读者进行课外研究和实验。

习题4.10 感知器实现“异或”

绘出下列输入数据并找出一条分割线，说明单个神经元无法做出“异或”决策。对输入

(0, 1) 和 (1, 0) 要产生正响应，对输入 (0, 0) 和 (1, 1) 要产生负响应。 122

习题4.11 感知器学习算法编程

编程实现对任意d维特征向量x的感知器学习算法。用2D向量进行测试，并说明它能够学习OR门和AND门，但无法学习XOR门。在下列两类合成的3D样本上进行测试：类别1是第一个卦限（x_1、x_2、x_3全为正）内的一些随机点集，类别2是其他任意卦限内的点集。

```
Class 1 = { ( 0 , 0.5 ), ( 0.5 , 0 ), ( 0 , 0 ), ( 0.25 , 0.25 ) }
Class 2 = { ( 0 , 1.5 ), ( 1.5 , 0 ), ( 0.5 , 1 ), ( 1 , 0.5 ) }
Initial gain= 0.5
Limit to number of passes= 5
Number of samples in Class1= 4; Number of samples in Class2= 4

Training phase begin with weights:        1     0.5     0.5

-----Adjust weights------: gain= 0.5
 pattern vector x =        1       0       1.5
Input Weights:       -1     0.5     0.5
Output Weights:      -1     0.5     1.25

-----Adjust weights------: gain= 0.5
 pattern vector x =        1       1.5     0
Input Weights:       -1     0.5     1.25
Output Weights:      -1     1.25    1.25

Weight Vector is:    -1     1.25        1.25     Classification for Class = 1

          Input Vector x / Response   / Error?

        1        0      0.5     -0.375     N
        1      0.5        0     -0.375     N
        1        0        0         -1     N
        1     0.25     0.25     -0.375     N

Weight Vector is:         -1     1.25     1.25     Classification for Class = 2

          Input Vector x / Response   / Error?

        1        0      1.5      0.875     N
        1      1.5        0      0.875     N
        1      0.5        1      0.875     N
        1        1      0.5      0.875     N

Errors for Class1: 0 Errors for Class2: 0
Final weights are:       -1     1.25     1.25
```

图4-18 计算机感知器学习程序的输出，学习两个线性可分类别间的线性判别决策

4.13.2 多层前向网络

前向神经网络是人工神经网络的一种特殊类型。网络中的每个神经元都位于某层l上。层l上神经元的输入来自层$l-1$上的所有神经元的输出，层l的输出又是层$l+1$上所有神经元的输入， 123
见图4-19。可以将最低层即第1层神经元的输入作为传感器的输入，将最高层L层神经元的输出作为分类结果。当输出y[c]最高时，则认为是类别c；或者所有的输出可认为是一种模糊分类的结果。层1和层L之间的神经元称为隐层神经元。由于任何一层到它的前一层没有反馈，所以称为"前向"。因此，ANN的工作类似于组合电路，它的输出是根据输入算出的，而没有利用对先前输入序列的记忆。 124

算法4.3 两个线性可分类别的感知器学习算法：计算权值向量w，区分类别1和类别2

S1和**S2**分别是n个样本的集合。

gain是当**x**被错分时调整**w**的比例因子。

max_passes是学习所有训练样本的最大遍数。

```
procedure Perceptron_Learning(gain, max_passes, S1, S2)
{
input sample sets S1 and S2;
choose weight vector w randomly
//设NE是错分类的样本总数。
NE = check_samples (S1, S2, w);
while (NE > 0 and passes < max_passes )
 {
    training_pass (S1, S2, w, gain);
    NE = check_samples (S1, S2, w);
    gain = 0.5*gain;
    passes = passes + 1;
 }
report number of errors NE and weight vector w;
}
 procedure training_pass (S1, S2, w, gain);
{
 for i from 1 to size of Sk
 {
    //标量积或点积 ∘ 计算AN 。
    take next x from S1;
    if (w∘x > 0) w = w−gain ∗x;
    take next x from S2;
    if (w∘x < 0)w = w + gain *x;
 }
}
```

前面的习题说明了单个的人工神经元可以实现等同于AND、OR和NOT逻辑门的运算。这意味着神经元的前向层可以实现任意的逻辑函数。这种网络功能强大，能模拟许多计算机程序的行为。而且由于神经元不局限于布尔值，它们能表示d维空间非常复杂的几何划分，因此能从训练样本中自适应地学习这种结构。图4-20说明前向神经网络如何进行异或计算，这一点对于单个神经元是不可能的。像习题中那样，第1层神经元实现AND和OR功能。最后一层只有一个神经元，它的权值向量为$\mathbf{w} = [0, -1, 1]$，当且仅当$-1x_1 + 1x_2$为正时输出为1。为了明白多层ANN是怎样实现复杂样本集的几何结构的，读者应当做后面的习题。

125

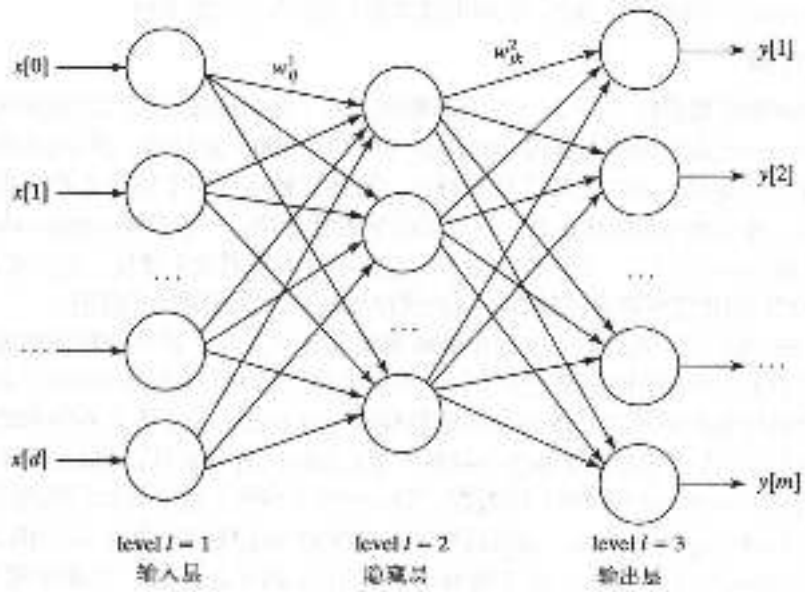


图4-19 多层前向人工神经网络。层l上神经元的输入来自层$l-1$上的所有神经元的输出，其输出又是层$l+1$上所有神经元的输入

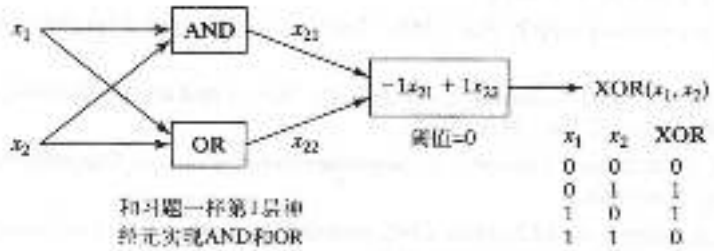


图4-20 利用前向人工神经网络实现XOR

在学习过程中，对于一系列样本，前向神经网络通过调整权值进行学习。称为反向传播算法的学习方法，从输出层向输入层反向传播分类错误。为了使输入/输出关系平滑，采用的是sigmoid传输函数而不是用阈值控制输出。反向传播算法的实现和使用可以在参考文献中找到。近来，反向传播算法与其他学习算法在不同方面的成功应用给模式识别和机器学习领域注入了新的活力。读者可查阅文献学习其他类型的网络以及它们的诸多应用。

习题4.12 用ANN实现2D三角形分类结构

构造一个前向人工神经网络，对于落在三角形（顶点是(3, 3), (6, 6)和(9, 1)）中的2D点x，它的输出是1；对于落在三角形之外的所有2D点，它的输出是0。利用阶跃函数$G(x)$。提示：在第1层上采用三个神经元来建立类别的边界，即三角形的边；第2层用一个神经元对第1层的三个输出进行综合。

习题4.13 用ANN实现三类别的分类

说明如何用2层前向网络，识别下列不相交类别的2D输入向量，类别1的向量在三角形内，类别2的向量在正方形内，类别3的向量在五角形内。利用上一个习题的结果，说明存在一个ANN可以识别每个类别与其他两个类别（不需要利用具体的直线或公式，只需调用*trangle*、

*square*和*pentagon*子网络即可）第2层的输出就是第1层输入向量的类别。

4.14 参考文献

Duda和Hart的早期著作（1973）（它有新版本）对于经典的统计模式识别问题和方法仍然极具价值。另一本经典的教科书是由Fukunaga（1972, 1990）著作的。对于d维特征向量的贝叶斯分类器理论的发展，可以查阅这些教科书，这些文献还说明了协方差矩阵在多维结构建模中的重要性。关于概率和统计，有几本优秀的导论性著作，一本是Hogg和Craig（1970）合著的，另一本是Feller（1957）的著作。参考这些文献可以研究分布理论，如正态或卡方分布。Jain等人（2000）的论文对许多近期有价值的统计模式识别文献进行了综述。

协方差矩阵的特征向量给出了空间中椭球聚类的自然方向，对应的特征值表示样本的散差。另外还给出了几种表征概率密度的方法。Schalkoff（1992）和Schurmann（1996）的近期
126 著作覆盖了除统计模式识别之外的句法和结构模式识别方法。关于人工神经网络全面而简要的介绍可以在Jain等人（1996）的综述中找到，更广泛的内容可参见Haykin（1994）、Hertz等人（1991）以及Schurmann（1996）的著作。Tanimoto（1995）的著作结合其他学习机制，是一篇很好的神经网络方面的著作，它还说明了怎样利用句法特征作为输入，用LISP语言实现了感知器学习和反向传播算法。关于感知器能做什么的理论研究，读者可参考Minsky和Papert（1989）的著作或者1969的原始版本。

1. Duda, R. O., and P. E. Hart. 1973. *Pattern Classification and Scene Analysis.* John Wiley & Sons, New York.
2. Duda, R. O., D. Stork, and P. Hart. 2000. *Pattern Classification.* John Wiley & Sons, New York.
3. Feller, W. 1957. *An Introduction to Probability Theory and Its Applications, vols. I and II.* John Wiley & Sons, New York.
4. Haykin, S. 1994. *Neural Networks: A Comprehensive Foundation.* Macmillan College Publishing, New York.
5. Hertz, J., A. Krogh, and R. Palmer. 1991. *Introduction to the Theory of Neural Computation.* Addison-Wesley, Reading, MA.
6. Hogg, R., and A. Craig. 1970. *Introduction to Mathematical Statistics.*
7. Jain, A. K., J. Mao, and K. M. Mohiuddin. 1996. Artificial neural networks: A tutorial. *IEEE Comput.* 29(3).
8. Jain, R. Duin, and J. Mao. 2000. Statistical pattern recognition, a review. *IEEE-TPAMI.* 22(1):4–37.
9. Fukunaga, K. 1990. *Introduction to Statistical Pattern Recognition,* 2nd ed. Academic Press, New York.
10. Kulkarni, A. 1994. *Artificial Neural Networks for Image Understanding.* Van Nostrand-Reinhold, New York.
11. Minsky, M., and S. Papert. 1989. *Perceptrons,* 2nd ed. MIT Press, Cambridge, MA.
12. Proakis, J. G. 1989. *Digital Communications.* McGraw-Hill, New York.
13. Quinlan, J. R. 1986. Induction of decision trees. *Machine Learning,* 1(1):81–106.
14. Schalkoff, R. 1992. *Pattern Recognition: Statistical, Structural, and Neural Approaches.* John Wiley & Sons, New York.
15. Schurmann, J. 1996. *Pattern Classification: A Unified View of Statistical and Neural Approaches.* John Wiley & Sons, New York.
16. Tanimoto, S. 1995. *The Elements of Artificial Intelligence with Common LISP,* 2nd ed. Computer Science Press, New York.

127

第5章 图像滤波与增强

本章讨论图像增强的方法。图像增强可以提高图像的视觉效果，也有利于进一步的自动处理。增强可以指减少图像中的噪声，也可以指强调或抑制图像中的某些细节。第1章已经介绍了图像滤波的两种方法。第一，在细菌图像中，从大的均匀区域中去除孤立的黑色或白色像素点。第二，利用反差算子增强图像中不同目标的边界，即提高目标和背景的对比度。

本章主要是图像处理（image processing）方面的内容，所有的方法都是对输入图像进行处理，并生成新的输出图像。其他经常用到的相关术语有滤波（filtering）、增强（enhancement）或调整（conditioning）。图像中包含着要抽取的信号或结构，也包含我们不感兴趣或不想要的干扰，这些干扰要想办法去掉。图像运算时，可以针对单个像素或者针对像素的局部邻域。我们已经知道如何把像素标记为目标点或者背景点、边界点或者非边界点。

图像处理的理论和方法足够写好几本书，这里仅详细介绍经典的图像处理方法。多数方法，都是根据输入图像中对应像素的邻域计算输出图像的像素值。但有的图像增强方法是全局性的，即根据输入图像的所有像素计算输出图像。两个最重要的概念是：（1）将图像邻域与模式或模板进行匹配（相关性（correlation））；（2）卷积（convolution），可以实现多种滤波运算的一种简单方法。 128

5.1 图像处理

在讨论方法之前，先看看存在哪些问题需要进行图像处理。以下是两大类问题。

5.1.1 改善图像质量

- 在非洲的狩猎旅行中，你拍摄到一张狮子追逐羚羊的照片。不巧的是，太阳位于被摄物体的后方，因而使得图片的光线显得过暗。增加低亮度像素点的亮度，保持高亮度点不变，这张照片就可以得到改善。
- 一张老照片有一条长的白色划痕，但其他部分完好。照片可以变成数字图像，并去除划痕。（参见图5-1）
- 扫描纸质文档并转化成文本文件。在进行字符识别之前，需要从背景中清除噪声像素点，字符中丢失的信息也要进行填充。

5.1.2 检测低层特征

- 生产直径3mm的电线，要用到视觉传感器测量电线直径的反馈信息。利用边缘算子确定电线两边的位置，边缘算子能够准确地识别电线和背景之间的边界 129
- 汽车自动驾驶系统，通过监测高速公路上的白线实现自动驾驶。在前视摄像机的视频帧中，通过找到对比度相反、方向相同的两条边线，就可以检测出两条白线。
- 把蓝图转化成CAD（计算机辅助设计）模型。其中需要把蓝图上的直线转化为图像中约一个像素宽的暗条纹。

本章主要讨论图像增强和图像恢复（见图5-2）的传统方法。开始之前先定义两个概念。

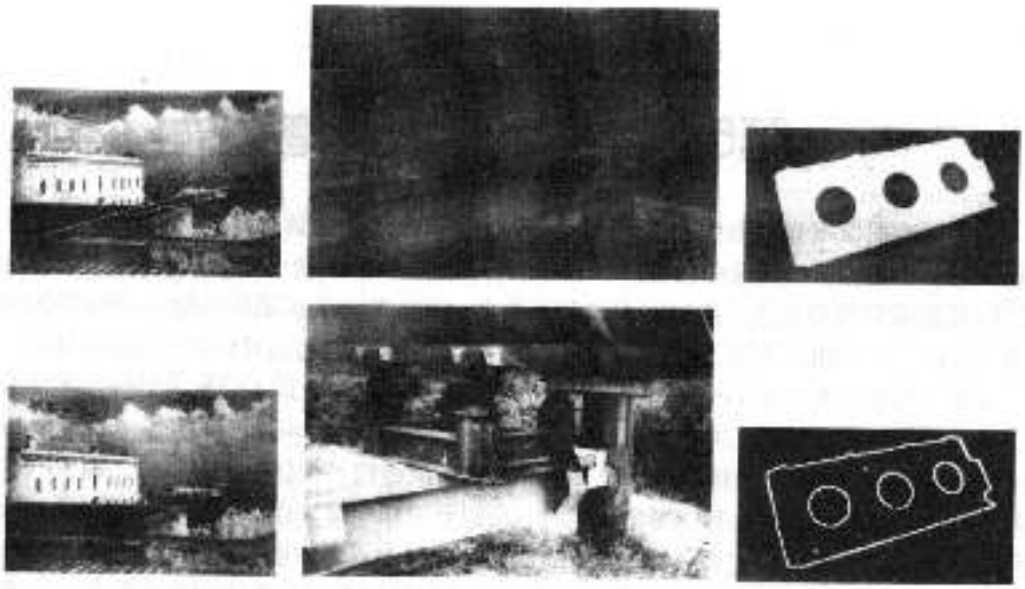


图 5-1

（左）San Juan原图中的划痕被去除
（中）Alaskan Pipeline的图片亮度作了重新调整后表现出更多的细节
（右）飞机零件的图像，进行了边缘增强，有利于自动识别和测量



图5-2（Shaoyun Chen和Anil Jain提供）

（左）原始的指纹图
（中）纹路检测及细化后的增强图像
（右）细微点特殊特征的识别，可与数据库中上百万个指纹图像进行匹配

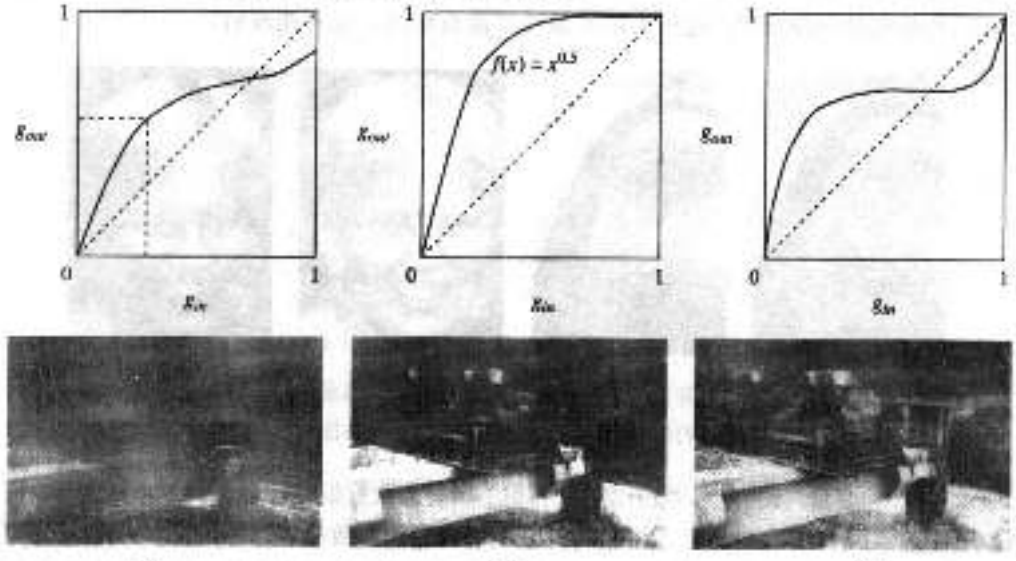
定义39 人或机器利用**图像增强**算子，提高图像中重要细节或目标的可检测性。这样的运算包括去噪、平滑、提高对比度以及边缘增强。

定义40 **图像恢复**试图将一幅受损图像恢复到理想状态。只有在理想图像形成和图像损坏的物理过程能够被理解和建模的情况下，图像恢复才有可能。恢复过程与损坏过程相反，可以将受损图像变换为理想图像。

5.2 灰度级映射

通过改变像素的亮度值来增强图像是一种常用的方法。大多数图像处理软件工具，都包含几种改变图像外观的方式，它们借助函数变换将输入的像素灰度值映射成一个新的输出值。

对这种方法进行扩展，由用户指定几块不同的图像区域，并对它们分别进行映射。对灰度值的重新映射通常称为扩展（stretching），因为一般都是将过暗的图像灰度值进行扩展，使其 130
分布在整个灰度值区间。图5-3说明一幅图像的亮度值被两个不同的映射函数扩展的结果。图5-3a表示原图以及映射函数的常用形式，图5-3b表示采用函数$f(x)=x^{0.5}$的亮度映射，它对所有亮度值进行非线性放大，低亮度值的放大程度大于高亮度值的放大程度。采用映射函数$f(x)=x^{\gamma}$时称为伽马（Gamma）校正，如果图像的物理畸变已知，要想将图像恢复到原来的形式，伽马校正或许是合适的理论模型，在图中情况下$\gamma=2.0$，是一个放大值。针对图中情况，取缩小值如$\gamma=0.3$是不实用的，因为场景中包含森林和管道本身的阴影。图5-3c显示的是更复杂的映射函数，通过交互方式进行确定。用户利用图像处理工具定义灰度级映射函数$g_{out}=f(g_{in})$，由用户控制鼠标在图像上取点，图像工具根据用户选择的点拟合出光滑的样条曲线。图5-3中的函数将一定范围内的亮度进行扩展或扩充，使输出表现出更多的细节变化。如果函数$f(x)$的斜率大于1，则在这些亮度范围内的图像变化就增大。



a) b) c)

图5-3 图像的亮度值被不同映射函数扩展的结果。注意不同的场景目标在不同图像中表现出的清晰度不同

（上行）亮度映射函数f

（下行）对原图进行变换后的输出图像

a）正常、偏暗和对比拉伸后管道图，使用的一般的亮度映射函数

b）$f(x)=x^{0.5}$的伽马校正，暗像素比亮像素得到更多的增强

c）利用交互式软件工具，由用户创建出上面的映射曲线，该曲线提高暗像素的像素值，降低亮像素的像素值

定义41 图像点算子（point operator），其输出像素仅由输入像素决定，$\mathbf{Out}[x,y]=f(\mathbf{In}[x,y])$，函数$f$可能依赖于全局性的参数。 131

定义42 对比度扩展（contrast stretching）算子是一种点算子，利用输入灰度的分段光滑函数$f(\mathbf{In}[x,y])$来增强图像的重要细节。

由于点算子将一个输入像素映射到一个输出像素，所以可按像素的任意顺序映射一幅图

像，或者并行映射各个像素。为了使人们更好地得到图形和新闻方面的服务，特殊亮度映射，包括图5-3中的非单调映射，在图像增强中非常有用。但在某些领域，如放射学，则必须要谨慎，不能改变有意义的亮度值，这些值是专家和精密传感器仔细校正好的。最后，单调灰度级扩展，对于有些机器视觉算法的性能可能提高不大（当灰度级$g_2>g_1$时$f(g_2)>f(g_1)$），但对于人类视觉，这种增强效果还是很明显的。

直方图均衡化

图像增强经常要用到直方图均衡化。该运算的两个要求是：（a）输出图像应当包含所有可能的灰度级；（b）输出图像在每个灰度级上有大致相等的像素个数。要求（a）有明确的意义，但要求（b）比较特殊，它的有效性必须凭经验判断。图5-4表示直方图均衡化结果，可以看见，灰度级的重新映射的确改变了一些区域的表观。例如拱桥的焊缝更容易看见。（用类似图5-3最右边的映射效果会更好，为什么？）脸部图像剪切自更大的一幅图像，剪切窗口中低亮度的像素不多，要求（b）使得大块均匀区域（如天空）重新映射成具有更多灰度级别的区域，表现出更强的纹理。这对于图像解释可能有帮助，也可能没有。



图5-4 直方图均衡化对灰度级进行映射，使输出图像的像素值分布在整个灰度范围，并且每个灰度值的像素个数大致相等。右边是直方图均衡化后的图像

要求（a）和要求（b）意味着输出图像利用了所有的灰度值，$z=z_1, z=z_2, \cdots, z=z_n$，每个灰
132 度级z_k大约被用了$q=(R\times C)/n$次，其中R和C分别是图像的行数和列数。为了定义扩展函数f，需要输入图像直方图$H_{in}[i]$，$H_{in}[i]$表示输入图像中具有灰度值z_i的像素个数。在输入图像直方图中增加i直至大约计算了q_1个像素，通过该方法找到第一个灰度级阈值t_1。所有满足灰度值$z_k<t_1-1$的输入图像像素在输出图像中将被映射成灰度值z_1。阈值t_1由下面的计算公式定义：

$$\sum_{i=1}^{t_1-1} H_{in}[i] \leqslant q_1 < \sum_{i=1}^{t_1} H_{in}[i].$$

t_1表示的是最小的灰度级，使原始直方图最多包括q个灰度值小于t_1的像素。第k个阈值t_k由下面的迭代公式定义：

$$\sum_{i=1}^{t_k-1} H_{in}[i] \leqslant (q_1+q_2+\cdots+q_k) < \sum_{i=1}^{t_k} H_{in}[i].$$

映射f的一种实现是查找表，从上述过程中很容易得到这样的映射表（lookup table）。在计算上面公式的过程中，只要不等式成立，阈值t_k就被放入（可能会重复地）数组$T[i]$。这样就有函数$z_{out}=f(z_{in})=T[z_{in}]$。

习题5.1

一幅200个像素的输入图像直方图如下：H_{in}=[0, 0, 20, 30, 5, 5, 40, 40, 30, 20, 10, 0, 0, 0, 0, 0]。(a) 利用直方图均衡化的公式（15个灰度级），则输出图像f(8)的值是多少？(b) 对f(11)重复问题 (a)。(c) 求输入图像的映射函数f的查找表形式T[i]。

习题5.2 直方图均衡化算法

用伪代码写出直方图均衡化的算法。保证定义所有用到的数据项和数据结构。

习题5.3 直方图均衡化程序

(a) 利用前面习题的伪代码，实现并测试直方图均衡化程序。(b) 针对不同的图像，对处理效果进行分析比较。

经常会出现这种情况，输出图像的灰度级范围大于输入图像的灰度级范围。这样对于任意函数f将灰度级重新映射到整个输出范围是不可能的。如果确实需要一个大致均匀的输出直方图，可用一个随机数发生器将输入值z_m映射到其邻域$T[z_m]$。上面的过程将$2q$个g级像素映射到输出灰度级g_i，而没有像素映射到灰度级g_i+1。我们可以模仿硬币落地的等概率事件，使g级的输入像素以相同的概率映射到g_i或g_i+1。 133

5.3 去除小图像区域

实际中常常需要去除图像中的小区域。一个小区域可能是噪声，或者是需要从图像描述中去掉的低层细节。改变单个像素的值，或者在抽取连通成分后去除小的连通成分，这些都是去除小区域的方法。

习题5.4

为了使输出的直方图更均匀，对随机函数f有什么要求？

5.3.1 去除盐椒噪声

绪论中简要讨论了从均匀区域去掉单个不规则像素的方法，在第3章中对这些方法进行了扩展。在亮区域内出现单个暗像素，或在暗区域内出现单个亮像素，这些都称为盐椒噪声。这种比喻是显而易见的。盐椒噪声是通过阈值进立二值图像的结果。盐点对应着在暗区域中的某些像素，这些像素通过了为检测亮像素而设定的阈值。椒点对应着在亮区域中的像素，但低于设定的阈值。表面材料变化、光照影响或者帧捕捉器中数/模转换的噪声，这些因素引起的分类错误都会产生盐椒效果。有些情况下，这些孤立像素点不是分类错误，而是与较大邻域形成对比的微小细节，如衬衫上的一粒纽扣，或者一块林间空地等，这些细节也许对所关心的问题来说无关紧要。

图5-5显示从细菌的二值图像中去掉盐椒噪声后的结果。用图中下面的模板对输入图像进行运算。如果输入图像中某邻域与左边模板匹配，则该邻域变换成由右边模板给出的邻域。该方法仅需要这两个模板。如果输入图像是经阈值化或其他分类过程得到的标记图像，则可采用更通用的模板，如图5-5的最下面一行所示，标记为L的像素孤立于其他标记为X的8-邻域像素，输出图像中则将该像素校正为X。L是图像中k个标记中的任意一个。该图说明了8-邻域和4-邻域都可用来进行这样的决策运算。在4-邻域情况下，不考虑4个角的像素。第3章讨论过，采用不同的邻域可导致不同的输出图像，如细薄图像情况。从图5-5可以看到，采用8-邻 134

域和采用4-邻域所产生的结果是有差异的。

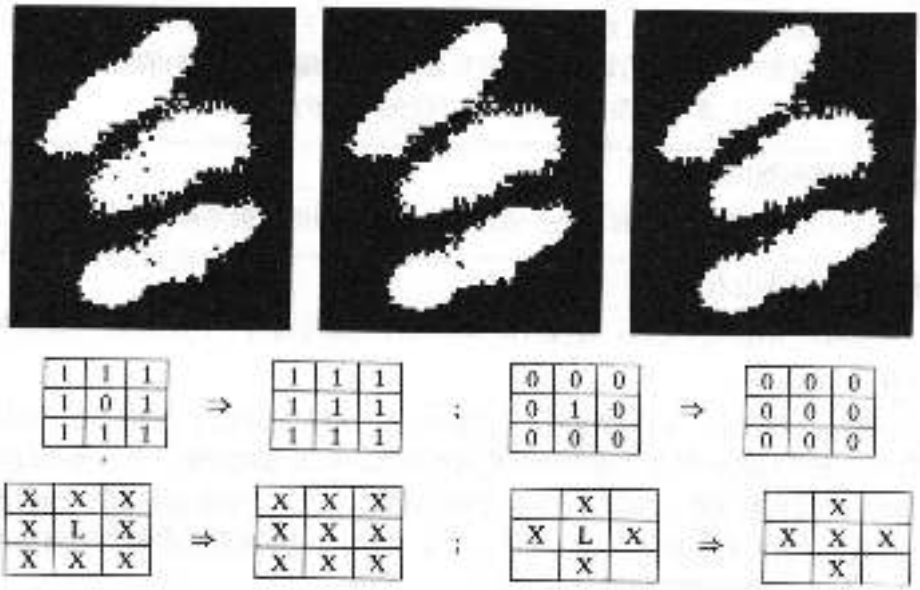


图5-5 从细菌的二值图像中去掉盐椒噪声后的结果。中间一行是二值像素邻域去噪模板，下面一行是针对一般标记图像去除孤立像素点的模板。（剪切自Frank Dazzo的细菌图像）

（左上）细菌的二值图像
（中上）采用8-邻域去除盐椒噪声的结果
（右上）采用4-邻域去除盐椒噪声的结果
左下是8-邻域决策模板
右下是4-邻域决策模板

5.3.2 去除小成分

第3章讨论了如何抽取二值图像的连通成分，并定义了大量的特征，这些特征根据构成成分的一组像素算出。图像描述是成分的集合，每个成分表示从背景抽取的区域，根据区域算出特征。通过运算，能够根据算出的特征从描述中去除任何成分，例如去掉像素数量很少的成分或者去掉非常细的成分。该处理能够去除细菌边界附近的一些噪声区域。如果不必或者不可能生成相应的输出图像，则可以把小区域从描述中删除。如果必须生成输出图像，就必须保留信息以便能恢复输入图像，并根据变化的区域对像素进行正确的再编码。图5-6表示去掉盐椒噪声及面积小于12个像素的小区域后的细菌图像。

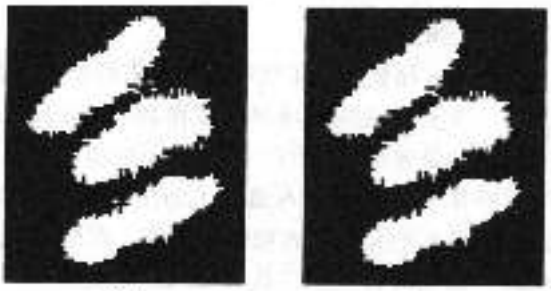


图 5-6

（左）用4-邻域模板去掉图5-5中盐椒噪声后的细菌图像
（右）去除小连通成分后的图像（原图由Frank Dazzo提供）

135

5.4 图像平滑

一幅图像常常既包含潜在的理想结构，也包含一些随机噪声或人为干扰，前者是要检测和描述的，而后者是希望去除的。例如一个简单模型，均匀目标的图像区域像素点具有值$g_i + N(0, \sigma)$，

其中g是理想成像条件下某个期望的灰度级，$N(0, \sigma)$是均值为0标准差为σ的高斯噪声。图5-7（左上）表示具有均匀区域的理想棋盘图。理想图像中加入高斯噪声得到中间的含噪图像。注意噪声值经过处理限制在区间[0, 255]。右上角的图是穿过图像中单行的像素值。

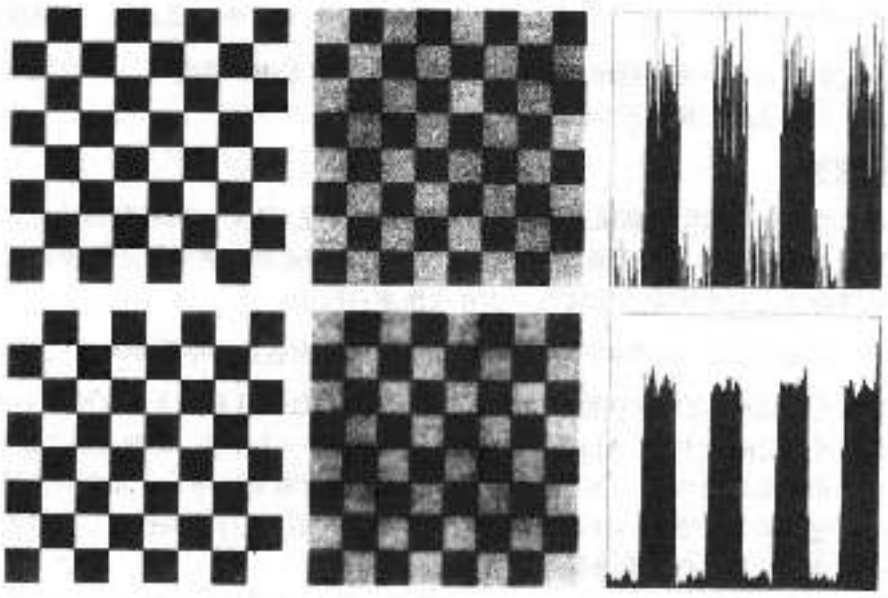


图 5-7

（左上）模拟生理想图像。黑色方块的像素值为0，白色方块的像素值为255

（中上）图像中加入了标准差为30的高斯噪声

（右上）从噪声图像的顶部开始第100行的像素值

（左下）根据图像直方图的谷值，对中上图像阈值化的结果，消除一些盐椒噪声。（中下）对每个像素点用其5×5的邻域平均化的结果

（右下）从噪声图像的顶部开始第100行的像素值

通过取邻域平均值的方法，可以减少区域内在正常亮度值上下浮动的噪声。

$$\text{输出图像}[r, c] = \text{输入图像}[r, c]\text{邻域的平均值} \tag{5-1}$$

$$Out[r, c] = \left(\sum_{i=-2}^{+2} \sum_{j=-2}^{+2} In[r+i, c+j] \right) \Big/ 25 \tag{5-2}$$

公式（5-2）定义了一个平滑滤波器，它对输入图像中的像素用5×5邻域内的25个像素值进行平均，得到一幅平滑的输出图像。图5-7（中下）表示对棋盘图像应用该方法的结果：图中右下角的图像行比右上角的输入图像行更光滑一些。该行的结果并不是只对该行进行平均，而是利用了图像中5行像素的值。同时注意到虽然平滑图像比原图干净些，但它不如原图清晰。

定义43 在像素的一个矩形邻域内进行等量加权，实现对图像的平滑处理，这种方法称为**盒形滤波**（box filter）。

与对所有输入像素进行等量加权不同，一种更好的方法是随着距中心像素$I[x_r, y_r]$的距离的增加而减小输入像素的权。高斯滤波（Gaussian filter）采用的就是这种方法，它是最常用的一种滤波器。在5.7节中将详细讨论它的特性。 136

定义44 当进行**高斯滤波**时，像素[x, y]根据下式进行加权：

$$g(x,y)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{d^2}{2\sigma^2}}$$

其中 $d=\sqrt{(x-x_c)^2+(y-y_c)^2}$ 是输出图像中邻域像素[x, y]到中心像素[x_c, y_c]的距离。

本章后面的内容，将更详细地讨论有关平滑的理论和方法，并在同一个框架下对边缘检测进行讨论。在这之前，我们先介绍常用的中值滤波。

5.5 中值滤波

对具有零均值噪声的均匀邻域进行平均化时，取均值是对I[x, y]的较好的估计。但当该邻域跨越两块区域的边界时，由于两块不同区域的样本参与运算，将导致边界模糊。流行的替换算法是中值滤波，它用像素点邻域的中值替代像素点的值。

137

定义45 设A[i]$_{i=0,\cdots,n-1}$是含n个实数的有序数组，则A中各数的**中值**是A[(n−1)/2]。

有时要区分考虑n为奇数或偶数的情况。当n为奇数，数组具有如上定义的唯一中值。当n为偶数，我们可定义两个中值，A[n/2]以及A[n/2−1]，或者一个中值，即取这两个值的平均值。虽然采用有序排列来定义中值，在实际中这n个值并不需要做全排序。对著名的快速排序算法进行修改，使它只对包含第$(n+1)/2$个元素在内的A的子数组进行递归排序。一旦排序支点元素位于原数组中间位置，整个集合的中值就可知道。

图5-8说明中值滤波既能平滑噪声区域，又能较好地保持区域间的边界结构。如果从白色方块内靠边缘的地方选择像素，该像素邻域的大部分值可能都是含噪的白色像素。如果真是这样，计算输出值时就用不到属于黑色块的邻域像素。同样，当计算黑色块边缘像素的输出值时，其邻域的大部分值可能是带噪声的黑色像素，这意味着计算输出值时就用不到属于白色区域的邻域样本。与求平均的平滑方法不同，中值滤波在平滑均匀区域的同时又保持了边缘结构。中值滤波也可去除盐椒噪声以及大多数其他的小型人为干扰。人为干扰使各种噪声值代替了理想的图像值。图5-9说明如何去掉结构化人为干扰，同时减少均匀区域的变化并保持区域间的边界。

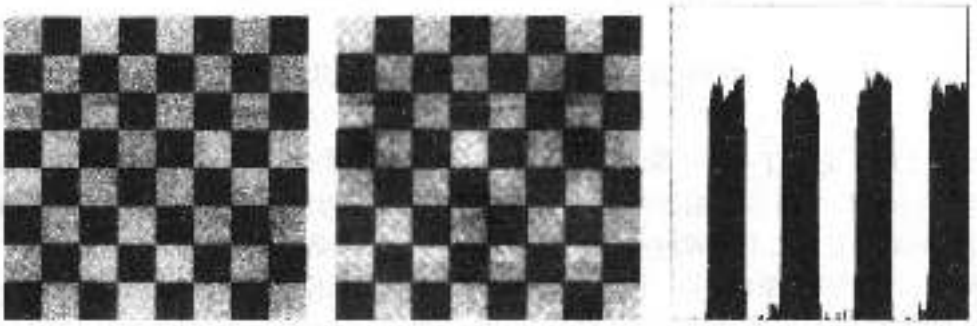


图 5-8

(左) 含噪棋盘图像

(中) 取中心像素5×5邻域的中值作为输出像素的值

(右) 从图像顶部开始的第100行的像素值，请与图5-7进行对比

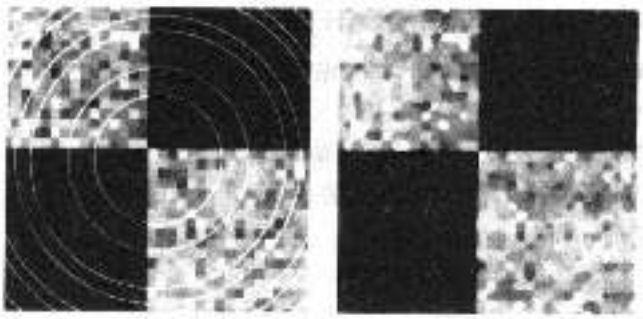


图 5-9

（左）输入图像，在四个像素均匀的区域加入了高斯噪声和人工光环干扰。

（右）应用7×7中值滤波的结果

计算中值比计算邻域的平均值需要用更多的时间，因为必须对邻域像素值进行排序，而且中值滤波不易通过专用硬件实现，而对于实时处理来说硬件实现是必要的，例如视频流水线的实时处理。但在许多图像分析任务中，中值滤波在图像增强方面作用巨大，这点时间耗费是值得的。 138

习题5.5 改进的快速排序算法1

（a）从众多数据结构和算法的教科书上找到传统快速排序算法的伪代码，修改算法使得一旦确定中值就返回。（b）相对于完整的排序算法，确定中值的算法计算量是多少？（c）用一种编程语言实现算法，并通过样例图像进行测试。

习题5.6 改进的快速排序算法2

利用上面的快速排序算法来检测图像函数中的跳变，如在棋盘图像中从黑色到白色方块的跳变，假设在数组A[n/2]处放置支点元素，找到了I[r, c]的邻域的中值。说明如何处理数组的其余部分，从而决定位置[r, c]的像素是否位于两块不同亮度区域的边界上。

从输入图像计算输出图像

前面举例说明了要进行哪些方面的图像增强，现在考虑如何对图像进行这些运算。下面的通用算法表示，用不同的滤波器对输入图像进行增强，并产生输出图像。

算法5.1表示简单的顺序计算过程，它以光栅扫描次序计算输出图像G的每个像素，并利用F[r, c]的邻域计算G[r, c]的像素值。显然，可以按任意顺序计算输出图像G的像素，而不必以行列为序。事实上，可以并行计算，这是因为输入图像不会因为任何邻域计算而改变。其次，过程*compute_using_neighbors*可通过盒形滤波或中值滤波的方法实现。对于盒形滤波，过程仅需要累加F[r, c]的$w \times h$个邻域像素的值，然后除以像素个数$w \times h$。为实现中值滤波，过程可以拷贝这$w \times h$个像素的值到一个局部数组A，然后进行部分排序得到中值。 139

可以使图像中只有h行在主存中同时存在。只对中间行r计算输出G[r, c]，然后在内存中输入新的一行，代替最旧的一行，计算下一个输出行G[r, c]。这个过程重复进行直至计算完所有可能的输出行。多年前，当计算机内存很小时，图像数据主要存储在磁盘上，很多算法一次只能处理图像的几行像素。今天，这种程序控制方式仍然存在价值，因为它可以用在图像处理板中，实现流水线处理结构。 140

算法5.1 根据输入图像像素F[r,c]的邻域计算输出图像像素G[r,c]

F[r,c]是行数为**MaxRow**列数为**MaxCol**的输入图像。
F不随算法而改变。
G[r,c]是行数为**MaxRow**列数为**MaxCol**的输出图像。
G的边界是那些邻域不全包含在**G**中的像素。
*w*和*h*是邻域的宽度和高度，单位为像素。

```
procedure enhance_image(F, G, w, h);
{
for r := 0 to MaxRow - 1
 for c := 0 to MaxCol - 1
  {
  if [r,c] is a border pixel then G[r,c] := F[r,c];
  else G[r,c] := compute_using_neighbors (F, r, c, w, h);
  };
}
procedure compute_using_neighbors (IN, r, c, w, h)
{
using all pixels within w/2 and h/2 of pixel IN[r,c],
compute a value to return to represent IN[r,c]
}
```

习题5.7

用一种编程语言实现算法5.1，对盒形滤波和中值滤波进行编程，并用一些图像如图5-9进行测试。

5.6 差分模板边缘检测

通过计算局部图像区域的亮度差异，可以检测出具有高对比度的图像点，例如不同目标之间或者场景各部分之间的边界，本节说明如何通过邻域模板检测出这些边缘。我们首先讨论一维信号，这不仅直观，而且也方便用公式表示，一维信号本身也是非常重要的内容。1D信号可以是2D图像的行或列，本节末尾讨论更通用的2D情况。

5.6.1 1D信号差分

图5-10显示如何利用模板计算信号的导数。设信号**S**是对函数*f*的采样序列，那么$f'(x_i) \approx (f(x_i) - f(x_{i-1}))/(x_i - x_{i-1})$，假设样本间距为$\Delta x = 1$，对**S**中的采样点，应用模板**M'** = [−1, 1]得到输出信号**S'**，通过这种方式来近似得到$f(x)$的导数，如图5-10所示。如图中所示，可以方便地认为**S'**的值为两样本点**S**值之差。如果**S'**[*i*]的绝对值较大，说明信号变化迅速，或者对比度较大。信号**S'**本身可以通过模板**M'**进行二次差分得到输出**S''**，**S''**对应着原始函数*f*的二阶导数。根据图5-10以及下面的公式，可以得出重要的结果：通过对原始样本序列**S**应用模板**M''**，可以近似得出函数的二阶导数。

141

$$\mathbf{S}'[i] = -S[i-1] + S[i] \tag{5-3}$$

$$\text{模板 } \mathbf{M}' = [-1, +1] \tag{5-4}$$

$$\mathbf{S}''[i] = -S'[i] + S'[i+1] \tag{5-5}$$

$$= -(S[i] - S[i-1]) + (S[i+1] - S[i]) \tag{5-6}$$

$$= S[i-1] - 2S[i] + S[i+1] \tag{5-7}$$

$$\text{模板 } \mathbf{M}'' = [1, -2, 1] \tag{5-8}$$

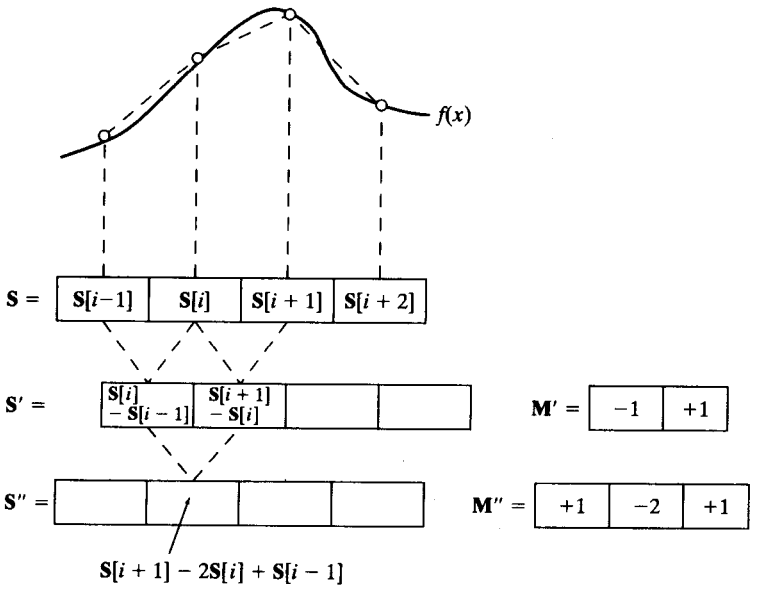


图 5-10

（左）一阶差分（S′）和二阶差分（S″）近似表示信号S的一阶导数和二阶导数

（右）模板M′和M″表示求导运算

如果只检测对比度大的点，通常采用对信号位$\mathbf{S}[i]$进行模板计算后的绝对值。如果这样，那么一阶导数模板可以是$\mathbf{M}' = [-1, 1]$或$[+1, -1]$，二阶导数模板可以是$\mathbf{M}'' = [+1, -2, +1]$或$[-1, +2, -1]$。很快将看到，对2D图像也存在着类似的情形。当只考虑幅值时，认为这些模板是一样的；当变化符号也重要时，就认为这些模板是不同的。 142

另一个常用的一阶导数模板如图5-11所示。这个模板有3个坐标，以信号点$\mathbf{S}[i]$为中心，通过模板计算信号穿过邻接值的差分。由于$\Delta x = 2$，如果不把结果除以2，将得出高于实际导数值的估计结果。另外，这个模板在理想跳变边缘处产生宽两个采样点的响应，如图5-11a~b所示。图5-12表示对采样信号应用二阶导数模板的响应。如图5-12所示，信号对比度可通过零交叉检测出来，零交叉方法确定两相邻信号值间变化的位置，并对信号变化进行放大。一阶和二阶导数信号共同揭示了许多局部信号的结构信息。图5-13显示了如何按同样的差分思想对信号进行平滑处理，下面对平滑模板和差分模板的一般特点做个对比。

导数模板的一些特性：

- 为了在对比度大的信号区域得到比较强的响应，导数模板的坐标符号相反。
- 导数模板的坐标和取零，使得恒值区域的响应为0。 143
- 一阶导数模板在对比度大的点产生较高的绝对值。

• 二阶导数模板在对比度大的点产生零交叉。

做为对比，平滑模板具有下列特性：

• 平滑模板的坐标都为正，它们的和为1，这样使恒值区域的输出与输入相同。

• 平滑和去噪的程度与模板的大小成正比。

• 跳变边缘的模糊程度与模板的大小成正比。

模板 **M = [−1, 0, 1]**

S_1			12	12	12	12	12	24	24	24	24	24
S_1	⊗	M	0	0	0	0	12	12	0	0	0	0

a）S_1是上跳变边缘

S_2			24	24	24	24	24	12	12	12	12	12
S_2	⊗	M	0	0	0	0	−12	−12	0	0	0	0

b）S_2是下跳变边缘

S_3			12	12	12	12	15	18	21	24	24	24
S_3	⊗	M	0	0	0	3	6	6	6	3	0	0

c）S_3是向上的斜坡

S_4			12	12	12	12	24	12	12	12	12	12
S_4	⊗	M	0	0	0	12	0	−12	0	0	0	0

d）S_4是亮脉冲或直线

图5-11 四种特殊信号的交叉相关结果，利用一阶导数边缘检测模板[−1, 0, 1]。注意，由于M的坐标之和是零，恒值区域上的输出一定是零

模板 **M = [−1, 2, −1]**

S_1			12	12	12	12	12	24	24	24	24	24
S_1	⊗	M	0	0	0	0	−12	12	0	0	0	0

a）S_1是上跳变边缘

S_2			24	24	24	24	24	12	12	12	12	12
S_2	⊗	M	0	0	0	0	12	−12	0	0	0	0

b）S_2是下跳变边缘

S_3			12	12	12	12	15	18	21	24	24	24
S_3	⊗	M	0	0	0	−3	0	0	0	3	0	0

c）S_3是向上的斜坡

S_4			12	12	12	12	24	12	12	12	12	12
S_4	⊗	M	0	0	0	−12	24	−12	0	0	0	0

d）S_4是亮脉冲或直线

图5-12 四种特殊信号的交叉相关结果，利用二阶导数边缘检测模板M[−1, 2, −1]。由于M的坐标之和是零，在恒值区域上的输出一定是零。注意输出中出现零交叉的地方，其中对应位置的输入信号发生变化的方式不同

盒形平滑模板 **M = [1/3, 1/3, 1/3]**

S_1			12	12	12	12	12	24	24	24	24	24
S_1	⊗	M	12	12	12	12	16	20	24	24	24	24

a) S_1是上跳变边缘

S_4			12	12	12	12	24	12	12	12	12	12
S_4	⊗	M	12	12	12	16	16	16	12	12	12	12

d) S_4是亮脉冲或直线

高斯平滑模板**M = [1/4, 1/2, 1/4]**

S_1			12	12	12	12	12	24	24	24	24	24
S_1	⊗	M	12	12	12	12	15	21	24	24	24	24

a) S_1是上跳变边缘

S_4			12	12	12	12	24	12	12	12	12	12
S_4	⊗	M	12	12	12	15	18	15	12	12	12	12

d) S_4是亮脉冲或直线

图 5-13

(上两行) 用盒形模板[1/3, 1/3, 1/3]平滑跳变和脉冲干扰

(下两行) 用高斯模板[1/4, 1/2, 1/4]平滑跳变和脉冲干扰

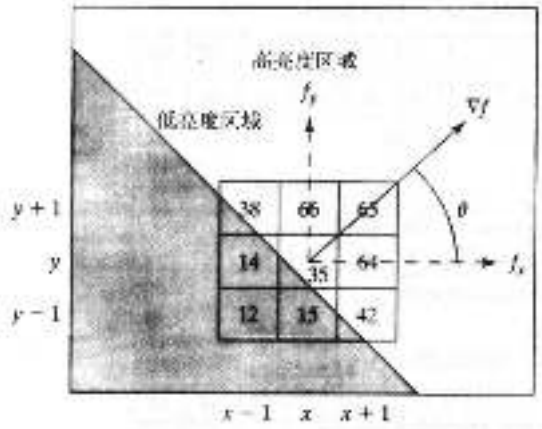
5.6.2 2D图像差分算子

2D图像函数$f(x, y)$的反差可能在任意方向出现。根据积分学，我们知道最大的变化沿着函数的梯度方向发生，图像平面的梯度方向为$\left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$。图5-14表明在数字图像中用离散近似的方法，可以非常直观地表示出这种变化。通过计算$(I[x+1, y] - I[x-1, y])/2$来估计位置 144
I[x, y]处沿x方向的反差，即用像素**[x, y]**左右邻域的亮度变化除以$\Delta x = 2$个像素单位。对图5-14所示的邻域，x方向的反差估计为 $(64 - 14)/2 = 25$。由于像素值含有噪声，并且边缘可能以任意角度通过像素阵列，因此应该求**[x, y]**邻域的三个不同反差估计值的平均值。

$$\begin{aligned}\partial f/\partial x \equiv f_x \approx \tfrac{1}{3}[&(I[x+1, y] - I[x-1, y])/2 \\ &+ (I[x+1, y-1] - I[x-1, y-1])/2 \\ &+ (I[x+1, y+1] - I[x-1, y+1])/2)]\end{aligned} \tag{5-9}$$

也就是对第y行及其上下两行在x方向的反差进行等量加权，并据此来估计x方向的反差。同样，y方向的反差估计如下:

$$\begin{aligned}\partial f/\partial y \equiv f_y \approx \tfrac{1}{3}[&(I[x, y+1] - I[x, y-1])/2 \\ &+ (I[x-1, y+1] - I[x-1, y-1])/2 \\ &+ (I[x+1, y+1] - I[x+1, y-1])/2)]\end{aligned} \tag{5-10}$$



$f_y = ((38 - 12)/2 + (66 - 15)/2 + (65 - 42)/2)/3 = (13 + 25 + 11)/3 = 16$

$f_x = ((65 - 38)/2 + (64 - 14)/2 + (42 - 12)/2)/3 = (13 + 25 + 15)/3 = 18$

$\theta = \tan^{-1}(16/18) = 0.727 \text{ rad} = 42°$

$|\nabla f| = (16^2 + 18^2)^{1/2} = 24$

图5-14 对于图像阵列的离散样本点，通过估计图像函数$f(x, y)$的梯度幅值和方向，来估计$[x, y]$反差的幅值和方向

为了节省计算时间，常常省略除以6的步骤，这样就得到成比例的估计结果。这两个反差算子模板在图5-15上部用M_x和M_y表示。图像函数的梯度，通过对像素$[x, y]$的8-邻域$N_8[x, y]$进行模板运算估计出来，如公式（5-11）至（5-14）所示。这些模板定义了Prewitt算子，这是由Judith Prewitt博士最先提出的，他利用这些算子来检测生物医学图像中的边缘。

145

$$\frac{\partial f}{\partial x} \approx (1/6)(M_x \circ N_8[x, y]) \tag{5-11}$$

$$\frac{\partial f}{\partial y} \approx (1/6)(M_y \circ N_8[x, y]) \tag{5-12}$$

$$|\nabla f| \approx \sqrt{\frac{\partial f}{\partial x}^2 + \frac{\partial f}{\partial y}^2} \tag{5-13}$$

$$\theta \approx \tan^{-1}\left(\frac{\partial f}{\partial y} \Big/ \frac{\partial f}{\partial x}\right) \tag{5-14}$$

在下一节对运算$M \cdot N$进行正式定义。在运算方法上，模板M与图像邻域N重叠，这样每个亮度值N_{ij}乘以权值M_{ij}，最终对结果进行相加。图5-15的中间一行表示两个类似的Sobel模板，它们的推导和含义都与Prewitt模板相同，只是中间点运算的权值应该是边缘点运算权值的两倍。

Roberts模板大小仅为2×2。这说明Roberts模板的效率更高，且更加局部化。这些模板通常称为Roberts交叉算子，它们实际上是计算4-邻域中心的梯度估计值，而不是中心像素。另外，算子的实际坐标系统与标准的行方向偏离45°。Robert交叉算子的应用如图5-16所示。原始输入图像是左上角的a图，由两个略微不

Prewitt: $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$; $M_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$

Sobel: $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$; $M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$

Roberts: $M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$; $M_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

图5-15 用于估计图像函数$f(x, y)$梯度的3×3模板。（上行）Prewitt模板；（中行）Sobel模板；（下行）Roberts模板

lel的Roberts算子得到的输出如图b和c所示，图d和e分别表示仅利用图像列方向和行方向的亮度差得到的结果，f表示行方向和列方向进行“或”运算的结果。定性地说，这是几种小邻域算子的运算结果，其中检测出很多边缘像素，但也有很多未检测出来，在带纹理的草地区域也有输出响应，但车库的上部信息丢失了，因为其亮度与天空的亮度一致。应该将Roberts算子的结果，与图5-16d～f所示的简单1D行和列模板相结合的结果进行比较。在计算梯度幅值时，一般要避免开方运算，代替的方法是求 $\max\left(\left|\frac{\partial f}{\partial x}\right|,\left|\frac{\partial f}{\partial y}\right|\right)$、$\left|\frac{\partial f}{\partial x}\right|+\left|\frac{\partial f}{\partial y}\right|$ 或者 $\left(\frac{\partial f^2}{\partial x}+\frac{\partial f^2}{\partial y}\right)/2$。比较图5-16b和c、f，说明避免开方运算是可行的，如果想知道实际梯度或梯度方向，就必须慎用这些估计方法。图5-17b表示利用Sobel 3×3算子计算均方梯度幅值的结果，图5-17c表示梯度方向的编码。原图中的小方块是8×8像素，Sobel算子检测出很多图像边缘，但不是全部。 146 147

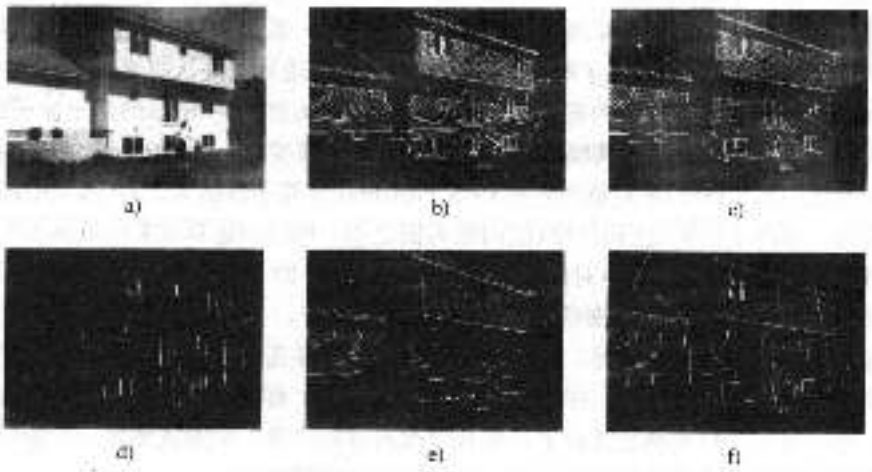


图5-16 Robert交叉算子的应用（图像由Ida Stockman提供）

a）原图
b）两个Roberts模板响应的绝对值总和的前5%
c）两个Roberts模板响应的均方值的前5%
d）y方向边缘模板[-1, +1]响应绝对值的前2%
e）x方向边缘模板[-1, +1]响应的绝对值的前3%
f）图像d和e的“或”运算结果，b、c和f相差甚微



图5-17 Sobel算子的应用

a）矢方块和圆环噪声的图像
b）3×3 Sobel算子的均方响应
c）用3×3 Sobel算子计算的梯度方向编码

习题5.8

如果需要真正的梯度幅值，Sobel模板为什么比Prewitt模板运算速度更快？

习题5.9 Prewitt模板的最优性*

证明Prewitt模板所提供的权值，实现了对亮度表面3×3邻域的最佳平面拟合。假设所有9个样本具有相同的权值。设3×3图像邻域的9个亮度值$I[r+i, c+j]$; $i, j = -1, 0, 1$由最小二乘平面模型$I[r, c] = z = pr + qc + z_0$拟合。（9个样本关于$r$和$c$等间距。）说明用Prewitt模板计算$p$和$q$的估计值，把$p$和$q$作为亮度函数的最小二乘平面拟合的偏导数。

图5-18b和c表示a中的室内场景图像中两行的亮度曲线。如图和曲线所示，b中所示的是下面一行的亮度，它表明该行穿过了四块黑暗区域，即（1）左边椅子上的大衣（列20至80），（2）位于中间的Prewitt博士的椅子和衣服（列170至240），（3）最右边椅子的阴影（列360至370）以及（4）电线（列430）。注意除了椅子和它的影子之间的边界，是从亮度220每隔约10个像素
148 缓慢下降到20之外，其他暗像素和亮像素之间的转换非常突然。c中所示的是上面一行的亮度，它表现出该行穿过画框、垫纸和两幅画时所经过亮度的明显变化。左边的画比右边的画表现出更多的亮度变化。图5-18d–f是对原图应用3×3 Prewitt梯度算子的结果。对于a–c中所示的相同的两个图像行，画出了列梯度f_c和行梯度f_r的绝对值之和。Prewitt运算结果的最高值与穿过的主要边界对应得非常好。但是，d中对应Prewitt博士所坐椅子的地方，即介于170和210之间的几个中等尖峰脉冲却难以解释。如图f所示，上面一行的反差，可以进行类似解释：主要目标边界与画框和垫纸的边界对应得很好，墙上最左边的画有许多亮度变化。一般说来，梯度算子能够很好地检测出孤立目标的边界，尽管存在一些一般问题。有时因为目标弯曲或渐变阴影导致边界丢失。另一方面，反差明显时经常产生几个像素宽的边界，后面就需要一个细化边界的步骤。梯度算子对纹理区域也产生响应，这一点将在第7章详细研究。

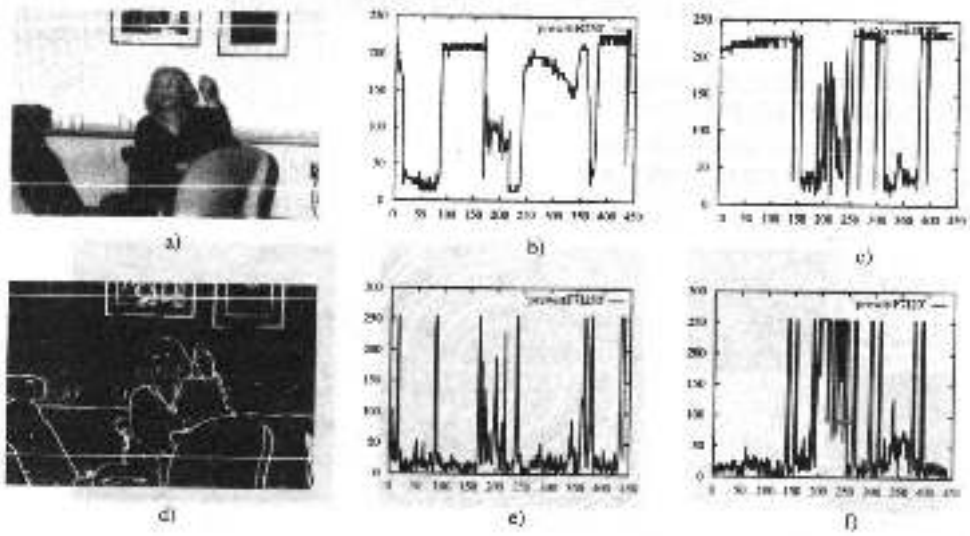


图 5-18

a) Judith Prewitt的图像，其中选择了两行
b) 沿着下面一行的亮度图
c) 沿着上面一行的亮度图
d) 利用Prewitt 3×3算子得到的d)的梯度图像
e) 梯度图像中下面一行的亮度图
f) e)梯度图像中上面一行的亮度图

5.7 高斯滤波与LOG边缘检测

高斯函数在许多数学领域都有重要的应用，包括图像滤波在内。本节，我们重点讲述它在图像平滑及平滑后的边缘检测方面的应用。

定义46 标准差为σ的一元**高斯函数**定义如下，其中c是比例因子:

$$g(x) = ce^{-\frac{x^2}{2\sigma^2}} \tag{5-15}$$

二元高斯函数定义为:

$$g(x, y) = ce^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{5-16}$$

这些公式与第4章定义的正态分布具有相同的结构，其中增加常量c是为了保证曲线下的面积为1。为了建立滤波模板，c一般取一个较大的数使所有的模板元素为整数。高斯函数以原点为中心，不需要正态分布中的定位参数μ。当信号或图像中包含该参数时，图像处理算法将通过平移去掉该参数。图5-19画出一元高斯函数，以及它的一阶和二阶导数，这些导数在滤波运算中也非常重要。计算导数的公式参见公式（5-17）~公式（5-22）。函数$g(x)$下面的面积为1，意味着它适合作为一个平滑滤波器，它对恒值区域无影响。$g(x)$是正的偶函数，而$g'(x)$等于$g(x)$乘以奇函数（$-x$）再除以σ^2。$g''(x)$揭示了更多的结构信息。公式（5-21）说明$g''(x)$是两个偶函数之差，中间下凸部分为负，该部分$x \approx 0$。由公式（5-22）可清楚地看到，二阶导数的零交叉发生在$x = \pm\sigma$处，这与图5-19中的情形是一致的。 149

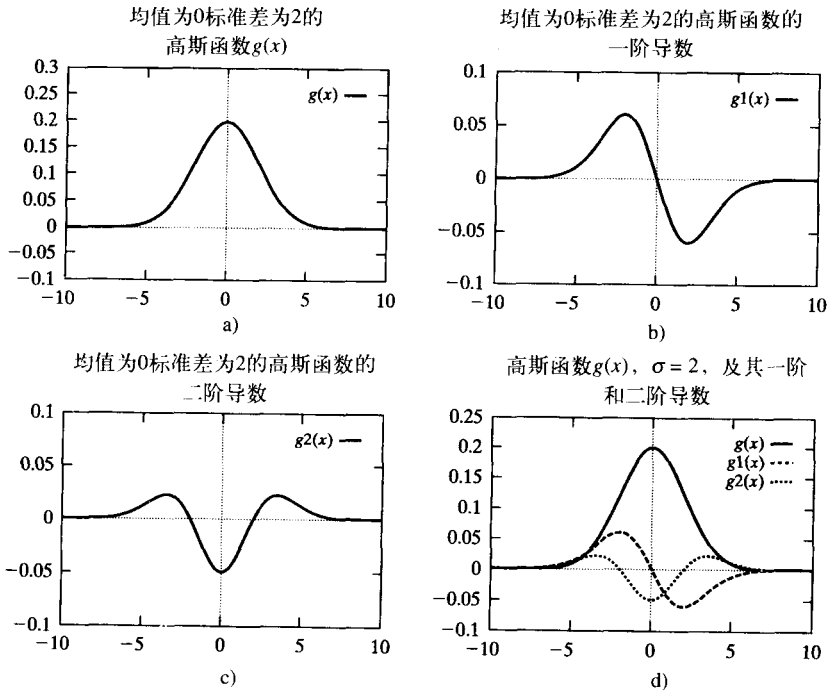


图5-19 一元高斯函数及其一阶、二阶导数

a）标准差σ=2的高斯函数$g(x)$

b）一阶导数$g'(x)$

c）二阶导数$g''(x)$，就像倒置的宽边帽的截面边缘

d）把三个图重叠到一起说明$g(x)$的拐点与$g'(x)$的极点和$g''(x)$的零交叉对应

$$g(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{x^2}{2\sigma^2}} \tag{5-17}$$

$$g'(x) = \frac{-1}{\sqrt{2\pi}\,\sigma^3} x e^{-\frac{x^2}{2\sigma^2}} \tag{5-18}$$

$$= \frac{-x}{\sigma^2} g(x) \tag{5-19}$$

$$g''(x) = \left(\frac{x^2}{\sqrt{2\pi}\,\sigma^5} - \frac{1}{\sqrt{2\pi}\,\sigma^3}\right) e^{-\frac{x^2}{2\sigma^2}} \tag{5-20}$$

$$= \frac{x^2}{\sigma^4} g(x) - \frac{1}{\sigma^2} g(x) \tag{5-21}$$

$$= \left(\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right) g(x) \tag{5-22}$$

高斯滤波的某些有用特性

1. 随着逐渐远离原点，权值逐渐减小到零。这表明离中心较近的图像值比远处的图像值更重要；标准差σ决定邻域的范围。总权值的95%包含在2σ的中间范围内。
2. 关于横坐标的对称性；把函数翻转进行卷积运算，产生同样的核。
3. 其傅里叶变换在频率域内表现为另一种高斯形式，这意味着与空间域高斯模板做卷积运算时，随着空间频率的提高，图像的高频成分逐渐减小。
4. 一维高斯函数的二阶导数$g''(x)$具有光滑的中间突出部分，该部分函数值为负，还有两个光滑的侧边突出部分，该部分值为正。零交叉位于$-\sigma$和$+\sigma$处，与$g(x)$的拐点和$g'(x)$的极值点对应。
5. 基于高斯-拉普拉斯算子的二阶导数滤波器称为LOG 滤波器。LOG 滤波器可用两个高斯函数之差来近似：$g''(x) \approx c_1 e^{-\frac{x^2}{2\sigma_1^2}} - c_2 e^{-\frac{x^2}{2\sigma_2^2}}$，该式通常称为DOG滤波器。在中间突出部分为正的情况下，必须有$\sigma_1 < \sigma_2$。要得到零交叉的正确位置，σ_2与σ_1密切相关，并且总负权值与总正权值达到平衡。
6. LOG滤波器特别适合检测两种亮度变化，即与中间突出部分重合的小斑点，以及与中间突出部分非常接近的大跳变边缘。

150 理解了一元高斯函数的特性，就可以直接建立相应的2D函数$g(x, y)$及其导数，只需用$r = \sqrt{x^2 + y^2}$替换1D中的x即可。1D形式绕垂直轴旋转可得到各向同性的2D函数形式，各向同性函数在任意过原点的切面上具有相同的1D高斯截面。其二阶导数形式好像一个宽边帽或称为墨西哥草帽。从数学推导上，帽子的空腔口沿$z = g(x, y)$轴向上，但在显示和滤波应用中空腔口一般朝下，即中间突出的部分为正，帽边为负。

两个不同的高斯平滑模板如图5-20所示。后面部分介绍边缘检测模板。

5.7.1 LOG边缘检测

151 LOG滤波器的两个不同模板参见图5-21和5-22。第一个是3 × 3的模板，是模板的最小实现形式，能够检测像素大小的图像细节。11 × 11的模板，对121个输入像素进行集成运算后得到输出，因此它适合较大的图像特征，而不适合较小的图像特征。如果利用硬件进行计算，集成121个像素要比集成9个像素多耗费许多时间。

$\mathbf{G_{3\times3}}=$

1	2	1
2	4	2
1	2	1

；

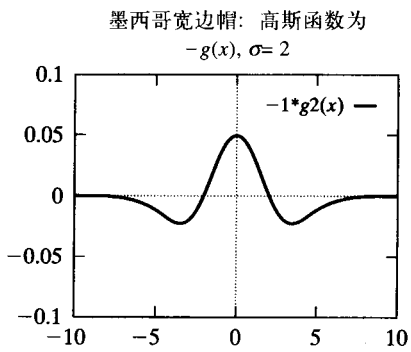
$\mathbf{G_{7\times7}}=$

1	3	7	9	7	3	1
3	12	26	33	26	12	3
7	26	55	70	55	26	7
9	33	70	90	70	33	9
7	26	55	70	55	26	7
3	12	26	33	26	12	3
1	3	7	9	7	3	1

图 5-20

（左）3 × 3近似高斯模板，由矩阵乘法[1, 2, 1]'⊗[1, 2, 1]得到

（右）$\sigma^2 = 2$的7 × 7近似高斯模板，对整数x和y利用公式（5-16）生成函数值，设$c = 90$使最小的模板元素为1



0	−1	0
−1	4	−1
0	−1	0

5	5	5	5	5	5
5	5	5	5	5	5
5	5	10	10	10	10
5	5	10	10	10	10
5	5	5	10	10	10
5	5	5	5	10	10

–	–	–	–	–	–
–	0	−5	−5	−5	–
–	−5	10	5	5	–
–	−5	10	0	0	–
–	0	−10	10	0	–
–	–	–	–	–	–

图 5-21

（上行）LOG滤波器的截面轮廓，以及3 × 3近似模板

（下行）输入图像及模板运算后的结果

0	0	0	−1	−1	−2	−1	−1	0	0	0
0	0	−2	−4	−8	−9	−8	−4	−2	0	0
0	−2	−7	−15	−22	−23	−22	15	−7	−2	0
−1	−4	−15	−24	−14	−1	−14	−24	−15	−4	−1
−1	−8	−22	−14	52	103	52	−14	−22	−8	−1
−2	−9	−23	−1	103	178	103	−1	−23	−9	−2
−1	−8	−22	−14	52	103	52	−14	−22	−8	−1
−1	−4	−15	−24	−14	−1	−14	−24	−15	−4	−1
0	−2	−7	−15	−22	−23	−22	15	−7	−2	0
0	0	−2	−4	−8	−9	−8	−4	−2	0	0
0	0	0	−1	−1	−2	−1	−1	0	0	0

图5-22 11 × 11的LOG近似模板，$\sigma^2 = 2$（取自Haralick and Shapiro, Volume I, page 349）

习题5.10 LOG滤波器的特性

设3×3图像邻域的9个亮度值，可用最小二乘平面模型$f(r,c)=z=pr+qc+z_0$很好地拟合。（9个样本关于r和c等间距。）证明简单的LOG模板

0	−1	0
−1	4	−1
0	−1	0

对该邻域产生零响应。即

152 LOG滤波器对恒值区域和斜坡变化都产生零响应。

5.7.2 人类视觉的边缘检测

现在讨论人工神经网络（ANN）结构，它能够以并行的方式实现LOG滤波运算。人工神经网络的行为与人类视觉系统的一些已知行为类似。另外猫和猴子的视觉系统产生的电信号也与神经网络的行为一致。图5-23表示对1D信

153 号的处理情况。视网膜细胞阵列感测到不同点的跳变边缘。第1层的细胞对第2层的细胞产生激励信号。每个第1层的细胞i和第2层的细胞j之间的物理连接具有一个连接权值w_{ij}，在细胞j中进行累算之前，这个权值与对应的激励相乘。细胞j的输出是$y_j=\sum_{i=1}^{N}w_{ij}x_i$，其中$x_i$是第$i$个第1层细胞的输出，$N$是第1层细胞的总个数。（实际上，只需要计算与第2层细胞j有直接连接的细胞i）。利用连接权值，有可能使得同样的细胞i对细胞j输入为正，对细胞$k\neq j$输入为负，这种情况是常见的。图5-23说明，对第二层的每个细胞j，其输出为$-a+2b-c$。这对应模板$[-1,2,-1]$，权值2用于中间的输入，而对于要抑制的输入a和b都用−1做权值。

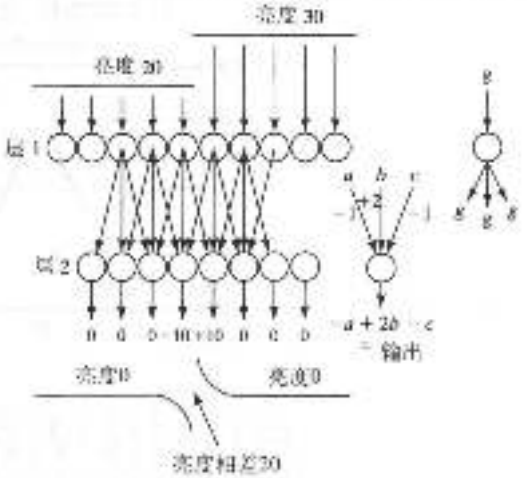


图5-23 利用ANN结构产生马赫带效应。视网膜细胞（层1）感知亮度，然后激励更高层（层2）的累算细胞

这种结构可以定义任意模板，对于滤波或特征检测中的交叉相关运算，允许以并行方式实现。心理学家马赫（Mach）注意到，人类感知两个区域之间的边缘时，就好像把边缘拉出来以夸大亮度的差异，如图5-23所示。注意该结构和模板在两个细胞之间的边缘处产生零交叉，其中一个产生正输出，另一个产生负输出。马赫带效应能改变连接面的感知形状，在通过被遮挡面显示多面体目标的计算机图形系统中，这种现象是很明显的。图5-24表示7个恒值区域，灰度级以步长32为间隔从31增加到255。你能感到它像3D网格，比如希腊神庙的陶立克柱子吗？

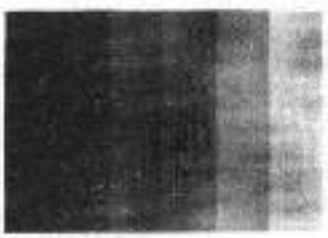


图5-24 由灰度值产生的七个恒值区域，灰度$31+32k$，$k=1,7$。由于马赫带效应，人们感到它像窗帘褶边或者网格

图5-25将图5-23扩展到2D图像。与累算细胞j连接的视网膜细胞集合组成细胞的感受野（receptive field）。利用二阶导数进行边缘检测，每个感受野有一个中心细胞集合，它们相对

细胞有正的权值w_0，还有一个负权值的周围细胞集合。视网膜细胞b和c在集算细胞A的感受野的中心，视网膜细胞a和d分布在周围，提供抑制性的输入。视网膜细胞d在集算细胞B的感受野的中心，细胞c分布在周围。中心权值与周边权值之和应该为0，这样集算细胞在恒值区域上就具有中性输出。因为中心和周边区域都是圆形的，所以当直线形区域边界以任意角度接近中心区域时，其输出都不是中性的。因此每个集算细胞是一个各向同性的边缘检测细胞。另外，如果与背景颜色不同的小区域在感受野的中心，集算细胞也会产生响应，因此该细胞也是一个点检测算子。图5-21表示最小的LOG模板与包含两块区域的图像求卷积的结果。图右边的结果显示，如何借助零交叉确定区域间的边界。与$\sigma^2 = 2$的LOG对应的11 × 11模板如图5-22所示。小模板能够检测到小区域间的边界，并对高曲率的边界敏感，但也会对噪声纹理产生响应。大模板具有明显的平滑效果，只对较光滑的大区域间的边界发生响应。

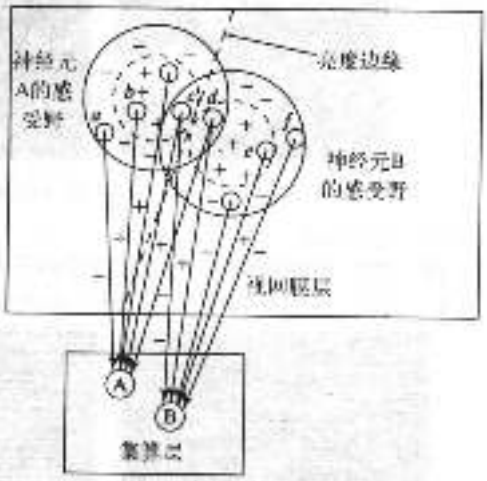


图5-25 LOG滤波器的3D ANN结构

154

习题5.11

给出上面论断的详细论据，即图5-25所示的集算细胞（a）对在感受野中心成像的反差点产生响应；（b）对两个贯穿过感受野中心的大区域之间的边界产生响应。

5.7.3 马尔-海尔德斯理论

大卫·马尔（David Marr）和埃伦·海尔德斯（Ellen Hildreth）提出，用LOG滤波器来解释人类视觉的低层行为。马尔提出人类低层视觉处理的目标是构造初始简图，初始简图指包含线、边缘和斑点的2D描述。（对双眼得到的初始简图进一步处理，以得到场景的3D解释。）为得到初始简图，Marr和Hildreth提出一种基于LOG滤波器的组织，其中LOG滤波器的参数σ取4个或5个不同的值。上述数学特性，成功解释了对人类知觉和对动物所做的实验结果。σ较大的LOG滤波器检测较宽边缘，σ较小的滤波器则集中检测小细节。在更高层次上协调不同尺度的输出结果，也许可以用大尺度检测指导小尺度的检测。后续工作出现了很多实用的尺度空间（scale space）方法，即对不同尺度检测算子的输出结果进行集成运算。

155

图5-26显示在两个不同层次上的高斯平滑结果。中间的图像很好地表达了主要目标及边缘，右边的图像则表现出更多的细节及噪声，注意轮船和沙子/水之间的边界在中间图像中未能体现出来，但在右边图像中有所体现。马尔的初始简图也包含对虚拟线段的描述，沿图像的曲线组成相类似的检测特征，这些特征构成虚拟线段。这些简图可能是虚线勾出的图像、一排灌木图像等。图5-27是一幅包含虚拟线的合成图像，以及两个不同LOG滤波器得到的输出结果。这两个LOG滤波器对线条端点都产生响应，一个对线条的边缘也有响应，另一个则没有响应。在图5-28的实际图像中可看到同样的道理，该图进行过阈值化处理，得到图示的纹理效果。最近对人类视觉系统和大脑的研究进展迅速。研究结果使对早期工作的解释变得复杂化。这些早期工作是Marr和Hildreth基于他们的数学理论完成的。不管怎样，多尺度高斯和LOG滤波器在计算机视觉方面得到了广泛的使用。

156



图 5-26（原图来自David Shaller 1998）

（左）输入图像。

（中）用$\sigma=4$的高斯滤波器平滑后再提取边缘的结果

（右）用$\sigma=1$的高斯滤波器平滑后再提取边缘的结果。小尺度高斯滤波的结果表现出更多的细节和噪声

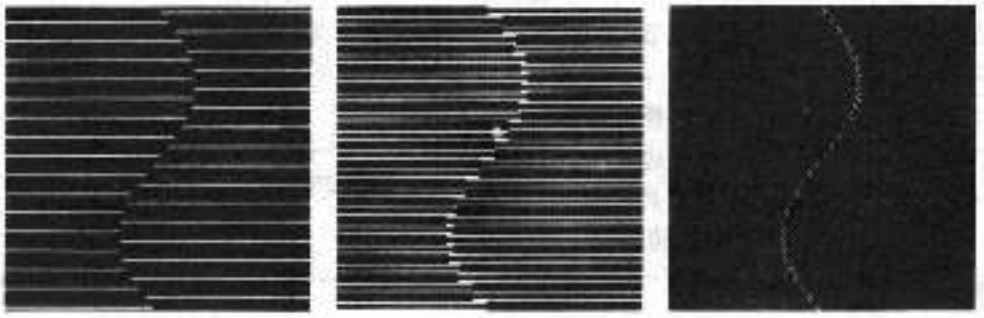


图 5-27

（左）线条端点形成的一条虚拟线，可能是两张包装纸接缝而形成的

（中）4×4的LOG滤波器对直线末端点产生响应

（右）另一个3×3 LOG滤波器仅对端点产生响应

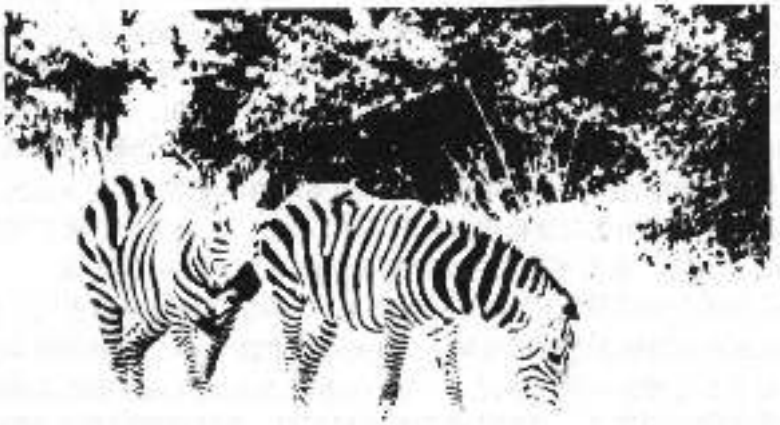


图5-28 阈值化后的图像。条纹两端形成的虚拟曲线构成了目标的边界。这些条纹线可以看成是水平放置的广义圆柱体的剖面（原图来自Eleanor Harding）

5.8 Canny边缘检测

Canny边缘检测算子是一个非常普遍和有效的算子，这里有必要对其做一下介绍，详细的讨论放在第10章。Canny算子首先对亮度图像进行平滑，然后从一个邻域到另一个邻域追踪具

有高梯度幅值的点，从而产生扩展的轮廓线段，图5-29表示，对实际复杂的室外图像进行边缘检测。图5-29中对圣路易斯拱门的轮廓检测效果很不错，利用参数$\sigma = 1$检测出拱门的一些金属缝隙，以及树木的一些内部变化，但采用$\sigma = 4$仅检测出这些目标的外部边界，如图5-29底部一行所示，算子隔离出许多棋盘状的纹理元素。为了进行比较，也给出了采用Roberts算子的结果，其中对梯度幅值采用较低的阈值，这个结果明显提取了场景（草地和篱笆）中的更多纹理元素，虽然其结构化程度不如Canny的输出结果。产生轮廓线段的算法将在10.3.2节中详细介绍。 157

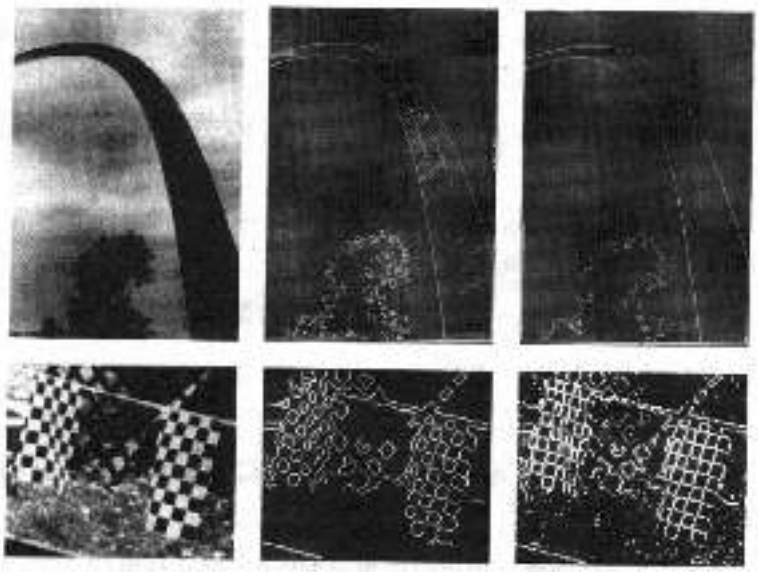


图 5-29

（左上）圣路易斯拱门的图像　　（左下）含纹理的图像
（中上）采用$\sigma = 1$的Canny算子的检测结果（中下）$\sigma = 1$的Canny算子的检测结果
（右上）$\sigma = 4$的Canny算子检测结果　　（右下）Roberts算子的结果，选择的阈值使梯度幅值前20%的像素通过

5.9 匹配滤波模板*

模板对于特定图像邻域的响应，与邻域和模板的相似性成正比。根据这一理论，我们现在知道如何针对特征设计模板，只需要设计和我们想检测的特征相似的模板。这种思想对边缘检测、纹理检测以及检测其他的特殊模式如孔或角点都是有用的。我们先利用一维信号来引入这个概念，一维信号本身也非常重要，它可以对应2D图像的行和列或者任意其他的穿过2D图像的分割线。有关概念和数学理论可以直接推广到2D情况。

5.9.1 向量空间

对于给定的$n \geq 1$，含n个实数坐标的所有向量的集合构成一个向量空间。下面介绍实用的向量空间运算。在研究分析几何或微积分时，读者可能已经涉及了$n = 2$或$n = 3$的向量。对于$n = 2$或$n = 3$，向量长度的定义，与从欧几里得平面几何和3D分析几何中使用的概念相同。在信号领域，长度与信号能量有关，能量定义为信号长度的平方，或者等价于所有坐标平方的和。后面会看到，信号能量是极其有用的概念。 158

定义47 信号$S = [s_1, s_2, \cdots, s_n]$的能量等于$\|S\|^2 = s_1^2 + s_2^2 + \cdots + s_n^2$。

注意在许多应用中，全范围的实值信号不会出现，因为有时坐标不可能为负值。例如，用12维的向量表示特定区域的12个月的降雨量，该向量就不应该有负坐标。类似地，图像每行的亮度值通常也保持在一个非负整数范围内。但我们将看到，这种情况下向量空间的概念仍然有用。通常为了做出某种解释会从所有的坐标中减去平均信号值，这就可能将某些坐标值变成0以下。另外模板中具有负值也是非常常见的。

用已定义的向量长度定义向量空间

设U和V是两个向量，u_i和v_i是实数，表示向量的坐标。a, b, c等是实数比例因子。

定义48 对于向量$U=[u_1, u_2, \cdots, u_n]$和$V=[v_1, v_2, \cdots, v_n]$，**向量的和**是向量$U\oplus V=[u_1+v_1, u_2+v_2, \cdots, u_n+v_n]$。

定义49 对于向量$V=[v_1, v_2, \cdots, v_n]$和实数（标量）$a$，**向量与标量的积**是向量$aV=[av_1, av_2, \cdots, av_n]$。

定义50 对于向量$U=[u_1, u_2, \cdots, u_n]$和$V=[v_1, v_2, \cdots, v_n]$，**向量的点集或者标量积**是向量$U\circ V=[u_1v_1+u_2v_2+\cdots+u_nv_n]$。

定义51 对于向量$V=[v_1, v_2, \cdots, v_n]$，它的**长度或者范数**是非负的实数

$\|V\|=V\circ V=(v_1v_1+v_2v_2+\cdots+v_nv_n)^{1/2}$。

定义52 当且仅当$U\circ V=0$时，称向量U和V**正交**。

定义53 向量$U=[u_1, u_2, \cdots, u_n]$和$V=[v_1, v_2, \cdots, v_n]$之间的**距离**，等于它们差的长度$d(U, V)=\|U-V\|$。

定义54 n维向量空间的**基**，由覆盖向量空间的n个独立向量$\{w_1, w_2, \cdots, w_n\}$组成。覆盖性质意味着任何向量$V$可以用基向量的线性组合表示，即$V=a_1w_1\oplus a_2w_2\oplus\cdots\oplus a_nw_n$。独立性质意味着任何一个基向量$w_i$都不能由其他基向量的线性组合表示。

[159]

上面所定义向量空间的特性

1. $U\oplus V=V\oplus U$
2. $U\oplus(V\oplus W)=(U\oplus V)\oplus W$
3. 存在向量O使得对所有的向量V，有$O\oplus V=V$
4. 对每个向量V，存在向量$(-1)V$使得$V\oplus(-1)V=O$
5. 对任意标量a, b和任意向量V，有$a(bV)=(ab)V$
6. 对任意标量a, b和任意向量V，有$(a+b)V=aV\oplus bV$
7. 对任意标量a和任意向量U与V，有$a(U\oplus V)=aU\oplus aV$
8. 对任意向量V，有$1V=V$
9. 对任意向量V，有$(-1V)\circ V=-\|V\|^2$

习题5.12

从列出的9个向量空间特性中任意选择5个，并证明它们是成立的。

5.9.2 利用正交基

向量空间的两个最重要的研究结果是：（1）每个向量可用唯一形式的基向量线性组合表示；（2）任何一组基向量都包含n个向量。用正交基表示任意向量V，具有更明确的含义，如下面的例子。

用基信号的线性组合表示信号的实例

考虑所有$n=3$的样本信号$[v_1, v_2, v_3]$的向量空间。以标准基来表示，任意向量$V=[v_1, v_2, v_3]=v_1[1,0,0]\oplus v_2[0,1,0]\oplus v_3[0,0,1]$，标准基向量互相正交并且具有单位长度，这样的基称为标准正交基。现在研究另外一个基向量集合$\{w_1, w_2, w_3\}$，其中$w_1=[-1,0,1]$，$w_2=[1,1,1]$，$w_3=[-1,2,-1]$，因为对$i\neq j$，有$w_i\circ w_j=0$，因此任意两个基向量是正交的。将它们变换到单位长度，得到新的基$\left\{\frac{1}{\sqrt{2}}[-1,0,1], \frac{1}{\sqrt{3}}[1,1,1], \frac{1}{\sqrt{6}}[-1,2,-1]\right\}$。现在用正交基表示信号$S=[10,15,20]$。信号$[10,15,20]$关于标准基，有

$$
\begin{aligned}
S\circ w_1 &= \frac{1}{\sqrt{2}}(-10+0+20)\\
S\circ w_2 &= \frac{1}{\sqrt{3}}(10+15+20)\\
S\circ w_3 &= \frac{1}{\sqrt{6}}(-10+30-20)\\
S &= (S\circ w_1)w_1\oplus(S\circ w_2)w_2\oplus(S\circ w_3)w_3\\
S &= (10/\sqrt{2})w_1\oplus(45/\sqrt{3})w_2\oplus 0w_3\\
\|S\|^2 &= 100+225+400=725\\
&= (10/\sqrt{2})^2+(45/\sqrt{3})^2+0^2=725
\end{aligned}
\tag{5-23}
$$

后面两个公式说明，当采用标准正交基时，通过对每个基向量上的分能量相加，很容易得到总能量。

该例说明如何用三个已知的基向量$[w_1, w_2, w_3]$表示信号$[10, 15, 20]$，我们已经看到这些基向量具有特殊的性质，一般地，设任意信号$S=[a_1, a_2, a_3]=a_1w_1\oplus a_2w_2\oplus a_3w_3$，那么$S\circ w_i=a_1(w_1\circ w_i)\oplus a_2(w_2\circ w_i)\oplus a_3(w_3\circ w_i)=a_i(w_i\circ w_i)=a_i$，其中$i\neq j$时，$w_i\circ w_j=0$；$i=j$时，$w_i\circ w_j=1$。因此，正交基是非常方便的。通过计算信号在每个基向量上的分能量，可以很容易地得到信号的总能量。例如用信号$S_2=[-5,0,5]$来重复上面的计算，S_2可以通过S减去S的平均信号值得到，即$S_2=S\oplus(-1[15,15,15])$。$S_2$与$S\circ w_1$相同，因为沿$[1,1,1]$的分量为0，$S_2$仅仅是$w_1$与一个标量的积，$S_2=(10/\sqrt{2})w_1=(10/\sqrt{2})((1/\sqrt{2}))[-1,0,1]=[-5,0,5]$，我们说$S_2$和$w_1$具有相同的模式。如果$w_1$是滤波器，则它与信号$S_2$匹配得非常好。从某种意义上，它也与信号$S$匹配得非常好。进一步拓展这个思想，但在开始拓展之前，要注意到，对n维信号向量空间，存在多个不同的有用的标准正交基。

160

习题5.13

（a）接着前面方框内的实例，用基$\left\{\frac{1}{\sqrt{2}}[-1,0,1], \frac{1}{\sqrt{3}}[1,1,1], \frac{1}{\sqrt{6}}[-1,2,-1]\right\}$表示向量$[10, 14, 15]$。（b）表示向量$[10, 19, 10]$，它与那个基向量最相似？为什么？

从向量和点积的特性，可以得到公式（5-24）所示的柯西-施瓦茨（Cauchy-Schwartz）不等式。它的基本意思是，单位向量的点积必定介于−1和1之间。这样，就得到决定两向量相似性的度量方式：若$U = V$，则得+1；若$U = -V$，则得−1。用规范化点积定义两个向量的夹角。这个夹角与在2D或3D空间中的三角计算结果是相同的。对$n \geq 3$，这个夹角或它的余弦值，作为衡量两向量相似性的抽象度量方式。如果两向量的规范化点积是0，则它们不相似；如果是1，则它们最相似；如果是−1，则它们互为相反，此时是否相似取决于实际问题。

161

5.9.3 柯西-施瓦茨不等式

对任意两个非零向量U和V，有

$$-1 \leq \frac{U \circ V}{\|U\|\|V\|} \leq +1 \tag{5-24}$$

定义55 设U和V是任意两个非零向量，那么U和V的**规范化点积**定义为$\left(\frac{U \circ V}{\|U\|\|V\|}\right)$。

定义56 设U和V是任意两个非零向量，那么U和V的**夹角**定义为$\cos^{-1}\left(\frac{U \circ V}{\|U\|\|V\|}\right)$。

习题5.14

画出下面的五个向量，并计算规范化点积，或者计算每对向量之间夹角的余弦值。这些向量是[5, 5]、[10, 10]、[−5, 5]、[−5, −5]、[−10, 10]，哪一对之间互相垂直？哪一对具有相同的方向？哪一对具有相反的方向？将相对方向与规范化点积的值进行比较。

5.9.4 m × n图像的向量空间

所有具有实值元素的$m \times n$矩阵的集合是维数为$m \times n$的向量空间，这里用模板和图像区域来解释向量空间理论，并说明如何应用向量空间理论。在本节，图像模型是在$m \times n$个离散采样点$[x, y]$的图像函数。我们主要针对2 × 2和3 × 3的矩阵，但每种情况都可以很容易地推广到任意大小的图像或模板。

5.9.5 2 × 2邻域的Robert基

亮度图像的2 × 2邻域结构，可以用图5-30所示的基来解释，我们将其称之为Roberts基。四个基向量中的两个在图5-15中表示过。如下面的习题所示，任意2 × 2的实值邻域都可由这四个基向量的和来唯一表示。比例因子的相对大小直接表明图像邻域和基向量的相似程度，因而可以用来解释邻域结构。图5-30给出了几个例子。

习题5.15

162 验证图5-30所示的Roberts基向量是正交的。

习题5.16

考虑所有2 × 2图像的向量空间，图像的像素值为实值。（a）确定a_i的值，使图像 $\begin{array}{|c|c|}\hline 10 & 5 \\ \hline 5 & 0 \\ \hline \end{array}$ 表示为四个Roberts基图像$\mathbf{W}_i$的线性组合。（b）解释为什么对任意的2 × 2图像总能找到唯一的a_i。

习题5.17

假设2 × 2图像 $\begin{array}{|c|c|}\hline a & b \\ \hline c & d \\ \hline \end{array}$ 具有能量e_1、e_2、e_3、e_4，它们分别沿着四个Roberts基向量$\mathbf{W}_1$、$\mathbf{W}_2$、

W_3、W_4。以a、b、c、d计算四个能量e的公式是什么？

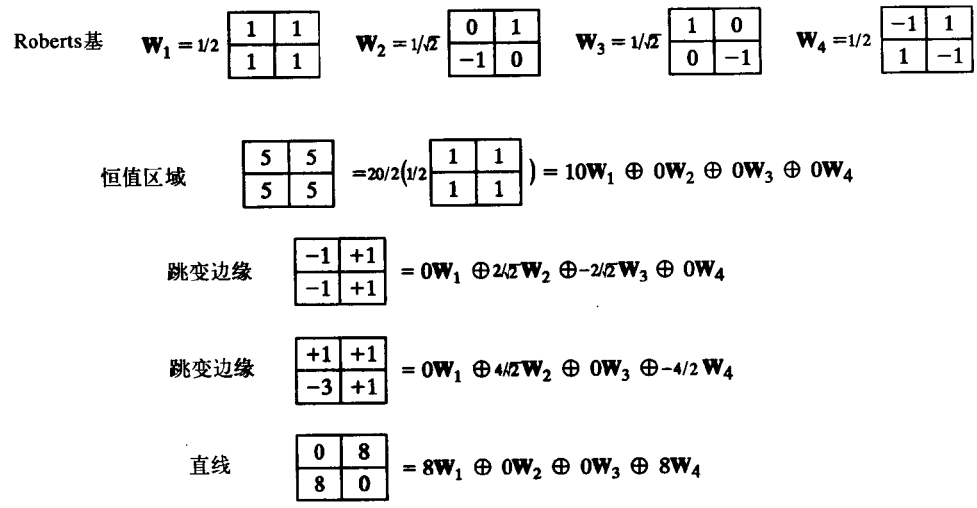
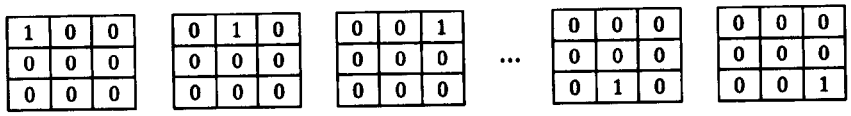


图 5-30

（第1行）所有2 × 2图像的基，其中包含两个Roberts梯度模板
（第2行）恒值区域与恒值图像存在倍数关系
（第3行）垂直跳变边缘仅在梯度模板上有能量
（第4行）对角跳变边缘沿匹配的梯度模板具有最大的能量
（第5行）直线模式沿恒值模板W_1和直线模板W_4具有能量

5.9.6 3 × 3邻域的Frei-Chen基

通常用于图像处理的模板大小为3 × 3或更大。3 × 3图像邻域的标准基如图5-31所示。标准基的一个优点是，用标准基扩展任意图像邻域的方法是显而易见的。但这种扩展对于邻域的2D结构提供不了任何信息。图5-32所示的Frei-Chen基包含一组标准正交模板，它们对3 × 3邻域的结构可以给出简单的解释。 163



所有3 × 3矩阵空间的9个标准基向量

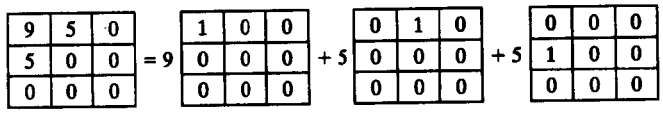


图5-31 任意3 × 3矩阵可表示为不超过9个的标准矩阵的加权和

（上行）九个基向量
（下行）用基表示的一个矩阵

用Frei-Chen基表示图像邻域允许将能量解释为梯度、波纹和直线等。当亮度结构与基向量或模板相似时能量就较高。每个基向量有一个特殊设计的结构。基向量W_1和W_2与Prewitt和Sobel梯度模板相似，基向量W_7和W_8与普通的3 × 3 Laplacian模板相似。对穿过3 × 3邻域的一像素宽的直线，直线模板响应很强烈；而两波纹模板模拟两个互相垂直的波，它们具有两个波峰、两个波谷和三个零交叉。由于要求集合是正交的，向量元素与前面孤立设计的模板略微不同。 164

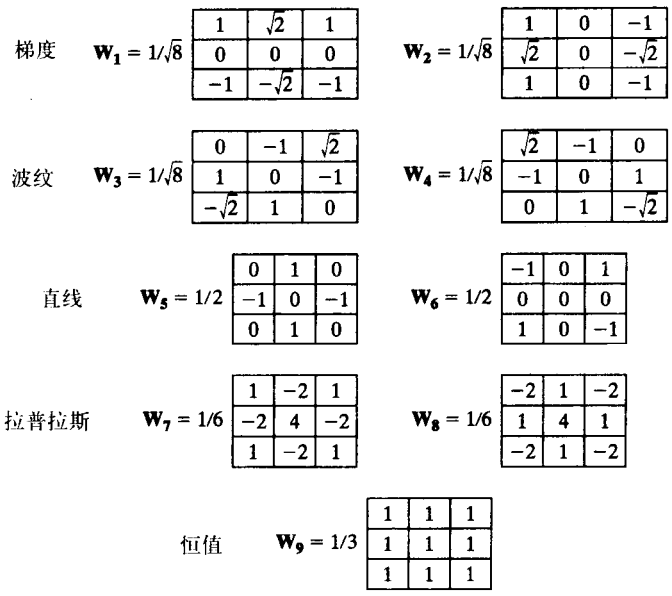


图5-32 所有3 × 3实值图像的Frei-Chen基

算法5.2计算一幅二值图像，检测出某个给定子空间中能量较大的亮度邻域结构。为了检测边缘，可以根据沿着基向量$\mathbf{W}_1$、$\mathbf{W}_2$的邻域能量选择像素，这通过设置**S** = {1, 1, 0, 0, 0, 0, 0, 0, 0}来表示。将亮度邻域投影到Frei-Chen基向量上，计算例子如下。

算法5.2 在选定子空间内检测具有高能量的邻域

F[r, c]是输入亮度图像；算法不改变**F**。

S是位向量，当且仅当感兴趣的子空间包括$\mathbf{W}_j$时，**S[j] = 1**。

*thresh*是要求的能量比阈值。

*noise*是噪声能量级别。

G[r, c]是输出图像，是一个二值图像，**G[r, c]** = 1表示**F[r, c]**在选定子空间**S**中有超过阈值的能量。

```
procedure detect_neighborhoods(F, G, S, thresh, noise);
{
for r := 0 to MaxRow - 1
for c := 0 to MaxCol - 1
  {
    if [r, c] is a border pixel then G[r, c] := 0;
    else G[r, c] := compute-using-basis (F, r, c, S, thresh, noise);
  }
}
procedure compute_using_basis(IN, r, c, thresh, noise)
{
\\N[r, c]是IN[i]中像素[r, c]的3 × 3邻域。
```

```
average_energy := N[r, c]∘W9 ;
subspace_energy := 0.0;
for j := 1 to 8
{
  if (S[j]) subspace_energy := subspace_energy + (N[r, c]∘Wj)^2 ;
}
if subspace_energy < noise return 0;
if subspace_energy/((N[r, c]∘N[r, c])−average_energy) < thresh return 0;
else return 1;
}
```

165

利用Frei-Chen基表示亮度邻域的例子

考虑亮度邻域 $N=\begin{array}{|c|c|c|}\hline 10&10&10\\\hline 10&10&5\\\hline 10&5&5\\\hline\end{array}$.

像前面一样，利用点积检测沿每个基向量的向量分量。由于是标准正交基，总的图像能量就是分量能量的和，N的结构可以用分量来解释。

$$N\circ\mathbf{W}_1=\frac{5+5\sqrt{2}}{\sqrt{8}}\approx 4.3;\ energy\approx 18$$

$$N\circ\mathbf{W}_2=\frac{5+5\sqrt{2}}{\sqrt{8}}\approx 4.3;\ energy\approx 18$$

$$N\circ\mathbf{W}_3=0;\ energy=0$$

$$N\circ\mathbf{W}_4=\frac{5\sqrt{2}-10}{\sqrt{8}}\approx -1;\ energy\approx 1$$

$$N\circ\mathbf{W}_5=0;\ energy=0$$

$$N\circ\mathbf{W}_6=2.5;\ energy\approx 6$$

$$N\circ\mathbf{W}_7=2.5;\ energy\approx 6$$

$$N\circ\mathbf{W}_8=0;\ energy=0$$

$$N\circ\mathbf{W}_9=25;\ energy=625$$

N中的总能量是N∘N = 675，其中625是沿$\mathbf{W}_9$向量方向上的平均亮度。其他方向上的所有能量是50，其中的72%即36是梯度基向量$\mathbf{W}_1$和$\mathbf{W}_2$上的，倘若对梯度子空间感兴趣，则邻域中心将标记为一个已被检测的特征。

习题5.18

验证图5-32所示的9个基向量是标准正交基。

习题5.19

(a) 以图5-32所示的基向量表示亮度邻域 $\begin{array}{|c|c|c|}\hline 0&0&1\\\hline 0&1&0\\\hline 1&0&0\\\hline\end{array}$ 。所有的能量都沿着直线基向量$\mathbf{W}_5$和

$\mathbf{W}_6$分布吗？（b）对于亮度邻域 $\begin{array}{|c|c|c|}\hline 0&1&0\\\hline 0&1&0\\\hline 0&1&0\\\hline\end{array}$，重复问题（a）。

166

习题5.20

（a）以图5-32所示的基向量表示亮度邻域 $\begin{array}{|c|c|c|}\hline 10&10&10\\\hline 10&20&10\\\hline 10&10&10\\\hline\end{array}$，所有的能量是如你所期望的那样

沿着某些基向量分布的吗？（b）对亮度邻域 $\begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ 的解释有什么不同吗？为什么？（c）什么样的图像邻域仅仅对$\mathbf{W}_7$和$\mathbf{W}_8$给出响应？

习题5.21

编程实现利用Frei Chen基检测像素，算法如上所述。允许程序的用户以9位的字符串输入感兴趣的子空间S。用户也可输入噪声能量级别和阈值，该阈值决定在所选子空间要求的最小能量。用实际图像测试你的程序，也要用上面习题所示的测试模式进行测试。

5.10 卷积和交叉相关*

前面的内容说明，检测可以利用将模板或图像模式与图像邻域相匹配的方法实现。另外，图像平滑也基于同样的道理。本节我们给出交叉相关和卷积这两种重要运算的定义，它们明确表示出模板在图像上的移动，并计算模板与每个图像邻域的点积。

5.10.1 模板运算定义

首先将简单的图像平滑重新定义为图像与平滑模板的交叉相关。利用盒形滤波器算出输出图像，对输入像素邻域内的各点进行等量加权就得到相应的输出像素。这等价于与权系数为 $\frac{1}{mn}$ 的$m\times n$图像模板进行点积运算，如图5-33所示的3×3模板。假设m和n都是奇数，除以2并忽略余数，公式（5-25）定义了利用模板$H[x, y]$从输入图像$F[x, y]$计算输出像素$G[x, y]$值的点积运算。在这个公式中，模板H以原点为中心，这样$H[0, 0]$是模板的中心像素，H对$F[x, y]$邻域像素的加权方式是显而易见的。另一个计算G中输出像素的公式是公式（5-26），它是

167

对公式（5-25）中的变量稍加改变得到的，它可以利用偶数维的模板$H[i, j]$。

定义57 图像$F[x, y]$和模板$H[x, y]$的**交叉相关**定义如下：

$$G[x, y] = F[x, y] \otimes H[x, y] = \sum_{i=-m/2}^{m/2}\sum_{j=-k/2}^{k/2} F[x+i, y+j]H[i, j] \tag{5-25}$$

参见图5-35。

为了实现该计算公式：

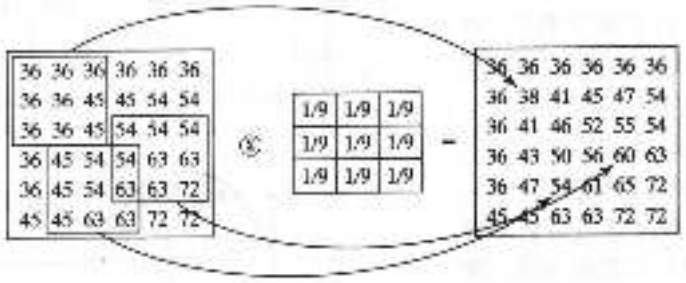
假设模板$H[x, y]$以原点为中心，这时负坐标是有意义的；

图像$F[x, y]$不必以原点为中心；

当$H[]$与$F[]$不完全重合时，结果$G[x, y]$必须以另一种方式定义。

另一种替换公式不要求模板维数为奇数，但应当看成是整幅图像的变换，而不只是以像素$G[x, y]$为中心的运算。

$$G[x, y] = \sum_{i=0}^{n-1} \sum_{j=0}^{k-1} F[x+i, y+j]H[i, j] \tag{5-26}$$



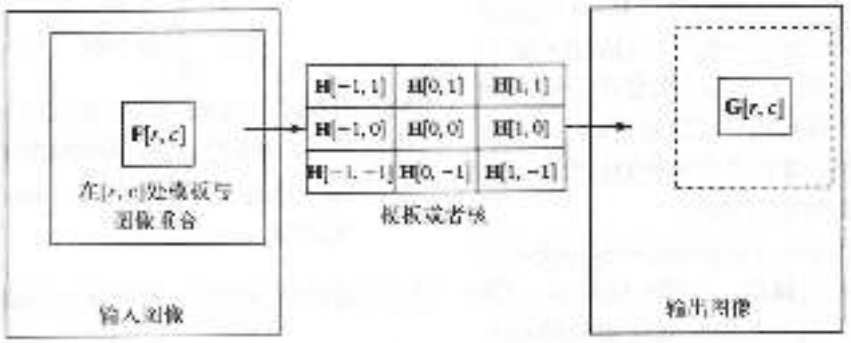


图5-33 用3×3盒形滤波器平滑图像，可以看成是输入图像邻域与盒形模板的点积运算，模板相当于一个等值小图像

习题5.22

假设图像**F**只有图像中心的像素值为1，其余全为0。**F**与图5-33所示的盒形模板做卷积能得到什么样的输出图像**G**?

习题5.23

设计一个模板，检测与X轴成30°角的边缘元素。模板不应该对其他方向的边缘元素或其他模式产生强烈响应。

习题5.24 角点检测

(a) 设计4个5×5的模板，检测与图像轴平行的任意矩形的角点。矩形可以比背景亮或比背景暗。(b) 你的模板是正交的吗? (c) 说明检测角点的决策过程，并证明其有效性。 168

5.10.2 卷积运算

定义58 函数$f(x, y)$和$h(x, y)$的卷积定义为:

$$g(x,y)=f(x,y)*h(x,y)$$
$$=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(x',y')h(x-x',y-y')\mathrm{d}x'\mathrm{d}y' \tag{5-27}$$

卷积与交叉相关密切相关，公式（5-27）对连续图像函数给出了卷积的正式定义。为了定义积分并能够实际使用，2D图像函数$f(x, y)$和$h(x, y)$在xy平面上的有限矩形外应当具有零像素值，且在其表面下的体积是有限的。对于滤波来说，核函数$h(x, y)$在某个矩形之外通常为0，该矩形要比支撑$f(x, y)$的矩形小得多。为了对空间频率f进行全局性分析，支撑h的矩形将包括f的所有支撑。在选读的5.11节傅里叶分析部分进行更详细地介绍。图5-34说明了对于1D信号计算两函数的卷积$g(x)$的步骤。核函数$h(x)$首先相对原点翻转，然后平移到点x，在x处计算$g(x)$。对输入函数$f(x')$和新的核函数$h(x'-x)$的乘积进行积分，最后得到$g(x)$。由于函数在区间$[a, b]$外为零，积分可限制到有限的区间。对于数字图像，卷积计算就是求乘积的离散和，而不是上面定义的连续积分。

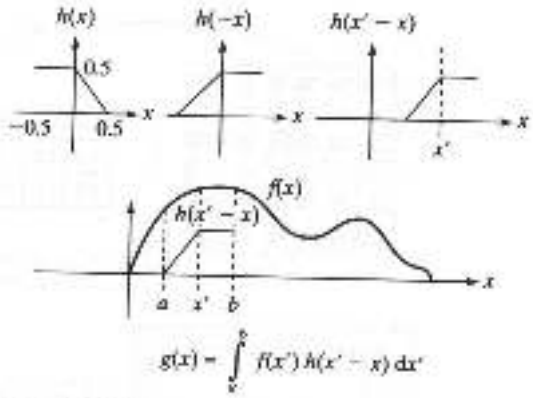


图5-34 计算信号$f(x)$和核$h(x)$的卷积，即$g(x)=f(x)*h(x)$。对任意点x，核h进行翻转然后平移到x，然后求$f(x)$和翻转平移后h的乘积之和，从而算出$g(x)$

169

交叉相关将模板或核直接平移到图像点$[x, y]$，而不经过翻转，如图5-35所示。否则，它就与卷积运算相同。从概念上来说，不考虑对核的翻转，而只是将核放在图像的某个位置，这样做会更加容易。如果核是对称的，则翻转后的核与原先的核是相同的，那么卷积的结果就与相关的结果相同。但对称主要是针对平滑模板和其他各向同性算子的，很多边缘检测的模板是不对称的。尽管卷积和交叉相关形式上不同，但由于它们之间的相似性，进行图像处理时常常认为它们都是“卷积”。本书用到的许多模板，都假设在应用于图像前不经过翻转。规范化交叉相关，将$\mathbf{G}[x, y]$除以$\mathbf{F}[x, y]$和$\mathbf{H}[x, y]$的幅值，结果就可以解释为是$\mathbf{F}$结构和$\mathbf{H}$结构进行匹配的结果，而不受比例因子的影响，这和我们前面讨论的一样。

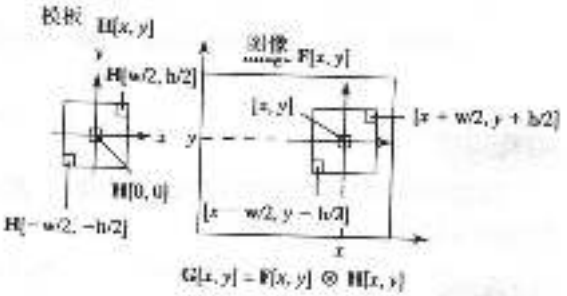


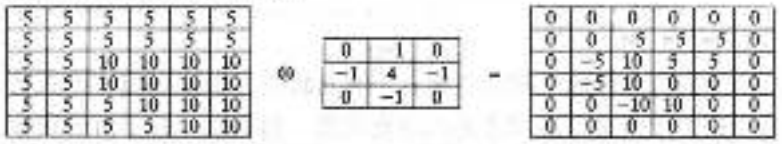
图5-35 计算图像$\mathbf{F}[x, y]$和模板$\mathbf{H}[x, y]$的交叉相关$\mathbf{G}[x, y]$即$\mathbf{G}[x, y]=\mathbf{F}[x, y]\otimes\mathbf{H}[x, y]$。为计算$\mathbf{G}[x, y]$，模板$\mathbf{H}[x, y]$中心放在输入图像点$\mathbf{F}[x, y]$的位置，求出$\mathbf{F}$图像值和$\mathbf{H}$上对应权值的乘积之和

170

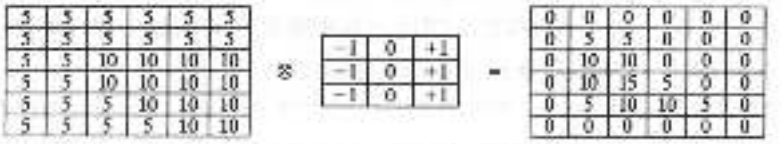
习题5.25 矩形检测

利用前面习题5.24的角点检测过程，编写程序检测图像中的矩形。（假设矩形的边与图像的边平行。）第一步应检测候选的矩形角点。第二步抽取含四个候选角点的子集，其中四个角

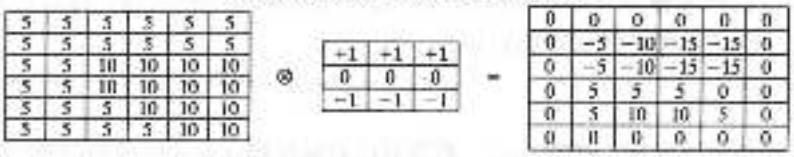
点根据几何约束组成合适的矩形。第三步是可选项，进一步检查四个角点，保证候选矩形内的亮度是统一的，并与背景形成对比。如果给一幅含噪声的棋盘图像，你的程序会出现什么结果？对一幅含噪的棋盘图像，如图5-7所示，以及一幅带有矩形窗的建筑图像，如图5-42所示，用这两幅图像对你的程序进行测试。



与LOG模板做交叉相关，边界处产生零交叉



与列导数模板做交叉相关，检测列边界



与行导数模板做交叉相关，检测行边界

图5-36 图像与多个模板进行交叉相关，增强了区域边界

（上）二阶导数算子在边界产生零交叉

（中）列（x）导数算子检测出列的变化

（右）行（y）导数算子检测出行的变化

习题5.26

已知$\mathbf{H}=\begin{bmatrix}1&0&-3\\0&4&0\\-3&0&1\end{bmatrix}$和$\mathbf{F}=\begin{bmatrix}5&5&5&5&5&5\\5&5&5&5&8&8\\5&5&5&8&8&8\\5&5&8&8&8&8\\8&8&8&8&8&8\end{bmatrix}$，计算$\mathbf{G}=\mathbf{F}\otimes\mathbf{H}$。 171

习题5.27 点扩展

已知核$\mathbf{H}=\begin{bmatrix}1&2&1\\2&5&2\\1&2&1\end{bmatrix}$，它与图像$\mathbf{F}[x,y]$作卷积的结果是什么？其中$\mathbf{F}[x_0,y_0]=1$，其他所有像素为0。

习题5.28

假设函数$h(x)$在$-1/2\leqslant x\leqslant 1/2$范围内的值为1，其他范围的值为0。假设函数$f(x)$在$10\leqslant x\leqslant 20$范围内的值为1，其他范围的值为0。（a）画出两个函数f和h。（b）计算并画出函数$g(x)=f(x)*h(x)$。（c）计算并画出函数$g(x)=h(x)*h(x)$。

5.10.3 并行计算

从卷积的定义可见，$g(x_1, y_1)$ 的计算与 $g(x_2, y_2)$ 的计算是独立的。事实上，所有的积分都可以同时、并行地进行计算。另外，对单个值 $g(x, y)$ 计算的积分可以同时构成所有的乘积，这使高度并行的系统成为可能。各种计算机结构都能够进行全部或部分的并行运算。

习题5.29

点彩画派以这样的方式作画：用画笔垂直于画布轻点，每次点出一个彩色点。这每个点类似于数字图像中的一个像素。参观者退后观察图像，将看到一幅平滑的图像。编程实现这种画法。程序应当提供一个颜色调色板和其他一些选项，如选择画笔的大小或在轻点时是否用“或”、“异或”运算。运行你的程序，创建一幅夜晚星空的图像。程序要能够将所画的图像数据存储成外部文件，这样以后就能接着修改这幅画。

习题5.30

编程实现模板和图像的卷积。程序应当从输入文件中以同样的格式读取图像和模板。可用上个习题产生的美术图片测试你的程序。

习题5.31

在搜索地球外智慧生命（SETI）时，希望通过扫描深层空间能够检测到感兴趣的信号。设信号S是以二进制表示的前100个质数序列，R是收到的信号，它比S长很多。假设R包含噪声并且由实值组成。为了检测S是否嵌入在R中，交叉相关或者规范化交叉相关能行吗？为什么？

5.11 正弦波空间频率分析*

傅里叶分析在信号处理中非常重要，很多书中都讨论了傅里叶分析的理论和应用，我们在此只做简单介绍，以介绍过的向量空间概念为基础。

数学家傅里叶将海平面想像成一组正弦波的和。由潮汐或轮船引起的大波浪的波长较长（频率低），由风或坠落物体引起的小波浪的波长较短（频率高）。图5-37的上面一行表示三个纯波，在1D空间$x\in[0, 512]$内周期个数分别为3、16、30。下面一行是两个函数，一个是上面的三个纯波之和，另一个是前两个纯波之和。这种多个波的集合可用来建立2D图像函数甚至3D密度函数。

利用傅里叶分析，把多数实际表面或者实际函数用正弦基来表示。沿着基向量的能量可以解释为所表征表面（函数）的结构。在表面的大块区域内有重复模式时，例如城市航测图像中的街区、大片水域的波浪、大片森林或农场的纹理等，傅里叶分析是比较实用的分析方法。这种思想可以扩展到整幅图像，或者图像的不同窗口，将它们用傅里叶基表示，然后对图像滤波，或根据不同基向量上的图像能量进行决策。例如，从图像中减去沿高频正弦波或余弦波的成分，则可以去除高频噪声。等价地，在重构空间图像时可以只增加低频波而忽略高频波。

5.11.1 傅里叶基

为了直观，假设一组标准正交正弦基图像（或图像函数）$E_k = E_{u,v}(x, y)$。这里，k和u、v是整数，确定了基向量的有限集合。关于参数u、v如何决定基向量很快我们就会清楚，但现在我们只用单个下标k，目的是把注意力集中在基本概念上。图5-37的下面一行表示由上面一个或两个纯余弦波相加得到的两个信号。利用更多的纯余弦波能够建立更复杂的函数。将图

5-38中的三个纯波进行叠加，每项前面乘一个系数，得到如图5-39所示的新的图像函数。利用傅里叶基函数E_i，任意图像函数都可以表示为$I(x,y)=\sum_{i=0}^{N-1}a_iE_i[x,y]$，和前面各节类似，$a_i$度量$I[x,y]$和$E_i[x,y]$的相似性，以及$I[x,y]$在特定成分波形上的能量。图像处理运算只针对$a_i$的值而不是亮度值$I[x,y]$进行。下面讨论三种主要运算。

172 ↕ 174

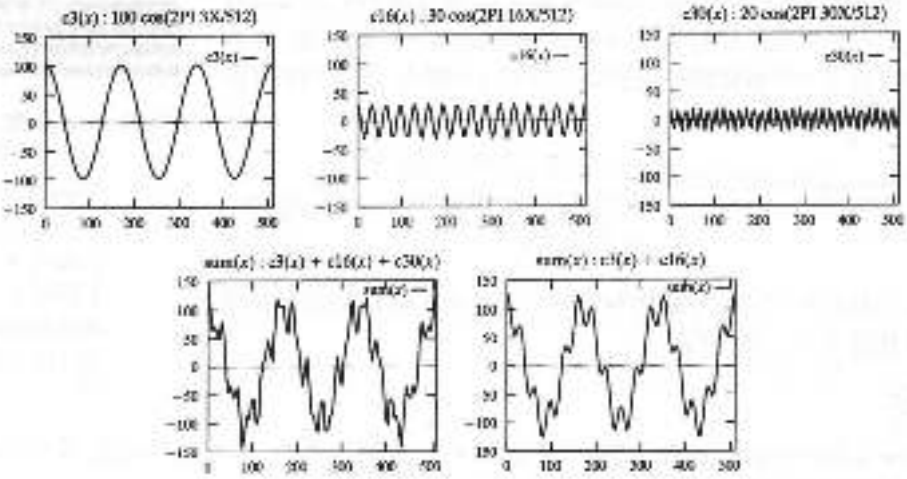


图 5-37

（上行）三个纯波，$100\cos\left(2\pi\frac{3x}{512}\right)$，$30\cos\left(2\pi\frac{16x}{512}\right)$和$20\cos\left(2\pi\frac{30x}{512}\right)$

（下行左边）三个纯波之和

（下行右边）前两个纯波之和

利用傅里叶基进行图像运算：

1. 利用傅里叶基，可去除图像或信号中的高频噪声。信号I表示为$\sum_i a_iE_i$。高频正弦E_i的系数a_i被置为0，用那些$a_i\neq 0$的剩余基函数之和来计算一个新的信号I'。

2. 傅里叶基可用于抽取纹理特征，可用这些纹理特征对图像区域中的目标进行分类。用傅里叶基表示图像或图像区域之后，可通过a_i计算特征，并用这些特征进行分类决策。一行行的水面波或止稼波就属于这样的过程，其中确定频率和纹理区域的方向时要用到a_i。

3. 傅里叶基可用于图像压缩。发送者可以发送a_i的子集，接收者通过求已知正弦成分的和重构近似的图像。如果需要，所有的a_i都可以发送，可以按照能量顺序或者按照频率顺序。接收者根据所得到的内容，可以在任意时间终止传送。

我们的目标是产生一组可用的图像函

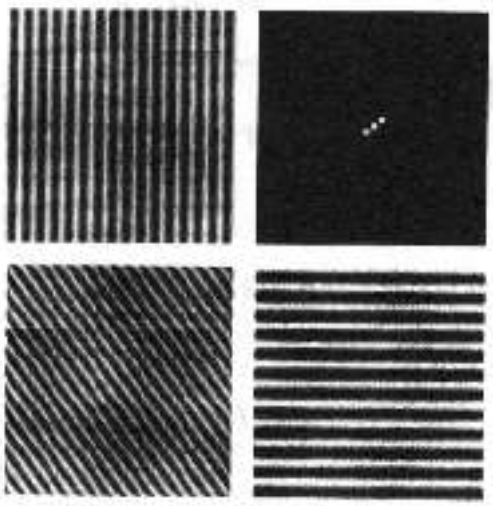


图5-38 空间域[x, y]上的不同正弦图像函数。左上图由公式100cos(2π (16x/512))+ 100产生，沿着x轴有16个周期。右下图由公式100cos(2π (12y/512)) + 100产生，沿着y轴有12个周期。左下图由公式100cos(2π(16x/ 512 + 12y/512)) + 100产生，注意左下图中的波是怎么与左上图和右下图中的波对应的。傅里叶功率谱如右上图所示

数的基，并弄明白在实际中如何使用。这需要关于连续函数的数学背景知识。对于方形xy平面，假设坐标系统的原点是图像函数的中心，数字图像$I[x, y]$由N^2个采样点组成。首先，建立一组不同频率的正弦波作为连续信号f的正交基。如果m、n是两个不同的整数，那么带频率参数的两个余弦波在区间$[-\pi, \pi]$上是正交的。读者通过完成下面的习题，验证函数集$\{1, \sin(mx), \cos(nx), \cdots\}$在区间$[-\pi, \pi]$上是一个正交集。余弦波的正交性服从公式（5-28），因为对所有的整数k，$\sin(k\pi)=0$。

$$\int_{-\pi}^{\pi}\cos(m\theta)\cos(n\theta)\mathrm{d}\theta=\frac{\sin(m-n)\pi}{2(m-n)}+\frac{\sin(m+n)(-\pi)}{2(m+n)}=0\ \text{对于}\ m^2\neq n^2 \tag{5-28}$$

余弦函数的N个空间对称样本组成一组向量，这组向量在前面定义的点积意义上是正交的。

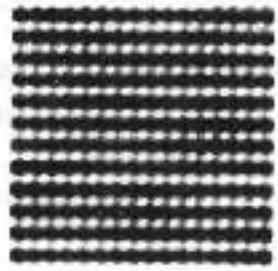


图5-39 图像函数$I[x, y]=100E_1+30E_2+10E_3$，其中$E_1$如图5-38的右下图所示，$E_2$如左上图所示，$E_3$如左下图所示（这可能是果园或泡沫漂充物的模型？）

习题5.32

考虑定义在区间$x\in[x_1, x_2]$上的所有连续函数。证明函数集合f, g, h, …和标量a, b, c, … 一起组成一个向量空间，证明下列性质：

$$f\oplus g=g\oplus f \qquad (f\oplus g)\oplus h=f\oplus(g\oplus h)$$
$$c(f\oplus g)=cf\oplus cg \qquad (a+b)f=af\oplus bf$$
$$(ab)f=a(bf) \qquad 1f=f$$
$$0f=\mathbf{0}$$

习题5.33

类似上面的习题，对于区间$x\in[x_1, x_2]$上的连续函数空间，定义点积和对应的范数如下：

$$f\circ g=\int_a^b f(x)g(x)\mathrm{d}x;\quad \|f\|=\sqrt{f\circ f} \tag{5-29}$$

证明点积具有如下四个性质：

$$(f\oplus g)\circ h=(f\circ g)+(g\circ h)$$
$$f\circ f\geqslant 0$$
$$f\circ f=0\Longleftrightarrow f=\mathbf{0}$$
$$f\circ g=g\circ f$$
$$(cf)\circ g=c(f\circ g)$$

习题5.34 奇函数和偶函数

如果$f(-x)=f(x)$，则该函数是偶函数。如果$f(-x)=-f(x)$，则该函数是奇函数。（a）证明$\cos(mx)$是偶函数，$\sin(nx)$是奇函数，其中m、n是非0整数。（b）设f和g分别是在区间$[-L, L]$上的奇函数和偶函数，证明$\int_{-L}^{L}f(x)g(x)\mathrm{d}x=0$。

习题5.35

利用习题5.33给出的点积定义，证明下列正弦函数f_k的集合在区间$[-\pi, \pi]$上是正交的。

$f_0(x) = 1$; $f_1(x) = \sin(x)$; $f_2(x) = \cos(x)$; $f_3(x) = \sin(2x)$; $f_4(x) = \cos(2x)$; $f_5(x) = \sin(3x)$; $f_6(x) = \cos(3x)$; …

5.11.2 2D图像函数

定义59 复值图像函数

$$\begin{aligned}\mathbf{E}_{u,v}(\mathbf{x}, \mathbf{y}) &= e^{-j\,2\pi(ux+vy)} \\ &= \cos(2\pi(ux + vy)) - j\sin(2\pi(ux + vy))\end{aligned} \tag{5-30}$$

其中u和v是图5-38所示的空间频率参数，$j = \sqrt{-1}$。

175
176

利用复值有其方便之处，这样对具有同样频率的余弦波和正弦波分别进行计算。其中正弦波与余弦波具有相同的结构，但相位相差1/4波长。当其中某个基函数与图像函数高度相关时，就意味着图像函数在频率u和v上具有较高的能量。傅里叶变换将图像函数转换成相关参数的阵列。我们首先讨论积分形式，然后给出数字图像的离散和形式。

定义60 2D傅里叶变换将一个空间域函数$f(x, y)$变换成u, v频域函数

$$\begin{aligned}F(u, v) &= I\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)E_{u,v}(x, y)\mathrm{d}x\,\mathrm{d}y \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)e^{-j2\pi(ux+vy)}\mathrm{d}x\,\mathrm{d}y\end{aligned} \tag{5-31}$$

函数f必须满足一定的条件。特别地，为了保证上式适定，图像函数要满足下列条件：积分$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}|f(x, y)|\mathrm{d}x\mathrm{d}y$的值是有限的，而且在某个矩形$R$之外有$f(x, y) = 0$，则公式中的无穷大积分就可以用$R$的上下限代替。另外，在$R$范围内，$f$的极点个数是有限的，而且没有无穷大间断点。

我们经常要用到功率谱，它综合了相同频率成分u，v下的正弦波和余弦波的能量。图5-38的右上图表示功率谱。

习题5.36

$\mathbf{F}(0, 0)$的特殊含义是什么？其中$\mathbf{F}(u, v)$是图像函数$f(x, y)$的傅里叶变换。

定义61 傅里叶功率谱计算公式为：

$$P(u, v) = (\mathrm{Real}(F(u, v))^2 + \mathrm{Imaginary}(F(u, v))^2)^{1/2} \tag{5-32}$$

图5-40显示，2D正弦波的实际波长与沿各轴的投影波长之间的关系。u是沿着X轴的频率，表示每单位长度的周期数，$1/u$是波长。v是沿着Y轴的频率，$1/v$是波长。λ是正弦波沿着它的中轴或传播方向的波长。通过以两种不同方式计算图5-40右图的三角形面积，得到下面的公式。该公式能够帮助我们了解功率谱提供了哪些关于原图的频率信息。

177

$$\begin{aligned}\lambda\sqrt{(1/u)^2 + (1/v)^2} &= (1/u)(1/v) \\ \lambda &= \frac{1}{\sqrt{u^2 + v^2}}\end{aligned} \tag{5-33}$$

假设图片的宽度和高度都为1。在图5-38中，图的宽度方向有$u = 16$个周期，所以波长是$1/u = 1/16$。同样，$1/v = 1/12$。应用公式（5-33），得到$\lambda = 1/20$。通过计算图5-38左下图中波的个数，我们看到沿着1.4单位长度的对角方向有27个波。于是沿着2D波实际方向产生的期望频率是27/1.4 ≈ 20。

图5-41显示图5-38中的三个正弦图像函数功率谱的主要响应。图5-38右上图的功率谱实际上表示在二个点产生的强烈响应，而不是一个点。首先注意到，$F(0, 0)$是$f(x, y)$的总能量。
178 由于图5-38的每个正弦波的均值是100而不是0，它们在0频率上具有较大的平均能量。另外根据定义$P(-u, -v) = P(u, v)$可以明显看出，功率谱关于原点$u = 0$，$v = 0$对称。图5-42显示四个真实图像的功率谱。

功率谱不必解释为一幅图像，而是原图的功率相对频率参数u和v的2D显示。事实上，光学设备可以计算这种变换，因此功率谱就可以实现为一幅物理图像。第2章中简单提到了一种传感器阵列，它分成扇区和环区（参见第2章图2-4c的ROSA结构）。由于旋转π角度时功率谱是对称的，如图5-42所示，因此可用扇区来采样有向功率，用环区采样与方向无关的频率带。这种采样方式也可通过软件完成。任何情况下，如果采样了n_r个环区和n_s个扇区，则得到$n_r + n_s$个特征，这些特征对于图像邻域的分类是有用的，这些特征是关于这些邻域的特征。

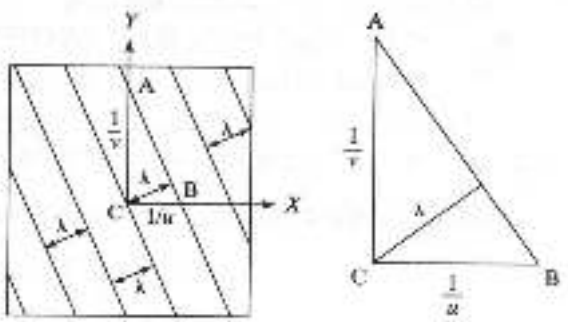


图5-40 正弦波在X轴和Y轴方向的波长$1/u_0$和$1/v_0$与2D波的波长λ之间的关系

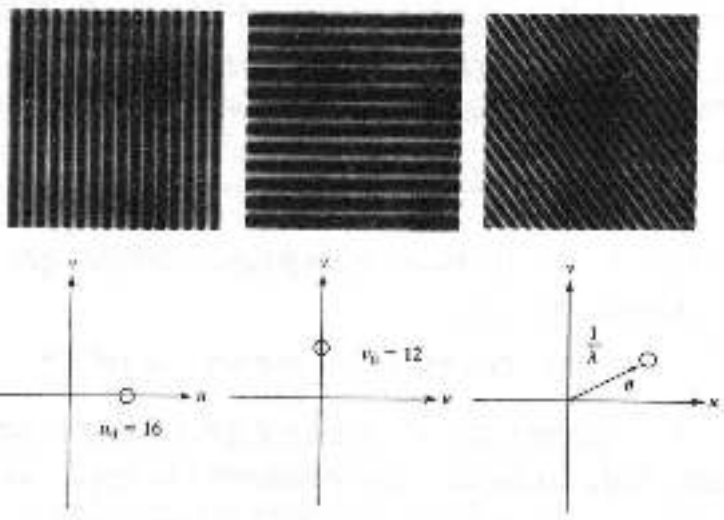


图 5-41

（上行）二个正弦波

（下行）它们在功率谱上的主要响应

5.11.3 离散傅里叶变换

179 数字图像中用到了离散傅里叶变换，或称为DFT，其定义参见公式（5-34）。我们已经知道，关于$N \times N$实值图像集合的基必须有N^2个基向量，每个基由一对频率参数u，v决定，它们

的范围从0到$N-1$，如下面的公式所示。

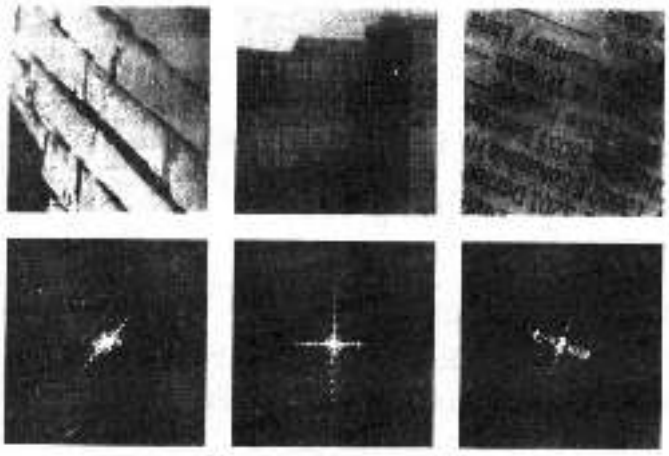


图5-42 三幅图像（上）及其功率谱（下）。砖块纹理的功率谱表明能量分布在多个频率的多个正弦波上，但主要方向是与6个黑缝垂直的方向，与X轴约成45°角。在与X轴成0°角的方向上有明显的能量分布，它们源自较短的垂直缝。建筑物的功率谱说明了在沿X和Y方向上波的高频能量。右边的图，取自一个电话本，与X轴成60°角的方向上分布有高频功率，它们表示文本行的纹理，垂直方向的能量分布得更宽，表示字符以及字符间距（砖块图像来自MIT媒体实验室Vis Tex数据库，Nairobi建筑物的图像由Ida Stockman提供）

定义62 离散傅里叶变换（DFT），将一幅具有$N \times N$个空间采样点的图像$\mathbf{I}[x, y]$变换到频域$N \times N$阵列$\mathbf{F}[u, v]$。

$$F[u, v] = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} I[x, y] e^{\frac{-2\pi j}{N}(ux+vy)} \tag{5-34}$$

为了计算单个频域元素（像素）$\mathbf{F}[u, v]$，只需计算整幅图像$\mathbf{I}[x, y]$和模板$E_{u,v}[x, y]$的点积，一般不是真正地建立频域图像，而是用u、v和所需的cos和sin函数隐含表示出来。同样也定义一个逆变换，将频域的$\mathbf{F}[u, v]$变换成空间图像$\mathbf{I}[x, y]$。虽然可以将变换$\mathbf{F}$显示为2D图像，但我们不认为它真是一幅图像，这样可以减少混淆。下面我们采用正式的术语即频率表示（frequency representation）。

定义63 离散傅里叶逆变换（IDFT），将一个$N \times N$的频率表示$\mathbf{F}[u, v]$变换到$N \times N$的空间图像$\mathbf{I}[x, y]$。

$$I[x, y] = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F[u, v] e^{\frac{2\pi j}{N}(ux+vy)} \tag{5-35}$$

首先把$\mathbf{I}[x, y]$正变换到$\mathbf{F}[x, y]$，我们期望通过逆变换能够得到原始图像。上面给出的一对定义，却不具备这一特性，证明就留作下面的习题。首先重点讨论DFT & IDFT的实际应用。为了存储或者传输图像，将图像变换成频率表示是有用的，通过逆变换可以恢复出输入图像。图像处理中，常常在逆变换之前对频率表示进行一些增强运算。例如，将代表高频波的$\mathbf{F}[u, v]$

中元素减小或置零，就可以减小或去除高频干扰。5.11.6节的卷积定理对这个过程进行了明确的理论解释。

习题5.37 复数的基本性质

利用定义$e^{j\theta}=\cos\theta+j\sin\theta$，（a）证明$(e^{j\theta})^n=\cos(n\theta)+j\sin(n\theta)$。（b）证明$x=e^{\frac{j2\pi}{N}}$是方程$x^N-1=0$在$x=0,1,\cdots,N-1$的解。（c）如果$x_0=1=e^{\frac{j2\cdot 0}{N}},\cdots,x_n=e^{\frac{j2\pi n}{N}}$是方程$x^N-1=0$的N个根，证明$x_1+x_2+x_3+\cdots+x_{N-1}=0$。

180

习题5.38 DFT/IDFT变换的可逆性证明

将公式（5-34）的$F[u,v]$代入到公式（5-35），我们希望得到原来的值$I[x,y]$。考虑下面的求和运算，其中x、y、s、t是[0，$N-1$]内的整型参数，它们和变换定义中的含义相同。

$$G(x,y,s,t)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}e^{\frac{j2\pi}{N}(ux+vy-us-vt)}$$

（a）证明如果$s=x$和$t=y$，那么$G(x,y,s,t)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}1=N^2$。（b）证明如果$s\neq x$或$t\neq y$，那么$G(x,y,s,t)=0$。（c）这是最主要的，即证明对变换后的结果再利用逆变换，能得到原来的图像。

5.11.4 带通滤波器

带通滤波是频域中常用的一种图像运算，如图5-43所示。用DFT将图像变换成它的频率表示，其中有的频率系数减小，可能为0，但有的系数保持不变。低通滤波器的原理图参见图5-43的左边所示，直观上，通过去除高频，然后借助公式（5-35）进行逆变换，将改变了的频率表示变换为平滑后的原始图像。如果不去除频率表示的元素，也可以通过求$F[u,v]$和2D高斯的点积，因为高斯函数对低频成分的加权值较高，对高频成分的加权值较低。图5-43中也显示了如何改变频率表示来实现高通和带通滤波。5.11.6节卷积定理部分，对这些运算进行了更深入的讨论。

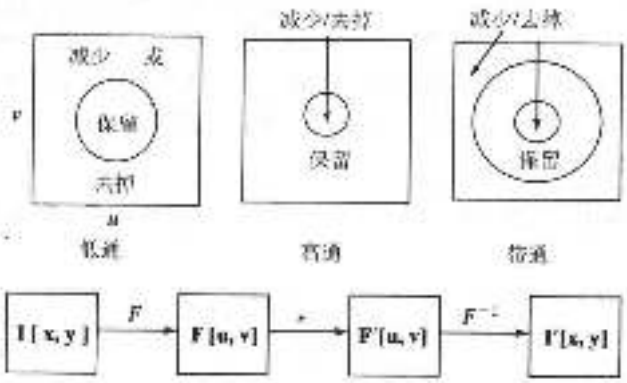


图5-43 带通滤波。先通过傅里叶变换将图像变换到频域（$F[u,v]$），然后再与带通滤波器相乘（×）。乘运算的结果使很多频率系数u、v为0，如上面一行所示。改善后的频率表示经过逆变换得到改善后的空间图像$I'[x,y]$

5.11.5 傅里叶变换讨论

快速傅里叶变换，对于不同的u、v对共享共同的运算，从而节省了计算时间，这种算法通常用于$2^m \times 2^m$的方形图像。尽管傅里叶变换在图像处理中得到普遍应用，但它可能引起图 181
像中的局部特征受到破坏，而这是不希望发生的。傅里叶变换是全局性的变换，每次计算$\mathbf{F}[u, v]$时都利用了所有的图像像素。例如，为了表示头发或草地，或者一些其他细微特征，必须要用到高频波。一方面，这样的高频信息可能被当做噪声滤掉。即使没有滤掉，也将通过高频波与整幅图像的点积来计算高频响应$\mathbf{F}[u_h, v_h]$。由于头发或草地区域只是整幅图像的一小部分，点积结果的高低，取决于其他部分的图像内容。在过去的十年中，明显的趋势是采用小波代替图像波。小波对于局部的图像变化更加敏感，同时也保留了全局性正弦波的一些主要优点。JPEG和其他图像压缩机制，在子图像上采用余弦波表示来减小数据大小。有时为了保持所需的局部图像细节，就要避免这样的压缩机制。

5.11.6 卷积定理*

本节，我们简要介绍卷积定理的证明过程，说明了两个函数在空间域的卷积与它们的频域表示逐点相乘等价。我们已经看到，这种等价关系具有重要的实用价值。

卷积定理:

如果$f(x, y)$和$h(x, y)$是关于空间参数x、y的满足一定条件的函数，那么$\mathbf{F}(f(x, y)*h(x, y)) \equiv \mathbf{F}((f*h)(x, y)) = \mathbf{F}(f(x, y))\mathbf{F}(h(x, y)) = \mathbf{F}(u, v)\mathbf{H}(u, v)$，其中$\mathbf{F}$是傅里叶变换算子，$*$是卷积算子。

在证明1D情况之前，先给出信号处理中常用的步骤（参加算法5.3）。图像$f(x, y)$所有点的卷积运算，也可以不用模板$h(x, y)$。

对常用的滤波器h，变换H可能是封闭的函数表达形式或者是内存中的存储阵列，这样就省略了步骤（2）。信号处理方面的教材一般包含变换对$<h, H>$的表格，不仅用图示方式也用函数形式进行说明，读者可以从中选择具有合适特性的滤波器。现在简要叙述1D卷积定理证明过程，同样步骤可以推广到2D情况。中间步骤的移位定理说明，当函数移位时变换将是怎样的。

算法5.3 借助傅里叶变换通过模板$h(x, y)$对图像$f(x, y)$进行滤波

(1) 对图像$f(x, y)$进行傅里叶变换得到它的频率表示$F(u, v)$;
(2) 对模板$h(x, y)$进行傅里叶变换得到它的频率表示$H(u, v)$;
(3) 对$F(u, v)$和$H(u, v)$逐点相乘得到$F'(u, v)$;
(4) 对$F'(u, v)$应用傅里叶逆变换得到滤波图像$f'(x, y)$。 182

移位定理: $\mathbf{F}(f(x - x_0)) = e^{-j2\pi ux_0}\mathbf{F}(f(x))$

通过定义 $\mathbf{F}(f(x-x_0)) \equiv \int_{-\infty}^{+\infty} f(x-x_0)e^{-j2\pi ux}dx$ ，进行变量替换$x' = x - x_0$，我们得到

$$\begin{aligned}\mathbf{F}(f(x-x_0)) &= \int_{-\infty}^{+\infty} f(x')e^{-j\ 2\pi u(x'+x_0)}dx' \\ &= \int_{-\infty}^{+\infty} e^{-j\ 2\pi ux_0} f(x')e^{-j\ 2\pi ux'}dx' \\ &= e^{-j\ 2\pi ux_0}\mathbf{F}(f(x))\end{aligned} \tag{5-36}$$

其中第一个因子对于变量x'的积分是恒定的。注意到

$$\left|e^{-j\,2\pi ux_0}\right|^2 = \cos^2(2\pi ux_0) + \sin^2(2\pi ux_0) = 1 \tag{5-37}$$

所以函数移位不改变$f(x)$或$\mathbf{F}(u)$的能量。

现在利用上面的结果简单对卷积定理进行证明。

$$\mathbf{F}((f*h)(x)) = \int_{x=-\infty}^{x=+\infty}\left(\int_{t=-\infty}^{t=+\infty} f(t)h(x-t)\mathrm{d}t\right)e^{-j\,2\pi ux}\mathrm{d}x. \tag{5-38}$$

利用对函数f和h所施加的约束条件，允许对积分顺序进行交换。

$$\mathbf{F}((f*h)(x)) = \int_{t=-\infty}^{t=+\infty} f(t)\left(\int_{t=-\infty}^{t=+\infty} h(x-t)e^{-j\,2\pi ux}\mathrm{d}x\right)\mathrm{d}t \tag{5-39}$$

利用移位定理，

$$\int_{x=-\infty}^{x=+\infty} h(x-t)e^{-j\,2\pi ux}\mathrm{d}x = e^{-j\,2\pi ut}\mathbf{H}(u) \tag{5-40}$$

其中$\mathbf{H}(u)$是$h(x)$的傅里叶变换，我们现在有

$$\begin{aligned}\mathbf{F}((f*h)(x)) &= \int_{t=-\infty}^{t=+\infty} f(t)(e^{-j\,2\pi ut}\mathbf{H}(u))\mathrm{d}t \\ &= \mathbf{H}(u)\int_{t=-\infty}^{t=+\infty} f(t)e^{-j\,2\pi ut}\mathrm{d}t \\ &= \mathbf{H}(u)\mathbf{F}(u) = \mathbf{F}(u)\mathbf{H}(u)\end{aligned} \tag{5-41}$$

183 1D卷积定理证毕。

习题5.39

模仿1D移位定理和卷积定理的证明过程，证明2D移位定理和卷积定理。

5.12 总结和讨论

本章内容很多，包含诸多方法和例子，我们来回顾一下主要概念是很重要的。首先讨论了增强图像外观的方法，目的是为了人们更容易理解图像，或者是为了自动处理的需要。有的方法对亮度级别重新映射以增强场景目标的外观，可以看出改善部分图像区域常常以降低其他区域的显示效果为代价。这些方法主要针对灰度图像，但大多数都可扩展到彩色图像，如果使用过图像增强工具的话，就能够知道这一点。我们对边缘增强也进行了讨论，它是人们理解图像的一种手段，希望艺术家的工具箱也因此已经变得更加丰富。

本章最重要的概念是利用模板或核来定义一个局部结构，然后应用于整个图像。卷积和交叉相关是两个非常有用和相关的技术，它们通过将输入图像亮度和对应的模板值逐点求积再相加，就得到在$\mathbf{I}[x, y]$处的处理结果。这些都是在理论和实际中很常见的线性操作。从前面的讨论可以看到，在特殊图像点对特殊模板的响应（相关），可以度量模板结构与图像邻域结构的相似程度。这个思想提供一种设计模板或滤波器的实用方法，可针对不同任务如平滑、边缘检测、角点检测甚至是纹理检测进行设计。

关于边缘检测的文献非常多，本章涉及几种不同的方法。要明白，对于特殊的机器视觉任务来说特殊的边缘检测是非常有用的。而许多开发者的梦想都尚未实现：对于目标边界这样的低层视觉问题产生一个统一的解，其中目标边界用已检测到的边缘描述表示。也许这个梦想是不现实的。毕竟，给出一幅汽车的图像，低层系统如何能知道图像中是否有汽车，是否是我们的兴趣所在，是动是静，或者我们是对检查汽车的表面划痕有兴趣，还是仅仅对识别汽车的品牌感兴趣？本章的边缘图像在很多方面都有用处。事实也是如此，后续章节的许多方法都要以边缘输入为基础。但我们对于边缘图也不应过分乐观，因为我们自己解释本章的图像时，利用了大量关于物体和世界的高层结构和知识。开发出的更高层方法必须具有容错性，因为边缘图像有间断、噪声和多层结构，这些问题使基于边缘的算法面临挑战。 184

5.13 参考文献

在图像处理和计算机视觉研究工作的早期，相关方面的专业期刊还没有出现，因此这方面的研究论文分散在各种不同的期刊上。 Larry Roberts于1965年出版了他的开创性学位论文，是关于3D积木世界中的目标识别问题。其中部分工作是利用算子检测边缘，这个算子现在称为Roberts算子。Roberts肯定想不到在他之后会出现大量的边缘检测工作。其他比较早期的工作，参见Prewitt在1970年和Kirsch在1971年发表的文献。Kirsch模板现在被认为是角检测模板而不是边缘检测模板。近期Shin等人（1998）推崇Canny边缘检测算子（1986），他们证明了Canny算子在性能和效率上都是最佳的，至少在根据运动恢复结构方面是如此。Huertas和Medioni的论文（1986） 从实用的角度深入研究了LOG滤波器的实现问题，并说明如何利用LOG滤波器使边缘的位置精度达到子像素级。边缘检测的工作可以容易地推广到3D立体图像，如Zucker和Hummel所做的工作（1981）。

Kreider等人（1966）的著作给出了线性代数方面的背景知识，主要是线性代数在函数分析中的应用，如我们用到的图像函数，这些和第1、2和7章的内容最相关。 第9章和第10章提供了用傅里叶级数逼进1D信号的背景知识，如果希望将傅里叶分析推广到2D图像函数，也要用到这些背景知识。可以参考Hecht和Zajac（1974）的光学教材，对图像的傅里叶解释是将图像看成一系列光波的叠加。由Cormen等人（1990）所著的算法一书，提供从n个数中选择第i个最小数的两种算法。其中一个算法的理论复杂度是$O(n)$，这使得中值滤波和盒形滤波具有相同的理论复杂度。

1. Canny, J. 1986. A computational approach to edge detection. *IEEE Trans. Pattern Anal. and Machine Intelligence,* 8(6):679–698.
2. Cormen, T., C. Leiserson, and R. Rivest. 1990. *Introduction to Algorithms.* MIT Press, Cambridge, MA.
3. Duda, R., P. Hart, and D. Stark. 2000. *Pattern Classification,* 2nd ed. John Wiley & Sons, New York.
4. Frei, W., and C-C. Chen. 1977. Fast boundary detection: a generalization and new algorithm. *IEEE Trans. Comput.,* C-26(10):988–998.
5. Hecht, E., and A. Zajac. 1974. *Optics.* Addison-Wesley, New York.
6. Huertas, A., and G. Medioni. 1986. Detection of intensity changes with subpixel accuracy using Laplacian-Gaussian masks. *IEEE-T-PAMI,* v. 8(5):651–664.
7. Kirsch, R. 1971. Computer determination of the constituent structure of biological images. *Comput. Biomed. Res.,* v. 4(3):315–328.

8. Kreider, D., R. Kuller, D. Ostberg, and F. Perkins. 1966. *An Introduction to Linear Analysis.* Addison-Wesley, New York.
185
9. Prewitt, J. 1970. Object enhancement and extraction. In *Picture Processing and Psychopictorics,* B. Lipkin and A. Rosenfeld, eds. Academic Press, New York, 75–149.
10. Roberts, L. 1965. Machine perception of three-dimensional solids. In *Optical and Electro-Optical Information Processing,* J. Tippett and others, eds. MIT Press, Cambridge, MA, 159–197.
11. Shin, M., D. Goldgof, and K. Bowyer. 1998. An objective comparison methodology of edge detection algorithms using a structure from motion task. In *Empirical Evaluation Techniques in Computer Vision,* K. Bowyer and P. Philips, eds. IEEE Computer Society Press, Los Alamitos, CA.
12. Zucker, S., and R. Hummel. 1981. A three-dimensional edge operator. *IEEE-T-PAMI,* v. 3:324–331.
186

第6章　颜色与明暗分析

色感对人类来说是非常重要的，色感不仅与光学物理有关，而且依赖于人眼和大脑对外界刺激进行融合处理的复杂过程。人类通过颜色信息辨别物体、材料、食品和地点，甚至一天中的某段时间。图6-1是同一场景但颜色编码不同的两幅图像。尽管两幅图中动物的形状是一样的，但右边的图像与左边的图像差异很大，观察者会把右边的一幅图看成是室内场景中的一只家猫，而不是草坪上的一只老虎。

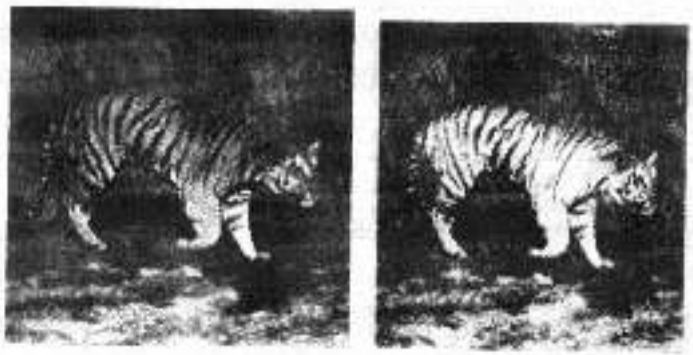


图6-1　（原图经Corel Stock Photos许可）参见彩图6-1

（左图）老虎在草地上的自然色图像

（右图）由于颜色的改变，对老虎的识别变得不太可靠。也许是只站在地毯上的家猫？

随着廉价设备性能的提高，利用机器进行颜色计算变得十分平常。现在已经有了彩色摄像机，彩色显示器和进行彩色图像处理的软件。和人类使用颜色的目的相同，机器也可以使用颜色。颜色信息能带来很多方便，因为它在图像像素上提供多个测度值，常常能够使分类变得更加简单而不需要做复杂的空间决策。

对颜色物理学和色感进行深入的研究需要大量的篇幅，这里我们只提供足够编程用的颜色基本知识，或者只作为阅读文献资料的一个指南。在介绍图像颜色编码的实用方法时，也会附带介绍颜色物理学的一些基本原理。随后给出一些基于颜色的目标识别和图像分割的实例及方法。

目标的明暗也要进行讨论。这个问题不只与目标颜色和光照有关，还与其他许多因素有关。这些因素包括物体表面的粗糙度、表面和光源以及观察者之间的角度、表面离光源及观察者之间的距离等。颜色与明暗效果，几个世纪以来一直是艺术作品的重要组成部分。对于计算机视觉算法中的场景解释来说也是非常重要的。

6.1　颜色物理学

波长λ在400－700nm之间的电磁辐射会刺激人体的感觉神经，从而产生色感（参见图6-2）。1nm等于10^{-9}m，也称作为毫微米。对蓝色光来说，每个波长是400×10^{-9}m，意味着每米长度上会有2.5×10^{6}个蓝波，或者每厘米长度上有25 000个蓝波。真空中的光速是3×10^{8} m/sec，

这相当于每秒0.75×10^{15}个蓝波的频率。这个频率是X射线的千分之一，是无线电波的10亿倍。

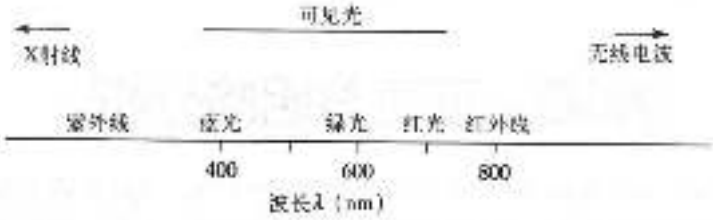


图6-2 电磁波谱中的可见光部分

在本章后面的内容中，我们提到波长或频率，只关心它们所产生的颜色性质。在人类感觉神经能感知的光谱范围之外，机器检测光辐射的能力是很强的。例如，特殊设备可以检测到短紫外波和极短的X射线。另外，许多固态摄像机能够检测到红外长波。无线电接收机会收到波长很长的无线电波。随着科学和工程技术的发展，已经研制出能够对像素进行测量的设备。这些设备能够把像素测度值转换为可见光谱，如X光机和红外（IR）卫星天气扫描仪就是
188 常见的两种设备。

习题6.1

假设一张纸厚0.004英寸。如用蓝光的波长做为单位，纸的厚度是多少？

6.1.1 感测被照射物体

图6-3显示点光源照射到一个物体表面的情况。光源的照射能量与物体表面分子相互作用的结果，使表面发出光能或者辐射出光。一部分能量又反射照射并刺激摄像头内的传感元件或者生物体眼睛内的敏感细胞。对物体颜色的感知或理解一般依赖如下三个因素：

- 不同波长的光能照射到物体表面。
- 物体表面对光的反射，这决定了物体表面怎样将入射光转化为反射光。
- 传感器的光敏特性，传感器接收来自物体表面的反射光照射。

定义64 可见光谱中所有波长的光按大致相同的能量比进行组合而形成**白光**。

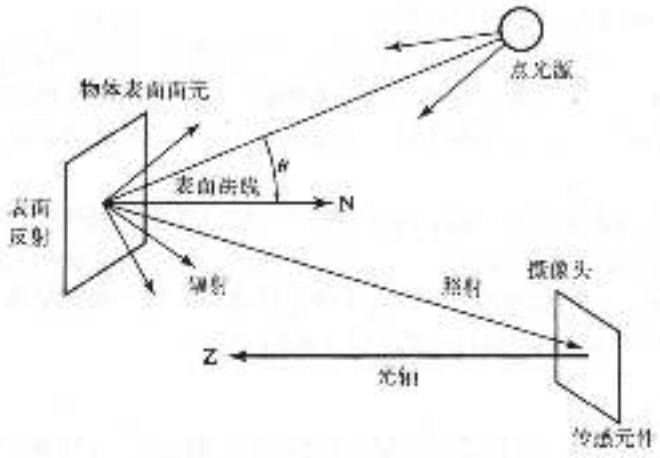


图6-3 光源光能经物体表面反射后照射到传感元件上

一个物体显示蓝色，是因为当白光照射到它的表面时，其表面材料显示蓝色。对于同一

个物体，当只用红色光照射时将显示紫色。一辆蓝色汽车在强烈的阳光（白光）照射下摸起来会觉得很热，汽车辐射出的能量在IR范围内，虽然人眼看不到这种IR能量，但是IR摄像头可以观测到。

189

6.1.2 其他因素

除了上面的三个主要因素外，还有其他几种物理学和人类色感方面的复杂因素。物体表面的镜面反射特性是不同的，也就是说它们像镜子的程度不一样，粗糙表面在所有方向上的反射能量是相同的，接收的能量或者强度与距离有关，离白色点光源距离较远的表面面元比距离较近的表面面元接收的能量要少。其效果与被照射物体和传感元件之间的距离关系类似，因此，相同表面材料的图像其像素强度会由于沿成像光线的距离不同而不同。对于表面反射到传感器的能量来说，表面面元相对光源的方向θ甚至比距离更重要。这些问题在本章末将会进行更详细的讨论。

习题6.2 强度随距离而变化

摄像头垂直于纸面安装，白炽灯从纸的另一面照射，拍摄图像并研究图像的强度，强度变化有多大？对于最明亮的像素点，随着距离的增加强度是不是有规律地减小？

习题6.3 强度随表面法线而变化

用一个排球代替一张纸重复上述实验，拍摄图像并研究图像强度，说明强度的变化情况及其规律。

6.1.3 感受器的敏感性

实际感受器只对一些光波有反应，而且对某些光波比对其他光波更加敏感。图6-4是抽样敏感曲线。三条曲线分别对应人眼的三类不同的锥状体，其中包含对不同光波敏感的不同化学色素。"human$_1$"曲线所对应的锥状体，对400－500nm之间的蓝光略为敏感。"human$_2$"曲线所对应的锥状体，对绿光非常敏感，而对较短的蓝波和较长的红波略微敏感。大脑对局部范围内的几种锥状体的反应进行融合，就产生了可见范围的色感。值得注意的是，虽然光的波长数目有无数个，但只要有三种感受器就可以了。许多其他有眼的动物只有一种或两种光感受器，产生的色感可能不是很丰富。固态传感元件常常在人类色感范围之外有非常好的敏感性。有一点需要牢记的是，有些时候随着天气的变暖，机器视觉系统看到的场景会和操作人员看到的不一样，这主要是由于对IR辐射的敏感性不同所造成的。

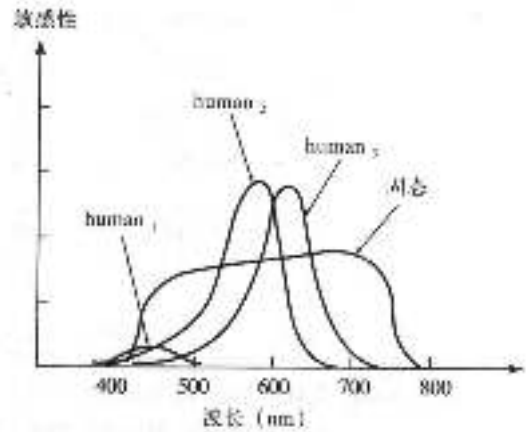


图6-4 人类三种锥状体色素细胞与固态传感器的敏感性比较

190

习题6.4 最喜欢的颜色

你有最喜欢的颜色吗？如果有，那是什么颜色？为什么？再问一下你周围的其余三个人，

他们最喜欢的颜色是什么。假设你得到的是多个答案，怎样解释这种情况？利用已学过的颜色物理学知识。

6.2 RGB三基色

仅仅通过使用三种类型的感受器，人就可以分辨出数千种颜色，具体更精确的数字还存在争议。图形学系统中的三基色RGB（red-green-blue）编码，常常用3字节表示，产生 $(2^8)^3$ 或者大概1600万种不同的颜色编码。为了更精确，我们说是1600万种编码而不是1600万种颜色，这是因为实际上人们并不能感知这么多不同的颜色。机器可以辨别出任何一对位编码不相同的颜色，但是这种编码也许能也许不能表现现实世界的显著差异。在3字节或者24位RGB像素表示中，红、绿、蓝各占一个字节。它们在内存中的存储顺序可以有变化。存储顺序与理论无关，但对编程的影响较大。显示设备的分辨率如果与人眼匹配，则称它使用的是真彩色。这至少需要16位，15位的编码系统可能是R、B、G各占5位，而16位的编码系统中绿色占6位，这样能更好的表示绿色，因为人们对绿色的敏感程度相对较大。

可见光谱中任意颜色的编码可以通过对三基色（RGB）进行组合得到，如图6-5所示。红色（255, 0, 0）和绿色（0, 255, 0）等量混合就会得到黄色（255, 255, 0）。与一种基色对应的数值表示该基色的强度。如果每种基色的强度都是最大值，那么结果就会产生白色。等比例的低强度三基色产生的颜色从灰色（c, c, c）一直到黑色（0, 0, 0），其中c为0到255的任意常数。在我们的算法中确定颜色值时，利用0到1范围的数值要比0到255更加方便，颜色值的取值范围是与设备无关的。

191

	RGB	CMY	HSI
红	(255, 0, 0)	(0,255,255)	(0.0 , 1.0, 255)
黄	(255,255, 0)	(0, 0,255)	(1.05, 1.0, 255)
	(100,100, 50)	(155,155,205)	(1.05, 0.5, 100)
绿	(0,255, 0)	(255, 0,255)	(2.09, 1.0, 255)
蓝	(0, 0,255)	(255,255, 0)	(4.19, 1.0, 255)
白	(255,255,255)	(0, 0, 0)	(-1.0, 0.0, 255)
灰	(192,192,192)	(63, 63, 63)	(-1.0, 0.0, 192)
	(127,127,127)	(128,128,128)	(-1.0, 0.0, 127)
	(63, 63, 63)	(192,192,192)	(-1.0, 0.0, 63)
	...		
黑	(0, 0, 0)	(255,255,255)	(-1.0, 0.0, 0)

图6-5 几种不同的三基色颜色编码系统。在算法中确定颜色值时，利用0到1范围的数值更加方便。HSI值是利用算法6.1由RGB变换得来的，其中$H\in[0.0, 2\pi]$，$S\in[0.0, 1.0]$，$I\in[0, 255]$，H和S采用字节编码

RGB系统是一个*加色系统*（additive color system），因为是向黑色（0, 0, 0）中加入不同成分形成新的颜色。这与RGB显示器（监视器）有着很好的对应。RGB显示器中有三种荧光粉能够发射出光线，三个相邻的荧光点构成一个像素，这些荧光点受到三束强度分别为c_1、c_2、c_3的电子束的轰击。人眼对三种荧光进行综合产生出颜色（c_1, c_2, c_3）的感觉。来自CRT屏幕上小片区域的三条光波，在物理上被叠加或者混合到一起。

假设颜色传感器把数字图像上的一个像素编码成（R，G，B），其中每个坐标的取值范围是[0，255]。公式（6-1）是一种对图像数据进行规范化处理的方法。这样做可以为计算机程序和人的判读带来方便，同时也方便进行颜色系统的转换，这一点将在后面讨论。想像一台彩色摄像机在光照发生变换的场景下工作。例如，物体表面上的点离光源的距离是不一样的，甚至对于某些光源来说有的点位于阴影之中。如聚集小汽车图像中的绿色像素，如果不先进行强度规范化处理，算法的结果将非常糟糕。

$$
\begin{aligned}
&\text{强度规范化} \quad I = (R+G+B)/3 \\
&\text{红色规范化} \quad r = R/(R+G+B) \\
&\text{绿色规范化} \quad g = G/(R+G+B) \\
&\text{蓝色规范化} \quad b = B/(R+G+B)
\end{aligned}
\tag{6-1}
$$

利用公式（6-1）的计算方法，规范化后的RGB值的和始终为1。还有其他的规范化方法，例如我们可以用（R，G，B）中的最大值做除数而不是用RGB的平均值。由于$r+g+b=1$，颜色坐标之间的关系就能够通过2D图方便地绘出，如图6-6所示。纯颜色值用三角形的三定点表示。例如消防红色在右下角（1，0）附近，草绿色位于上面（0，1）处，而白色位于中心（1/3，1/3）。在图6-6中，蓝轴b与r轴和g轴垂直，方向由纸面向外，这样三角形实际上是通过点[1，0，0]，[0，1，0]和[0，0，1]的三维坐标系中的一个薄面。对于三角形内部不同的$r-g$取值，蓝色值可以通过$b=1-r-g$算出。

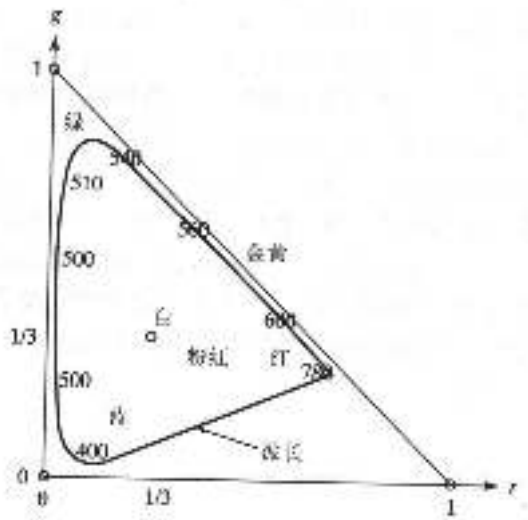


图6-6 规范化RGB坐标系的颜色三角形。蓝轴b与r轴和g轴垂直，方向由纸面向外。这样三角形实际上是通过点[1，0，0]、[0，1，0]和[0，0，1]的三维坐标系中的一个薄面。对于三角形内部不同的$r-g$取值，蓝色值可以通过$b=1-r-g$算出

习题6.5 颜色编码实验

得到一幅RGB彩色图像，并利用图像工具认真观察。把绿色和蓝色的编码字段进行交换，对结果进行分析说明。把所有蓝色值加倍，对结果进行分析说明。

6.3 其他基色系统

其他一些基色系统，有的适用于产生彩色的设备，有的符合人类色感。有的基色只是其他基色线性变换的结果，有的不是。

6.3.1 CMY减色系统

CMY减色系统是在白纸上印刷的模型，它是从白色值上减去某个数值，而不是像RGB系统那样向黑色值上加上某个数值。在图6-5中，CMY编码位于RGB编码的右边。CMY是"Cyan-Magenta-Yellow"的缩写，这是CMY系统的三基色，对应三种墨水。青色吸收红光照射，品红色吸收绿光，黄色吸收蓝光，因此当印好的图像被白光照射时会产生合适的反射。该系统被称为减色系统，因为是为了吸收而编码。部分颜色的编码情况为：白色编码（0，0，0），因为白色光不会被吸收；黑色编码（255，255，255），因为白光的所有成分都会被吸

收；黄色编码（0，0，255），因为入射白光中的蓝色成分容易被墨水吸收，从而留下了红色和绿色的成分，就产生了黄色的感觉。

6.3.2 HSI系统

HSI（色调,饱和度,强度）系统对颜色信息进行编码，从两个色度（chromaticity）编码值中分离出总强度I，这两个色度是色调H和饱和度S。图6-7中的颜色立方体与图6-6中的RGB三角形有关。在立方体表示中，每个r、g、b值可以独立在[0.0, 1.0]范围内编码。如果沿主对角线对立方体进行投影，就得到图6-8中左边的六边形。在这个表示方法中，原来沿着颜色立方体对角线的灰色现在都投影到中心白色点，而红色点[1，0，0]现在则位于右边的角上，绿色点[0，1，0]位于六边形的左上角。图6-8的右边是称为六棱锥（hexacone）的3D颜色表示法。三维表示法允许把前面立方体的对角线看成是一条竖直的强度轴**I**。定义色调**H**的角度范围是离红色轴0到2π之间，其中纯红色的角度为0，纯绿色的角度为$2\pi/3$,纯蓝色的角度为$4\pi/3$。为了在这个颜色空间中完全确定一个点，饱和度**S**是第三个坐标值。饱和度是颜色纯度或者色调的模型，用1来表示完全纯净或完全饱和色；用0表示完全不饱和色调，也就是说有一些灰色成分。

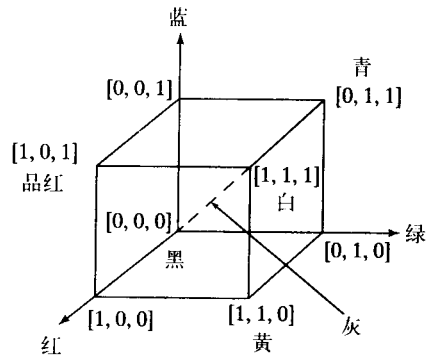
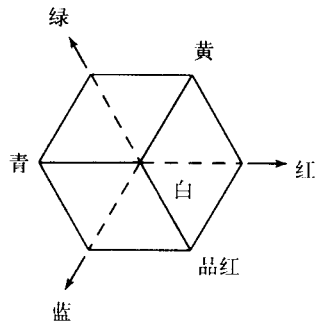


图6-7 规范化RGB坐标系的颜色立方体。图6-6中的三角形是对过点[1,0,0]、[0,1,0]与[0,0,1]的平面进行投影的结果



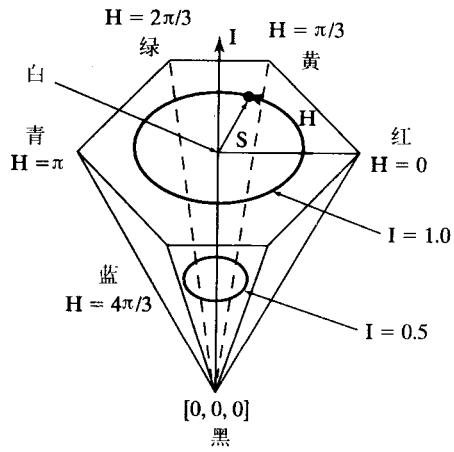


图6-8 HSI颜色六棱锥表示。左边是RGB立方体的投影，与过点[0, 0, 0]、[1, 1, 1]的对角线垂直,颜色名字标在六边形的顶点处。右边是一个六棱锥，表示HSI颜色坐标系，强度（I）是垂直轴；色调（H）是从0到2π的角度，红色位于0.0；饱和度（S）的范围是0到1，其值与纯度或者不同于白色的程度有关，S＝0.0的颜色都对应在I轴上

HSI系统有时候也会被称为HSV系统，在HSV系统中用值（Value）代替强度（Intensity）。对于图形学设计人员，HSI系统更为方便一些，因为它提供了对亮度和色调的直接控制。彩色蜡笔被放在中间且靠近I轴的地方，而深色和浓色则在六棱锥的外围。HSI也可以对计算机视

觉算法提供更好的支持，因为它可以对照明进行规范化处理，还可以聚焦在两个色度参数上，这两个色度参数与该物体表面的固有特性密切相关，而不是与照射光源密切相关。 194

在算法6-1中，给出了从RGB坐标系到HSI坐标系的推导过程。这个算法可以对输入值（r，g，b）进行转化，这些输入值来自3D颜色立方体，或者经公式（6-1）规范化处理过，甚至是图6-5左列的RGB字节编码值。强度I的输出值范围与输入值的取值范围相同。当强度$I=0$时，饱和度S并没有定义；当$S=0$时色调H也没有定义。H的范围是[0，2π]。而为了确定数学变换公式，要用到平方根和反余弦运算。算法6.1使用很简单的运算方法，因此即使把一整幅图上的所有像素从一种编码转化成另一种编码时，算法运行起来也是非常快的。图6-5的右边给出了算法6.1的输出结果。

算法6.1 RGB编码到HSI编码的转换

```
R, G, B： RGB的输入值，范围全部是[0,1]或者是[0,255];
I: 与输入范围相同的强度输出值;
S: 饱和度输出值，范围[0, 1];
H: 色调输出值，范围[0, 2π], 如果S＝0则值为－1;
R, G, B, H, S, I都是浮点数;
procedure RGB_to_HSI(in R,G,B; out H,S,I)
{
I := max ( R, G, B );
min := min ( R, G, B );
if (I ⩾ 0.0) then S := (I − min )/I else S := 0.0;
if (S ⩽ 0.0) then { H := −1.0; return; }
\\根据RGB成分的相对大小计算色调。
diff := I − min;
\\是红轴±60度内的点吗?
if (R = I) then H := (π/3)*(G − B)/diff;
\\是绿轴±60度内的点吗?
else if (G = I) then H := (2*π/3) + π/3 *(B − R)/diff;
\\是蓝轴 ± 60度内的点吗?
else if (B = I) then H := (4*π/3) + π/3*(R − G)/diff;
if (H ⩽ 0.0) H := H + 2π;
}
```

习题6.6

利用算法6.1，（a）把RGB码（100,150,200）转化成HSI码，（b）把RGB码（0.0, 1.0, 0.0）转化成HSI码。

参考前面的图6-6，看看HSI的值与颜色三角形有什么关系。色调与光的主波长有关，并且近似对应图6-6中三角形边上一点，λ的较低值在400nm附近，起始于原点，沿g轴上升到大约520nm，然后沿直角三角形的斜边向下进一步增加到800nm。色调与白色中心到三角形边上某点（r, g）的角度对应。在图6-6中，50%饱和金色的H和S值位于白色点与金色点中间。图

6-6是对画家彩色调色板的一个近似。

图6-9通过改变饱和度对一幅图像进行变换。原始输入图像在左边，中间图像是对所有像素的饱和度S增加40%后的效果，右边图像是对S降低20%后的效果。比较我们的实验，右边图像的颜色看起来像洗过一样，而中间图像的颜色又显得调整过分。有一点要注意到，即使机器视觉系统工作在强度变化的白光下，在三幅图像中色调H是不变的，所以色调应是颜色分割的一个可靠特征。



图6-9 （Frank Biocca提供）参见彩图6-9

（左图）输入的RGB图像
（中图）饱和度S增加40%
（右图）饱和度S降低20%

习题6.7

设计算法，利用如下解析几何的方法，把[0, 1]范围内的r、g、b颜色坐标转换成H、S、I坐标。从点[r, g, b]向颜色立方体过[0, 0, 0]和[1, 1, 1]的对角线引垂线，计算相应的H、S、I值。

195, 196

6.3.3 电视信号的YIQ与YUV系统

美国国家电视标准委员会（NTSC）电视标准采用的编码体制是一个亮度参数Y和二个色度参数I与Q。在黑白电视中只用亮度参数，而在彩色电视中二个参数都要用到。从RGB到YIQ的近似线性变换由公式（6-2）给出，实际上，对Y的编码比对I与Q的编码用到的位数更多，因为人类视觉系统对亮度（强度）要比对色度更加敏感。

$$\begin{aligned} \text{亮度} \quad & Y = 0.30R + 0.59G + 0.11B \\ \text{红-青} \quad & I = 0.60R - 0.28G - 0.32B \\ \text{品红-绿} \quad & Q = 0.21R - 0.52G + 0.31B \end{aligned} \tag{6-2}$$

YUV编码用于一些数字视频产品，以及压缩算法如JPEG和MPEG中。RGB到YUV的转换公式如下：

$$\begin{aligned} Y &= 0.30R + 0.59G + 0.11B \\ U &= 0.493 * (B - Y) \\ V &= 0.877 * (R - Y) \end{aligned} \tag{6-3}$$

对于数字图像与视频压缩来说，采用YIQ和YUV比采用其他颜色编码系统更加合适，因为亮度与色度可以用不同的位数进行编码，这在RGB系统中是不可能的。

习题6.8 颜色编码转换

对于彩色摄像机，假设一像素的RGB编码值是（200, 50, 100），其中255是最高值（能量最大）。（a）HSI系统中等价的三个值是什么？（b）YIQ系统中等价的三个值是什么？

习题6.9

从RGB到YIQ的变换可逆吗？如果可逆，计算出逆变换。 [197]

习题6.10 图像重编码

假设你有一台显示器和观看RGB图像的软件，做如下的实验。首先建立HSI图像，使得右上1/4是饱和红色，左下角是饱和黄色，左上角是50%饱和蓝色，右下角是50%饱和绿色。利用算法6.1把RGB图像转换成HSI图像，并把HSI图像转换成RGB图像。把图像显示出来，并研究4个1/4图像区的颜色。

6.3.4 基于颜色的分类

在许多应用中，像素颜色包含很多与分类有关的信息。在6.5节介绍的人类皮肤颜色模型，长期用于从彩色图像中寻找人脸。但是这个过程有时也会出错。例如一个棕色纸板盒。其图像像素就能够通过皮肤颜色的测试，也许需要用区域形状特征把多面体纸板盒的表面与椭圆形状的人脸区分开。在图6-10中，通过保留与训练样本像素接近的像素，从图像中抽取出白色区域。样本像素从标记符号中得到。出现的几个不希望生成的区域，主要是由其他白色物体及镜面反射所产生。特征识别算法可以对很多特征进行识别，并去掉多数不想要的成分。

总的说来，对一个单独像素的颜色进行解读容易出错。图6-9左边的原幅图像是用带闪光灯的摄像机拍摄的，内含一些白色小块，是由于镜面反射造成的（在6.6.3节讲解）。如果对黄色的定义范围扩大，分类器就会把这些白色小块放到黄色成分之中，蓝色杯子镜面反射的像素也有可能包括到黄色之中。在颜色空间的特殊区域，颜色解读会出现问题。当饱和度接近0时，色调的计算和解读就不可靠；当亮度较低时，饱和度的解读也不可靠。 [198]



图6-10 从左边的彩色图像中分割出白色像素。白色像素的单个连通成分用第3章的颜色算法任意标记。（David Moore提供分析）参见彩图6-10

习题6.11

做下面的实验，证明当饱和度或者亮度接近0时，从RGB到HSI的转换是不稳定的。通过编程实现算法6.1。（a）对于大L及$\Delta L_i \in \{-2, -1, 1, 2\}$，把RGB码$(L + \Delta L_R, L + \Delta L_G, L + \Delta L_B)$转换成HSI码。$H$的值一样吗？（b）对于小$L$(10左右)，$\Delta$同上，重复该实验。$S$的值一样吗？

6.4 颜色直方图

在图像检索或者目标识别中，可用颜色直方图表示一幅彩色图像。直方图统计每种像素

的数目，每个像素只需访问一次，并在直方图的合适箱格上添加一个增量，就能够快速生成直方图。利用颜色直方图在图像数据库中进行图像检索的内容将在第8章讨论。颜色直方图对于平移、绕成像轴的旋转、小的离轴旋转，尺度变化和部分遮挡等是相对不变的。这里我们简单介绍颜色直方图的方法，最初的颜色直方图匹配算法是由Swain和Ballard于1991年提出的，主要用在目标识别中。

用直方图来表示彩色图像的一种简单方法是，把每个RGB颜色码中的最高两位连起来。直方图将有$2^6 = 64$个箱格。分别计算三种颜色直方图也是可以的，一种颜色对应一个直方图，再把它们组合成总的直方图。例如，把三个独立的RGB直方图量化成16级，将总共产生$k = 48$个箱格的直方图，就像Jain和Vailaya（1996）所用的直方图那样。图6-11是两幅彩色图以及根据他们的方法所生成的直方图。

图像直方图$h(I)$和模型直方图$h(M)$的交叉值定义为，对于K个对应箱格，将所有$h(I)$与$h(M)$中的较小者相加，如公式（6-4）所示。将交叉值除以模型的像素数进行规范化处理，就得到匹配值。这个匹配值是图像中含有多少模型中的颜色的一种测度，它不会因为图像中的背景像素在模型中不存在而减小。也可定义其他类似的测度，例如，可以用箱格中的数值除以像素总数从而把直方图规范化成频数，然后利用欧几里德（Euclidean）距离对两幅图像进行比较。

$$\text{intersection}(h(I), h(M)) = \sum_{j=1}^{K} \min[h(I)[j], h(M)[j]]$$

$$\text{match}(h(I), h(M)) = \frac{\sum_{j=1}^{K} \min[h(I)[j], h(M)[j]]}{\sum_{j=1}^{K} h(M)[j]} \tag{6-4}$$

199

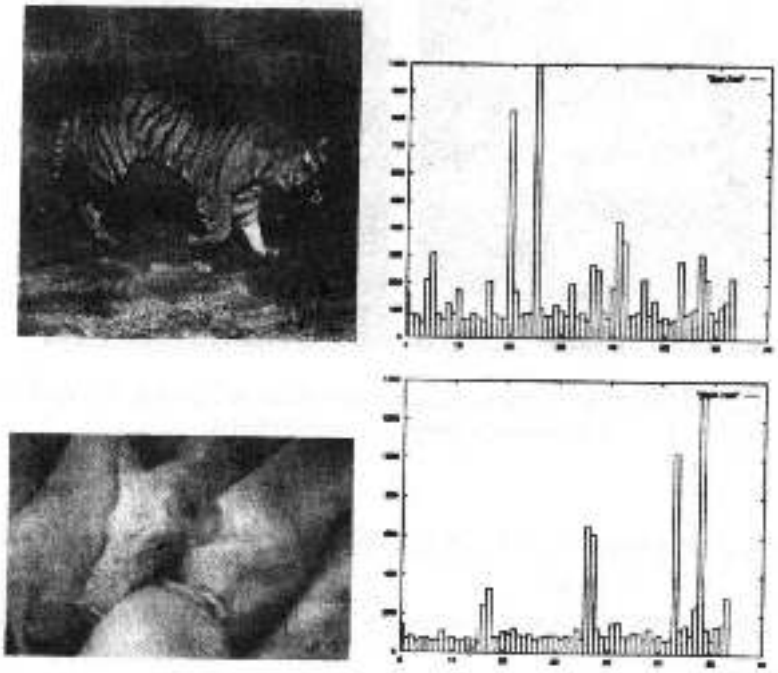


图6-11 彩色图像及其64箱格的直方图。（直方图由A. Vailaya提供，图片经Corel Stock Photos许可）参见彩图6-11

实验表明，在上面提到的几种变化情况下以及采用不同的空间量化方法，直方图匹配值能够很好地表示图像的类似程度。Swain 和Ballard 也提出反投影（backprojection）算法，可以确定与模型中目标大小近似的区域在图像中的位置，该图像与模型直方图是最佳匹配的。这样，他们提出两种基于颜色的算法，一种是为了识别，这时图像中包含已知目标，另一种是决定的目标位置。如果图像是在不同光照条件下得到的，那么首先就应该去掉强度影响。人们也应该考虑到对直方图进行平滑处理，使得反射光potentially有小的移动时仍能得到较好的匹配。另一种方法是匹配累积分布而不是频数本身。

习题6.12 匹配像素互换的图像

已知图像A，我们把A中的像素位置进行随机交换产生图像B。（可以像下面这样做：首先把A拷贝到B，然后对于B中的像素I[r, c]，随机选择像素I[x, y]，把I[r, c]与I[x, y]进行位置交换。）对A和B的直方图进行匹配，会产生怎样的匹配结果？ 200

习题6.13 商品识别

拿出3支香蕉、3个橘子、3个红色苹果、3个绿色苹果、3个绿色辣椒和3个红色西红柿。对于这6类物品，分别拍出三幅图像，每次都把3个物品的位置做一下调整，这样就得到18幅图像。对每幅图像求彩色直方图。用每类集合中的第一幅直方图作为模型（总共有6个模型），然后计算每个模型与其他12幅直方图之间的匹配值。对结果进行分析。根据你的实验结果，说明超市商品识别系统能够识别放在收银台天平上的物品吗？

6.5 颜色分割

现在我们讨论从彩色图像中寻找人脸，彩色图像由工作站前的摄像头拍取。做这个工作的最终目标是为了实现更好的人机交互。设计的算法能够找到与人脸对应的主要区域。首先是训练阶段，用不同的人脸样本确定人脸像素的本质特征；其次，根据新图像中人类像素的(r, g)值落入训练数据的位置，来识别这些人脸像素。图6-12绘出了包含不同人脸图像的像素(r, g)，用的是经公式（6-1）规范化后的红、绿值。通过第4章介绍的方法确定边界，很容易定出6类像素。其中三幅图中包含人脸，一个主要类和两个由阴影及胡须产生的像素类。 201

识别人脸区域主要分三个步骤。第1步的输入是用1、2、3…、7做标记的标记图像，这是根据训练数据进行分类的结果（标记7用于表示不属于其他6类中任何一类的像素）。图6-13的中间是两幅不同人脸的标记图像，多数属于背景的像素做了正确标记，同样多数属于人脸的像素也做了正确标记。但是有许多错误标记的小区域。然后根据各部分相对主要人脸区域的大小和位置，对它们进行整合，加入到主要区域或者把它们删除。首先进行连通成分处理，就像第3章讲的那样，把标记4、5或6的像素做为前景像素。第2步选择最大的合适的成分做为人脸目标。这一步根据处理的许多例子利用启发式学习，丢弃太小或太大的成分。一般留下不到100个成分，而把多数成分划归到阴影类中。第3步去掉剩下的成分或把它们合并到选中的人脸目标之中。应用基于人脸知识的几种启发式方法，同时假设景物中只有一张人脸。例子结果显示在图6-13的右侧。程序执行速度很快，足以应付这些计算，大约是实时每秒30次。包括计算眼睛和鼻子的位置，这些在书中还没有涉及到。这个例子推广出其他许多问题。一个关键的阶段是原始图像中几千种颜色码的聚类运算，目的是为了得到只有少量标记的标记图像。在人脸抽取的例子中，聚类通过手工实现，但有时要进行自动聚类。分割问题将在第10章进行详细介绍。 202

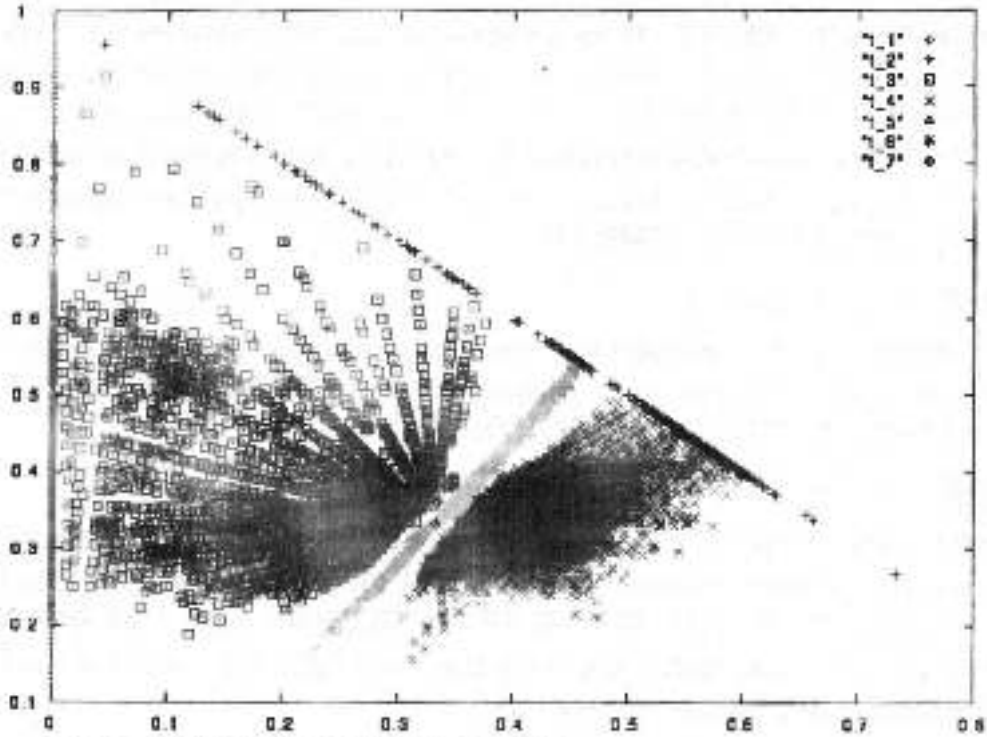


图6-12 通过训练得到的皮肤颜色类别。水平轴是R_{norm}，垂直轴是G_{norm}。1_4类是主要的人脸颜色，1_5和1_6是次要的人脸类，它们与人脸上的阴影和胡须区域有关。（V. Bakic提供）参见彩图6-12



图6-13 人脸抽取实例。（图像由V. Bakic提供）参见彩图6-13

（左图）输入图像
（中图）标记图像
（右图）抽取的人脸区域的边界

6.6 明暗分析

在光物理学和人类感知方面，存在几种因素使问题变得复杂化。各种表面的镜面反射特性是不同的，也就是说，它们像一面镜子的程度不一样。理想的镜面反射，把入射能反射到

沿反射线的受限锥体内。理想散射表面在各个方向上的反射能是相同的。因此一个表面在反射入射光方面不仅与光的波长有关，还与方向有关。此外，表面接收辐射的能量或强度还与距离有关，离白色点光源较远的表面面元要比距离较近的表面面元接收的能量小。其效果与被照射物体与传感器元素之间的距离关系类似。因此，图像强度将由于沿成像光线的距离不同而不同。表面面元相对光源的方向θ也很重要。

6.6.1 来自单一光源的照射

远处单一光源照射到目标表面的情况如图6-14所示。通常，我们无法找到可以观察表面的视点位置，所以只考虑表面如何被光源照射。假设光源离得很远，从被照射物体表面的所有面元到光源的方向，可以用一个单位长度的方向向量**s**来表示。到达表面面元A_j的单位面积的光能（强度i），与表面面元的面积以及表面面元与照明方向**s**之间夹角的余弦之积成正比。夹角的余弦为$\mathbf{n}\circ\mathbf{s}$，其中**n**是表面面元$A_j$处的单位法线向量。这样表面面元接收的入射光强度的数学模型为:

$$\text{入射强度 } i \sim \mathbf{n}\circ\mathbf{s} \tag{6-5}$$

203

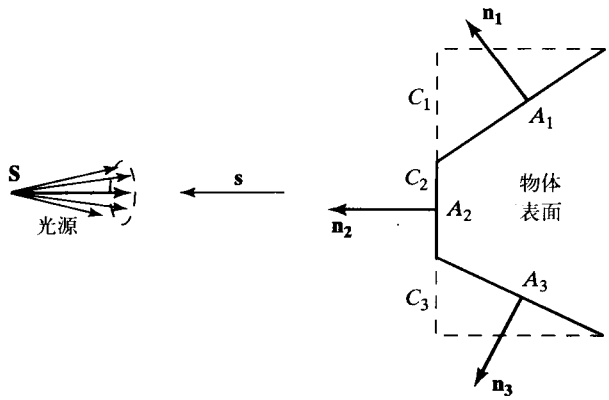


图6-14 物体表面面元A_j受到光源**S**的照射，接收的能量在垂直于光源方向上的投影$C_j = A_j\cos\theta_j$成正比。接收的照射强度就是$i\sim\mathbf{n}\circ\mathbf{s}$，其中**n**是表面的单位法线向量，**s**是指向光源的单位方向，θ_j是表面法线向量$\mathbf{n}_j$与**s**之间的夹角

表面接收的照射能量直接与光源的功率成正比，光源的功率可能知道也可能不知道。也许光源向各个方向发射能量，或者像聚光灯那样只向一个锥形区域发光。两种情况下光源的功率都用每球面度的瓦特数表示，或者说是以光源为中心的单位球体锥形角的单位面积所发出的能量。这个简单的表面面元辐照模型可以很容易扩展到曲面情况，只要考虑矩形表面面元达到无穷小程度。表面面元对入射光的反射部分称为表面面元的*反射率*（albedo）。

定义65 表面面元的**反射率**是指反射的总照度与接收的总照度之比。

我们已经假设反射率是表面的固有属性，对一些表面来说情况不是这样，因为反射亮度的一部分将随着光照与表面法线的相对方向不同而不同。

6.6.2 漫反射

现在对上述模型进行扩展，考虑来自物体表面的反射，此外建立表面面元对应视点位置V的外观模型。图6-15 显示的是*漫反射或朗伯反射*（diffuse, Lambertian reflection）。在以表面面元为中心的半球体所有方向上对到达表面面元的光能进行平均反射。表面漫反射与光的波

长具有一定的关系。反射光强度与入射光强度成正比，常量系数是表面的反射率，深色表面的反射率较小，光亮表面的反射率较大。

$$\text{漫反射强度} I \sim \mathbf{n}_f \circ \mathbf{s} \tag{6-6}$$

定义66 **漫反射**表面在所有方向上均匀地反射光线，结果从所有视点看它都有一样的亮度。

重要的特征就是，当从半球体所有方向上观察，表面面元具有同样的亮度，因为它的亮度与观察者的位置无关。参考图6-15，无论是从位置V_1还是位置V_2观察，表面面元A_1将有同样的亮度。同样地，无论是从位置V_1还是位置V_2观察，表面面元A_2也将有同样的亮度。如果这三个表面面元由相同的材料构成，它们有同样的反射率，那么A_2将看起来比A_1更亮一些，而A_1将比A_3更亮一些，因为这些表面与照明方向所成的角度不同。无论是从位置V_1还是位置V_2观察，表面面元A_3将根本看不到。（如果$\mathbf{n} \circ \mathbf{v} < 0$，$\mathbf{v}$是到观察者的方向，则看不到表面面元$A_3$。）

204

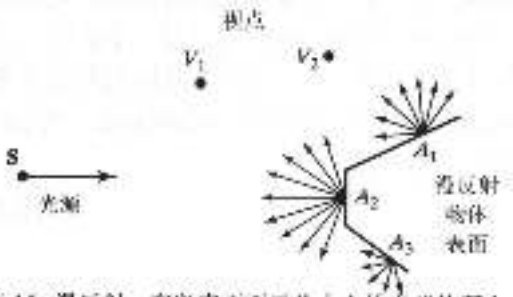


图6-15 漫反射。在以表面面元为中心的半球体所有方向上均匀分布反射能，这样对于表面可见的所有视点，整个平面将表现出均匀的亮度

图6-16给出了一个漫反射的例子，显示了鸡蛋和黑陶花瓶的反射光强度。图像中一行像素的强度分布很像一条余弦曲线，表明物体表面的形状与反射光密切相关，正如公式（6-6）所表达的那样。

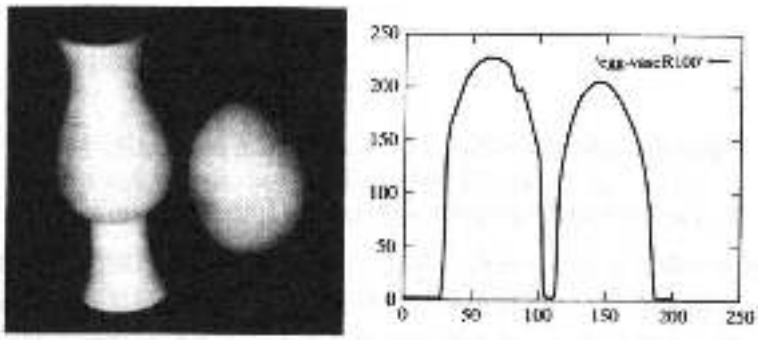


图6-16 朗伯物体的漫反射，一个花瓶和鸡蛋，以及通过白点的一行像素的强度分布图。强度与物体的形状密切相关（Deborah Trytten许可）

习题6.14

考虑漫反射材料构成的多面体，让表面F正对远处的光源S，使另一表面A与F相邻，看起来只有F的一半亮。表面A与表面F之间的法线夹角是多少？

6.6.3 镜面反射

很多光滑表面的行为很像一面镜子，把大部分入射光沿反射线反射出去，如图6-17所示。反射线（R）与表面的法线（N）和入射线（S）在同一个平面上，并且入射角等于反射角。理想镜面将沿方向R把从光源S接收的光能全部反射出去。此外，反射能量与入射光具有相同

的波长构成。而与目标表面的实际颜色无关。因此红苹果在它反射白色光源的地方将会具有白色亮区或者闪光。公式（6-7）是计算机图形学中常用的镜面反射数学模型。公式（6-8）定义了怎样根据表面法线和光源方向计算反射光线**R**。参数α称为表面反光参数，α的值为100，对于很亮的表面α的值更大一些，注意随着α的增加，当ϕ逐渐远离0时，$\cos\phi^{\alpha}$下降很快。 205

$$\text{镜面反射强度 } i \sim (\mathbf{R} \circ \mathbf{V})^{\alpha} \tag{6-7}$$

$$\mathbf{R} = 2\mathbf{N}(\mathbf{N} \circ (-\mathbf{S})) \oplus \mathbf{S} \tag{6-8}$$

定义67 **镜面反射**，象镜子一样的反射。表面反射的光能在绕反射线的紧锥体内反射出去。此外，反射光的波长构成与光源类似，与表面颜色无关。

定义68 **物体上的高亮区**，是对光源进行镜面反射所造成的亮点区。高亮区预示着物体的材质是蜡、金属或玻璃等。

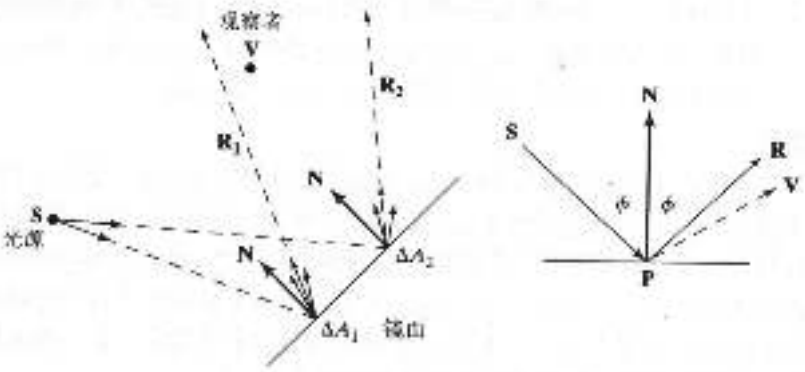


图6-17 镜面反射或者类似镜面的反射，反射能分布于绕反射线**R**的窄锥体内。视点**V**接收来自表面面元ΔA_1的一些反射能，而很少接收来自表面面元ΔA_2的反射能。在**V**处接收的强度是$\cos^{\alpha}(\mathbf{R},\mathbf{V})$，其中**R**是反射线，**V**是从表面面元到视点的方向，α是反光参数

6.6.4 随距离增大而变暗

光能到达表面的强度随表面离光源的距离变大而减小。当然，地球比水星接收太阳的强烈照射要小。这种现象的模型见图6-18。假设光源单位时间内发出恒值的能流，包含光源的任何球面一定在单位时间内拦截同样多的能量。因为球的表面积与半径的平方成正比，单位面积的能量一定与半径的平方成反比。这样物体表面接收的入射光强度将随到光源距离的平方而下降。在图6-18中把这个距离记为d_1。同样的模型应用到物体表面面元的反射光能上，空间V处的观察者将观察到表面亮度与观察者到表面面元的距离d_2的平方成反比。这种反比平方模型一般用在计算机图形学中，用来计算要绘制表面的明暗变化，使得用户能够感觉到3D距离或深度。 206

习题6-15

一个发明家想把下面的设备卖给交警，以检测夜间的车速。这个设备在t_1和t_2时刻发射很短的闪光，并用传感器测量来自汽车的反射信息，根据反射强度，用图6-18的原理计算两时

刻之间产生的距离d_1和d_2。汽车的速度简单地按这段时间内的距离变化来计算。对这种仪器的设计思想进行评价，它能工作吗？

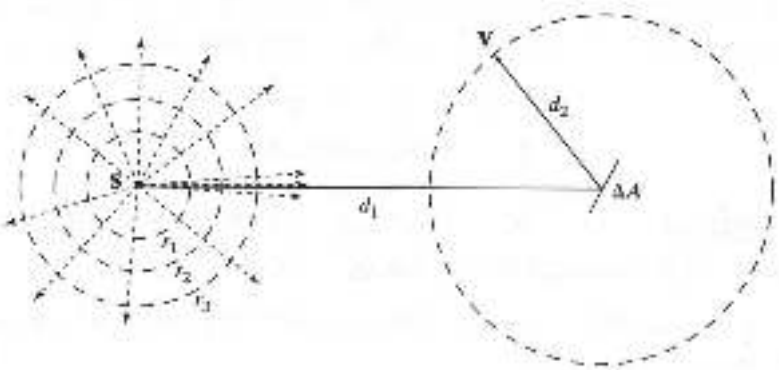


图6-18 点光源通过任意封闭的球面发出的总能量是一样的，因此表面上每单位面积的能量，或者说是强度一定与球面半径（d_1）的平方成反比。同样，表面元反射的光能强度一定随观察者到表面距离（d_2）的增大而减小

6.6.5 复杂因素

对于大多数表面，理想的反射模型应同时包含漫反射和镜面反射。如果我们用闪光灯照射苹果并进行观察，实际上我们看到的是有微白高亮区的微红色物体。微红色反射来自漫反射，而高亮区来自镜面反射。如果全部苹果都是镜面，那么我们就不能看到苹果的大部分表面。

通常有许多光源照射一个场景，而且有更多的表面面元对这些光源的光进行反射。我们除了能说出场景中存在环境光之外，也许不能说出所有的能量交换。在计算机图形学中，当
207 对一个表面进行明暗处理时，利用环境光是常见的方式。

定义69 环境光，是由多个光源产生的，经许多表面交叉反射后，在场景中的每个地方存在的稳态光能。

有的表面实际上能发出光。这些物体也许是电灯灯泡或者是先吸收一种能量再发射出可见光的物体。它们不仅反射光也发射光。最后，所有的发射和反射现象都与波长有关。光源发射含不同波长的整个光谱（除非是单色激光器），表面反射或者吸收某些波长的能量比对其他波长更多一些。可以生产出仪器来测量这些波长的存在，例如多谱扫描仪能够对来自单个表面面元的反射产生200个颜色值。但对于人类，我们可以只用三种颜色值如RGB或HSI合成一种可见光。计算机图形学一般只用RGB成分描述照射亮度和表面反射。

习题6.16

一名业余摄影师日落后在大峡谷边上给朋友拍了一幅照片，尽管用了闪光灯，朋友的照片拍得也很好，但美丽的大峡谷背景却几乎是黑色的，为什么？

6.6.6 Phong明暗模型*

在计算机图形学中常用的着色模型是Phong明暗模型，它解释了几种现象：（a）环境光，（b）漫反射，（c）镜面反射和（d）随距离而变暗，其中（b），（c）和（d）是针对独立光源来说的。假设表面面元在图像点I[x，y]成像的详细情况，以及所有光源的位置和性质是已知的，用$K_{a\lambda}$表示漫反射，用$K_{s\lambda}$表示镜面反射，其中$K_{s\lambda}$是对不同波长λ的反射系数向量（通常

RGB值与三个系数有关），那么该表面面元的反射性质表示为：

$$I_\lambda[x,y]=I_{a\lambda}K_{d\lambda}+\sum_{m=1}^{M}\left(\frac{1}{cd_m^2}I_{m\lambda}[K_{d\lambda}(\mathbf{n}\circ\mathbf{s})+K_{s\lambda}(\mathbf{R}_m\circ\mathbf{V})^\alpha]\right) \tag{6-9}$$

公式（6-9）用到了环境光强度$I_{a\lambda}$和一组M个光源强度$I_{m\lambda}$。可以认为这个公式是一个向量方程，对单个波长λ按类似的公式计算。$I_{a\lambda}$是波长λ的周围光的强度，$I_{m\lambda}$是波长λ的光源m的强度。第m个光源离表面面元的距离是d_m，经表面面元反射产生的反射线是$\mathbf{R}_m$。

6.6.7 基于明暗的人类感知

毫无疑问，人类对三维物体形状的感知离不开表面的明暗信息。尽管上面的照射和反射模型比较简单，但描述的现象说明我们对明暗变化是有感觉的。这个简化模型在计算机图形学中很重要，而且为了加快对被照射表面的绘制速度用了各种近似方法。在受控环境中，计算机视觉系统甚至可以用上面的公式通过明暗分析计算表面形状，这些方法在第13章进行讨论。对于图6-16中的物体，我们对公式标定之后，就可以算出表面点的法线方向。在不受控场景如户外场景中，对不同现象进行解释就很困难。 208

6.7 相关话题*

6.7.1 颜色应用

与只用图像强度、纹理或形状特征相比，颜色特征使一些模式识别问题变得非常简单。颜色测度是局部的，不需要聚集算法和形状分析。例如习题6.13中的问题，在商店自动收费或者配送中心质检系统中，像素级颜色信息在水果与蔬菜分类方面得到了长期应用。另一个例子是建立滤波器去掉WWW中的色情图片。6.5节描述的人脸识别算法，首先根据训练数据进行皮肤颜色检测，然后划分出皮肤像素的区域，并且计算皮肤区域之间的几何关系。如果裸体部分占了一幅图的大部分，那么这幅图就被屏蔽掉。在访问图像数据库以及理解显微镜拍摄的生物图像方面，颜色特征都是很有用处的，这一点将在第8章做详细介绍。

6.7.2 人类的色感机制

了解人类的色感机制是很重要的，主要有以下两方面原因：首先，人类视觉系统通常是研究和模仿的有效系统；其次，图形和图像显示的主要目的是进行人机交互。机器视觉工程师常常希望知道如何才能复制或取代人类的视觉能力，而图形图像学家总想弄明白如何才能做到最佳的人机交互。

总的说来，人类对颜色是有偏爱的。例如，墙的颜色通常刷成不饱和色而不是饱和色，红色趋于刺激，而蓝色趋于放松。大约8%的人是色盲，这意味着应仔细进行颜色选择以方便信息交流。在人类的视网膜内，红绿感受器的数目远大于蓝色感受器数目，特别是在高分辨中央凹中蓝色感受器的数目非常少。因此很多颜色计算在神经元内进行，神经元对来自感受器的输入信息进行集成处理。在神经元处理方面，已经提出各种各样的理论解释颜色处理机制。这种较高级的处理机制，人们还没有完全弄明白，人类的视觉处理机制仍在研究之中。虽然对于显示器上单个像素的颜色并不能准确感知，但即使在光照变化，包括只有两个主波长的光照情况下，人类仍然能够很好地判断一个展开面的颜色。基于边缘的强度处理（第5章）比颜色处理要快，在颜色处理完成之前前者就有了目标识别的结果。理论上常常强调，人类的颜色处理机制是如何在较原始的强度处理基础上发展的。读者可以通过阅读参考材料以及其他相关材料，对人类视觉感知这一广泛领域进行更深入的探索。 209

6.7.3 多谱图像

如第2章讨论的那样，得到一个像素的3个颜色值的传感器是一个多谱传感器。然而，在人类感觉不到的电磁波段，传感器却可以感知到，例如红外波段。在卫星图像的IR波段中，热的沥青路显示亮色，而冷的水域显示暗色。在利用简单程序对表面图像进行分类时，计算一个像素的多个测量值常常是有用的。扫描系统可能较贵，因为必须对它进行认真设计，才能保证辐射的几个频带确实来自同一个表面面元。可以对MRI的扫描参数（参考第2章）进行修改以得到多幅3D图像，对于被扫描体积的每个体素，能有效产生m个强度。这n个测量值可用来确定这种体素是否是脂肪、血液或肌肉组织等。应提醒读者知道，要得到一个3D的MRI体积数据可能需要整整一小时的时间，这意味着由于运动的影响会测到一些噪声，特别是在不同组织的边界附近，由于循环或者呼吸所引起的微小运动，在边界处抽样的材料元素很可能在扫描过程中发生变化。

6.7.4 主题图像

*主题图像*用*伪彩色*将图像中不同属性的材料分开，或者将图像中的不同区域分开。例如，地图或者卫星图像的像素可以根据人的假设做标记，河流是蓝色的，郊区是紫色的，道路是红色的。这些并不是传感器拍到的自然颜色，但是在我们的文化中对这些图像内容已经形成共识。天气图显示温度主题，红色表示热，蓝色表示冷。同样，主题图像可以对表面深度、局部表面方向或者几何形状、纹理、一些特征的密度或者任何其他标量或标称分类进行编码。图6-13的中间两幅图是主题图像，实际颜色空间中的黄、蓝和紫色仅仅是为了区别三个类别。重要的是要记住主题图像显示的不是实际物理传感器的数据，而是经转换或者分类后的数据，目的是为了人类能够进行更好的观察。

6.8 参考文献

更详细的光线处理和光学分析，可以参考Hecht和Zajac（1974）的著作。实用的数字颜色编码参考了Murray和VanRyper（1994）的著作，读者可从这本书中找到数字图像存储的文件
210 格式方面的详细内容。在Foley（1996）等人编写的计算机图形学一书中，详细介绍了彩色显示的硬件设计问题，特别是彩色显示的阴罩技术。Levine（1985）的著作讨论了几种不同的生物视觉系统，以及它们做为测量仪器的特点。Overington（1992）的著作就信号处理技术做了更详细的讨论。Livingston（1988）在心理著作方面开了一个好头。彩色直方图匹配的内容参考了Swain和Ballard（1991）的论文，以及Jain和Vailaya（1996）的论文。人脸抽取的细节可以在Bakic和Stockman（1999）的技术报告中找到。用MRI对人脑的多谱分析请参考Taxt和Lundervold（1994）发表的论文。

1. Bakic, V., and G. Stockman. 1999. Menu selection by facial aspect. *Proc. Vision Interface '99* (19–21 May 1999). Trois Rivieres, Quebec.
2. Fleck, M., D. Forsyth, and C. Pregler. 1966. Finding naked people. *Proc. Euro. Conf. Comput. Vision.* Springer-Verlag, New York, 593–602.
3. Foley, J., A. van Dam, S. Feiner, and J. Hughes. 1996. *Computer Graphics: Principles and Practice,* 2nd Ed in C. Addison-Wesley, New York.
4. Hecht, E., and A. Zajac. 1974. *Optics.* Addison-Wesley, New York.
5. Jain, A., and A. Vailaya. 1996. Image retrieval using color and shape. *Pattern Recog.,* v. 29(8):1233–1244.
6. Levine, M. 1985. *Vision in Man and Machine.* McGraw-Hill, New York.

7. Livingstone, M. 1988. Art, illusion and the visual system. *Sci. Am.* (Jan. 1988), 78–85.
8. Murray, J., and W. VanRyper. 1994. *Encyclopedia of Graphical File Formats.* O'Reilly and Associates, Sebastopol, CA.
9. Overington, I. 1992. *Computer Vision: A Unified, Biologically-Inspired Approach.* Elsevier, Amsterdam.
10. Swain, M., and D. Ballard. 1991. Color indexing. *Inter. J. Comput. Vision,* v. 7(1): 11–32.
11. Taxt, T., and A. Lundervold. 1994. Multispectral analysis of the brain in magnetic resonance imaging. *Proc. IEEE Workshop on Biomed. Image Anal.* (24–25 June 1994), 211 Seattle, 33–42.

第7章　纹理分析

纹理是另一种图像特征，可用来将图像分割成感兴趣的区域，并对这些区域进行分类。在有的图像中，纹理可以定义区域的特性，且对于获得正确的分析是非常关键的。图 7-1中的图像有三种显著不同的纹理：老虎的纹理，灌木丛的纹理以及水域的纹理。这些纹理可以量化表示，并用来识别物体所属的类别。



图7-1　包含不同区域的图像，每个区域都有一种明显的纹理（版权属于Corel Stock Photos）

纹理给我们提供图像中颜色或亮度的空间分布信息。假设区域的直方图表示它有50% 的白色像素和50% 的黑色像素，图7-2 表示具有这样的亮度分布的三个不同的图像，它们可被认为是三种不同的纹理。最左边的图像有两大块：一个白块和一个黑块，中间图像有18个白色小块和18个黑色小块组成棋盘状，最右边的图像有六个长条块，三个白块和三个黑块，组成条状。

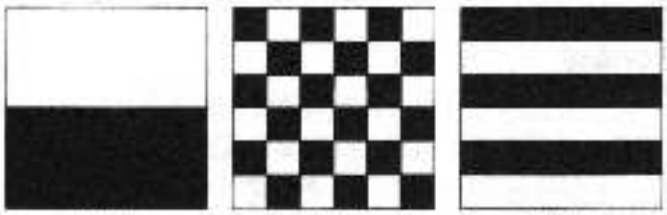


图7-2　三种不同的纹理具有相同的黑白比例分布

图7-2的纹理是人为创建的，包含由黑色块和白色块构造的几何模式。纹理在自然场景中很常见，尤其是室外场景，既包含自然目标又包含人工目标。沙子、石头、草地、叶子、砖块以及许许多多的物体创建了诸多纹理图像。图7-3显示了这样的一些自然纹理。注意两个不同的砖块纹理以及两个不同的叶子纹理看起来都非常不同。所以，仅仅用只有物体类别无法描述纹理。本章讨论什么是纹理、纹理的表示和计算，以及纹理在图像分析中的使用。 212

7.1　纹理、纹理素和统计

图7-2的人工纹理由基本的白色或黑色矩形块组成。在棋盘图中，黑白小方块在2D网格中交替出现。在条状模式中，区域是由长条块在垂直方向以交替颜色组成。分割这些单一颜色的区域以及识别这些简单的模式是很容易的。

现在，考虑图7-3的两种叶子纹理。第一个是许多的小圆叶子，第二个是少量的较大而突出的叶子。这些叶子的空间分布难以用文字描述，而且是不规则的，但是图像的一些性质，

使人们认为图像中存在某种明显的分布。

纹理分析的部分难点是准确地定义什么是纹理。主要有两种定义方法：

213 1. **结构方法**：纹理是具有某种规则或重复关系的基本纹理素（texel）的集合。

2. **统计方法**：纹理是区域中亮度值分布的一种定量度量方法。

第一种方法具有一定魅力，它对于人工创建的有规则的模式是有效的，但第二种方法更通用，更易于计算，在实际中更常用。

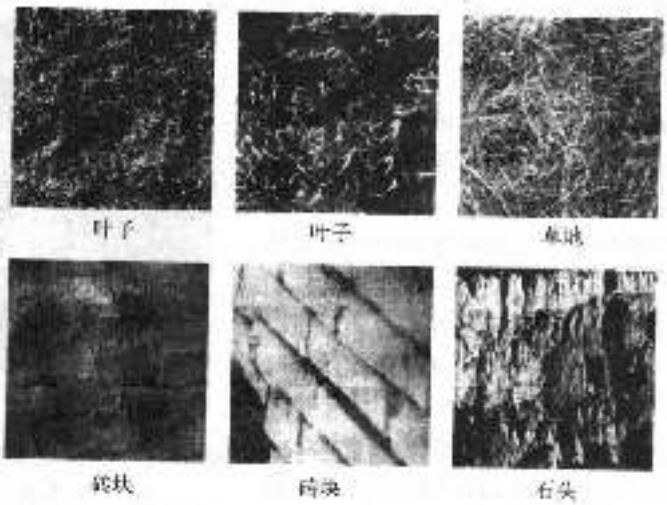


图7-3 自然纹理（来自MIT媒体实验室VisTex数据库：
http://vismod.www.media.mit.edu/vismod/imagery/VisionTexture/vistex.html）

7.2 基于纹理素的描述

纹理可认为是一组具有某种特殊空间关系的纹理素的集合。因而，纹理的结构描述包括纹理素的描述以及空间关系的定义。当然，纹理素必须是可分割的，纹理素之间的空间关系必须是能有效计算的。Tuceryan和Jain提出了一种非常好的基于几何的描述方法。纹理素是可通过一些简单的步骤如阈值化等抽取的图像区域。纹理素之间的空间关系特性，根据下面纹理素的Voronoi图得到。

假设已经抽取出一组纹理素，且每个纹理素都可用一个有意义的点来表示，例如它的重心。设S是这些点的集合。对S中的任意点对P和Q，可以构造连接这两点的线段的垂直平分线。这个垂直平分线将平面分成两个半平面，其中一个是距离P较近的点的集合，另一个是距离Q较近的点的集合，相对于P和Q的垂直平分线，设$H^Q(P)$是距离P较近的半平面。对S中的每个
214 点Q我们都可以重复这个过程。P的Voronoi多边形中的所有点，距离P比距离S中的其他点更近。P的Voronoi多边形定义如下：

$$V(P)=\bigcap_{Q\in S,Q\neq P} H^Q(P)$$

图7-4显示一组圆形纹理素的Voronoi多边形。对于内部的纹理素，该模式表现为六边形；对于位于图像边界的纹理素则表现为不同的形状。

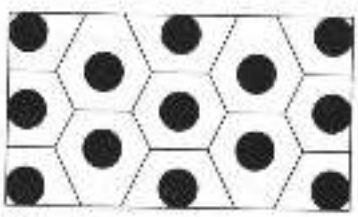


图7-4 一组圆形纹理素的Voronoi图

一旦从图像中抽取出纹理素，并计算出它们的Voronoi

图，就可以计算多边形的形状特征，利用这些形状特征将多边形分类。这些类别确定了各纹理区域。图7-4所示的模式类型扩展到大图像中将产生一个纹理均匀的区域，其特点是具有规则六边形的形状特征。

习题7.1 基于纹理素的描述

寻找或创建5幅图像，使图像纹理具有明显的纹理素，这些纹理素可以利用简单方法进行检测，例如根据灰度值或颜色范围取阈值。至少要找到一种包含了多种纹理素的纹理。画出该图像小部分区域的Voronoi图。

7.3 定量纹理测度

绝大多数情况下，在实际图像中分割纹理素比分割人工生成的模式要困难得多。相反，描述纹理的数量或统计值可从灰度值（或颜色）本身计算出来。这种方法虽然直观性较差，但计算方便，可有效用于纹理的分割和识别。

7.3.1 边缘密度和方向

由于边缘检测是众所周知的、便于应用的特征检测方法，所以把边缘检测作为纹理分析的第一步是很自然的。在给定大小的区域内，边缘像素点的个数在某种程度上反映了区域的纹理分布密集度。边缘的方向一般也有助于刻画纹理模式，它们往往是边缘检测过程的另一个结果。 215

考虑含有N个像素的区域。如果对该区域应用基于梯度的边缘检测算子，对每个像素p产生两个输出：1）梯度幅值$Mag(p)$和2）梯度方向$Dir(p)$，如在第5章中所定义的。一种非常简单的纹理特征是*每单位面积的边缘数*（edgeness per unit area），对于某个阈值T，该纹理特征定义如下：

$$F_{edgeness}=\frac{|\{p \mid Mag(p) \geqslant T\}|}{N} \tag{7-1}$$

每单位面积的边缘数度量了纹理分布的密集度，但不包括纹理的方向。

对这个测度进行扩展，使其既包含密集度又包含方向，可以采用梯度幅值和梯度方向两种直方图。设$H_{mag}(R)$表示区域R的梯度幅值的规范化直方图，H_{dir}表示区域R的梯度方向的规范化直方图。这些直方图的箱格数都是固定的小数目（如10），这些箱格表示幅度的组类和方向的组类。直方图都根据区域R的大小N_R进行了规范化。那么

$$F_{mag\,dir}=(H_{mag}(R), H_{dir}(R)) \tag{7-2}$$

是关于区域R中纹理的定量描述。

观察图7-5所示的两幅5×5图像。左边的图像比右边的图像具有更多的边缘，它有25个像素，每个像素内就有一条边缘，那么它的每单位面积的边缘数就是1.0。右边的图像在25个像素内共有6条边缘，那么它的每单位面积的边缘数是0.24。对于梯度幅值直方图，假设有两个箱格，分别代表暗边缘和亮边缘。对于梯度方向直方图，采用三个箱格，分别代表水平、垂直和对角方向的边缘。左边的图像有6条暗边缘和19条亮边缘，那么它的规范化梯度幅值直方图是（0.24,0.76），意味着24%的边缘是暗边缘，76%的边缘是亮边缘。它有12条水平边缘和13条垂直边缘，没有对角边缘，这样它的规范化梯度方向的直方图是（0.48, 0.52, 0.0），意味着48%的边缘是水平的，52%是垂直的，对角方向占0%。右边的图像无暗边缘，有6条亮边缘，

216 它的规范化梯度幅值直方图是（0.0, 0.24）。它没有水平和垂直边缘，只有6条对角边缘，这样它的规范化梯度方向直方图是（0.0, 0.0, 0.24）。对于这两幅图像，每单位面积的边缘数足以把它们分开，但直方图测度在一般意义上提供了一个更有力的描述机制。两个n-箱格直方图H_1和H_2可通过计算它们的L_1距离来进行比较。

$$L_1(H_1, H_2) = \sum_{i=1}^{n} |H_1[i] - H_2[i]| \qquad (7\text{-}3)$$

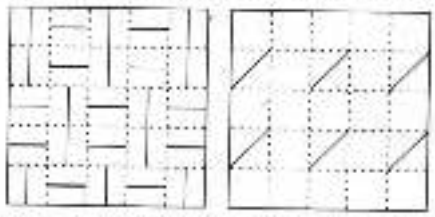


图7-5 具有不同边缘和边缘方向统计特征的两幅图像

习题7.2 基于边缘的纹理测度

获得一组有许多人为结构的图像，它们含有清晰明确的边缘。编写程序，利用公式（7-2）计算每幅图像的纹理测度$F_{<x,y,G>}$，并利用公式（7-3）的L_1距离进行比较。

7.3.2 局部二值分解

另一个简单但有效的纹理测度是局部二值分解。对图像中的每个像素p，检查它的8个邻点，看是否有比p大的亮度值。从8个邻点得到的结果用于构造8位二进制数$b_1b_2b_3b_4b_5b_6b_7b_8$，如果第i个邻点的亮度值小于或等于p的亮度值，则$b_i = 0$，反之$b_i = 1$。用这些数字的直方图表示图像纹理。对于两幅图像或区域，可通过计算上面定义的直方图间的L_1距离进行比较。

习题7.3 LPB纹理测度

利用前面习题的图像，另写一个程序计算每幅图像的LBP纹理测度直方图。利用该测度计算图像对之间的L_1距离，与你前面得到的结果进行比较。

7.3.3 共生矩阵和特征

共生（co-occurrence）矩阵是一个二维的阵列C，其中的行和列表示可能的图像值V的集合。例如，对于灰度图像，V是可能的灰度值的集合；对于彩色图像，V是可能的颜色值的集合。$C(i, j)$表示值i与值j以某种指定的空间关系共同出现的次数。例如，定义空间关系为值i紧接着值j的右边出现。为更精确起见，我们特别考虑V是一组灰度值的集合且空间关系由向量d确定的情况，d描述了值为i的像素和值为j的像素之间的位移关系。

217 设d是一个位移向量(dr, dc)，其中dr是行方向的位移（向下），dc是列方向的位移（向右）。设V是灰度值的集合，图像I的灰度共生矩阵C_d定义如下：

$$C_d[i, j] = |\{[r, c] \mid I[r, c] = i \text{以及} I[r + dr, c + dc] = j\}| \qquad (7\text{-}4)$$

图7-6显示共生矩阵的概念，用到了4×4的图像I以及三个不同的共生矩阵$C_{[0,1]}$，$C_{[1,0]}$和$C_{[1,1]}$。

在$C_{[0,1]}$中，注意位置[1, 0]的值为2，表示图像中$j = 0$直接出现在$i = 1$的右边两次。位置[0,1]的值为0，表示图像中$j = 1$从未紧接着$i = 0$的右边出现。共生矩阵中的最大值是4，位于[0,0]处，表示图像中0出现在另一个0的右边4次。

习题7.4 共生矩阵

对图7-6构造灰度共生矩阵$C_{[1,2]}$，$C_{[2,1]}$和$C_{[2,2]}$。

标准的灰度共生矩阵有两个重要的变形，第一个是规范化的灰度共生矩阵N_d，定义如下：

$$N_d[i,j]=\frac{C_d[i,j]}{\sum_i\sum_j C_d[i,j]} \tag{7-5}$$

将共生矩阵的值规范化到0和1之间，这样在大矩阵中就可以把这些值理解为概率。第二个是对称灰度共生矩阵S_d，定义如下：

$$S_d[i,j]=C_d[i,j]+C_{-d}[i,j] \tag{7-6}$$

将一对对称的连接组合在一起。

218

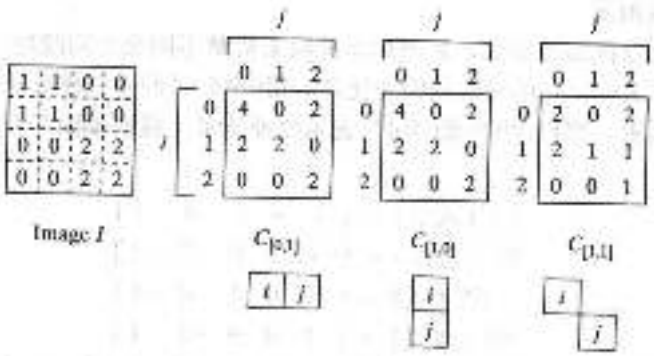


图7-6 灰度图像的三种不同的共生矩阵

习题7.5 规范化共生矩阵

对图7-7的图像，计算规范化共生矩阵$N_{[1,-1]}$，假设黑色像素的灰度值为0，灰色像素的灰度值为1，白色像素的灰度值为2，规范化共生矩阵如何表示图像的纹理模式？

共生矩阵可捕捉纹理特征，但不利于进一步的分析，比如对两种纹理的比较。从共生矩阵中计算数值特征，可以用更紧凑的方法表示纹理。下面是从规范化共生矩阵中推导出的标准特征。

$$\text{能量}=\sum_i\sum_j N_d^2[i,j] \tag{7-7}$$

$$\text{熵}=-\sum_i\sum_j N_d[i,j]\log_2 N_d[i,j] \tag{7-8}$$

$$\text{对比度}=\sum_i\sum_j (i-j)^2 N_d[i,j] \tag{7-9}$$

$$\text{均匀性}=\sum_i\sum_j \frac{N_d[i,j]}{1+|i-j|} \tag{7-10}$$

$$\text{相关性}=\frac{\sum_i\sum_j (i-\mu_i)(j-\mu_j)N_d[i,j]}{\sigma_i\sigma_j} \tag{7-11}$$

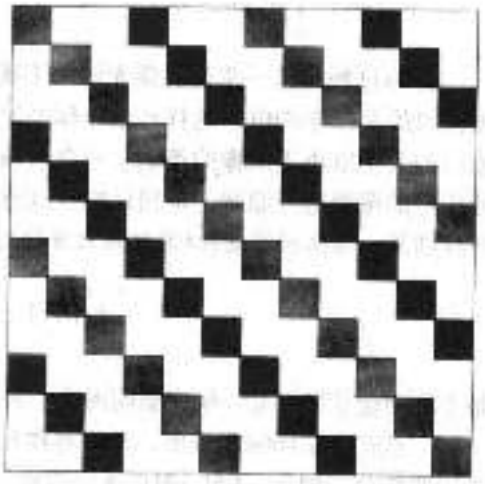


图7-7 具有对角线纹理模式的图像

219

其中μ_i、μ_j是行和列的均值，σ_i、σ_j是行和列的标准差。和$N_x(i)$、$N_y(j)$定义为：

$$N_d[i]=\sum_j N_d[i,j]$$

$$N_d[j] = \sum_i N_d[i, j]$$

从共生矩阵中推导纹理测度的一个问题是如何选择位移向量d。Zucker和Terzopoulos建议的方法是，利用χ^2统计测试来选择具有最多的结构的d值，也就是使下列值最大化:

$$\chi^2(d) = \left(\sum_i \sum_j \frac{N_d^2[i, j]}{N_d[i]N_d[j]} - 1 \right)$$

7.3.4 Laws纹理能量测度

另一种生成纹理特征的方法是，利用局部模板来检测不同类型的纹理。Laws提出了纹理能量方法，度量一个大小固定的窗口内的变化量。用9个5 × 5的卷积模板计算纹理能量，对于被分析图像的每个像素，都用含9个数的向量表示纹理能量。模板根据下面的向量算出，这与第5章的内容类似。

$$\begin{aligned}
\text{L5（水平）} &= [\ 1 \quad 4 \quad 6 \quad 4 \quad 1\ ] \\
\text{E5（边缘）} &= [-1 \ -2 \quad 0 \quad 2 \quad 1\ ] \\
\text{S5（点）} &= [-1 \quad 0 \quad 2 \quad 0 \ -1\ ] \\
\text{R5（波纹）} &= [\ 1 \ -4 \quad 6 \ -4 \quad 1\ ]
\end{aligned}$$

向量的名字代表了它们的含义。L5向量表示加权中心的局部均值。E5向量检测边缘，S5向量检测点，R5向量检测波纹。计算向量对的外积得到2D卷积模板。例如，模板E5L5是按下面方式计算E5和L5的乘积得到的:

$$\begin{bmatrix} -1 \\ -2 \\ 0 \\ 2 \\ 1 \end{bmatrix} \times \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix} = \begin{bmatrix} -1 & -4 & -6 & -4 & -1 \\ -2 & -8 & -12 & -8 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 8 & 12 & 8 & 2 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

Laws过程的第一步是去除光照的干扰：通过在图像上移动一个小窗口，从每个像素中减去窗口的局部平均值，这样产生一幅经过预处理的图像，其中每个邻域的平均亮度值接近0。窗口的大小取决于图像的类别，对自然场景采用15 × 15的窗口。然后用16个5 × 5的模板对预
220 处理后的图像进行滤波，得到16幅经过滤波的图像。设$F_k[i, j]$是在像素$[i, j]$处用第k个模板的滤波结果，那么对滤波器k的纹理能量图E_k定义如下:

$$E_k[r, c] = \sum_{j=c-7}^{c+7} \sum_{i=r-7}^{r+7} |F_k[i, j]| \tag{7-12}$$

每个纹理能量图都是一幅完整的图像，表示用第k个模板对输入图像进行处理。

一旦产生了16幅能量图，某些对称对则可以互相组合，最终产生9个图，每一对用它们的平均值代替。例如，E5L5测量水平边缘，L5E5测量垂直边缘。这两个图的平均值则测量总边缘。9个合成的能量图是:

L5E5/E5L5	L5S5/S5L5
L5R5/R5L5	E5E5
E5S5/S5E5	E5R5/R5E5

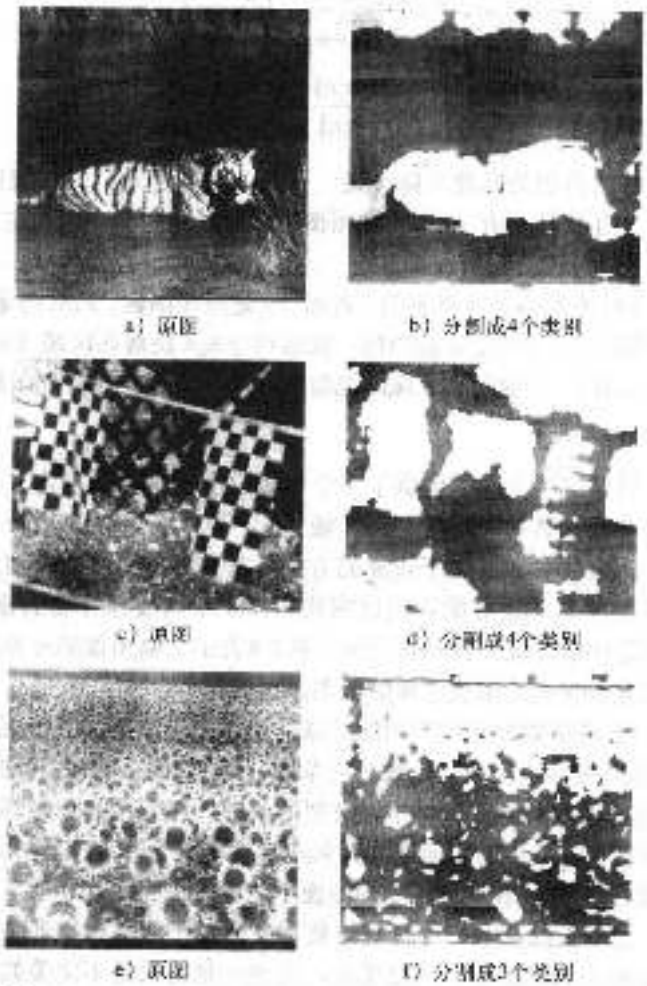
S5S5 S5R5/R5S5

R5R5

所有处理的结果给出9个能量图，或者从概念上说，是一幅图像，它的每个像素点都有含9个纹理特性的向量来描述。表7-1表示图7-3中草地、石头和砖块图像中主要纹理的9个纹理特征。用这些纹理特征可将图像聚类成纹理均匀的区域，图7-8显示多纹理图像聚类后的分割图。

表7-1 图7-3中图像的Laws纹理能量测度

图像	E5E5	S5S5	R5R5	E5L5	S5L5	R5L5	S5E5	R5E5	R5S5
叶子1	250.9	140.0	1209.2	709.6	512.3	1516.2	187.5	568.8	430.0
叶子2	257.7	121.4	988.7	820.6	510.1	1186.4	172.9	439.6	328.9
草地	192.8	107.2	1076.9	586.9	410.5	1208.5	144.0	444.8	338.1
砖块1	128.1	60.2	512.7	442.1	273.8	724.8	86.6	248.1	176.3
砖块2	72.4	28.6	214.2	263.6	120.9	271.5	43.2	93.3	68.5
石头	224.6	103.2	766.8	812.8	506.4	1311.0	150.4	413.5	281.1



a）原图 b）分割成4个类别

c）原图 d）分割成4个类别

e）原图 f）分割成3个类别

图7-8 利用Laws纹理能量测度分割图像。（原图来自Corel Stock Photos和MIT媒体实验室VisTex数据库）参见彩图7-8

习题7.6 Laws纹理能量测度

编程计算Laws纹理能量测度，输入是灰度图像，输出是9幅图像，每幅对应一种纹理能量测度。获得一组既有人造纹理又有自然纹理的图像，对它们进行一系列的测试。对于每次测试，设其中一幅为测试图像，其他图像为数据库图像，编写交互式前端程序，允许用户选择测试图像的一个像素，然后在数据库图像中寻找那些与所选像素纹理类似的图像，相似性度量采用9个纹理能量测度的L_1距离。某方法是把测试图像像素的9个值，与图像库中每幅图像的每个像素的9个值进行比较，一旦某个像素与测试像素具有足够类似的纹理能量测度，则选择该图像。你能想出一种更有效的方法吗？

7.3.5 自相关和功率谱

图像的自相关函数可用来检测纹理元素的重复模式，描述纹理的精细度和粗糙度。利用第5章的思想，一幅$(N+1)\times(N+1)$图像对于位移$d=(dr, dc)$的自相关函数$\rho(dr, dc)$按如下公式给出：

$$\rho(dr,dc)=\frac{\sum_{r=0}^{N}\sum_{c=0}^{N}I[r,c]I[r+dr,c+dc]}{\sum_{r=0}^{N}\sum_{c=0}^{N}I^2[r,c]} \tag{7-13}$$

$$=\frac{I[r,c]\circ I_d[r,c]}{I[r,c]\circ I[r,c]} \tag{7-14}$$

如果纹理较粗，那么自相关函数下降缓慢；否则，下降迅速。对于规则的纹理，自相关函数将有波峰和波谷。由于$I[r+dr, c+dc]$在图像的边界处未定义，必须定义一种方法计算这些虚的图像值。

自相关函数与傅里叶变换的功率谱相关。若$I(r, c)$是图像函数，$F(u, v)$是它的傅里叶变换，则$|F(u, v)|^2$定义为功率谱，其中$|\cdot|$是复数的模。频域可分成n_r块圆环区域（对频率信息）以及n_θ扇形区域（对方向信息），计算每块区域的总能量来产生一组纹理特征，如第5章所介绍的。

7.4 纹理分割

任何纹理测度，只要对每个像素提供了一个值或一个向量值，描述了该像素点邻域的纹理，都可用于将图像分割成具有相似纹理的区域。和任何其他分割算法一样，纹理分割算法可分成两大类：基于区域的方法和基于边界的方法。基于区域的方法试图将具有相似纹理特性的像素点分组或聚类。基于边界的方法试图找到介于不同纹理分布的像素间的纹理边界。我们把分割算法留到第10章图像分割时再讨论。图7-8表示几幅图像的分割结果，其中利用了Laws纹理能量测度以及将像素点组成区域的聚类算法。

在图7-8a和b中，老虎图像被分成不同的区域，表示老虎、水和其他混合区域。在图7-8c和d中，多目标图像被分成多块区域，大致对应草地、两面旗帜、黑色的网状篱笆和背景。在图7-8e和f中，向日葵图像被分割成三种不同类型的纹理：图像顶部和底部的黑暗边界、在田野远处的小朵向日葵花，以及在田野近处的大朵向日葵花。表7-2表示每幅图像主要区域的平均Laws纹理能量测度。表7-3比较了几幅不同图像中老虎区域的Laws测度。

向日葵图像中，一些大的花朵，深色花心被划分为深色的边界纹理，这是因为用来计算纹理的模板比那些大花朵的花心小。一般说来，这些分割结果是不完美的，它们受到算子的限制。同时利用颜色和纹理进行的分割可取得更好的效果，但自然场景分割还是悬而未决的问题。对于一般更复杂的分割处理参见第10章。

表7-2 图7-8中图像主要区域的Laws纹理能量测度

区域	E5E5	S5S5	R5R5	E5L5	S5L5	R5L5	S5E5	R5E5	R5S5
老虎	168.1	84.0	807.7	553.7	354.4	910.6	116.3	339.2	257.4
水	68.5	36.9	366.8	218.7	149.3	459.4	48.6	159.1	117.3
植被	258.1	113.0	787.7	1057.6	702.2	2056.3	182.4	611.5	350.8
葡萄	189.5	80.7	624.3	701.7	377.5	803.1	120.6	297.5	215.0
草	206.5	103.6	1031.7	625.2	428.3	1153.6	146.0	427.5	323.6
小花朵	114.9	48.6	289.1	402.6	241.3	484.3	73.6	158.2	109.3
大花朵	76.7	28.8	177.1	301.5	158.4	270.0	45.6	89.7	62.9
边界	15.3	6.4	64.4	92.3	36.3	74.5	9.3	26.1	19.5

表7-3 几幅不同图像中老虎区域的Laws纹理能量测度

图像	E5E5	S5S5	R5R5	E5L5	S5L5	R5L5	S5E5	R5E5	R5S5
老虎1	171.2	96.8	1156.8	599.4	378.9	1162.6	124.5	423.8	332.3
老虎2a	146.3	79.4	801.1	441.8	302.8	996.9	106.5	345.6	256.7
老虎2b	177.8	96.8	1177.8	531.6	358.1	1080.3	128.2	421.3	334.2
老虎3	168.8	92.2	966.3	527.3	354.1	1072.5	124.0	389.8	289.8
老虎4	168.1	84.0	807.7	553.7	354.4	910.6	116.3	339.2	257.4
老虎5	146.9	80.7	868.7	474.8	326.2	1011.3	108.2	355.5	266.7
老虎6	170.1	86.8	913.4	551.1	351.3	1180.0	119.5	412.5	295.2
老虎7	156.3	84.8	954.0	461.8	323.8	1017.7	114.0	372.3	278.6

习题7.7 纹理分割

编程计算一幅图像的Laws纹理能量测度，研究它们在纹理分割中的效果。另编写交互式前端程序，允许用户在图像中画出块状区域，每个区域包含一种类别的纹理，如草地或天空。对每块区域，计算九个纹理特征的平均值。用表格列出每种纹理类别的名字和该类别对应的九个平均值。比较不同类别的分类结果。

7.5 参考文献

纹理分析是计算机视觉中研究时间最长的领域之一，可以追溯到20世纪60年代末和70年代初在遥感方面的应用研究。Haralick、Shanmugam和Dinstein（1973）提出了共生矩阵特征，用于分析遥感图像。Zucker和Terzopoulos（1980）针对采用共生矩阵技术确定最佳位移量的问题提出了统计检验。Trivedi（1984）利用灰度共生矩阵特征进行目标检测。

Julesz（1975）做了一组现在很著名的实验，实验人类对纹理的感知情况。Tumura、Mori和Yamawaki（1978）提出了与人类视觉感知对应的纹理特征。Tomita、Shirai和Tsuji（1982）以及Tuceryan和Jain（1990）提出了纹理分析的结构方法。Wang和He（1990）提出了光谱方法。而Cross和Jain（1983）提出了马尔可夫随机场方法。Laws纹理能量模板能计算数值纹理特征，在实际中得到广泛应用。由Weszka、Dyer和Rosenfeld（1976）所著的综述文章论述了该领域的早期工作。Tuceryan和Jain（1994）的著作全面描述了1995年之前的研究方法。Leung和Malik（1999）最近的工作，提出一种从训练样本中学习纹理基元的新方法。

224

1. Cross, G. R., and A. K. Jain. 1983. Markov random field texture models. *IEEE Trans. Pattern Anal. Machine Intelligence,* v. PAMI-5:25–39.
2. Haralick, R. M., K. Shanmugam, and I. Dinstein. 1973. Textural features for image classification. *IEEE Trans. Systems, Man, and Cybernetics,* v. 3:610–621.
3. Julesz, B. 1975. Experiments in the visual perception of texture. *Sci. Am.,* 34–43.
4. Laws, K. 1980. Rapid texture identification. *SPIE Image Processing for Missile Guidance,* v. 238:376–380.
5. Leung, T., and J. Malik. 1999. Recognizing surfaces using three-dimensional textons. *Int. Conf. Comput. Vision* (Sept. 1999), 1010–1017.
6. Tamura, H., S. Mori, and T. Yamawaki. 1978. Textural features corresponding to visual perception. *IEEE Trans. Systems, Man, and Cybernetics,* v. 8(6):460–473.
7. Tomita, F., Y. Shirai, and S. Tsuji. 1982. Description of textures by a structural analysis. *IEEE Trans. Pattern Anal. Machine Intelligence,* v. PAMI-4:183–191.
8. Trivedi, M. M. 1984. Object detection based on gray level cooccurrence. *Comput. Vision, Graphics, and Image Proc.,* v. 28:199–219.
9. Tuceryan, M., and A. K. Jain. 1994. Texture analysis. In *Handbook of Pattern Recognition and Vision,* C. H. Chen, L. F. Pau, and P. S. P. Wang, Eds. World Scientific Publishing Co., Singapore, 235–276.
10. Tuceryan, M., and A. K. Jain. 1990. Texture segmentation using Voronoi polygons. *IEEE Trans. Pattern Anal. Machine Intelligence,* v. 12(2):211–216.
11. Wang, L., and D. C. He. 1990. Texture classification using texture spectrum. *Pattern Recog. Lett.,* v. 13:905–910.
12. Weszka, J., C. R. Dyer, and A. Rosenfeld. 1976. A comparative study of texture measures for terrain classification. *IEEE Trans. Systems, Man, and Cybernetics,* v. SMC-6: 269–285.
13. Zucker, S. W., and D. Terzopoulos. 1980. Finding structure in co-occurrence matrices for texture analysis. *Comput. Graphics and Image Proc.,* v. 2:286–308.

225

第8章 基于内容的图像检索

大容量的内存外存设备越来越便宜，处理器的运算能力越来越强，这使得建立和使用图像数据库从期望变成了现实。图像数据库可用来存储艺术收藏品图像、卫星图像、医学图像以及普通图片。使用图像库的目的各种各样。例如艺术收藏家可能想找到某位艺术家的作品，或者找出曾经见过的某幅图像的作者；医学图像库使用者可能是学习解剖学的学生，或者是正在寻找某种疾病的样例图片的医生；普通图片则可为一篇文章或一本图书提供合适的插图。总之，图像检索的应用范围很广泛，用户可能在寻找马的图像，或者日落的图像，甚至是在查找抽象的概念，比如"爱"。

图像库包含上万幅甚至上百万幅图像，数量非常巨大。在多数情况下，仅提供关键词进行检索。这些关键词由人来进行标注、分类并输入数据库系统。而图像可以根据内容进行检索，所谓内容是指图像的颜色分布、纹理、区域形状或者目标类别等。尽管图像分割和识别算法仍然处于初级状态，目前已经建立起图像检索商用系统和研究系统，并且这些系统有的已经投入使用，这些系统经常在万维网上可以演示。这一章讨论基于内容的图像检索方法，而不是关键词检索方法。

8.1 图像数据库实例

有的图像数据库的建立，只是为了说明图像检索系统是如何工作的。IBM的图像内容查询（QBIC）数据库就是这样的系统。QBIC是一个研究性系统，后来IBM把它开发成一个商业系统，并向市场销售。QBIC基于图像的视觉内容进行检索，利用了诸如颜色百分比、颜色分布和纹理等特征。Virage公司开发出一个具有竞争力的产品，即Virage搜索引擎。它可以基于颜色、组成、纹理和结构来检索图像。这些及其他图像搜索引擎都可用于检索其他机构提供的数据库。例如，旧金山的精品艺术博物馆（Fine Arts Museums）允许用QBIC检索他们的图像库，其中包括很多数字化绘画。图8-1a就是该图像库中雷诺阿的一幅绘画。类似的数字艺术图像库在世界的各大城市都已经建立起来。



a）雷诺阿的绘画



b）紫水晶图像

图8-1 数字图像实例。（皮埃尔·奥古斯特·雷诺阿的绘画，Beaulieu的风景，1893，经旧金山精品艺术博物馆许可，Mildred Anna Williams收藏，1944.9。紫水晶图像经Smithsonian学院许可，1992）参见彩图8-1

除了艺术品收藏库，还有普通的图片收藏库。私人用户也许想把这些图片作为产品或者作为文章中的插图，这些图片经过私人允许就可以使用。这种图像库中最大的是Corbis Archive图像库，它包含一千七百多万的图像，其中近一百万是数字图像，并且这个数字还在增长。该图库试图捕捉人类的所有表情和感觉，它包含诸如历史、艺术、娱乐、科学、工业和动物等类别。Corbis提供基于关键词

和基于浏览的图像检索方式。另一个公司Getty Images，提供几个在线的分类图像数据库，可根据关键词进行检索。

除了艺术作品和摄影图片外，图像库中还包括科学和医学图像。美国国家医药图书馆提供的图片包含：X射线，CT扫描图，MRI核磁共振图像，以及从男性和女性尸体上定时抽取的彩色切片，可给人们提供大约1万4千幅图像进行医学研究。美国国家航空航天局（NASA）
227 建立了一个巨大的卫星图像库，并有偿提供给公众使用。美国地质勘探局（USGS）提供Web搜索功能，为寻找和定购数据库包括数字卫星图像和航测图像的用户服务。另外，万维网本身也是一个包含文本和大量图像的数据库。Web图像搜索引擎正在开发，将根据关键词以及一定程度的图像内容进行检索。

8.2 图像数据库查询

要检索一个图像数据库，必须根据某种方式进行检索，而不是对整个图像库从头到尾搜索一遍。一般来说，公司在建立图像库时，都有一个选择过程和分类过程，选择过程决定哪些图像应该加入数据库，分类过程为选定的图像分配类别和关键词。万维网上的图像通常都有一个标题，根据这些标题可以自动抽取出关键词。

对于关系数据库系统，实体可通过其文本属性来检索。用来检索图像的属性包括一般类别、目标名称、人名、创建的日期和来源等。可根据这些属性建立图像索引，查询时就可以快速查出结果。这种文本查询方式可用SQL关系数据库语言来描述，SQL可用于所有的标准关系数据库。例如，查询任务：

```
SELECT * FROM IMAGEDB
WHERE CATEGORY = 'GEMS' AND SOURCE = 'SMITHSONIAN'
```

将从称为IMAGEDB的集合中寻找并返回满足下面条件的图像：这些图像的CATEGORY属性是“GEMS”，且SOURCE属性是“SMITHSONIAN”。目的是检索Smithsonian学院的宝石图像。图8-1中的b图就是从集合中找到的一幅紫水晶图像。检索结果中还有很多其他的宝石图像。为了允许更多的选择性检索，需要为每幅图像存储一个关键词集合。在关系数据库中，每幅图像的KEYWORD是一个多值属性。例如这幅紫水晶图像可能有这些关键词：“AMETHYST”、“CRYSTAL”和“PURPLE”，针对用户的需求，可以根据这三者或它们的任意组合进行查询。例如，SQL查询

```
SELECT * FROM IMAGEDB
WHERE CATEGORY = 'GEMS' AND SOURCE = 'SMITHSONIAN'
      AND (KEYWORD = 'AMETHYST' OR KEYWORD = 'CRYSTAL'
           OR KEYWORD = 'PURPLE')
```

将从集合IMAGEDB中检索满足下面条件的图像：这些图像的CATEGORY属性是“GEMS”且SOURCE属性是“SMITHSONIAN”，并且KEYWORD的值为“AMETHYST”或“CRYSTAL”或“PURPLE”。这样的检索结果将不仅仅是紫水晶图像；用户可以浏览返回的
228 图像集合并选择图像。

关键词检索的能力是有限的。人工标注关键词耗费财力，而且有可能遗漏一些有助于检索的关键词。对于网上数据库，利用HTML标题可以自动标注关键词，但索引能力同样是有限的。此外，自动获取关键词的检索方法，返回的检索结果可能与用户期望的结果大相径庭。

图8-2表示在网上检索“pigs”关键词得到的两幅图像。

仅依靠关键词检索是不够的。下面研究其他图像检索方法，这些方法可以代替关键词检索或者与关键词检索相结合。



a）包含猪的图像　　b）不包含猪的图像

图8-2　关键词检索返回猪的图像（图像a的版权来自Corbis. Credit line:©51 Clive Druett; Papilio/CORBIS）

习题8.1　关键词查询

设计SQL查询，要检索出图8-2a但不会检索出图8-2b。采用任何你认为合适的类别和关键词。

8.3　示例查询

示例查询（QBE）是数据库查询术语，形式上通过填充表格中的数值和限制条件来实现查询，系统可将其转化为SQL语句。第一个QBE系统是IBM开发的。微软公司的Access也属于这类系统。在标准关系数据库中，属性值主要是文本或数据。示例查询仅仅是为用户提供了一个方便的接口，没有任何特殊的功能。

对于图像数据库，示例查询的思想很有意义。与输入一个查询不同，图像数据库用户能够提供给系统一个样例图像，或者在屏幕上交互地画出一幅，或者仅仅是勾勒出目标的轮廓。检索系统应能返回与此相似的图像或者包含相似目标的图像。这是所有基于内容的图像检索的目标。每个检索系统都有各自的方式来定义查询任务、判断查询图像和数据库图像的相似性以及选择要返回的图像。

为了使讨论具有一般性，我们考虑用一幅例图和一些约束条件来进行查询。例图可以是一幅数字图像、一幅用户画的草图、一幅线条图或者是一个空集（在这种情况下，检索结果仅需要满足约束条件）。约束条件可以是出现在检索系统中的关键词，或者指明应该出现在图像中的目标，甚至是目标间的空间关系。最常见的情况，查询图像是一幅数字图像，检索系统按照一定的图像距离测度比较库中图像与查询图像的相似性。当返回的距离是0时，库中图像与查询图像完全匹配，距离值大于0表示与查询图像有不同程度的相似性。图像搜索引擎通常返回一个图像集合，按照距离大小进行排序。图8-3表示基于颜色分布距离测度的QBIC检索结果。图中显示的是与查询图像最相似的8幅图像，其中左上角为查询图像，因为与查询图像最相似的总是其自身。 229

8.4　图像距离度量

判断库中图像与查询图像在多大程度上相似，这取决于采用什么样的距离测度。主要有四类相似性度量方法：

（1）颜色相似性
（2）纹理相似性
（3）形状相似性
230 （4）目标和目标关系相似性

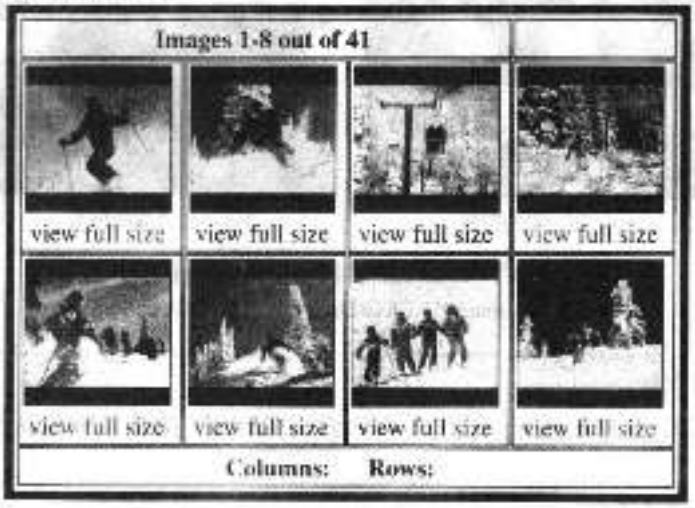


图8-3 基于颜色分布相似性的QBIC检索结果。查询图像是位于左上角的图像。（Egames提供）参见彩图8-3

8.4.1 颜色相似性度量

颜色相似性度量一般比较简单。它比较一幅图像的颜色与另一幅图像的颜色或与一个定义的查询概念比较。例如，QBIC允许用户通过颜色百分比进行查询。用户从颜色表中选择最多5种颜色，并指明希望每种颜色所占的百分比。QBIC寻找与这些颜色百分比最接近的图像。图像中颜色的位置不是检索时考虑的因素。图8-4表示对40%红色30%黄色和10%黑色的查询任务返回的图像集合。这些图像在颜色上很相似，但却有不同的构成。

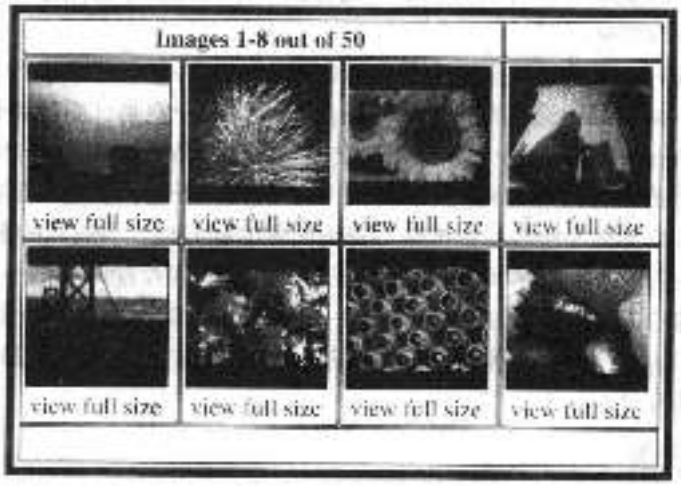


图8-4 基于颜色百分比的QBIC检索结果。查询定义为40%的红色、30%的黄色和10%的黑色。（Egames提供）参见彩图8-4

一种相关技术是颜色直方图匹配，这在第6章中曾经讨论过。在第16章的Veggie Vision系统中将会用到。用户提供一幅样例图像，并要求系统返回与该图像颜色直方图距离最小的图像。颜色直方图距离应包含某种度量方式，以衡量两种不同颜色的相似程度。例如，QBIC将颜色直方图距离定义为：

$$d_{hist}(I, Q) = (h(I) - h(Q))^T A(h(I) - h(Q)) \tag{8-1}$$

其中$h(I)$和$h(Q)$分别是图像I和Q的K维直方图，A是一个$K \times K$的相似度矩阵。在这个矩阵中，颜色越相似，相似度的值越接近于1，反之，颜色差别越大，相似度的值越接近于0。 231

颜色分布是另一种距离度量方法。一般开始时用一幅空栅格图表示查询图像，然后从颜色表中为每个方格选择颜色。图8-5中，用户从左上角所示的颜色矩阵中选择了两种颜色，并对右上角的6×6栅格涂色。图中所示的图像，是利用简单的颜色分布距离测度得到与查询图像最相似的图像。如图8-3所示，也可以选用一幅例图，使系统返回与例图具有相似颜色空间分布的图像。

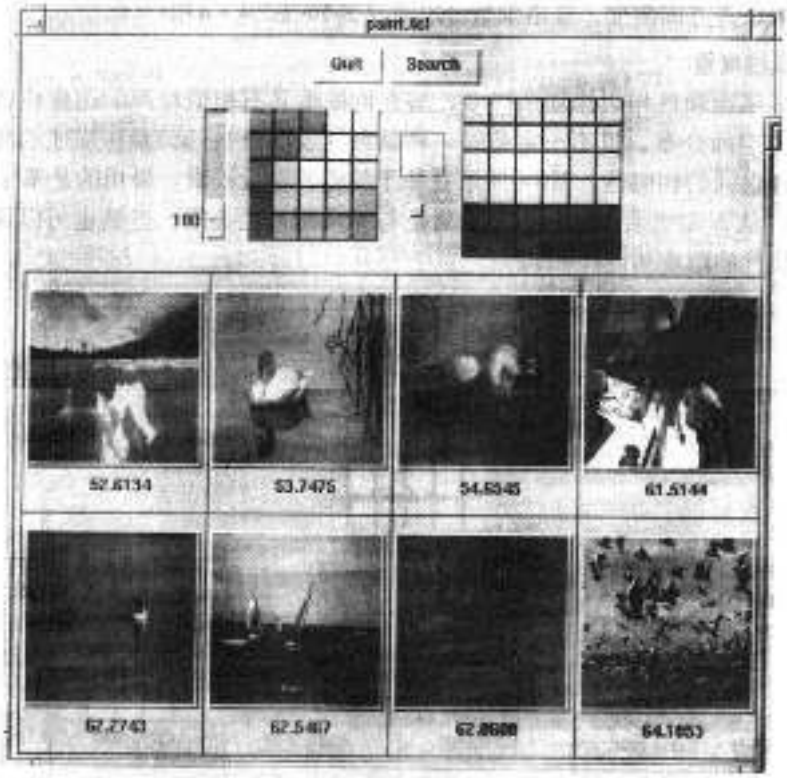


图8-5 图像库检索结果，其中查询图像是涂色的栅格。（图像来自MIT媒体实验室的VisTex数据库：https://vismod.www.media.mit.edu/vismod/imagery/VisionTexture/vistex.html；参见彩图8-5）

栅格颜色分布度量要用到栅格颜色距离测度$\hat{d}_{color}$，来比较查询图像的每个方格与可能匹配的图像对应方格之间的相似性，并将结果组合为统一的图像距离：

$$d_{gridded\_color}(I, Q) = \sum_g \hat{d}_{color}(C^I(g), C^Q(g)) \tag{8-2}$$

232

其中$C^I(g)$表示库中图像I的第g个方格的颜色，$C^I(Q)$表示在查询图像Q中对应的方格g的颜色。

至于方格本身的颜色表示可简可繁，以下是几种合适的表示方法：

（1）方格的平均颜色值

（2）颜色的均值和标准差

（3）颜色的多箱格直方图

方格距离测度$\hat{d}$必须是关于颜色表示的有意义的距离。例如，如果颜色均值用一个三元组（R, G, B）来表示，那么就可以选择测度 $\hat{d}=\left\|(R^Q,G^Q,B^Q)-(R^I,G^I,B^I)\right\|^2$，当然这个选择未必是最好的。一些系统不是比较（R, G, B）的值，而是将颜色空间划分成3D箱格的集合，并用一张表来表示箱格数之间相似性。这与前面的QBIC直方图距离采用的是相同的技术。

习题8.2 颜色直方图距离

设计4×4×4的颜色直方图距离测度，用来比较两幅图像，可以比较整幅图像，也可以比较子图。用该基本测度实现基于栅格的距离测度，这个栅格测度允许用户定义栅格的维数，并可将每对方格的距离组合为一个统一的距离，如公式（8.2）所示。用多对颜色图像，计算你定义的栅格颜色直方图测度，栅格维数分别定义为1×1、4×4和8×8。

8.4.2 纹理相似性度量

纹理相似性要比颜色相似性复杂得多。与查询图像具有相似纹理的图像应该具有相同的颜色（或灰色）空间分布，但不一定是同一种颜色（或灰色）。第7章所描述的纹理测度可用来判断两种纹理之间的相似性。图8-6所示是基于纹理的图像检索，采用的是基于Laws纹理能量的距离测度。从查询结果可见，这个距离是与图像颜色无关的。当然也可以设计同时包括颜色和纹理相似性的距离测度。

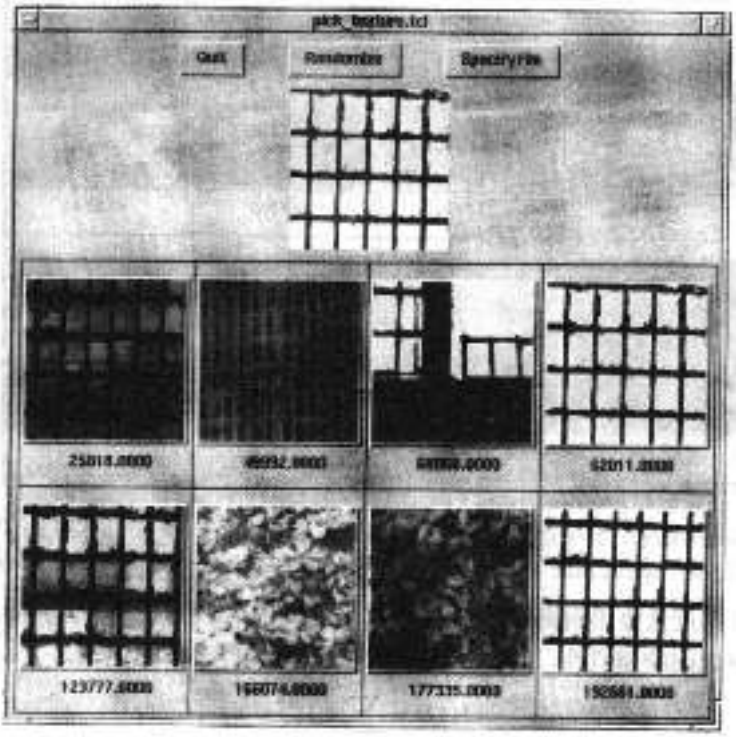


图8-6 基于纹理相似性的图像库检索结果。（来自MIT媒体实验室的VisTex数据库：http://vismod.www.media.mit.edu/vismod/imagery/VisionTexture/vistex.html）参见彩图8-6

纹理距离度量包括以下两方面：

（1）纹理的表示。

（2）基于这种表示的相似性的定义。

最常用的纹理表示是纹理描述向量，该向量表示一幅图像或图像某个区域的纹理数。纹理描述向量的实例，如Haralick的五个共生纹理特征组成的向量，以及Laws的九个纹理能量特征组成的向量。虽然纹理描述向量可用来表征整幅图像的纹理，但只适合描述单一纹理的 233
图像。对于一般图像，在像素点周围的邻域（15 × 15）内计算纹理描述向量，然后用聚类算法对像素点聚类，该算法对每个不同的纹理类别赋予一个唯一的标号。

如果得到了像素点的纹理描述向量，并将像素点标记为隶属某个纹理类别，就可以定义不同的纹理距离。最简单的纹理距离是挑选-点击（pick-and-click）距离。用户点击查询图像纹理区域的某个像素点来选择纹理，或者从预先设定的集合中选择纹理，选择的纹理通过纹理描述向量来表示，将它与图像库中的纹理描述向量进行比较，距离测度定义如下：

$$d_{\text{pick\_and\_click}}(I, Q) = \min_{i \in I} \|T(i) - T(Q)\|^2 \qquad (8\text{-}3)$$ 234

其中$T(i)$是图像I的第i个像素点的纹理描述向量，$T(Q)$是选定像素点的纹理描述向量或是要查询类别的纹理描述向量。虽然计算看似复杂，但如果用聚类过程得到的纹理类别列表来表示库中的图像，则可以避免大多数计算。对每幅库中的图像，查询图像的纹理描述向量仅仅需要与列表中的纹理描述向量做比较。加上索引使检索更快。

挑选-点击距离要求用户选定纹理，对查询图像无法进行自动运算。更一般的纹理测量根据前面讨论的颜色栅格测量推广而来。查询图像被划分成栅格，对每一个栅格计算纹理描述向量。对于库中图像进行同样的计算过程。基于栅格的纹理距离定义如下：

$$d_{\text{gridded\_texture}}(I, Q) = \sum_{g} \hat{d}_{\text{texture}}(T^I(g), T^Q(g)) \qquad (8\text{-}4)$$

其中$\hat{d}_{\text{texture}}$可以是欧氏距离或其他距离测度，纹理直方图距离可以参照颜色直方图距离进行定义。对于每个纹理类别，直方图确定了特征描述向量落在该纹理类别的像素点数目。计算纹理直方图简单有趣的方法是利用相交的直线段对。直线检测器（见第10章）用于检测图像中的直线段。找出那些相交或几乎相交的直线段对，并计算每对直线段的夹角。通过这些角度变量产生描述图像的纹理直方图。

习题8.3 纹理距离测度

从第7章选择几种不同的纹理测度，并实现为图像距离测度，可用该测度对查询图像的子图纹理与库中图像的子图纹理进行比较。然后编程实现栅格纹理距离测度，要能够调用任何一个图像距离测度对每个方格进行比较。对于一组图像，利用每种图像距离测度，针对大小不同的栅格进行比较。测试的图像库中，每幅图像都包含几片不同纹理区域。

8.4.3 形状相似性度量

颜色和纹理都反映了图像的全局属性，其中所用的距离度量方式，试图确定某幅图像中是否含有某种颜色和纹理，以及这些颜色和纹理的位置是否和查询图像中的位置对应。形状不是一种图像属性，问一幅图像的形状是什么样没有任何意义。但形状对于图像中的一个特定区域是有意义的。形状比颜色和纹理都进了一步，因为在形状相似性度量中需要进行区域识别。在许多情况下，需要手工完成这个过程，但有的领域也可以用自动分割的方法。基于

形状的检索要得到广泛应用，首先应该解决分割问题。分割问题将在第10章讨论，下面讨论形状匹配。

二维形状识别是图像分析的一个重要方向。第3章定义了图像区域的一些特征。由于这些特征是对整个形状而言的，所以又称为全局形状特征。根据全局特征可以用第4章讲的统计模式识别方法比较两个形状。形状匹配也可采用结构方法，这时形状由其基元及其空间关系进行描述。由于这种表示是一个关系图，图匹配的方法也可用于形状匹配。图匹配是种有效的方法，因为它以空间关系为基础，这种空间关系对于大多数二维变换具有不变性。但图匹配是个非常缓慢的过程，计算时间与基元个数成指数关系。基于内容的图像检索，需要快速确定图像中的形状在多大程度上与查询的形状相似。一般要求形状匹配方法具有平移不变性和尺寸不变性。有时也希望其具有旋转不变性，这样图像中的目标无论是正常方向还是发生旋转都可以识别出来。不过在图像检索中不是总要求旋转不变性，因为很多场景中的目标一般都处于正常的方向，例如室外场景中的建筑物、树木和车等。

形状度量方法大量存在于计算机视觉的相关文献中，既有粗糙的全局度量方法，它们对目标识别有所帮助但无法最终完成识别；也有非常细致的度量方法，可以寻找具有特定形状的目标。形状直方图是简单的度量方法，它能排除不可能匹配的形状，但也会返回不正确的检索结果，就像颜色直方图一样。基于边界的方法要具体些，它通过某种方法表示形状的边界并寻找具有类似边界的形状。简图匹配方法更加具体，不仅寻找与查询匹配的单个目标的边界，而且寻找与查询匹配的单目标或多目标图像区域，其中查询简图由用户绘出或由用户提供。现在我们分别讨论这几类方法。

1.形状直方图

由于直方图距离计算简便快速，并且已用于颜色和纹理匹配，自然就会想到将其应用于形状匹配。主要问题是用什么变量定义直方图。把形状看作是一个二值图像中值为1的像素点组成的区域，其他部分的像素点为0，这种直方图匹配方法是利用形状的水平和垂直投影作投影匹配。假设形状有n行m列，每一行和每一列都是直方图的一个箱格。储存在箱格中的数值就是该行或该列上值为1的像素点的个数。这样就构成了$n+m$个箱格的直方图，它仅适用于具有同样尺寸的形状。为了使投影匹配具有尺寸不变性，行箱格数和列箱格数可以保持不变，通过确定形状从左上角到右下角的箱格数，就可以保证平移不变性。投影匹配不只有旋转不变性，但也适用于旋转角度小或几何畸变小的情况。实现旋转不变性的一种方法是，求出最佳拟合的椭圆的轴（第3章中讨论过），然后旋转形状直到椭圆的主轴沿竖直方向为止。因为我们不知道哪一端是形状的上部，所以必须尝试两种可能的旋转方向。另外，如果主轴和次轴差不多等长，就要考虑四种可能的旋转方向。投影匹配已经成功地用于标志检索。

235
236

另一种方法是构造形状边界上每个像素的正切角的直方图。这种方法具有尺寸和平移不变性，但不具有旋转不变性，因为正切角是根据形状的固定方向计算出来的。解决这个问题有几种不同的方法。一种方法是与上面描述的类似，将形状根据主轴进行旋转，另一种简单方法是将直方图进行旋转。如果直方图具有K个箱格，则有K种可能的旋转。一旦算出直方图距离过大，则可快速排除不正确的旋转。或者把具有最大值的箱格作为第一个箱格，并用该值对直方图进行规范化。考虑到可能出现的噪声和畸变，需要多试几个最大箱格。

习题8.4 形状直方图

编程实现形状直方图距离测度，要求利用形状每个边界像素的正切角计算直方图，通过

旋转查询图像的直方图，使距离测度具有旋转不变性，使每个箱格都有一次机会作为第一个箱格，结果取这些旋转后直方图的最小距离，利用这个距离测度，比较你从实际图像中抽取的形状，抽取形状的方法可以是阈值化或者采用交互的方式得到。

2.边界匹配

边界匹配算法要求对查询形状和图像形状的边界进行抽取和表示。边界可以表示为一系列的像素点或者由多边形来近似。对于一系列像素点，经典的匹配方法是利用傅里叶描述子来比较两个形状。在连续数学中，傅里叶描述子是形状边界函数的傅里叶级数展开的系数。在离散情况下，形状由一系列点表示，如m个点$< V_0, V_1, \cdots, V_{n-1} >$。从这个点序列，可以得到单位向量系列：

$$v_k = \frac{V_{k+1} - V_k}{|V_{k+1} - V_k|} \tag{8-5}$$

以及累积差系列：

$$l_k = \sum_{i=1}^{k} |V_i - V_{i-1}|, \quad k > 0$$
$$l_0 = 0 \tag{8-6}$$

傅里叶描述子$\{a_{-M}, \cdots, a_0, \cdots, a_M\}$可由下式计算：

$$a_n = \frac{1}{L\left(\frac{n2\pi}{L}\right)^2} \sum_{k=1}^{m} (v_{k-1} - v_k) e^{-jn(2\pi/L)l_k} \tag{8-7}$$

可用这些描述子定义形状的距离测度。设Q是查询形状，I是要与Q进行比较的图像形状，设$\{a_n^Q\}$是查询形状的傅里叶描述子，$\{a_n^I\}$是一般图像的傅里叶描述子。傅里叶距离测度由下式表示：

$$d_{Fourier}(I, Q) = \left[\sum_{n=-M}^{M} |a_n^I - a_n^Q|^2\right]^{\frac{1}{2}} \tag{8-8}$$

如上所述，该距离仅具有平移不变性。如果需要其他不变性，该距离可与其他数值计算方法结合，这些数值计算包括缩放变换、旋转变换和求使$d_{Fourier}(\mathbf{I}, \mathbf{Q})$最小的起始点等。

如果用多边形表示边界，就可以算出每边的长度和各边之间的夹角，可用这些表示形状。形状可由一系列的连接点$< X_i, Y_i, \alpha_i >$表示，其中(X_i, Y_i)表示一对直线的交点坐标，α_i表示它们之间的夹角。用一系列连接点$\mathbf{Q} = \mathbf{Q}_1, \mathbf{Q}_2, \cdots, \mathbf{Q}_n$表示查询目标Q的边界，同样用$\mathbf{I} = \mathbf{I}_1, \mathbf{I}_2, \cdots, \mathbf{I}_n$表示图像I的边界，我们的目的是寻找从Q到I的映射。该映射使查询图像中的线段与图像I中具有近似长度的线段匹配，并且查询图像中相邻线段间的夹角α与待查图像中相邻线段间的夹角α'匹配。

另一种边界匹配技术是弹性匹配，也就是将查询形状变形使其尽可能地与待查图像形状接近。查询形状与待查形状之间的距离取决于两个方面：（1）使查询形状变形直至与图像形状最佳匹配所需要的能量；（2）变形后的查询形状与图像形状实际匹配程度的测度。如图8-7所示，是利用弹性匹配检索到的马的图像，其中查询图像是用户手工绘出的大致轮廓，用该轮廓表示要检索的形状。

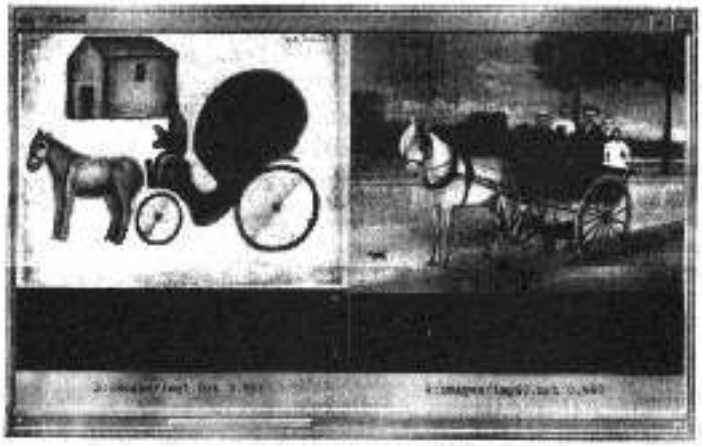
习题8.5 边界匹配

尽管有许多边界形状匹配算法，但它们在基于内容的图像检索方面并未得到普遍应用，你能解释为什么吗？

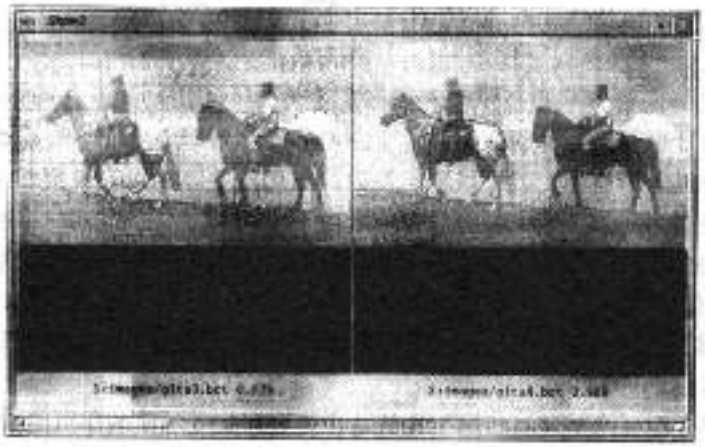
237



a) 用户绘制的查询形状



b) 两幅检索出来的图像



c) 另外两幅检索出来的图像，包含两匹马

图8-7 弹性匹配图像检索结果（Alberto Del Bimbo提供）

3. 简图匹配

简图匹配系统，允许用户输入一幅包含图像中主要边缘的简图，然后寻找含有相匹配边缘的彩色图像或灰度图像。在ART MUSEUM系统中，数据库包含名画的彩色图像，彩色图像经过下面的预处理得到中间形式的图像，称为抽象图像。

1. 应用仿射变换将图像缩小到预定的大小，如64×64个像素点，并用中值滤波来去除噪

声。得到的结果是规范化的图像。

2. 利用基于梯度的边缘检测算法检测边缘。边缘检测分成两步进行：首先，基于梯度均值和标准差取全局性域值得到全局性边缘；其次，根据局部计算的域值从全局边缘中选出局部的边缘。得到的结果称为精细边缘图像。

3. 对精细边缘图像进行细化和收缩，最终的结果称为抽象图像。这是原始图像比较清晰的边缘简图。

当用户输入简图进行查询时，它也要经过大小规范化、二值化、细化、收缩的转换过程。这些处理的结果称为线性简图。现在要求线性简图必须与抽象图像匹配。匹配方法是基于相关的算法。两幅图像被分成栅格，对查询图像的每个栅格块，计算它与库中图像对应的栅格块之间的局部相关性。为了使算法更稳健，对库中图像的栅格位置进行几次移位，分别计算相关性，其中的最大相关值作为查询结果。最后的相似性测度是所有局部相关性之和。距离测度与相似测度成反比。采用前面的符号，表示如下：

$$d_{sketch}(I, Q) = \frac{1}{\sum_g \max_x [\hat{d}_{correlation}(shift_x(A^I(g)), L^Q(g))]} \tag{8-9}$$

其中$A^I(g)$指库中图像I的抽象图像栅格块g，$shift(A^I(g))$指同一幅抽象图像栅格块g的移位栅格，$L^Q(g)$指查询图像Q的线性简图栅格块g。

习题8.6 简图匹配

根据ART MUSEUM的规则，设计并实现一种简图匹配的距离测度。根据用户的简图，检索一组图像。

8.4.4 目标检测及空间关系度量

虽然第一个图像搜索引擎提供了多数距离测度，涉及颜色、纹理和形状等，但这些并不是多数终端用户想要的。终端用户倾向于查询包含某类实体的图像，这可能是某种特殊的目标，比如人物或狗；或者可能是抽象的概念，比如快乐或贫穷。第一代目标识别系统，用到了象人脸、人体和马这样的目标。为使图像检索能够使用这方面的技术，要求对目标识别作进一步的研究。

1. 人脸检测

人脸检测非常重要，因为它可以帮助我们检索包含人物的图像。当然人脸检测也非常困难，因为图像中的人脸可能是任意大小、任意位置，正面或其他角度，并存在不同的肤色。卡内基梅隆大学研发的系统，采用了多分辨率的方法来解决大小问题。它将彩色图像转化成灰度图像以避免肤色的差异，再对亮度规范化，通过直方图均衡化扩展灰度级范围。然后采用神经网络分类器进行识别。该分类器事先采用16 000幅人脸和非人脸图像进行了训练。神经网络的输入是20 × 20=400图像像素点的亮度值，输出是人脸或非人脸两大类别。虽然难以从神经网络中抽取一种准确的算法，敏感性分析表明对网络行为影响最大的是20 × 20图像中的眼睛，其次是鼻子，再次是嘴。这种方法效果不错，可检测出多数的正面人脸，当然不是全部，如图8-8所示。该方法无法推广到其他目标，除非在它们的灰度图像中具有和眼睛、鼻子、嘴同样排列方式的特殊模式。

238 ~ 240

2. 人体检测

另一种检测目标的方法是，在图像中寻找具有与该目标相关的颜色和纹理区域，该领域

的研究最初是为了检测裸体图像，用于滤掉查询结果中的色情图片。由Fleck、Forsyth和Bregler（1996）提出的方法包括两个步骤：（1）找到可能的大块肉体区域；（2）对这些区域聚类，检测可能的人体。



图8-8 基于神经元网络检测人脸

人体过滤器在像素级上进行。通过如下变换将初始的RGB图像变换到以10为底的对数空间：

$$I = L(G) \tag{8-10}$$

$$R_g = L(R) - L(G) \tag{8-11}$$

$$B_y = L(B) - \frac{L(G) + L(R)}{2} \tag{8-12}$$

其中$L(x)$由下式定义：

$$L(x) = 105 \log_{10}(x + 1 + n) \tag{8-13}$$

n表示[0, 1]内的随机噪声。I分量用来产生如下的纹理幅度图：

241

$$texture = med_2(|I - med_1(I)|) \tag{8-14}$$

其中med_2和med_1分别是两个不同尺寸的中值滤波器（med_2是med_1的1.5倍）。纹理幅度图用来检测低纹理的区域，因为图像中的皮肤很可能具有光滑的纹理。

用色调和饱和度选择那些颜色与皮肤匹配的区域。在计算前R_g和B_y图像也经过中值滤波。从以10为底的对数空间到色调和饱和度的转换如下：

$$hue = atan(R_g, B_y) \tag{8-15}$$

$$saturation = \sqrt{R_g^2 + B_y^2} \tag{8-16}$$

如果像素落在以下两个区间之一，则标记为皮肤像素点。

1. texture < 5,110 < hue < 150,20 < saturation < 60

2. texture < 5,130 < hue < 170,30 < saturation < 130

注意有关常量与原始工作有关，用户可根据不同的数据集和实际情况进行修改。

皮肤图是一个二值数组，其中值为1的像素是皮肤像素，值为0的是非皮肤像素。对该数组进行形态闭运算处理，将得到更清楚的结果。一旦找到图像中的皮肤区域，可再作如下检

查：（1）肉体区域如果足够大（占图像的30%），可认为是色情图片；（2）区域之间具有合适的空间关系，可认为是人体部分。

习题8.7 人体和人脸检测

设计人体检测器寻找肉色区域。选出规定大小以及更大的区域，查找面部特征的证据，特别要按眼睛、鼻子和嘴的顺序查找。基于找到的特征，对每个区域计算属于人脸的概率。

3. 空间关系

一旦识别出目标，它们的空间关系也可以确定，可用某种形式化的方法进行查询，这需要一组具有预定空间关系的命名目标。这是图像检索过程的最后一步。参考伯克利大学的Forsyth等人（1996）的近期工作以及圣巴巴拉大学的Ma和Manjunath（1997）所做的类似工作。研究人员成功利用颜色和纹理将图像分割成区域，这些区域对应着目标或场景中的背景。对于与众不同的老虎和斑马，它们具有特殊的颜色和纹理模式，就可以采用这种方式找到目标。像丛林、天空或海滩这样的背景也可以分割出来。图8-9显示这种分割的实例。原始彩色图像如左图所示，区域分割后的图像如中间图所示。右图是符号表示方式，用椭圆表示感兴趣的区域。这种表示方法可用来构造一个关系图，它的节点是区域类别，连接边表示空间关系。可用相关匹配技术来建立图像检索的关系距离测度。虽然在此我们没有对该系统进行更深入的讨论，但Del Bimbo开发的检索系统，其输入就是这种表示方式。该系统允许用户选中具有一定空间关系的图标，把图标放在查询屏上作为查询输入，系统返回具有这些关系的对应目标的图像。图8-10是空间查询系统的检索实例。 242



图8-9 从图像中抽取目标和空间关系并用于检索。（原图像经Corel Stock Photos许可）参见彩图8-9

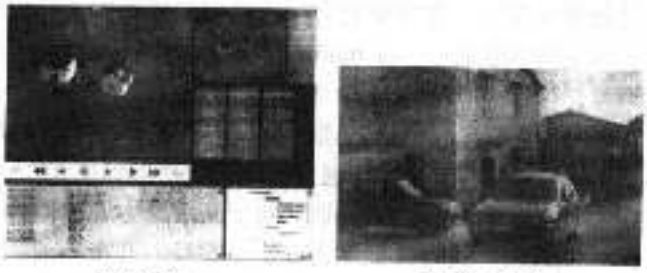


图8-10 空间关系查询的结果（图片由Alberto Del Bimbo提供，经IEEE授权，翻印自"Symbolic Description and Visual Querying of Image Sequences using Spatio-Temporal Logic," by A. Del Bimbo, E. Vicario, D. Zingoni, *IEEE Transactions on Knowledge and Data Engineering*, vol.7, no.4, Aug. 1995. © 1995 IEEE）

习题8.8 根据目标和目标关系进行检索

找到或编写程序，基于颜色和纹理（如果纹理可用）把彩色图像分割成区域。用一组训练图像测试程序，其中每类目标，如老虎、天空、丛林在几幅图像中都存在。用已知区域的颜色和纹理特征训练分类算法，编写程序，通过分割算法和分类器对输入图像生成标记区域，并计算区域对之间上、下、左、右和邻接的空间关系，然后编写交互式前端程序，允许用户
243 输入一个图结构，其中节点是训练集的目标，连接边是要求的关系。程序应返回满足查询条件的所有数据库图像。

8.5 数据库组织

和其他大型数据库一样，大型图像数据库的数据量很大，查询图像时遍历整个图像库是不合适的。如果希望对于每次查询，都只需搜索库中的部分图像，那么图像库中的图像必须按一定规律进行组织和索引。在多数关系数据库系统中，有很多标准方法检索数字数据和文本数据。空间数据也有一套检索方法，这些方法也在使用，例如地理信息系统。当前的研究系统，正在研究如何为基于内容的图像检索系统建立图像索引。

8.5.1 标准索引

在大多数关系数据库中，用户可指定一个属性，根据这个属性建立索引。通常这个属性是与每个数据记录相关的重要键值。例如，如果某个数据库包含着某公司员工的记录，那么社会安全号码就是用户的一个属性，可用作数据的索引。由于每人都只有一个社会安全号码，该属性就称为主键（primary key）。如果该属性数据经常被其他属性访问，例如员工的姓，那么就可对该属性本身再建立一个索引。

在关系数据库中，一个索引就是一个数据结构，根据索引系统能够找到给定的属性值，并迅速找到数据库中具有该属性值的所有记录。在关系数据库中有两种常用的索引类型：散列索引和B-树索引。散列索引可以快速找到具有查询属性值的数据记录。B-树或B^+-树索引能够快速查找属性值落在查询指定范围内的记录。

1. 散列索引

散列索引应用散列表理论访问数据库中的大量记录。假设存在一个庞大的键值集合，而只有一小部分同时出现在数据库中，假设数据库是包含N个记录的文件，每个记录包含多个字段，其中一个字段存放键值。散列索引的访问机制是通过散列函数实现的，散列函数把每个键值与文件内的一个地址对应起来，文件中包含（或指向）该特殊键值的数据库记录。如果键值是数字，一个简单的散列函数是$f(x) = x \bmod N$（x对N取模），也就是把x除以N所得的余数作为要访问记录的记录号。图8-11显示具有数字键值的数据库散列索引。这个散列表有0到9十个位置（实际散列表会比这大的多），散列函数取$f(x) = x \bmod 10$（x对10取模）。如图所示，
244 当检索所有键值为45的记录时，在散列表中，通过散列函数映射到位置5。

如果每个键值对应散列表中的一个位置，那么对于任何给定的键值，访问散列表所用的时间就是一个常量。但一般来说，这种情况不会发生。相反，几个不同的键值很可能对应到相同的位置，称这种现象为冲突（collision）。解决冲突的方法，可以在所有数据结构的教科书中找到。图中所示的方法是采用一个包含所有记录的链表，其中的键值对应散列表的同一位置，最后的结果是，当访问数据库时都要进行一些搜索，而不是简单的直接读取。但如果散列函数合适，散列表又不是很满，访问时间仍然近似为常数。散列索引的性质，决定了它最适于查询准确约束的情况，即$KEY = VALUE$，而不适于查询键值属于某个范围的情况。

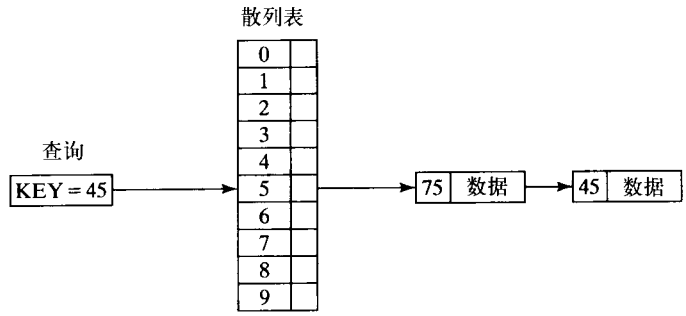


图8-11 散列索引实例

2. B⁺-树索引

B-树和B⁺-树都是平衡的多路搜索树，可用于建立索引，并适用于范围查询。B-树的内部节点和叶子节点都带有键值和数据，而B⁺-树只有叶子节点带有数据。下面我们将集中讨论B⁺-树，因为数据库中的数据和索引应该分开。

p叉搜索树的每个节点最多含有$p-1$个键值和p个指针。B⁺-树的内部节点和叶子节点具有不同的形式。B⁺-树的内部节点遵从如下约束条件:

（1）它的形式为$\langle P_1, K_1, P_2, K_2, \cdots, P_{q-1}, K_{q-1}, P_q \rangle$，其中$P_i$代表指向其他节点的指针，$K_i$代表键值。直观上，$P_{i-1}$所指向的子树包含的所有键值都小于等于$K_i$，而$P_i$所指向的子树包含的所有键值都大于$K_i$。

（2）非根节点至少有（$p/2$）个子树指针。

（3）根节点至少有2个子树指针。

B⁺-树的叶子节点遵从如下约束条件:

（1）它的形式为$\langle K_1, Pr_1, K_2, Pr_2, \ldots, K_{q-1}, P_{rq-1}, P_{next} \rangle$，其中$K_i$代表键值，$Pr_i$是一个数据指针，$P_{next}$指向下一个叶子节点。

（2）Pr_i指向一个键值为K_i的记录；当索引的搜索字段不是文件的键值时，Pr_i则指向符合条件的多记录数据块。 245

（3）每个叶子节点有（$p/2$）个值。

（4）所有的叶子节点在树的同一层次上。

B⁺-树内部节点的叉数p可能和叶子节点不同，目的是为了让每种类型的节点与物理存储块相适应，这些物理存储块是指从磁盘传输到计算机内存的单位数据量。

为在B⁺-树中找到给定的键值或值的范围，查询系统从根节点开始搜索。系统将节点读入内存，对该节点的键值进行二分搜寻。如果找到该节点处的两个相邻的键值，而且这两键值之间包含给定的键值，那么这两键值之间的指针所指向的子树将包含给定的键值或给定键值范围的最小值。如果给定的键值小于节点中的第一个键值，该键值左边的指针指向正确的子树。同样，如果给定的键值大于节点中的最后一个键值，该键值右边的指针指向正确的子树。一旦确定了合适的子树，它就成为进一步搜索的树根节点。检索系统重复进行这些运算直到得到叶子节点为止。

在叶子节点中，再次执行二分搜寻以找到给定的键值或键值的起始值K_i。相关指针P_i指向包含该键值的数据记录。如果仅查找一个键值，则现在可以返回相关记录。如果这是一个范围搜索，数据记录的P_{next}指针可用于寻找余下的数据记录，直到遇到给定键值范围的末端为止。

图8-12是一个B⁺-树实例，通过数字键值和图像数据检索数据库记录。内部节点用实线矩形框表示，叶子节点用虚线矩形框表示。根节点指向三个不同的子树：键值小于等于100的子树；键值介于100和200之间的子树；键值大于200的子树。如图所示，键值介于100和200之间的子树，其根节点指向两个叶子节点：一个键值小于110的叶子节点和一个键值介于110和150
246 之间的叶子节点。叶子节点包含实际的键值和相关的图像数据文件。

B⁺-树灵活有效，广泛应用于关系数据库系统。它们可用于图像数据库系统来检索与图像有关的单个数值或文本字段。它们不适用于检索多维数据。

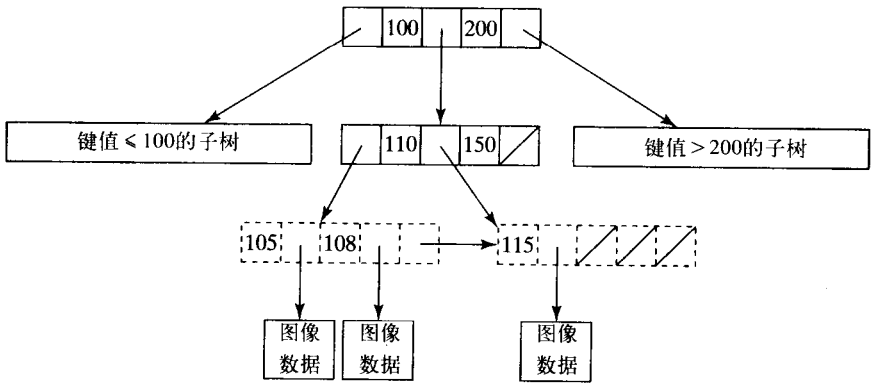


图8-12 B⁺-树索引实例

8.5.2 空间索引

空间信息系统包含的数据是多维的。针对空间索引已经提出了许多数据结构。四叉树是一种分层结构，每个节点具有4个分区，即在树的每一层将2D数据的搜索空间分成4个子区(quadrant)。四叉树可用于表示二值图像中的区域。K-d树是对二叉搜索树的扩展，支持对k维数据的搜索。R-树是B-树向更高维数的扩展，适用于各种空间信息系统。在R-树中，用一个n维的最小边界矩形（MBR）检索一个数据对象，它限制对象所占据的空间。每个实际数据对象用唯一的标识符（ID）表示。R-树的叶子节点包含数据对象的ID。内部节点包含形如MBR、CHILD的实体，其中CHILD是指向R-树中更低层节点的指针，MBR覆盖了更低层节点实体的
247 所有矩形。图8-13显示2D对象集的R-树索引。矩形的分布取决于树构造的顺序以及所用的R-树构造算法。R-树存在其他一些变形，如R⁺-树和R*树。

8.5.3 基于内容的多距离测度图像索引

上述方法利用简单的距离测度进行图像检索，其中只使用单一特征或者少数几个特征，不适用于大型通用系统。大型系统允许用户选择多个基本距离测度以及集成的方法。这种系统需要更灵活的组织和索引形式。如果基本度量采用公制单位，那么三角不等式可以提供非标准的索引方法。三角不等式说明，如果Q是一幅查询图像，I是数据库中的一幅图像，K是特意选择的关键图像，那么对于任意图像距离测度d，下列关系成立：

$$d(I, Q) \geqslant |d(I, K) - d(Q, K)|$$

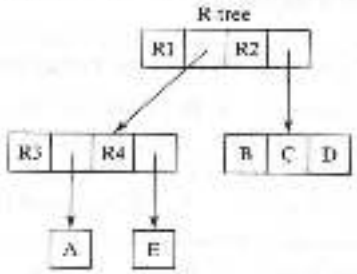
将数据库中的图像和查询图像都与第三幅关键图像做比较，可以得到查询图像与数据库图像之间距离的下界。

首先考虑单个距离测度d的情况。从数据库中选择一组关键图像，直观上，这些图像应代

表数据库中的不同场景类别。查询图像Q与每个关键图像$K_1, K_2, \cdots, K_M$比较，得到一组距离$d(Q, K_1), d(Q, K_2), \cdots, d(Q, K_M)$。假设用户已经规定，要返回所有离查询图像$Q$的距离小于$T$的图像，那么对于每个关键图像$K_i$，所有满足下式的图像$I$就可以立即排除，因为$d(I, Q)$肯定大于$T$。

$$|d(I, K_i) - d(Q, K_i)| > T$$

三叉树数据结构就利用这种方法，并通过与查询图像的直接比较排除数据库中的大多数图像。对这项技术进行扩展，可动态定义距离测度，该距离测度是基本距离测度的线性组合或者布尔组合。



索引矩形集

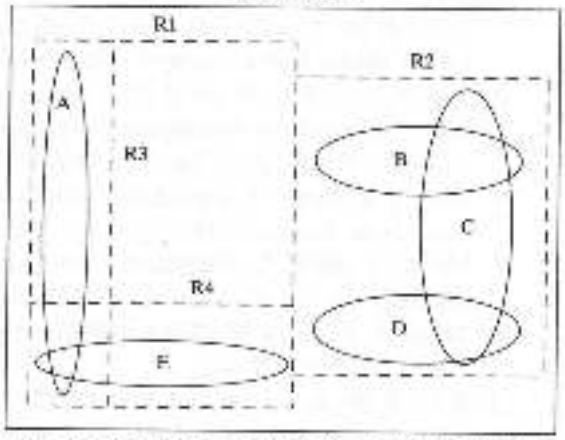


图8-13 2D数据的R-树索引实例。数据表示数据对象，索引矩形用虚线表示。矩形R1进一步分成矩形R3和R4，每个矩形都包含一个数据对象。矩形R2未作进一步的分解，包含三个数据对象：B、C和D

习题8.9 索引

假设一组图像根据Laws的纹理能量测度建立索引，解释如何利用R-树作为系统的索引机制。

8.6 参考文献

基于内容的图像检索是计算机视觉技术相对较新的一个应用领域。第一个通用商业化系统，QBIC，是由Niblack及其所在的研究小组（1993）在IBM Almaden研发出来的。它利用颜色、纹理和形状特征进行检索。另一个早期系统是Kato等人（1992）的ART MUSEUM系统，它基于用户画出的简图从图像库中检索绘画。Minka和Picard（1996）提出如何通过用户选择（相关反馈）来提高检索系统的性能。Del Bimbo、Pala和Santini（1994）研究出的系统通过弹性匹配来检索目标图像，其中的目标与用户定义的形状相匹配。

Fleck、Forsyth和Bregler（1996）提出了一种算法，利用颜色和区域间的空间关系来查找图像中的裸体人物。后来对这些工作进一步扩展，用于查找其他目标，如马。Rowley、Baluja和Kanade（1996）提出一种利用神经网络检测人脸的方法，该神经网络用上千幅图像训练过。伯克利大学的Carson、Belongie、Greenspan和Malik（1997）以及圣巴巴拉大学的Ma和Manjunath（1997）提出了基于区域的检索方法，利用基于颜色和纹理的分割算法来产生感兴趣的区域。Baeza-Yates及其小组（1994）以及Berman（1994）都针对近似匹配提出了树型结构。Berman和Shapiro（1999）对这些技术进行扩展并发出一个图像检索系统。Samet的著作（1990）中描述了通用的空间数据结构。

1. Baeza-Yates, R., W. Cunto, U. Manber, and S. Wu. 1994. Proximity matching using fixed queries trees. *Combinatorial Pattern Matching*. Springer-Verlag, New York, 198–212.
2. Berman, A. P. 1994. A new data structure for fast approximate matching. technical

report 94-03-02. Department of Computer Science and Engineering, University of Washington.

3. Berman, A. P., and L. G. Shapiro. 1999. A flexible image database system for content-based retrieval. *Comput. Vision and Image Understanding,* v. 75(1–2):175–195.
4. Carson, C., S. Belongie, H. Greenspan, and J. Malik. 1997. Region-based image querying. *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries,* San Juan, Puerto Rico.
5. Del Bimbo, A., E. Vicario, and D. Zingoni. 1993. Sequence retrieval by contents through spatio temporal indexing. *IEEE Symp. Visual Lang.,* 88–92.
6. Del Bimbo, A., P. Pala, and S. Santini. 1994. Visual image retrieval by elastic deformation of object sketches. *IEEE Symp. Visual Lang.,* 216–223.
7. Fleck, M. M., D. A. Forsyth, and C. Bregler. 1996. Finding naked people. *Proc. Euro. Conf. Comput. Vision.* Springer-Verlag, New York, 593–602.
8. Forsyth, D. A., J. Malik, M. M. Fleck, H. Greenspan, T. Leung, S. Belongie, C. Carson, and C. Bregler. 1996. Finding pictures of objects in large collections of images. *Proc. 2nd Inter. Workshop on Object Representation in Comput. Vision* (April 1996).
9. Kato, T., T. Kurita, N. Otsu, and K. Hirata. 1992. A sketch retrieval method for full color image database. *11th Inter. Conf. Pattern Recog.,* 530–533.
10. Ma, W. Y., and B. S. Manjunath. 1999. Netra: a toolbox for navigating large image databases. *Multimedia Systems,* v. 7(3):184–198.
11. Minka, T. P., and R. W. Picard. 1996. Interactive learning with a society of models. *Proc. CVPR-96,* 447–452.
12. Niblack, W., and others. 1993. The QBIC project: Querying images by content using color, texture, and shape. *SPIE Proc. Storage and Retrieval for Image and Video Databases,* 173–187.
13. Rowley, H., S. Baluja, and T. Kanade. 1996. *Human Face Detection in Visual Scenes.* Carnegie-Mellon University, Pittsburgh, PA.
14. Samet, H. 1990. *The Design and Analysis of Spatial Data Structure.* Addison-Wesley, Reading, MA.

第9章　二维运动分析

图像序列能够反映场景的动态变化。人们通过视频观看行家的动作，就可以学习打高尔夫球；或者每隔几小时摄取一次植物根部的图像，然后观察得到的图像序列，可以更好地理解根部的生长情况。场景中的物体运动、观测器运动、或者物体与观测器同时运动，是产生图像运动现象的原因。图像序列中的运动特征，能够用来检测其中的运动目标，或计算目标的运动轨迹。在观测器运动而环境静止的情况下，又可以通过图像中的变化计算观测器在环境中的运动情况。

同样，图像中的像素变化包含着重要的特征，这些特征可用于目标检测与识别。图像运动能够揭示物体的形状以及其他特性，如运动速度或功能。对物体随时间运动的情况进行分析，可以说是我们研究的最终目标，例如对交通流量进行控制，或者对装有新假肢人员的步态进行分析。目前人们保存了大量的视频信息，记录着各种事件和场景结构。有必要寻找合适的分割技术，把这些图像序列变成有意义的事件或场景，以方便访问、分析和修改。

本章主要讨论基于二维图像和视频序列的运动检测，以及图像特征的抽取方法。针对前面提到的应用问题，探讨有关解决办法。第13章讨论如何基于二维图像进行三维结构和运动的分析。

9.1　运动现象及应用

我们需要对图像序列中的各种运动情况进行分析，并对有关应用问题进行讨论。要研究的内容不只是检测出一个运动目标，还要研究多运动目标情况下的运动分析和形状分析。

运动大体划分为以下四种情况，其中术语*摄像机*（camera）与术语*观测器*（observer）可以互换使用。

- 摄像机静止，单个目标运动，背景不变。
- 摄像机静止，多个目标运动，背景不变。
- 摄像机运动，场景不变。
- 摄像机运动，多个目标运动。

最简单的情况是，传感器是静止的，而且目标所在的背景也是不变的。目标在背景中的运动，引起图像中与之对应部分的像素发生变化。对这些像素进行分析能够揭示目标的形状及其运动的速度和路径。这种传感器一般用在安全防护场合。家庭应用中，常常利用这种传感器检测快速运动的目标，这种快速运动可能是户主回到家中或着是陌生人闯入室内所产生的，这时电灯就会自动打开。这种简单的运动传感器，也可用在制造业，用来检查工作室内某个零件是否存在；或用在交通控制系统中，检测车辆的运动。

静止摄像机捕捉的数据，也可用于分析一个或多个目标的运动情况。为得到目标运动的轨迹或路径，需要随时对运动目标进行跟踪，这反过来又可以揭示目标的行为。比如利用摄像记录可对进入营业大厅或工作室人员的行为进行分析。同时采用几台摄像机，能够得到同一目标不同视点的图像，并据此计算出它的三维路径。在对运动员或病人的康复情况进行分析时，常常要用到多台摄像机。目前正在开发的一个系统中，对网球比赛中运动员和网球的

轨迹进行跟踪，并对比赛的各要素进行分析。

即使外界三维环境不变，摄像机自身的运动也会引起图像发生变化。这种运动模式有以下几方面的优点。首先，比单一视点有更大的观察范围，比如摄像机扫视时，就能得到关于场景的全景视野；其次，能够计算目标的相对深度，因为近处目标的图像要比远处目标的图像变化快；再次，能够感知或测量近处目标的三维形状，多视点观察可采用与双目立体视觉类似的三角计算。处理与分析视频或电影内容时，当摄像机扫视或者变焦时，需要及时进行点的检测。这时我们可能对景物的内容不感兴趣，而对观察景物的方式感兴趣。

最困难的运动问题是，不仅传感器在运动，而且场景中的多个目标也在运动，根本无法确定背景中的哪些部分是不变化的。移动机器人在繁忙的交通要道中行驶就属于这种情况。另一种有意思的情况是，为了跟踪工作室内不同的运动目标，几个摄像机之间要保持联络，使所得的图像之间存在一定的对应关系。

下一节介绍各种图像分析方法，主要针对含两幅以上图像的图像序列，目的是对运动引
252 起的图像变化进行检测，或者对物体本身及其运动进行分析。

习题9.1

寻找控制自动开灯的运动检测器。这些设备一般安装在车库或住所的入口处。验证当你快速进入室内时电灯会自动打开。(a) 如果你非常慢地移动，看看运动检测器是不是没有反应？(b) 这说明运动检测器是如何工作的？(c) 这与电影侏罗纪公园中的霸王龙雷克斯有相关的地方吗？

9.2 图像相减

在第1章引入了图像相减的概念，用于检测背景不变情况下的运动目标。假设视频摄像机以每秒30帧摄取传送带图像，其中传送带背景为黑色。如果较亮的物体在摄像机视野前移过，物体的前边与后边在相邻的两帧图像中只有几个像素的位移。如果相邻的两帧图像相减，即I_{t-1}减去I_t，这些边将保留下来，并且明显区分于背景值。

图9-1显示工作室监视系统中相邻两帧图像的差分结果，两幅图像之间间隔几秒钟。这个例子中的背景图像通过大量的视频帧算出，背景是不均匀的。一个人进入工作室，引起图像中的某个部分发生变化，通过图像相减可以检测出这个人的存在。如图9-1所示，图中的边界框内部是检测到的变化区域。进一步分析该边界框，可以得出目标的形状甚至所属的类型。图9-1中中间的那幅图实际上显示出三个不同的变化区域，分别是：(1) 人，(2) 被人打开的门，(3) 计算机显示器。可以事先为监视系统提供目标的位置信息，甚至可以提前确定要重点监视或忽略的部分。例如，应该对门进行重点监视，而忽略显示器部分的变化。该技术可
253 用于停车场监视与记录、街道交通流量监视、室内人员监视等。

利用相减法检测图像变化的步骤见算法9.1，其中涉及到的运算方法在第3章中已经给出。

算法9.1 利用图像相减检测两幅图像之间的变化

输入$\mathbf{I}_t[r, c]$与$\mathbf{I}_{t-\Delta}[r, c]$，是两幅黑门图像，时间间隔$\Delta$秒。

输入τ是亮度阈值。

输出$\mathbf{I}_{out}[r, c]$是二值图像，$\mathbf{B}$表示边界框的集合。

1.对于两幅输入图像上的所有像素$[r, c]$，

如果（$|I_t[r,c]-I_{t-\Delta t}[r,c]|>\tau$）则$I_{out}[r,c]=1$

否则$I_{out}[r,c]=0$。

2.抽取I_{out}上的连通成分。

3.去掉噪声小区域。

4.用小圆盘形的结构元对I_{out}进行闭运算，与邻域融合。

5.计算像素变化区域的边界框。

6.返回$I_{out}[r,c]$及像素变化区域的边界框集B。



图9-1 （S.-W.Chen提供）

（左图）一个人在工作室里

（中图）图像相减结果，存在三个变化区域：背景被人挡住的区域、门和显示器处

（右图）由人引起的变化更显著，其他两处变化是预料之中的，因此可忽略掉

习题9.2

该习题需要一台工作站，工作站上配有摄像机及图像存取软件，编写程序，对你的桌子上方进行监视（工作站附近）。程序应能捕捉图像，并计算每幅图像的直方图，每当直方图有显著变化时，系统应发出警报。针对各种静态场景，以及从桌子上取放物体时，测试所编程序的运行情况。

9.3 计算运动向量

三维场景中点的运动，引起投影到图像上的对应点的运动。图9-2是三种典型情况。静态摄像机焦距减小或焦距不变而摄像机逐渐远离景物时，可引起图像收缩。其中沿光轴方向有一点的图像不发生变化，该点称为收缩中心（focus of contraction）。静态摄像机焦距增加或摄像机逐渐接近景物中心（focus of expansion）可引起图像膨胀，其中膨胀中心的图像不发生变化。摄像机扫视或我们的头部转动，会引起三维点在图像上的对应点产生移动，如图9-2中的右图所示。

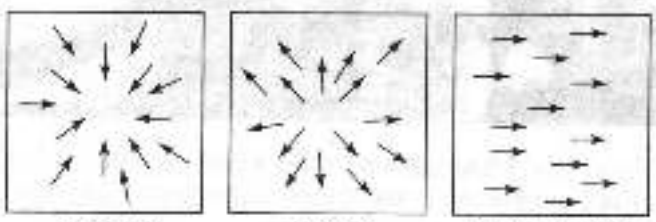


图9-2 焦距变化和扫视引起图像特征的变化。焦距变化效果与我们远离和接近景物时所看到的情景类似。镜头扫视与我们转动头部时所看到的情景类似

定义70 三维空间中点的运动，投影到图像中对应一个二维向量，由这些二维向量构成的二维阵列称为图像**运动场**（motion field）（参见图9-2）。图像运动向量是指，
254 三维空间中点的运动投影到图像空间所对应的位移。对应同一个三维运动点，在t时刻的图像点到$(t+\Delta)$时刻的图像点之间，形成运动向量，或者说运动向量与t时刻的瞬时速度估计值对应。

定义71 当传感器接近目标时，图像上有一个特殊的点，所有的运动场向量都从该点发出，这个点就是**膨胀中心**。膨胀中心一般对应着传感器前移时所注视的三维点。当传感器远离目标时，图像上也有一个特殊的点，所有的运动向量都会聚于该点，这个点就是**收缩中心**。收缩中心一般对应着传感器后退时所注视的三维点。

运动场的计算不仅能用于目标识别，又能用于运动分析。为了计算运动向量，一般要附加如下两个约束条件之一，但约束性不是很强。一是估计三维点P在(t_1, t_2)期间的运动特性时，假设该点及其周围的亮度基本保持不变；或者是假设在(t_1, t_2)期间，图像上物体边缘处的亮度差别基本保持不变。

定义72 对应点附近的图像亮度相对不变时所得到的运动场称为**图像流**（image flow）。

下面给出两种计算图像流的方法。首先我们先介绍一个基于运动场技术的视频游戏。

9.3.1 Decathlete游戏

在日本相模原市的三菱电子和在马萨诸塞州剑桥市的三菱电子研究室的研究人员，用运动分析方法去控制Sega Saturn Decathlete游戏。他们采用一台低分辨率摄像头，用计算图像流的方法取代了键盘。游戏中实际运动员的手臂运动控制着仿真运动员的运动。在例子中，仿真运动员正与另一位进行跨栏比赛。在图9-3中，左边是运动员，正通过手臂做跑步动作。他
255 运动得越快，仿真运动员跑得也越快。运动员举起双拳表示跳越时，仿真运动员也要适时地跳过栏架。在图9-3的右边，显示器中显示的是仿真比赛的现场情况。图的右下角是一台摄像机，用来捕捉实际运动员的手势。图中的另外两个人正在欣赏这个操作过程。



图9-3 左边的人通过手臂做出跑步动作控制跨栏比赛。右边显示仿真比赛的现场。右下角是一台摄像机，用来捕捉运动员的手势运动，据此人控制仿真运动员的跑步速度和跳越动作（参考IEEE Computer Graphics,vol.18, no.3,May-June,1998，IEEE授权）

图9-4用于控制跨栏比赛的运动分析示意图，图9-4a是运动分析的简单示意图，图9-4b是

对a图的注解图。注意a图的左上角是摄像机摄取的运动员做出跑步动作的一帧视频，而a图的左边靠中间位置是从多帧视频中抽取的运动向量。a图的左下角是根据视频得出的在水平方向上的平均运动轨迹，a图的中间是垂直方向上的平均运动轨迹。



图9-4 控制跨栏比赛的运动分析方法。在a图中，左上角显示运动员做出跑步动作的一帧视频；左边靠中间位置是从多帧视频图像中抽取的运动向量；中间的垂直运动曲线表示跨栏动作（参考IEEE Computer Graphics, vol.18, no.3,May-June,1998，IEEE授权）

当运动员“跑”和“跳”的时候，摄像头必须能看到运动员的双下。跑步动作引起水平方向的平均运动发生变化。变化的频率说明运动员的速度有多快，并据此控制仿真运动员的速度。跳越动作造成垂直方向的平均速度大于一个阈值，于是向仿真运动员发出跳越命令。这种方法的空间分辨率很低但时间分辨率很高。

Decathlete游戏中用到的简单运动分析方法，也许能作为一般的计算机手势接口。例如，未来计算机系统的输入方式允许使用美国手语，或者是小语种手语。

9.3.2 点对应

通过确定t_1时刻和$(t_1+\Delta)$时刻两幅图像上的对应点，可以计算出稀疏运动场。选用的这些点要有一定的特殊性，要在两幅图中都能识别出来而且能确定它们在图像中的位置。无论是彩色图像还是黑白图像，都应选择角点或高度兴趣点（high interest point）。在进行彩色图像分割时，可能要用到持续运动区域的中心。检测角点时可以用模板法，如Kirsch边缘算子，或者Frie-Chen算子集中（第5章）的波纹模板，也可用兴趣算子（interest operator）。该算子计算以**P**为中心的邻域在垂直、水平和两对角线方向上的亮度变化，只有当这四个变化值中的最小值超过一个阈值时，**P**点才能做为一个兴趣点。具体参见算法9.2。另一种基于纹理的算子作为习题9.3的设计内容。 256

算法9.2 检测感兴趣的图像点

```
procedure detect_corner_points(I, V);
{
    \I[r, c]是MaxRow行，MaxCol列的输入图像。
    \V是从I中搜索出的兴趣点的集合，是算法的输出。
    \t是兴趣算子的阈值。
    \w是兴趣算子邻域宽度的一半。
```

```
    for r := 0 to MaxRow−1
  for c := 0 to MaxCol−1
        {
        if I[r, c] is a border pixel then break;
        else if (interest_operator (I, r, c, w) ≥ τ1)then add
        [(r, c), (r, c)]to set V;
    \\第二个(r,c)保存后面发现的向量前端。
        }
}
real procedure interest_operator(I, r, c, w)
{
    \\w是算子窗口宽度的一半。
    \\参见习题9.3中的纹理兴趣算子。
    v1 := variance of intensity of horizontal pixels I1[r, c−w] ... I1[r, c+w];
    v2 :=variance of intensity of vertical pixels I1[r−w, c] ... I1[r + w, c];
    v3 := variance of intensity of diagonal pixels I1[r−w, c−w] ... I1[r + w, c + w];
    v4 := variance of intensity of diagonal pixels I1[r−w, c + w] ... I1[r + w, c−w];
    return minimum{v1,v2,v3,v4};
}
```

习题9.3 纹理兴趣算子

试验下面的兴趣算子。它基于$n \times n$邻域的纹理。首先用3×3或2×2模板计算整幅输入图像的梯度幅度。然后，把幅度图阈值化产生二值图像。只有当二值图像中$B[r, c]$的$n \times n$的邻域在四个主要方向上的变化显著时，原始图像的像素点$[r, c]$才是兴趣点。在方向$[\Delta r, \Delta c] = [0, 1]$、$[1, 0]$、$[1, 1]$、$[1, -1]$上的变化量，等于以$B[r, c]$为中心的$n \times n$邻域内所有像素的$B[r, c] \otimes B[r + \Delta r, c + \Delta c]$之和，其中$\otimes$是异或算子，当且仅当两项输入不同时结果才为1。如上所述把$B[r, c]$处四个变化量的最小值赋给IN[r,c]，就得到一幅兴趣图像(interest image)。用几幅黑白图像包括棋盘图像测试你的算子。

一旦时刻图像I_1上的兴趣点集$\{P_i\}$确定下来，就要开始找出$(t + \Delta)$时刻图像I_2上的对应点。我们的方法不是先检测图像I_2上的兴趣点再判断对应性，而是直接对I_2进行搜索来确定I_1上的点在I_2上的对应位置。用第5章介绍的交叉相关法可以实现这个要求。已知I_1上的兴趣点P_i，对于I_1上的P_i邻域，在I_2上寻找与之最相关的邻域，前提是假设运动量是有限的。图9-5是在I_2上搜索最佳邻

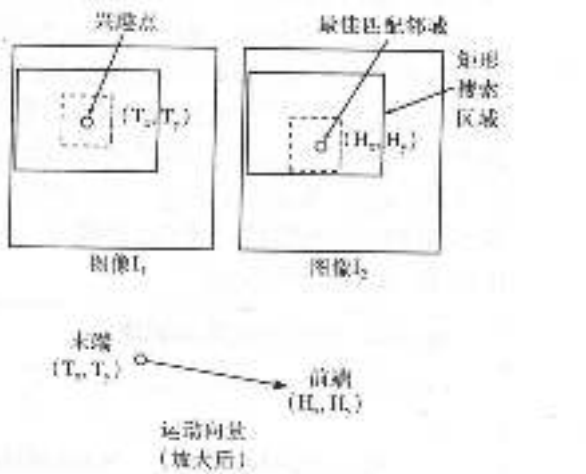


图9-5 对图像I_1中的每一个兴趣点(T_x, T_y)，搜索I_2中与(T_x, T_y)邻域最佳匹配的矩形区域。如果匹配得好，它就成为运动向量的前端(H_x, H_y)

域的方法，该邻域与I_1的P_j邻域最佳匹配。把I_2上的最佳相关邻域的中心$P_k=[P_{kr}, P_{kc}]$作为对应点，该点将成为运动向量的前端，$P_j=[P_{jr}, P_{jc}]$是向量的末端。对P_k的搜索限制在图像行$P_{jr}-R\cdots P_{jr}+R$与列$P_{jc}-C\cdots P_{jc}+C$之间的矩形$C\times R$区域内。搜索区域小时，会加快搜索速度和减小歧义性，但只有当目标的运动速度在一定范围内，算法才是可用的。算法步骤参见算法9.3。图9-6显示算法的应用情况。使三个纹理明显的图片在一个纹理不太明显的背景前运动，就生成了实验图像。

257
~
258

算法9.3 计算两幅图像中兴趣点产生的运动向量

$\mathbf{I}_1$[r, c]和$\mathbf{I}_2$[r, c]是MaxRow行、MaxCol列的输入图像。

V是输出运动向量的集合$\{[(T_x,T_y), (H_x, H_y)]_i\}$，

其中(T_x, T_y)、(H_x, H_y)分别为运动向量的末端和前端。

```
procedure extract_motion_field(I1, I2, V)
{
\\检测匹配的角点,返回运动向量V。
\\τ2是对邻域进行交叉相关运算的阈值。
        detect_corner_points(I1, V);
        for all vectors [(Tx ,Ty), (Ux , Uy)] in V
          match := best_match(I1, I2 ,Tx , Ty, Hx, Hy);
          if (match < τ2) then delete [(Tx, Ty), (Ux, Uy)] from V;
          else replace [(Tx ,Ty), (Ux, Uy)] with [(Tx, Ty), (Hx, Hy)] in V;
}
real procedure best_match(I1, I2 ,Tx, Ty, Hx, Hy);
\\(Hx, Hy) 作为I2中最佳匹配邻域的中心返回，该邻域与I1中以(Tx, Ty)为中心的邻域匹配。
\\sh与sw确定搜索的矩形范围:h与w确定邻域的范围。
{
        \\第一次指示还没找到最佳匹配。
        Hx:= −1; Hy:= −1; best:= 0.0;
        for r:= Ty−sh to Ty + sh
          for c:= Tx−sw to Tx + sw
            {
              \\如第5章所描述的，把I1中的N与I2中的N进行交叉相关。
              match := cross_correlate(I1, I2, Tx, Ty, r, c, h, w);
              if (match > best) then
                {
                  Hy := r; Hx : = c; best := match;
                }
            }
}
```

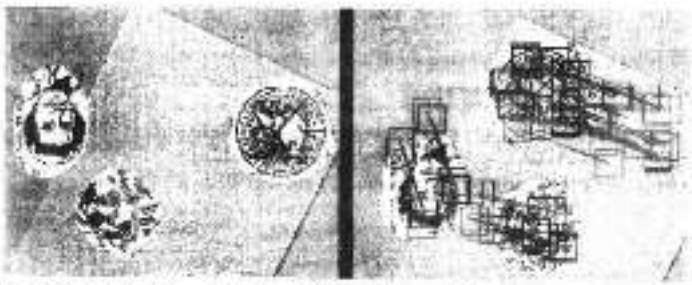


图9-6 算法9.3的应用结果。左边是t_1时刻的图像，右边是带有运动分析结果的t_2时刻的图像，红色方框表示原始邻域的位置，是对左图运用兴趣算子检测得出的。蓝色方框表示右图中与左图最佳匹配的邻域。一组绿线是表示运动向量，分别对应三个运动目标。最左边的目标向下偏右一点的方向运动，最下面的目标向右偏下一点的方向运动，最右边的目标向左偏上一点的方向运动。（分析由Adam T. Clark提供）参见彩图9-6

可在算法9.3中加入迭代控制，每次分析两帧图像，最后连续跟踪多帧图像的特征点。第$(t+\Delta)$帧上识别到的角点，能代替前面第t帧上识别到的角点，以及新的用于交叉相关运算的邻域，这个新邻域也许发生了变化。只要动态场景中重要点的邻域是逐渐变化的，就可以用这种方式对这些重要的特征点进行跟踪。一般我们也要考虑角点被挡住以及新角点出现的可能。这些内容将在第9.4节讨论。

习题9.4

考虑标准棋盘图像。(a) 设计角点检测器，只检测方块的四个角点，方块内部及沿两方块之间边线上的点不检测。(b) 摄像机静止，让棋盘图像慢慢移动，拍摄几幅图像。(c) 利用这些图像测试你设计的角点检测器，并给出检测结果，看看在每幅图上正确检测出的角点数和错误检测出的角点数。(d) 对于几对运动量不大的图像，实现并测试算法9.3。

259
260

9.3.3 MPEG视频压缩

MPEG视频压缩技术采用复数运算，最高以200:1的压缩比压缩视频流。可以注意到MPEG运动图像压缩方法与算法9.3有类似之处。MPEG的子目标不是计算运动场，而是利用预测编码压缩图像序列，即从一些图像帧预测出另一些图像帧。重要的不在于运动向量正确表达了运动目标，而在于能够从一幅图像的邻域高质量预测出另一幅图像的邻域。MPEG编码器用运动向量取代一帧中的整个16×16的图像块，运动向量确定了与前面某帧图像最佳匹配的16×16亮度块的位置。图9-7表示MPEG压缩算法中用到的运动

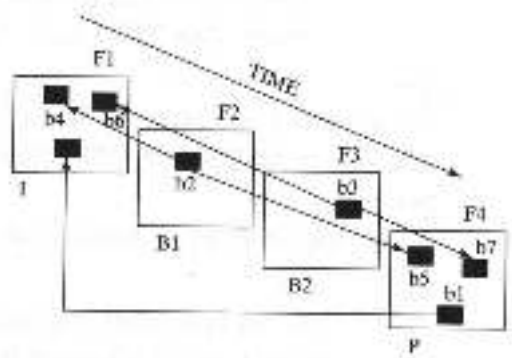


图9-7 MPEG方法中采用运动向量压缩视频序列的简单示意图，序列中包括四帧图像：F1、F2、F3、F4。F1作为独立帧（I），用JPEG单帧静态图像编码法编码。对F4，根据F1用运动向量附加差分块进行P帧预测编码：16×16的像素块（b1）在F1中的对应位置，通过运动向量附加差分块的方法进行确定。中间帧B1与B2帧完全用运动向量插值法，把F1帧中的像素块（b4）与F4帧中的像素块（b5）进行平均，重构16×16的像素块（b2）。尽管中间帧F2和F3最初是在F4之前形成的，也只有当F4帧被解码之后，才能对中间帧F2与F3进行解码。中间帧的压缩率最高，因为每个16×16的像素块只用两个运动向量表示。I帧的压缩率最低

估计方法。图题文字对该方案进行了详细解释。没有采用显著图像点，而是用了均匀的栅格方块，通过搜索视频序列中前面的图像，从而找到与这些方块相匹配的区域。图中只显示了少数几个方块的计算过程。理想情况下，每个16 × 16的方块B_k可用一个向量$[V_{x_k}, V_{y_k}]$代替，编码器通过这个向量来确定前一帧的最佳匹配亮度块的位置，如果两个亮度块之间有差异，则可用少量的数位表示这个差异并进行传送。

尽管MPEG中运动向量的设计是为了压缩的目的，而不是为了运动分析，研究人员已经开始实验用运动向量建立运动场。优点是，MPEG编码器现在能实时计算这些向量，而且已经 261
用于视频流中。未来的信号编解码器也许真的能提供用于运动分析的运动场。

习题9.5

假设视频序列中每帧图像是320 × 240的 8位黑白图像。

(a) 中间帧的MPEG编码的输出是什么？

(b) 表示这种输出需要多少字节？

(c) 相对原始图像，中间帧的压缩比是多少？

9.3.4 图像流计算*

现有方法已经能够估计图像上所有点而不只是兴趣点的图像流。我们来研究一种经典的方法，它至少根据前后两帧图像同时算出时空梯度。图9-8是理想情况下的一个例子，表示物体在摄像机面前运动时摄像机所观察到的场景。左下角是t_1时刻的图像a，显示一个三角形物体；在t_2时刻的图像b中，可以看出三角形物体向上运动了一段位移。从这个简单的例子出发，引出我们在研究图像流数学模型时需要做出的几个假设。

- 假设在$[t_1, t_2]$时间段内，目标物体的反射率和光照度不变化。
- 假设在这段时间内，目标离摄像机或光源的距离没有显著变化。
- 假设在t_1时刻的亮度邻域$N_{x,y}$，在t_2时刻能被观察到，新的位置是$N_{x+\Delta x, y+\Delta y}$。

对实际图像来说这几条假设并不是很强的约束条件，但有时计算图像流向量时是必须满足的。我们用一个简单的离散型例子引出图像流理论，后面从带连续空间参数的连续亮度函数$f(x, y)$，导出图像流方程。

习题9.6

参考图9-8，亮度函数为$f(x, y, t)$。考虑t_1时刻图像上空间坐标是$x = y = 4$的像素，即三角形上部亮度9、7、5之间的亮度为7的像素。计算图像函数在$x = y = 4$处的空间偏导数$\partial f/\partial x$与$\partial f/\partial y$，$t = t_1$，用3 × 3邻域。计算在$x = y = 4$处的时间偏导数$\partial f/\partial t$，$t = t_1$。用什么方法合适？

3333333333	3333333333
3333333333	3333333333
3333333333	3333333333
3373333333	3397533333
3397533333	3399753333
3399753333	3399975333
3399975333	3333333333
3333333333	3333333333
a) t_1	b) t_2

262

图9-8 图像流例子。一个亮度值较大的三角形，从t_1时刻到t_2时刻向上移动了一个像素。背景亮度值是3，而物体的亮度值是9

9.3.5 图像流方程*

根据上面的假设条件，推导出图像流方程（image flow equation），并讨论如何用图像流方程计算图像流向量。对连续亮度函数

$f(x, y, t)$ 在任意点 (x, y, t) 的小邻域内进行泰勒（Taylor）级数展开。

$$f(x + \Delta x, y + \Delta y, t + \Delta t) = f(x, y, t) + \frac{\partial f}{\partial x}\Delta x + \frac{\partial f}{\partial y}\Delta y + \frac{\partial f}{\partial t}\Delta t + h.o.t. \tag{9-1}$$

注意公式（9-1）是对多变量函数的近似表达式，如果只有一个变量则表示为$f(x + \Delta x) = f(x) + f'(x)\Delta x$。对于 (x, y, t) 附近的小邻域，我们忽略公式（9-1）中的高阶项$h.o.t.$，只考虑线性项。下一个重要的步骤参见图9-9，需要求的图像流向量$\mathbf{V}=[\Delta x, \Delta y]$，使$t_1$时刻 (x, y) 处邻域N_1的亮度与t_2时刻 $(x+\Delta x, y+\Delta y)$ 处邻域N_2的亮度一致，这个假设意味着

$$f(x + \Delta x, y + \Delta y, t + \Delta t) = f(x, y, t) \tag{9-2}$$

根据公式（9-1）与公式（9-2），并忽略高阶项，可得到图像流方程如下：

$$-\frac{\partial f}{\partial t}\Delta t = \frac{\partial f}{\partial x}\Delta x + \frac{\partial f}{\partial y}\Delta y = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right] \circ [\Delta x, \Delta y] = \nabla f \circ [\Delta x, \Delta y] \tag{9-3}$$

图像流方程并不能保证流向量$\mathbf{V}$有唯一解，但提供了一个线性约束方程。其实，也许有多个N_2邻域与N_1邻域的亮度一致。图9-10显示当受限于以点 (x, y) 为中心的一个小邻域或孔径（aperture）时，有多个流向量存在的可能性。针对以$\mathbf{P}$为中心的小孔径，点P有可能移动到$\mathbf{R}$、$\mathbf{Q}$或线段$\mathbf{QR}$上的其他位置。图9-11显示方块目标四条边缘的运动情况。一般地，我们不明确指出物体的边缘，但是图9-9仍然适用于等亮度的轮廓曲线。图中的边缘线应是轮廓的切线，局部范围近似为轮廓线。

263

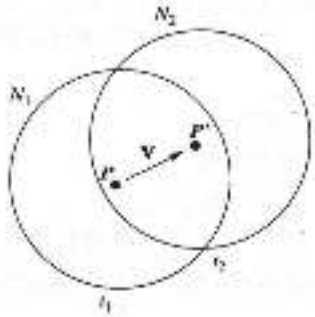


图9-9 在V方向上产生的运动，t_1时刻邻域N_1的亮度与t_2时刻邻域N_2的亮度一致

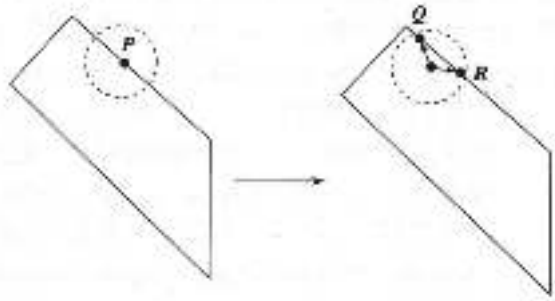


图9-10 从t_1时刻到t_2时刻，亮度边缘向右运动。但由于匹配用的邻域或孔径尺寸有限，点P有可能移动到R、Q或线段QR上的其他位置

我们可以对图9-10做如下解释。观察到点P的变化，这个变化可用梯度$\frac{\partial f}{\partial t}\Delta t$确定。这个变化等于空间梯度$\nabla f$与流向量$\mathbf{V}$的点积。$|\mathbf{V}|$可以很小如等于到新边缘的垂直距离，也可以很大，这时流向量的方向与空间梯度方向很不一致。当一条绳索被很快向上拉起，水平方向有些微小的振动，结果造成图像边缘的位置变化很小时，$|\mathbf{V}|$就很大。

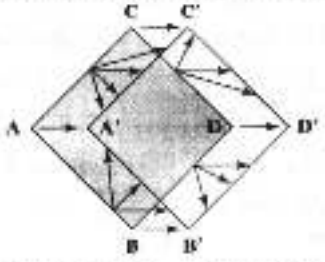
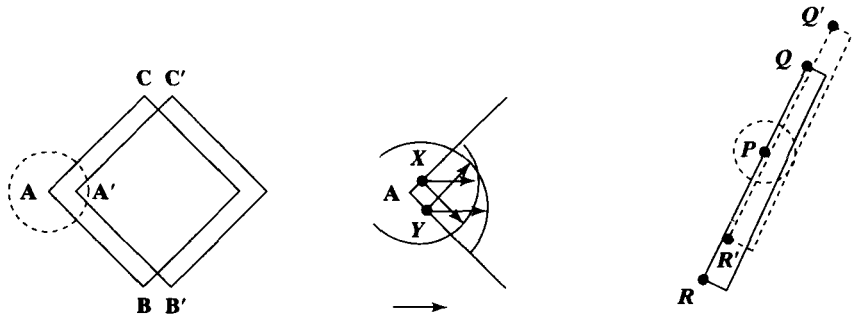


图9-11 块状目标向右运动。t_1时刻末端在边缘上的运动向量，受到线性关系的约束，使向量前端位于t_2时刻的边缘上，在角A、B、C、D处的一般性约束产生了右移（move right）现象，于是由于边缘的连贯性使这种右移被推广到所有的边缘点

9.3.6 利用传播约束求解图像流*

图像流方程提供一种约束，这种约束对每个像素位置都适用。根据一致性假设，邻近像素应具有相似的流向量。图9-12显示如何用邻近约束降低运动方向的歧义性。图9-12b是对a中运动方块的角A邻域的放大。图像流方程把点X处的运动方向θ_x限制在$5\pi/4$与$\pi/4$之间，把点Y处的运动方向θ_y限制在$-\pi/4$与$3\pi/4$之间。如果点X与点Y属于同一个刚性物体，则X与Y处的流向量就被限制在上面两个范围的交集内，即$-\pi/4$与$\pi/4$之间。 264

图9-11与图9-12强调两点：第一，只有在兴趣点即角点处，才能用小孔径约束可靠地计算图像流；第二，在角点处，对流向量的约束可以推广到边缘位置，而如图9-12c所示，对于离角点比较远如边缘上的点P处，可能需要许多次迭代才能接近一个合适的值。利用随机像素图像进行光流计算的实验做了很多。进一步的研究表明，这样的图像也许比高度结构化的图像计算起来更加容易，因为它的邻域更有可能是独有的。二维松弛法在第11章进行讨论。用微分方程求解图像流的方法参见Horn与 Schunck于1981年发表的论文，见9.6节的参考文献部分。



a) b) c)

图 9-12

a）正方形目标右移

b）角A的放大图显示，根据两个邻近图像流方程得到的约束，使方向的歧义范围降低到$\pi/2$

c）极端孔径问题：长条形目标沿长度方向运动，位于点P的孔径及其邻近处，运动方向的歧义值是π

9.4 计算运动点路径

前面讨论了识别t_1时刻图像上的兴趣点，并查找该点在下一帧即t_2时刻图像上对应点的方法。如果点周围的亮度邻域具有独特的纹理，那么我们就能用规范化交叉相关技术随时跟踪这一点。另外，领域知识也能使图像序列中的目标跟踪变得容易，比如跟踪网球比赛中的桔黄色网球，或者跟踪工作站前面的粉色人脸。

现在考虑一般的情况，即运动点附近的纹理或颜色不是独有的，这样就必须通过运动本身的特性得到这些点的轨迹。图9-13显示三个目标物经6个时刻的光滑运动轨迹。在考虑一般情况之前，先提出三个具体的问题。第一，考虑装有许多网球的盒子掉到地上，要根据视频序列计算每只球的轨迹；第二，在流体中混入荧光粒子，研究流体通过容器的流动特性，并拍摄粒子随时间的运动情况。假设每个粒子在图像中看起来是一样的；第三，计算人们在街道上的行走路径。人们的穿着反映在图像上也许是唯一的，但一些人在图像中具有类似的外观，这确实是很有可能的。 265

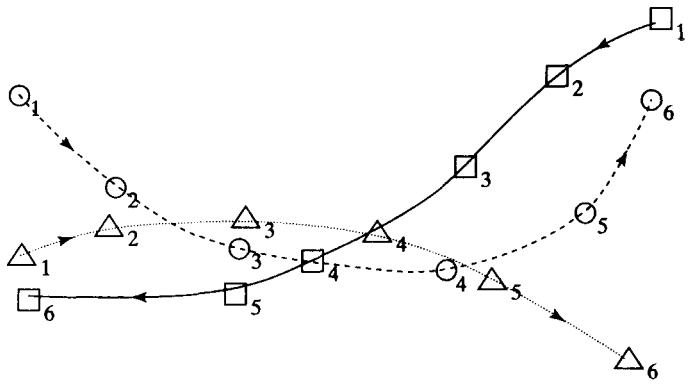


图9-13 三个目标○、△、□的运动轨迹。图中显示了六个时刻每个目标的位置。○和△从左向右运动，□从右向左运动

可以利用如下针对三维空间实际物体的一般性假设:

(1) 物体位置随时间的变化是平稳的。

(2) 物体运动速度随时间的变化是平稳的，包括速度的大小与方向。

(3) 某个时刻物体在空间中只有一个位置。

(4) 两个物体不可能在同一时刻占有同一个位置。

前三个假设对于三维空间的二维投影是成立的，平稳的三维运动产生光滑的二维运动轨迹。第四个假设在投影情况下就不再成立，因为一个物体会遮挡另一个物体，当只用一台摄像机时就会出现问题。实验表明，根据运动物体的图像序列，人类能识别出物体并能分析它的运动情况。在Johansson于1976年做的著名实验中，让光线照射人体的各个部分。当人体静止时，观察者不能确定眼前的目标是一个人；但当人体运动时，观察者能够很容易地认识到眼前的目标就是一个人。

下面我们给出Sethi与Jain在1987年设计的一个算法，借助上面的四个假设，计算图像序列中通过观测点的最光滑的一组路径。首先给出单条路径光滑度的数学定义；然后定义最光
266 滑的m条路径集，因为这组路径的m个光滑值之和是最优的；最后定义**贪婪交换算法**（greedy exchange algorithm），该算法在每个时刻进行最优的赋值分配，用迭代的方法把路径m从t_1时刻延伸到t_n时刻。

定义73 如果在时刻$t = 1, 2, \cdots, n$，目标i在摄像机的视野范围内，则目标i的图像点序列$T_i = (p_{i,1}, p_{i,2}, \cdots, p_{i,t}, \cdots, p_{i,n})$就称为**$i$的轨迹**。

轨迹上任意两点之间的差分向量定义为:

$$V_{i,t} = p_{i,t+1} - p_{i,t} \tag{9-4}$$

我们可以根据接近或离开轨迹上点$p_{i,t}$的向量差分，来定义该点的光滑值。方向的光滑性通过向量的点积来度量，速度大小的光滑性通过向量幅度的几何平均值与平均幅度之比来度量。

$$S_{i,t} = \omega\left(\frac{V_{i,t-1} \circ V_{i,t}}{|v_{i,t-1}||V_{i,t}|}\right) + (1-\omega)\left(\frac{2\sqrt{|V_{i,t-1}||V_{i,t}|}}{|V_{i,t-1}| + |V_{i,t}|}\right) \tag{9-5}$$

其中权系数w取值范围是$0 \leqslant w \leqslant 1$，结果使得$0 \leqslant S_{i,t} \leqslant 1$（参看本节习题）。注意对于直线

轨迹，所有空间点的差分向量是一样的，而且公式（9-5）的结果为1.0，这就是最优的点光滑值。方向或速度大小的变化使$S_{i,t}$的值变小。假设m个独立的点是从n帧图像的每一帧中抽取出来的，后面将会看到可以放宽这个假设。第一帧上的点标记为$i = 1, 2, \cdots, m$。问题是如何建立具有最大总光滑值的m条轨迹T_i。在公式（9-6）中，总光滑度定义为所有m条路径上所有内部点的光滑度之和。

$$\text{总光滑度 } T_s = \sum_{i=1}^{m} \sum_{t=2}^{n-1} S_{i,t} \tag{9-6}$$

习题9.7

假设$w = 0.5$，方向与速度大小的加权系数相同。（a）证明具有单位边长的规则六边形，它的每个顶点的光滑度是0.75。（b）正方形顶点的光滑度是多少？

习题9.8

（a）利用柯西-施瓦茨不等式（见第5章），证明计算$S_{i,t}$的公式中 $\dfrac{V_{i,t-1} \cdot V_{i,t}}{|V_{i,t-1}||V_{i,t}|}$ 的值界于0与1之间。（b）证明两个正数x与y，其几何平均值 $\sqrt{xy}$ 不超过其算术平均值$(x + y)/2$。并据此证明 $\dfrac{2\sqrt{|V_{i,t-1}||V_{i,t}|}}{|V_{i,t-1}|+|V_{i,t}|}$ 的值界于0与1之间。（c）证明只要w界于0与1之间，则公式（9-5）中的$S_{i,t}$界于0与1之间。

267

习题9.9

下面两种情况下，4点轨迹的总光滑度是多少？（a）沿边长为s的六边形的四边；（b）沿边长为s的正方形的四边。

算法9.4从n帧序列图像计算m条轨迹。不保证时间T_s最小，但实验结果表明算法有很好的效果。首先参考图9-14的简单例子，直观上对算法进行初步的了解。表9-1列出所关心路径的光滑度。在第1帧中可以对点进行随机标号，例如物体□$_1$ = 1 = T[1, 1]，物体○$_1$ = 1 = T[2, 1]，然后把轨迹扩展到后面各帧中的最近点T[1, 2] = ○$_2$，该点是最近的点，用排除法后T[2, 2] = □$_2$。在转换到实际轨迹时我们犯了一个错误。我们在时刻$t = 3$时利用预测进行最近邻赋值之后，再计算这两条路径的总光滑度。从表9-1的前两行可以看出，这两条路径的总光滑度是0.97 + 0.98 = 1.95。如果把赋值操作T[1, 2] = ○$_2$与T[2, 2] = □$_2$进行交换，可得到更好的光滑值0.99 + 0.99 = 1.98。交换之后，到$t = 2$时的轨迹是（□$_1$，□$_2$）和（○$_1$，○$_2$）。最近点的初始赋值将在时刻$t = 3, 4$时给出最佳的光滑值，不需要交换赋值。但是当$t = 5$时，最近点赋值将产生轨迹（□$_1$，□$_2$，□$_3$，□$_4$，○$_5$）和（○$_1$，○$_2$，○$_3$，○$_4$，□$_5$），最后两个赋值相交换后，在中间的三个轨迹点得到的总光滑度会提高，从2.84 + 2.91 = 5.75提高到2.89 + 2.94 = 5.83，所以图9-14显示的最后标号是正确的。

算法9.4　贪婪交换算法：输入各时刻的二维点集，计算光滑路径

P[i, t]保存帧序列$t = 1,2,\cdots,n$上的二维点,点的标号为$i = 1, 2, \cdots, m$。

T[i, t]，输出轨迹集合，m行n列。

T[i, t] = k的意思是，目标i被看作是第t帧中的第k个点。

1. 初始化：通过最近邻连接建立m条完全路径。

(a) 第一帧：对所有i，设置目标标号$\mathbf{T}[i, 1] = i$;

(b) 其他帧：对于$t = 2, 3, \ldots, n$，使$\mathbf{T}[i, t] = k$,其中点$\mathbf{P}[k, t]$是离点$\mathbf{P}[\mathbf{T}[i, t-1], t-1]$最近的点，该点还没有赋值。

2. 交换循环：for t :=2 to $n-1$

(a) 对$j \neq k$的所有点对(j, k)，计算由于$\mathbf{T}[j, t]$与$\mathbf{T}[k, t]$赋值交换带来的光滑度的提高量;

(b) 进行交换,使光滑度最大程度提高。如果总的光滑度不提高，则不交换;

(c) 如果做了交换则置位交换标志。

3. 终止测试：如果上述循环中进行了交换，则清零交换标志，重复交换循环。

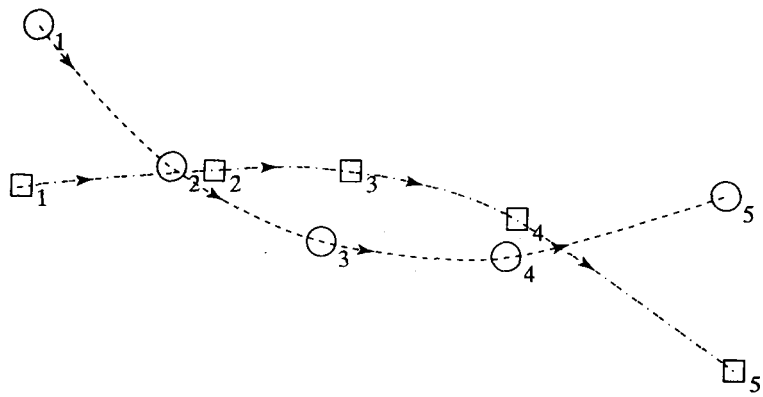


图9-14 两个目标的轨迹，在前5个位置中，○和□沿着图像流向量运动。跟踪器可能认为$\square_2$继承的是$\bigcirc_1$，而$\bigcirc_5$有可能是序列$\square_1$，$\square_2$，$\square_3$，$\square_4$的最后一点

表9-1 图9-4的路径光滑度

t = 1	t = 2	t = 3	t = 4	t = 5	光滑度
$\bigcirc_1$(112 262)	$\square_2$(206 185)	$\bigcirc_3$(250137)			0.97
$\square_1$(106 175)	$\bigcirc_2$(180 188)	$\square_3$(280 185)			0.98
$\bigcirc_1$(112 262)	$\bigcirc_2$(180188)	$\bigcirc_3$(250137)			0.99
$\square_1$(106 175)	$\square_2$(206 185)	$\square_3$(280 185)			0.99
$\bigcirc_1$(112 262)	$\bigcirc_2$(180 188)	$\bigcirc_3$(250137)	$\bigcirc_4$(360137)		1.89
$\square_1$(106 175)	$\square_2$(206 185)	$\square_3$(280 185)	$\square_4$(365 156)		1.96
$\bigcirc_1$(112 262)	$\bigcirc_2$(180 188)	$\bigcirc_3$(250137)	$\bigcirc_4$(360137)	$\square_5$(482 80)	2.84
$\square_1$(106 175)	$\square_2$(206 185)	$\square_3$(280 185)	$\square_4$(365 156)	$\bigcirc_5$(478 170)	2.91
$\bigcirc_1$(122 262)	$\bigcirc_2$(180 188)	$\bigcirc_3$(250137)	$\bigcirc_4$(360137)	$\bigcirc_5$(478 170)	2.89
$\square_1$(106 175)	$\square_2$(206 185)	$\square_3$(280 185)	$\square_4$(365 156)	$\square_5$(482 80)	2.94

在每次应用光滑指标之前，算法9-4初始化n个点的m条完全路径。交换循环的次数是可变的，目的是通过交换两条路径之间的点去提高光滑度。如果在任意时刻t，通过交换使光滑度有了提高（而且总是最大的提高），那么重复整个交换循环。总的来说，在每一时刻t，可能

发生（$\binom{n}{2}$）次交换。算法至少需要进行$(n-2)\binom{n}{2}$次运算。每增加一次交换循环就需要做更多的运算。当超过$1.0m(t-2)$时，光滑度不可能再提高，因此能提高光滑度的循环次数是有限的。这时就应终止算法。

如果在$t=1$帧的赋值是任意的，总共要考虑m^{n-1}种可能的路径。这是一个很大的数目。贪婪交换算法每次只交换两个赋值运算，而不考虑两个以上的赋值运算。因此可能得不到全局最小值。可以对算法9.4进行修改，只用一帧或三帧进行初始化，并且在得到新的一帧和抽取特征点时算法是连续的。如果能得到所有帧上的所有点，可在交换循环中用前向和后向处理方法对算法进行改进。也可对算法进行扩展，处理在两帧之间有新点出现与旧点消失的情况。这主要是由于一个运动物体被另一个遮挡所造成的。在那些少于m个点的图像帧中可以用虚假点（ghost point）进行补充。

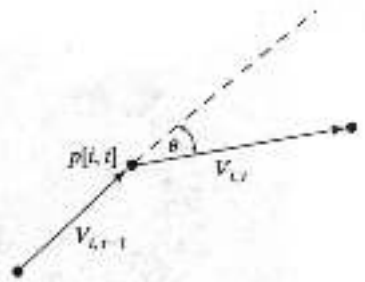


图9-15 接近和离开轨迹上点$p[i,t]$的向量

习题9.10

下面三个点的集合是从6帧视频图像中抽取的，并对应于图9-13中的数据。用贪婪交换算法确定最光滑的三条轨迹。

t=1	t=2	t=3	t=4	t=5	t=6
(483 270)	(155 152)	(237 137)	(292 128)	(383 117)	(475 220)
(107 225)	(420 237)	(242 156)	(358 125)	(437 156)	(108 108)
(110 133)	(160 175)	(370 180)	(310 145)	(234 112)	(462 75)

习题9.11

在下列情况下，你认为贪婪交换算法能成功地根据图像序列中的点构造出轨迹吗？请解释为什么。(a) 旋转木马视频中，木马上下运动。(b) 从人行道拍取的街道视频，摄像机前正好有两辆汽车以35MPH的速度驶过，两辆车的运动方向相反。(c) 关于两个台球发生碰撞的高速影片。运动的白球击打静止的红球。碰撞之后，白球静止，而红球得到了白球的所有动量。

268 ~ 270

面向特殊问题的集成跟踪

算法9.4表明只用一般光滑性约束所做的工作。在特殊应用中，需要更多的信息以提高跟踪的稳健性和跟踪速度。如果m个点所对应的特征都能够得到，在光滑性计算中就可以包括特征匹配。进一步，对t时刻及以前的这部分轨迹进行拟合，可以预测第$t+1$帧轨迹点的位置，这可以大大降低用交叉相关方法进行点搜索的工作量。在最近的研究文献中可以找到这些算法。Maes等人（1996）通过计算手、脚和头的轨迹来跟踪人的运动。手、脚和头是运动人体的侧面影像上曲率大而突出的部分。Bakic和Stockman（1999）用一台与工作站相连的摄像机跟踪人脸，眼和鼻子，目的是为了控制鼠标的光标。图9-16显示的是在当前帧中检测到的特征，得出光标在8×8菜单选择阵列中的位置。第二行第三列中的笑脸表示用户选中的这一项。由于系统集成了领域知识，处理速度可以达到每秒15帧以上。利用人脸的颜色信息在图像中识别出人脸，利用人脸的结构知识确定眼睛和鼻子的位置。另外，根据眼与鼻子的轨迹预测

271

在下一帧图像上的什么地方寻找特征，如果在预测到的邻域内找到了特征，就不用作全局性人脸检测了。

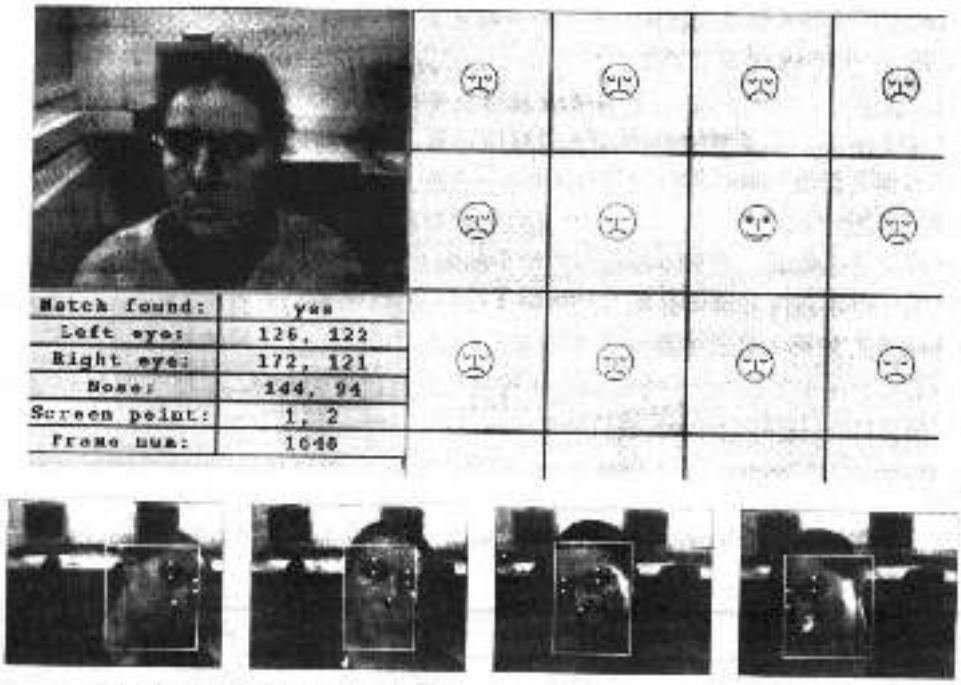


图9-16 跟踪工作站用户的眼与鼻子，做到没有鼠标的情况下控制光标的移动（图像由Vera Bakic提供）

（上图）在菜单中选择要检测的人脸特征

（下图）人脸图像序列，显示检测和跟踪眼与鼻子的过程

电影泰坦尼克号中有很多例子，综合利用计算机图形学与计算机视觉技术，把实际图像与合成的图像相结合。原来是一艘模型船的图像，然后通过在甲板上添加运动模型对图像进行增强。一名女演员穿着20世纪早期飘曳的古典裙装，拍摄出一系列关于她的运动图像。实际服装上装有许多小灯，目的是为了方便检测运动序列中的特征点，然后根据这些运动点的轨迹，修改模型人与模型服装的运动，再把它们加到模型船图像中的不同位置。这部巨型电影的每一帧都花费了计算机和工作人员的大量时间，所以不是所有的措施都一定是个自动的。

9.5 检测视频中的显著变化

视频序列可以记录几分钟或几小时的监视录像，电视新闻的不同抓拍镜头，完成的记录文献或电影等。把视频序列分割成子序列，并储存到数字图书馆，人们能够随意访问，这一点变得日益重要。本节讨论视频序列分解和分析的重要概念和方法。首先要对视频或其他图像序列中的几方面变化进行定义。

- **场景变化**（scene change）指环境的变化。例如，从饭店场景到街道场景。希望这种变化是整个背景的总变化。一般通过下面的一种摄像特效，摄取10到15帧以上的图像，制造出场景的变化效果。
- **镜头切换**（shot change）是在同一场景中，显著改变摄像机的视点。一般通过变换摄像机实现这种效果。如在饭店场景一台摄像机拍摄男演员A正在说话，而另一台摄像机拍

摄桌子对面的男演员B的反应。

- **摄像机扫视**（camera pan），摄像机水平扫视景物。如果摄像机从右向左扫过，物体就像是从左边进入，穿过图像到了右边，最后从右边出去。从静态景物全景序列的连续帧中计算出的运动向量，方向将仍然是从左到右。
- **摄像机变焦**（camera zoom），随时间改变焦距，以放大其部分景物的图像或缩小景物的图像并包括更多的周围背景。
- **特效处理**（Camera effect），利用淡变、溶变、擦除等效果把一幅图像转化成另一幅图像。淡出（fade out）是指由原始图像逐渐变黑或变白的连续过程，而淡入（fade in）是指从由黑或白屏逐渐变成某视频图像的连续过程。通过淡出视频A再淡入视频B，可以实现从视频A到视频B的转换。溶变（dissolve）是指经过若干个图像帧，A中的像素逐渐变成图像B中的像素。一种溶变方式是对A和B中的像素进行加权处理，如A中像素的权系数取$(1-t/T)$，B中像素的权系数取t/T，其中帧号$t=0,\cdots,T$。擦除（wipe）效果通过改变A和B在帧中显示的区域大小使B逐渐代替A。想像汽车前面的雨刷，假设A显示在雨刷的一边，B显示在另一边。在两个区域之间使用垂直、水平或者对角分界线实现擦除效果。或者，B一开始出现在一个小圆区域内，逐渐变大最后占满整帧范围。 272

习题9.12

写出伪码算法，用擦除法把视频原始资料A渗入视频原始资料B中。原始资料A是图像序列$A_t[r,c]$，原始资料B是图像序列$B_t[r,c]$。（a）假设擦除的实现是，从时刻t_1到时刻t_2用斜率为1的对角线，在时刻t_1从像素点[0, 0]（左上角）出发，在时刻t_2结束于像素点$[M-1, N-1]$。（b）假设擦除的实现是，圆形区域从帧中心开始逐渐变大，在时刻t_1圆圈的半径是0，在时刻t_2圆圈的半径大到与帧的边缘相切。

9.5.1 视频序列分割

分割的目的是，把一个较长的视频序列分割成单个场景的子序列。例如一个30分钟的电视新闻节目中，将有几个10到15秒的片断，片断中的镜头对着新闻广播员，他正在桌子旁报道新闻；而背景是不变的办公室场景，但可能有摄像机变焦效果。该片断之后，一般屏幕会过渡到其他纪实性视频资料，也许是关于洪涝的报道，运动会的精彩镜头，一次会议实况或者政府官员在漫步视察。一般要报道的事件包括若干个不同的镜头，它们之间要有过渡。这种过渡可用于视频分割，根据图像特征随时出现的显著变化，可以检测出这种过渡。

计算序列中两帧图像I_t与$I_{t+\Delta}$之间的差别，一种最简易的方法是计算对应点之间的平均差，如公式（9-7）所示。根据不同的摄像特效，时间间隔Δ可以是一帧或者更多帧。

$$d_{pixel}(I_t, I_{t+\Delta}) = \frac{\sum_{r=0}^{MaxRow-1}\sum_{c=0}^{MaxCol-1} |I_t[r,c] - I_{t+\Delta}[r,c]|}{MaxRow \times MaxCol} \tag{9-7}$$

对于基本稳定的拍摄，即使摄像机轻微摇晃或目标运动存在很小的偏差，公式（9-7）也会产生比较大的偏差从而误导我们。Kasturi和Jain在1991年提出的更稳健的改进方法是，把图像分割成比较大的模块，测试是不是多数模块在两幅图中基本上是一样的。公式（9-8）定义的似然比，用来估计对应模块的亮度是否有显著变化。设图像I_1中的模块B_1的亮度均值与方差分别是u_1和v_1，图像I_2中的模块B_2的亮度均值与方差分别是u_2和v_2。根据公式（9-8）中的似然比对模块差进行定义。如果足够多的模块差为0，则结果证明两幅图像基本上来自同一个镜 273

头。显然，当图像纹理复杂，以及帧与帧之间存在不稳定时，为去除摄像机轻微摇晃造成的影响，用公式（9-8）比用公式（9-7）效果更好。

$$
\begin{aligned}
r &= \frac{\left[\left(\frac{v_1+v_2}{2}\right)+\left(\frac{v_1-v_2}{2}\right)^2\right]^2}{v_1 v_2} \\
d_{block}(B_1, B_2) &= 1 \text{ 当 } r > \tau_r \\
&= 0 \text{ 当 } r \leqslant \tau_r \\
d(I_1, I_2) &= \sum_{B_{1i} \subset I_1,\, B_{2i} \subset I_2} d_{block}(B_{1i}, B_{2i})
\end{aligned}
\tag{9-8}
$$

两图之差可用他们的直方图之差表示，如第8章中的$d_{\chi^2}(I, Q)$。在我们现在的讨论中，I相当于图像I_1，Q相当于图像I_2。64级直方图足够了。对于彩色视频帧，在[0,63]范围的值可通过连接红绿蓝颜色值的高两位得到。直方图比较比前面的方法要快，而且是场景一般特征的更好的表示。由于直方图总体上避开了空间一致性检查，当两幅图像的直方图相同而总体空间分布不同，或者实际上是来自两个不同的镜头时，就会出现错误结果。

图9-17显示来自同一视频记录的四帧图像，上面两帧发生在场景切换之前，下面两帧发生在场景切换之后。图9-18是根据图9-17的前三帧计算出的直方图。左面的两个直方图类似，这意味着对应的两帧图像可能来自同一个镜头，右面的直方图与左面的两个显著不同，说明图9-17中的第三帧来自可能不同的镜头。



图9-17 同一视频记录中的四帧图像。上面两帧与下面两帧之间存在镜头切换（经Springer-Verlag允许，Zhang 等人1993年再版）

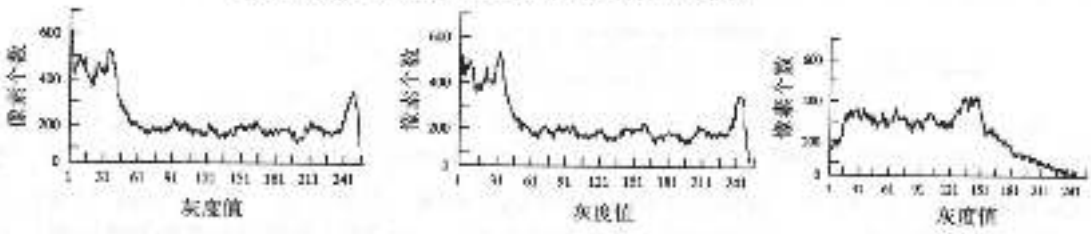


图9-18 图9-17前三帧的直方图。上面的两直方图类似，它们对应的两帧图像也类似。下面的直方图与前两个明显不同，表示它对应的图像帧与前两帧不同（经Springer-Verlag允许，Zhang 等人1993年再版）

9.5.2 忽略摄影特效

如果只是由于某种摄影特效，如摄像机扫视或者变焦，引起相邻两帧图像具有显著差异，这时我们不想对视频序列进行分割。9.5.1节的镜头过渡可用来进行简单的运动分析，从而决定这些效果是否能够忽略。通过计算运动向量可检测出镜头的水平扫视，并且确定运动向量是否在某个模式方向及幅度附近发生聚类。对算法9.3的输出V进行简单分析就能做到这一点。

根据运动场周围的运动矢量可检测出变焦效果。周边处的运动向量之和近似为0说明存在膨胀或收缩的情况。只用到了运动场的周边说明膨胀中心（FOE）或者收缩中心（FOC）不在运动场的中心附近。假设运动向量是利用MPEG算法中的块匹配技术算出的。那么由I_1和I_2确定的运动场的最上和最下块中具有运动向量。上下相对的两运动向量的垂直分量之差，要大于这两个运动向量的任何一个，如图9-19所示。对于运动向量的水平分量也有类似的关系。利用这些启发性的方法，就能够合理地检测出膨胀或收缩效果。然而，根据块匹配得到的运动场的质量却有所下降，因为随着变焦速度加快比例尺度会发生变化。

274

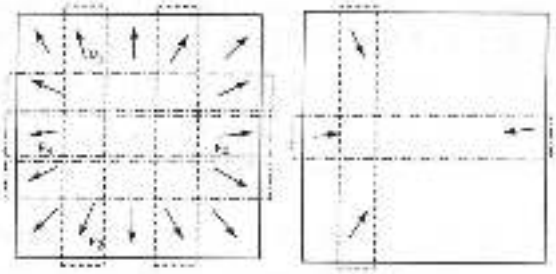


图9-19 检测摄像机变焦的启发式方法：通过比较运动场周边处的运动向量。两上下相对的运动向量的垂直分量之差，要大于这两个运动向量的任何一个，即$|v_{1x}-v_{2x}|>\max(|v_{1x}|,|v_{2x}|)$。同样，对于水平方向的相对运动向量有$|v_{1y}-v_{2y}|>\max(|v_{1y}|,|v_{2y}|)$。这种关系对膨胀（左）及收缩（右）都成立

习题9.13

获得同一场景视频的前后两帧图像。(a) 计算平均像素差，定义见公式（9-7）。(b) 把图像分成$2\times 2=4$的模块，计算模块差之和，定义见公式（9-8）。

275

9.5.3 存储视频子序列

一旦把一个较长的视频序列分割成有意义的子序列，就可以把这些子序列存储在视频库中以供查询和检索。访问视频库可以用第8章讨论的一些方法。可用第8章介绍的通过识别和利用关键帧（key frame）访问数据库。将来我们可能会更进一步，如进行自动运动分析，并进行图符行为标记如*running*、*fighting*和*debating*。为了标记著名人士，需要进行人脸识别；或者进行一般的目标识别，提供标记如*horse*、*house*。与静态图像相比，视频包含许多帧图像，尽管这意味着计算负担很重，但运动分析所提供的信息提高了把目标从背景分离的能力，以及对目标进行分类的能力。

习题9.14

考虑本章前面讨论的应用，即根据比赛的视频序列对网球比赛进行分析。(a) 程序应输出什么样的行为和事件？(b) 程序应输出什么定量数据？

9.6 参考文献

跟踪网球运动员及网球的例子，主要基于贝尔实验室Pingali、Jean和Carlbom在1998年的工作。Freeman等人于1998年发表的论文中，描述了几个实验结果，把计算机视觉技术与已有

的应用技术相结合，设计出一种手势接口。文中还给出一种快速运动估计算法。Kage等人（1999）详细介绍了快速运动估计算法，并对游戏接口进行了详细地讨论。视频分解与索引内容是根据Zhang等人（1993）以及Smolier、Zhang （1996）的工作。根据平凡点的图像帧计算光滑轨迹是以Sethi、Jain （1987）的工作为基础的，他们的工作是在特定的计算机视觉时代做出的，那时人们普遍从几个常规假设出发，研究能计算出什么结果。更近的工作，如Maes等人（1996）、Darrell等人（1998）、Bakic与Stockman（1999）集中讨论特殊问题的有关知识,目的是为了加速计算过程并使其更加稳健。Ayers与Shah （1998）的工作显示如何根据与监视应用相关的语义对运动与变化进行解释。

1. Ayers, D., and M. Shah. 1998. Recognizing human actions in a static room. *Proc. 4th IEEE Workshop on Applications of Computer Vision,* Princeton, NJ (19–21 Oct. 1998), 42–47.

276
2. Bakic, V., and G. Stockman. 1999. Menu selection by facial aspect. *Proc. Vision Interface '99,* Quebec, Canada (18–21 May 1999),18–21.

277
3. Darrell, T. 1998. A radial cumulative similarity transform for robust image correspondence. *Proc. IEEE CVPR,* Santa Barbara, CA (June 1998), 656–662.
4. Darrell, T., G. Gordon, M. Harville, and J. Woodfill. 1998. Integrated person tracking using stereo, color, and pattern detection. *Proc. IEEE CVPR,* Santa Barbara, CA (June 1998), 601–608.
5. Freeman, W., D. Anderson, P. Beardsley, C. Dodge, M. Roth, C. Weissman, W. Yerazunis, H. Kage, K. Kyuma, Y. Miyake, and K. Tanaka. 1998. Computer vision for interactive computer graphics. *IEEE Comput. Graphics and Applications,* v. 18(3) (May–June 1998), 42–53.
6. Horn, B., and B. Schunck. 1981. Determining optical flow. *Artificial Intelligence,* v. 17:185–203.
7. Johansson, G. 1964. Perception of motion and changing form. *Scandanavian J. Psychology,* v. 5:181–208.
8. Kage, H., W. T. Freeman, Y. Miyake, E. Funatsu, K. Tanaka, and K. Kyuma. 1999. Artificial retina chips as on-chip image processors and gesture-oriented interfaces. *Optical Engineering,* 38(12):1979–1988.
9. Kasturi, R., and R. Jain. 1991. Dynamic vision. In *Computer Vision Principles,* R. Kasturi and R. Jain, eds. IEEE Computer Society Press, Washington, D.C., 469–480.
10. Maes, P., T. Darrell, B. Blumberg, and A. Pentland, 1996, *The ALIVE System: Wireless, Full-Body, Interaction with Autonomous Agents.* ACM Multimedia Systems: Special Issue on Multimedia and Multisensory Virtual Worlds, Sprint.
11. Pingali, G., Y. Jean, and I. Carlbom. 1998. Real time tracking for enhanced tennis broadcasts. *Proc. IEEE CVPR,* Santa Barbara, CA (June 1998), 260–265.
12. Salari, V., and I. Sethi. 1990. Correspondence of feature points in presence of occlusion. *IEEE Trans. on Pattern Analysis and Machine Intelligence,* v. 12(1):87–91.
13. Sethi, I., and R. Jain. 1987. Finding trajectories of feature points in a monocular image sequence. *IEEE Trans. on Pattern Analysis and Machine Intelligence,* v. 9(1):56–73.
14. Smolier, S., and H-J Zhang. 1996. Video indexing and retrieval. In *Multimedia Systems and Techniques,* B. Furht, ed. Kluwer Academic Publishers, Boston, 293–322.

278
15. Zhang, H-J., A. Kankanhalli, and S. Smoliar. 1993. Automatic partitioning of full-motion video. *Multimedia Systems,* v. 1(1):10–28.

第10章　图像分割

图像分割是指把一幅图像分成不同的区域。许多分割任务的目标是让图像区域代表一定的含义，如卫星图像中表示住宅、郊外和森林的区域。在图像分析任务中，区域可以用组成区域的边界像素集表示，例如3D工业目标图像中的直线段或圆弧段。区域也可定义为既有边界又有特殊形状的像素集合，如圆、椭圆或多边形。当兴趣区域不覆盖整幅图像时，我们仍然可以将图像分成兴趣前景区域和可忽略的背景区域。

分割有两个目的。第一个目的是将图像分割成部分以便进一步分析。在简单情况下，对环境进行控制，使分割过程能可靠地抽取出要进行分析的部分。例如，在第6章关于颜色的讨论中，提出了一种从彩色视频图像分割人脸的算法。如果人物的衣服及房间的背景与人脸具有不同的颜色分量的话，这个分割是可靠的。在复杂情况下，如从灰度航测图像中抽取出完整的公路网络，分割问题就会非常困难，可能要应用大量领域方面的知识。

分割的第二个目的是改变图像的表示方法。必须对图像像素进行组织，形成更高级的表示单元，使这种高级表示单元比像素表示更有意义，或者更有利于进一步的分析。关键问题是能否找到一种通用的自下而上的分割方法，适应不同领域而又不需要任何专门的领域知识。本章讨论的分割方法可以用于许多不同的领域。下面将讨论基于区域的表示单元和基于曲线的表示单元。一种分割系统适用于所有问题，这样的前景看起来非常黯淡。经验表明，实际利用机器视觉时必须能够从众多方法中进行选择，或者根据具体的领域知识确定一种方案。



图10-1　参见彩图10-1

（左）橄榄球图像

（右）分割成区域的图像，每个区域是颜色相似的连通像素集合

本章讨论几种分割算法，包括经典的区域增长法、聚类算法，以及直线和圆弧检测法。图10-1显示将一幅橄榄球比赛的彩色图像分割成具有近似颜色的区域。图10-2是从玩具积木图像中抽取直线段的结果，注意这两种情况的分割结果，以人类的标准来看离完美相差很远，但是，这些分割结果可以作为更高层自动处理的有效输入，例如，可以根据衣服的数字识别橄榄球运动员，或者根据线段识别要装配的零件。

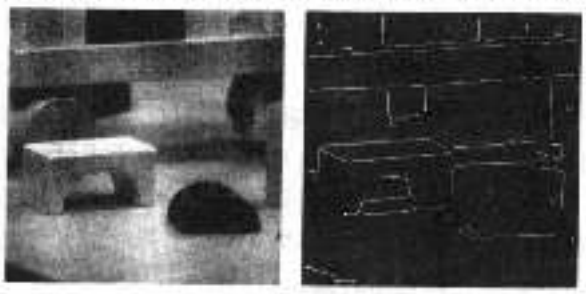


图　10-2

（左）积木图像

（右）用ORT（Object Recognition Toolkit）工具包抽取出的线段图像

10.1　区域分割

- 图像分割后的区域应在某些特征方面表现得一致和同质，如灰度、颜色或纹理。

• 区域内部分布单一，不能有太多的孔。

280 • 对于区域内部的同一特征，相邻区域间应具有明显的差别。

• 分割边界应该是光滑不粗糙，且空间位置准确。

同时满足所有这些要求是有困难的，因为严格一致和同质的区域一般都充满了孔且边界粗糙。坚持相邻区域的值有明显差别的话，会导致区域融合到一起并且使边界丢失。另外，人类感觉均匀的区域，在分割系统获得的低层特征上未必是均匀的，这时可能需要利用高层的知识。本章要讨论的分割算法，可用来分割各种图像，并为各种高层分析服务。

10.1.1 聚类方法

在模式识别中，聚类是将模式向量的集合分成多个子集的过程，这些子集称为聚类(cluster)。例如，如果模式向量是实数对，如图10-3所示的点，聚类则是寻找在二维欧氏空间中互相接近的点的子集。

聚类方法有很多。我们来讨论图像分割中用到的几种聚类算法，包括经典聚类算法、简单的直方图算法、Ohlander的递归直方图算法以及Shi的图分割技术。

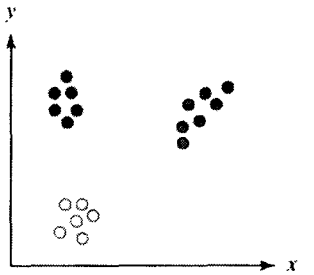


图 10-3 在欧氏度量空间可被分成三类的点集。每个聚类由某种意义上相接近的点组成。图中的几种类别用填充模式不同的圆圈表示

1. 经典聚类算法

聚类的一般问题是将向量集分成几组，每组具有相似的值。在图像分析中，向量代表一些像素，有时代表像素周围的邻域。这些向量的元素包括:

(1) 强度值

(2) RGB值及由此推出的颜色特征

(3) 计算得到的特征

(4) 纹理度量值

任何与像素相关的特征都可用来对像素分组。基于这些度量空间值，把像素分门别类，

281 就很容易利用第3章的连通成分标记找到连通区域。

传统聚类中，有K个类别$C_1, C_2, \cdots, C_K$，均值分别为$m_1, m_2, \cdots, m_K$。最小二乘误差测度(least square error measure)定义为

$$D = \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - m_k\|^2$$

上式检验数据与指定类别的接近程度。最小二乘聚类过程，考虑所有K个类别的可能划分，选择使D最小的那一种。该方法从计算量上来说是不可行的，一般采用近似的方法。重要的问题为是否预先知道K。许多算法都假设参数K由用户提供，另有一些算法则试图根据一些指标找到最佳的K，例如保持每类方差小于某个指定的数值。

2. 迭代K-均值聚类

K-均值(K-means)算法是一种简单的迭代爬山算法，描述如下。

算法10.1 对一组n维向量进行K-均值聚类

1. 令ic(迭代次数)为1;
2. 随机选取K个均值$m_1(1), m_2(1), \cdots, m_K(1)$;

3. 对每个向量x_i计算$D(x_i, m_k(ic))$，$k = 1, \cdots, K$，将x_i分配给具有最近均值的聚类C_j；
4. ic加1，更新均值得到新的集合$m_1(ic), m_2(ic), \cdots, m_K(ic)$；
5. 重复第3步到第4步，直到对所有的k，都有$C_k(ic) = C_k(ic + 1)$。

该算法可以保证能终止，但不能保证最小二乘意义上的全局最优。对第2步进行修改，把向量集随机分成K个聚类，并计算它们的均值。第5步的终止条件修改为，当迭代中改变聚类的向量百分比非常小时终止。图10-4显示的是，对图10-1中的橄榄球图像在RGB空间应用K-均值聚类算法的结果。



图10-4 参见彩图10-4

（左）橄榄球图像

（右）利用K均值聚类，得到$K = 6$种不同灰度当聚类结果，6个聚类对应6种颜色：深绿色、绿色、深蓝色、白色、棕色和黑色

3. isodata聚类

Isodata聚类（isodata clustering）是另一种迭代算法，它利用了标分合并的技术。假设有K个聚类$C_1, C_2, \cdots, C_K$，均值分别为$m_1, m_2, \cdots, m_K$，设$\sum_k$是聚类k的协方差矩阵（定义如下）。如果x_i是如下形式的向量：

$$x_i = [v_1, v_2, \cdots, v_n] \quad 282$$

那么均值向量m_k表示为：

$$m_k = [m_{k1}, m_{k2}, \cdots, m_{kn}]$$

$\sum_k$定义如下：

$$\Sigma_k = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{12} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ \sigma_{1k} & \sigma_{2k} & \cdots & \sigma_{nn} \end{bmatrix} \tag{10-1}$$

其中$\sigma_{ii} = \sigma_i^2$是向量的第i个元素v_i的方差$\sigma_{ij} = \rho_{ij}\sigma_i\sigma_j$，是向量的第$i$个元素和第$j$个元素的协方差。（$\rho_{ij}$是第$i$个元素和第$j$个元素的相关系数，$\sigma_i$是第$i$个元素的标准差，$\sigma_j$是第$j$个元素的标准差。）

图10-5表示的是，对图10-1中橄榄球图像在RGB空间应用isodata聚类算法（由算法10.2描述）的结果。聚类图像是连通成分标记过程的输入，产生如图10-1所示的分割结果。isodata聚类的阈值τ_0设为RGB颜色空间立方体边长的10%。 283

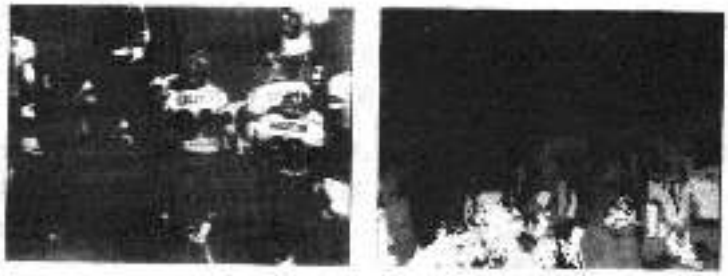


图10-5 参见彩图10-5

（左）橄榄球图像

（右）利用isodata聚类，得到$K=5$时不同灰度的聚类结果。5个聚类对应5种颜色：绿色、深蓝色、白色、银色和黑色

算法10.2 对一组n维向量进行isodata聚类

1. 将x_i分配到使下式最小的聚类l中

$$D_\Sigma = [x_i - m_l]^t \Sigma_i^{-1} [x_i - m_l].$$

2. 如果下式成立，合并聚类i和j

$$|m_i - m_j| < \tau_v$$

其中τ_v是方差阈值。

3. 如果$\sum_k$的最大特征值大于τ_s，则拆分聚类k。

4. 如果对于每个聚类i有

$$|m_i(t) - m_i(t+1)| < \epsilon$$

或者如果达到最大迭代次数，则停止迭代。

习题10.1 isodata与K-均值聚类

对于橄榄球图像，isodata算法的结果比K-均值算法的结果更好，因为它正确地将图像顶部的深绿区域与接近底部的深绿区域分成一类。思考为什么isodata算法的性能能够优于K-均值算法？

4. 简单直方图聚类

迭代分割重组方案需要多次遍历图像数据，而直方图方法仅遍历图像数据一次，因此在度量空间聚类技术中是一种耗时最少的算法。

直方图模式搜索（Histogram mode seeking）是一种度量空间聚类过程，其中假设图像中的同类目标是度量空间（即直方图）中的聚类，将聚类映射回图像区域就可实现图像分割，其中聚类标号的最大连通成分构成图像区域。对于灰度图像，首先确定直方图的波谷，谷与谷之间的间隔就是各个聚类，这样就实现了度量空间的聚类。像素值属于第i个间隔的像素用下标i进行标记，其所属分区是所有像素标记为i的连通成分之一。第3章讨论的自动阈值化技术，是针对双模式直方图的模式搜索实例。

灰度图像一般具有多模式的直方图，这样任何自动阈值化技术，都必须寻找图像中的波峰以及将波峰分开的波谷。这个任务说起来容易做起来难。图10-6是积木灰度图像的直方图。简单的波谷搜索算法，可能把该直方图判断为双模式，并在39和79之间的某个地方取一个阈值。利用试错阈值选择法则产生出3个阈值，得到图10-7所示的4幅阈值化图像，它们表示出

图像中有意义的区域，于是就提出了面向知识的阈值化(knowledge directed thresholding)技术，即阈值选择既与直方图有关，又与区域的质量/有效性有关。

习题10.2 直方图模式搜索

编写程序，确定多模式直方图的模式。首先利用第3章的Otsu方法将直方图分成两部分，然后如果可能，将每部分再分成两部分。分别用灰度图像和彩色图像进行测试。

5. Ohlander递归直方图聚类

Ohlander等人（1978）用递归的方式对直方图聚类思想进行了改进。首先对整幅图执行直方图模式搜索，然后再对所得聚类的每块区域进行模式搜索，直到得到的区域无法做进一步分解为止。一开始他们定义一个模板，覆盖图像中的所有像素。给定任意模板，计算图像上被覆盖区域的直方图。对该直方图应用度量空间聚类技术，生成一组聚类。然后对图像中的像素进行聚类标注。如果只有一个度量空间聚类，就终止当前模板。如果不止一个聚类，就对每个聚类进行连通成分标记运算，对应每个聚类标号会产生几个连通区域，用每个连通成分生成一个新模板，新模板放在模板栈中。模板栈中的模板表示需要进一步分割的区域。在迭代过程中，栈中的下一个模板覆盖要进行直方图运算的像素。对每个新模板重复聚类直到栈空为止。图10-8显示这个聚类过程，我们称之为面向直方图的空间递归聚类。

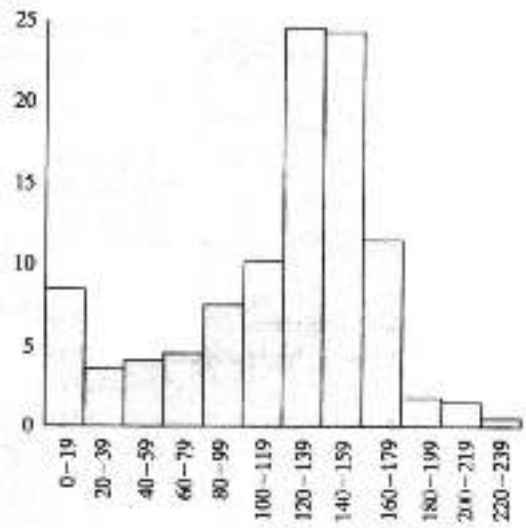


图10-6 表10-2积木图像的直方图



图10-7 根据积木图像的直方图，人工选择3个阈值，得到4幅阈值化图像

对于一般的彩色图像，Ohta、Kanade和Sakai（1980）建议不要直接对红、绿、蓝（RGB）颜色变量计算直方图，而应该先进行变换，该变换接近于Karhunen-Loeve（主成分）变换，再计算各变量的直方图，其中变换方式为 $(R+G+B)/3$，$(R-B)/2$和 $(2G-R-B)/4$。

6. Shi的图分割技术*

Ohlander/Ohta算法，对于包含人工目标和单色区域的简单颜色场景效果不错，但对于复杂的自然场景使用效果则不尽人意，因为这类图像中带纹理的部分存在大量的小块区域。Shi和Malik（1997）提出一种分割方法，利用颜色、纹理或者结合使用颜色和纹理及其他特性进行分割。他们将分割问题形式化为图分割问题，并提出一种新的图分割方法，该方法将问题化解为求如下特征向量和特征值的问题。

284
285

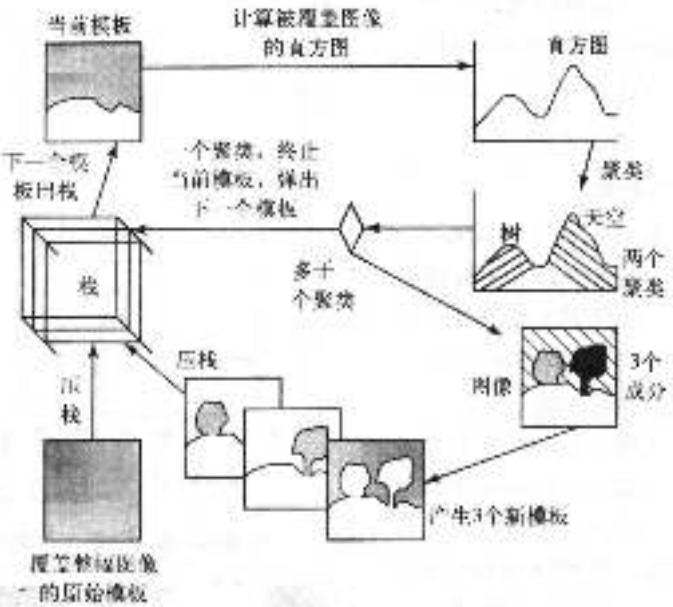


图10-8 面向直方图的空间递归聚类。原始图像有4个区域：草地、天空和两棵树。当前模板（左上角所示）识别出包含天空和树的区域。对它的直方图聚类产生颜色空间的两个聚类：一个是天空，一个是树。天空聚类成一个连通成分，树聚类成两个连通成分。每个连通成分成为新的模板，被压入模板栈中以便进一步地分割

设$G = (V, E)$是一个图，它的节点是度量空间中的点，它的每条边都有一个权值$w(i, j)$，表示节点i和j的相似度。分割的目标是将顶点划分成不相交的集合$V_1, V_2, \cdots, V_m$，这样使得集
286 合内的相似度较高，而集合间的相似度较低。

图$G = (V, E)$可被分成两个不相交的图，其节点集合分别记为A和B，方法是去掉A中节点到B中节点之间的连接边。两个集合A和B之间的不相似程度，可用去掉边的权值之和来表示，这个总权值称为切痕（cut）。

$$cut(A, B) = \sum_{u \in A, v \in B} w(u, v) \tag{10-2}$$

把分割问题形式化的一种方法，是寻找图中的最小切痕（minimum cut），不断重复这个步骤直到区域足够一致。但是最小切痕准则倾向于分割成较小的孤立节点的集合，这在寻找相同颜色或纹理的大块区域时没有作用。Shi根据$cut(A, B)$的定义提出了规范化切痕（normalized cut, Ncut），A和整个顶点集合V的关联度（association）定义为：

$$asso(A, V) = \sum_{u \in A, t \in V} w(u, t) \tag{10-3}$$

287

则规范化切痕定义为：

$$Ncut(A, B) = \frac{cut(A, B)}{asso(A, V)} + \frac{cut(A, B)}{asso(B, V)} \tag{10-4}$$

根据这个定义，分割出较小孤立点集的切痕将不具有较小的规范化切痕值，使规范化切

痕值较小的划分在图像分割中更加实用。另外，总规范化关联度（normalized association）由下式给出:

$$Nasso(A, B) = \frac{asso(A, A)}{asso(A, V)} + \frac{asso(B, B)}{asso(B, V)} \tag{10-5}$$

上式表示给定集合内的节点之间相连接的紧密程度。它与规范化切痕具有如下关系:

$$Ncut(A, B) = 2 - Nasso(A, B) \tag{10-6}$$

在分割过程中可以根据需要使用上述定义中的任何一个。

给出规范化切痕和总规范化关联度的定义，还需要通过分割像素集合实现对图像的分割计算。Shi的分割过程参见算法10.3。

Shi利用该算法对图像进行分割，分别基于图像亮度、颜色和纹理信息。连接边的权值 $w(i, j)$ 定义为:

$$w(i, j) = e^{\frac{-\|F(i)-F(j)\|_2}{\sigma_I}} * \begin{cases} e^{\frac{-\|X(i)-X(j)\|_2}{\sigma_X}} & \text{当} \|X(i) - X(j)\|_2 < r \\ 0 & \text{其他} \end{cases} \tag{10-7}$$

其中

- $X(i)$ 是节点i的空间位置。
- $F(i)$ 是基于亮度、颜色和纹理信息的特征向量，定义如下:

$F(i) = I(i)$，图像亮度值，用于分割亮度图像。

$F(i) = [v, v \cdot s \cdot \sin(h), v \cdot s \cdot \cos(h)](i)$，其中$h$、$s$和$v$是HSV值，用于颜色分割。

$F(i) = [|I * f_1|, \cdots, |I * f_n|](i)$，其中$f_i$是在不同尺度和方向上的高斯滤波器的二次差分（difference of difference of Gaussian, DOOG），用于纹理分割。

注意对大于预定像素数r的节点对i和j，权值$w(i, j)$ 设为0。

算法10.3利用颜色和纹理信息能够得到很好的图像分割结果。图10-9显示该算法对自然图像的分割效果。虽然分割结果很好，但算法过于复杂，对实时系统不适用。 288

算法10.3 Shi的聚类过程。图的节点表示像素，图的边表示像素对之间的相似程度

1. 建立权连接图$G = (V, E)$，其节点集V是图像像素的集合，边集合E是权值为$w(i, j)$的一组边的集合，$w(i, j)$表示从节点i到j之间的边连接权，通过该权值计算i的度量空间向量与j的度量空间向量之间的相似度。N表示节点集合V的大小。定义向量d，其分量$d(i)$如下

$$d(i) = \sum_j w(i, j) \tag{10-8}$$

这样$d(i)$表示从节点i到所有其他节点的总连接权。设D是一个$N \times N$的对角矩阵，其对角向量为d。设W是一个$N \times N$的对称矩阵，$W(i, j) = w(i, j)$。

2. 设x是一个向量，其元素定义为

$$x_i = \begin{cases} 1 & \text{当node } i \text{在} A \text{中} \\ -1 & \text{其他} \end{cases} \tag{10-9}$$

设y是对x的连续逼近，定义为

$$y = (1 + x) - \frac{\sum_{x_i>0} d_i}{\sum_{x_i<0} d_i}(1 - x) \tag{10-10}$$

求下列矩阵方程的特征向量y和特征值λ

$$(D-W)y=\lambda Dy \tag{10-11}$$

3. 利用第二小的特征值对应的特征向量将图分成两部分，找到使得规范化切痕最小的划分点[⊖]。

4. 通过检查切痕的稳定性并保证规范化切痕低于预定的阈值，决定是否需要对当前的划分结果做进一步的分割。

5. 如果必要，对分割后的部分再次进行划分。

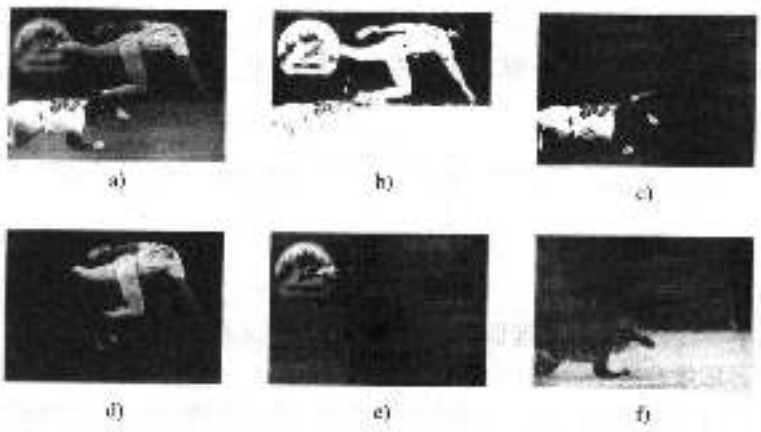


图10-9 a是原始灰度图像，用Shi的分割方法得到区域图像b－f。在结果图b中，选择的区域是深色背景区域，用黑色表示。在其他结果图中，选择的区域用原来的灰度值表示，其余部分用黑色（由Jianbo Shi提供）

10.1.2 区域增长

与划分图像不同，区域增长（region grower）从图像某个位置（通常是左上角）开始，并使每块区域变大，直到被比较的像素与区域像素具有显著差异为止。一般通过用统计检验来决定是否具有显著差异。Haralick与Shapiro（1985）提出下面的区域增长算法，称为Haralick区域增长算法。该算法假设区域是具有相同群体均值和方差的连通像素集合。

289

设某像素的亮度值为y，其邻域用R表示，邻域内包含N个像素。定义区域均值$\overline{X}$和散度S^2为：

$$\overline{X}=\frac{1}{N}\sum_{[r,c]\in R}I[r,c] \tag{10-12}$$

以及

$$S^2=\sum_{[r,c]\in R}(I[r,c]-\overline{X})^2. \tag{10-13}$$

假设R中的所有像素与测试像素y是相互独立的，且具有相同的分布态，下面的统计量服从T_{N-1}分布，

$$T=\left[\frac{(N-1)N}{(N+1)}(y-\overline{X})^2/S^2\right]^{\frac{1}{2}} \tag{10-14}$$

⊖ Shi认为广义特征系统的第二最小特征向量是规范化切痕问题的实值解。

如果T足够小，y就加入到区域R，利用y对均值和散度进行更新。新的均值和散度如下：

$$\overline{X}_{new} \leftarrow (N\overline{X}_{old} + y)/(N+1) \tag{10-15}$$

以及

$$S^2_{new} \leftarrow S^2_{old} + (y - \overline{X}_{new})^2 + N(\overline{X}_{new} - \overline{X}_{old})^2. \tag{10-16}$$

如果T过高，y值不太可能是属于R中的像素，如果y与所有的邻域都不同，那么它就开始一个新的区域。稍严格的连接指标，不仅要求y必须与邻域的均值足够接近，而且要求该区域 290
中的一个邻点必须与y的值足够接近。

为给出显著不同的精确涵义，可以利用α水平统计进行显著性测试，分数α表示自由度为$N-1$的T统计超过值$t_{N-1}(\alpha)$的概率。如果观测到的T大于$t_{N-1}(\alpha)$，那么就说差别是显著的。如果像素和分割区域确实来自同一群体，那么测试提供不正确答案的概率是α。

显著水平α是用户提供的一个参数，对较小的自由度，$t_{N-1}(\alpha)$的值较高；对较大的自由度，$t_{N-1}(\alpha)$的值较低。如果区域散度是相等的，区域越大，像素值离区域均值就越接近，这样才能将像素合并到区域中，这种行为有阻止大区域吸收其他像素的趋势，当区域变大时也有阻止区域均值漂移的趋势。图10-10显示Haralick区域增长的运算过程。

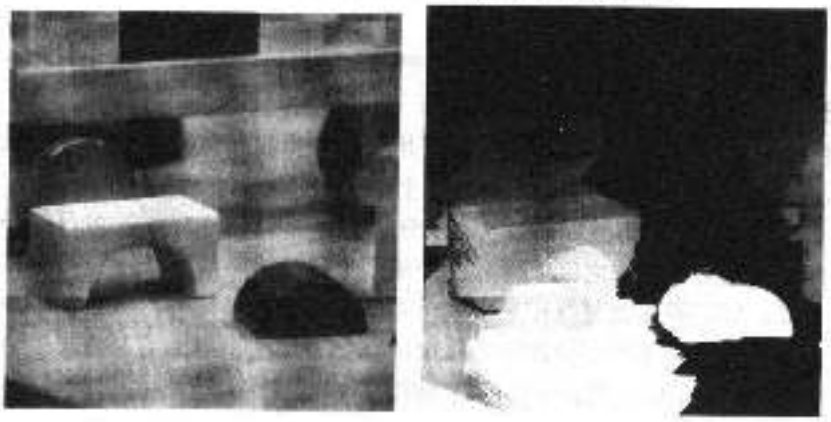


图10-10 （积木图像由John Illingworth和Ata Etemadi提供，分割运算采用GIPSY图像处理系统）
（左）积木图像
（右）利用Haralick区域增长算法得到的分割区域

习题10.3 区域增长

编程实现Haralick区域增长算法，并用它分割灰度图像。

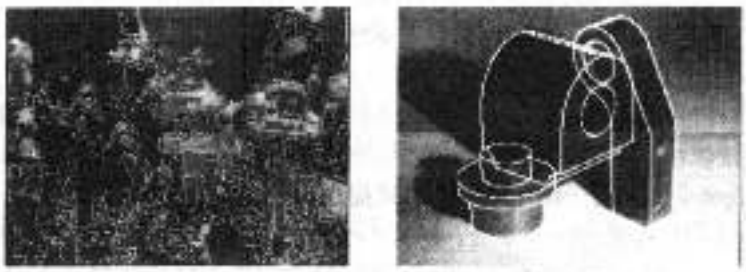
10.2 区域表示

每种生成图像区域的算法，必须有相应的方法对图像区域进行存储以备后用。存储方式包括原图上的覆盖图、标记图像、边界编码、四叉树结构和特征表。标记图像是最常用的表示方法。下面介绍这几种表示方法。 291

10.2.1 覆盖图

覆盖图是显示图像分割区域的一种方法，它在原图上覆盖一种或多种颜色。许多图像处理系统都提供这种操作，作为图像输出过程的一部分。通常，原图是灰度图像，覆盖的颜色

是与灰度明显不同的颜色，如红色或白色。为显示分割得到的区域，可将区域边界的像素变换成白色，并显示变换后的灰度图像。有时为了使区域边界更明显，可使边界的宽度多于一个像素。图10-11a显示所选暗区域的边界，包括深蓝色的裁判员外衣和运动员的编号，覆盖在原始的灰度图像上。覆盖图的另一种应用是突出图像中的某种特征。图10-11b是第1章的工业零件图像，其中识别到的目标模型投影覆盖在原始灰度图像上。



a) b)

图10-11 覆盖图举例

a）选择的区域边界覆盖在橄榄球图像上

b）3D目标线框模型覆盖在工业零件图像上（Mauro Costa提供）

10.2.2 标记图像

标记图像是一种很好的区域表示方法，可用于进一步的图像处理过程。其思想是为每块检测到的区域赋予一个唯一的标号（一般是一个整数），并建立一幅图像，其中区域内的所有像素都用唯一的标号作为像素值。多数连通成分算法（见第3章）的输出就是标记图像。在有的运算中，标记图像可作为选定区域像素的模板，从而算出区域的特征，如面积或最佳拟合椭圆的主轴长度。标记图像也可以用灰度或伪彩色显示。如果标号的整数值较小，灰度图像显示时看起来都是黑色，可通过拉伸标记图像或直方图均衡化得到更好的灰度分布。本章前面的橄榄球分割图像就是以灰度表示的标记图像。

10.2.3 边界编码

区域也可用边界而不是图像来表示，这些边界存储为某种数据结构。最简单的形式是区域边界像素的线性链表（参见本章后面的边界抽取过程，从标记图像抽取区域边界）。点链表的一种变形是**弗里曼链码**（Freeman chain code），它可根据点链表以任何量化程度进行信息编码，这比原来的点链表占用更少的空间。概念上看，被编码的边界覆盖在一个方格上，方格的边长决定了编码的分辨率。从曲线的起始点开始，利用与边界点最近的栅格交点定义直线段，该直线段把相邻的两个栅格点连接起来，用一个小整数对这些直线段的方向进行编码，该整数取值范围是从0到编码用到的邻点个数。图10-12显示的是8-邻域的链码，0°的直线段编码为0，45°的直线段编码为1，以此类推下去，直到315°的直线段编码为7。图中的小六边形表示闭合曲线的开始，其余的栅格交点以菱形表示。起始点的坐标加上链码，足以在所选栅格分辨率上重构出该曲线。链码不仅节省空间，也可用于曲线自身的后续处理，如基于形状的目标识别。当一块区域不仅有一个外边界，且有一个或多个内孔边界时，可分别用链码表示每个边界。

292

当不需要抽取边界时，边界像素可用直线段近似，形成对边界的多边形逼近（polygonal

approximation)，如图10-12的右下图所示。这种表示可节省空间并简化处理边界的算法。 293

10.2.4 四叉树

四叉树（quadtree）是另一种节省空间的区域表示方法，它对整个区域编码，而不只是边界。一般对每个感兴趣区域都用一个四叉树结构来表示。每个四叉树的节点表示图像中的一个方块区域，它具有三个标记之一：满（full）、空（empty）和混合（mixed）。如果节点标记为满，那么该节点表示的方块区域中的每个像素都是感兴趣区域的像素；如果节点标记为空，那么在方块区域与感兴趣区域之间没有交集；如果节点标记为混合，那么方块区域中有的像素是感兴趣区域中的像素，而有一些则不是。四叉树中只有混合节点有子节点。满节点和空节点都是叶子节点。图10-13显示图像区域的四叉树表示方法。区域看起来呈块状，因为图像的分辨率仅仅是8×8，这就产生一个四层的四叉树。要使曲线边界光滑，则需要更多的层数。地理信息系统中就采用四叉树表示地图区域。

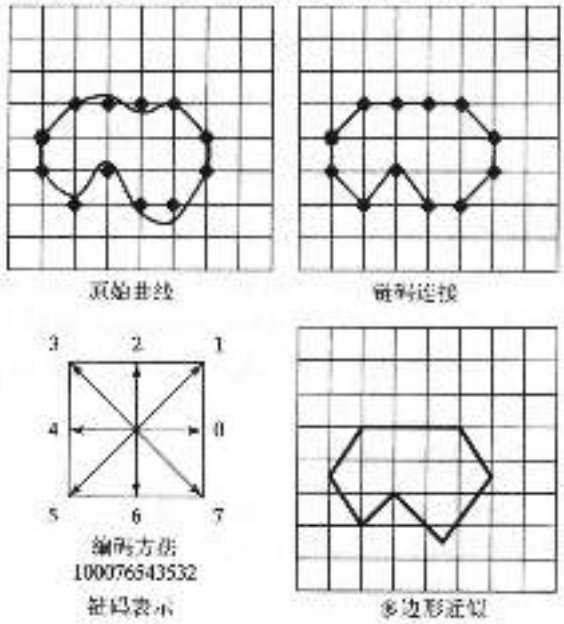


图10-12 两种边界编码方法：链码和多边形逼近。链码编码采用8个符号表示直线段的8个可能的角度，这些直线段逼近栅格上的曲线。多边形逼近采用直线段来拟合原始曲线，直线段的端点具有实值坐标，并不受原始栅格点的限制

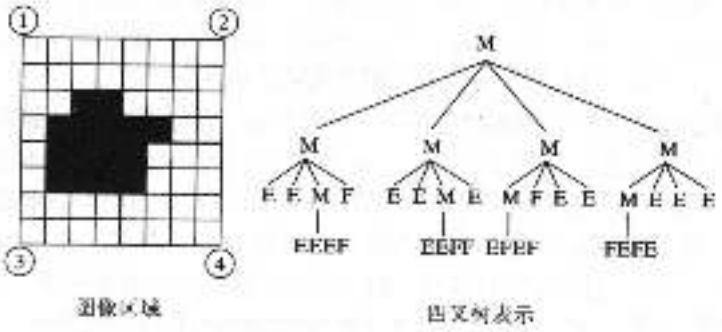


图10-13 图像区域的四叉树表示。对于树的第一层，节点有四个子节点，分别对应左上、右上、左下和右下分区，如图中圆圈中的数字所示。M = mixed，E = empty，F = full

10.2.5 特征表

有时希望用区域特征来表示区域，而不是用它的像素来表示。这种情况下的表示就称为特征表（property table）。在关系数据库的意义上它是一个表，其中行表示图像中的每块区域，列表示感兴趣的特征。特征可以是区域的大小、形状、亮度、颜色或者纹理。在第3章、第6章和第7章中描述的特征都是可能的选择。例如，在基于内容的图像检索系统中，区域可能通过面积、最佳拟合的椭圆主轴和次轴之比、两种主要颜色、一种或多种纹理测度等来表示。

294 特征表可以增加内容，以包括或者指向区域的链码编码或四叉树表示。

习题10.4 计算面积和周界

考虑以（a）标记图像和（b）链码表示的图像区域。

1. 给出计算区域面积和周界的算法。

2. 给出算法的运行时间。

习题10.5 测试像素是否在区域中

考虑用（a）标记图像和（b）边界的多边形逼近表示的图像区域。

1. 对每种情况，给出测试像素$[r, c]$是否属于该区域的算法。

2. 给出算法的运行次数，用合适的参数表示，如区域中的像素个数或多边形逼近的线段个数。

10.3 轮廓分割

有的图像分析直接针对区域进行运算，有的针对区域边界或其他结构，如直线段或圆弧段。本节讨论如何从图像中抽取这些结构。

10.3.1 区域边界跟踪

一旦确定了区域，如通过分割得到的区域或者连通成分标记的区域，就可以抽取出区域的边界。对于小尺寸图像，抽取边界很容易。扫描图像，对每个连通成分，建立第一个边界像素的列表，然后对每块区域，从第一个边界像素开始，沿着顺时针方向跟踪连通成分的边界，直到回到第一个边界像素为止。对于不在内存中的大尺寸图像，由于要访问大量的外存设备，这时利用简单的边界跟踪算法，会造成过多的I/O操作。

下面介绍一种称为边界查找（border）的算法，它从左到右、从上到下扫描一遍图像，就能抽取出所有区域的边界。该算法输入是标记图像，输出是区域边界像素顺时针方向的坐标列表。这个算法很灵活，对它稍加修改就可用于选择特定区域的边界。

边界查找算法的输入是一幅标记图像，其像素值表示区域的标记。假设用背景区域标记表示属于背景区域的像素，这些背景区域可能不是连通的，它们的边界不需要检测。边界查找算法不是对一块区域的边界跟踪完成后再移向下一块区域，而是对图像进行从左到右、从上到下的扫描，搜集组成区域边界连接线段的边界像素链。在算法执行期间，共当前区域
295 （current region）的部分边界已经扫描过，但尚未产生输出，但过去区域（past region）已经完全扫描过并生成了边界输出，而未来区域（future region）尚未扫描到。

数据结构包括当前区域的边界像素链。由于图像中可能有大量的区域标记，但一次最多只能有$2 \times number_of_columns$个区域处于活跃状态，可以采用一个散列表，已知区域标记时就能够快速访问区域链。（$2 \times number_of_columns$是安全上限，实际区域数会少一些。）当完成对一个区域的扫描并产生输出后，则从散列表中去除该区域。如果在扫描中遇到一个新区域，则将它加入散列表。区域散列表的入口指向该区域链的连接表。区域链是关于像素位置的连接表，可以从始点或终点开始生长。

跟踪算法一次检查标记图像的三行，即正在处理的当前行、上一行和下一行。对于图像的最上一行和最下一行，添加两行虚拟的背景像素，这样所有行都可按同样方法处理。对于NLINES × NPIXELS的标记图像S的算法参见算法10.4。

在这个过程中，S是标记图像的名字，这样S[R, C]是当前被扫描的像素值（LABEL）。如果这是一个新标记，就将它加入到当前区域标记的集合CURRENT中。NEIGHB是具有标记LABEL的像素[R, C]的邻点列表。函数*pixeltype*检查[R, C]和它的邻点值，决定[R, C]是否是非背景的边界像素。如果是，这个过程搜索具有标记LABEL的区域链，其末尾有[R, C]的邻点，如果找到一个，用过程*add*把[R, C]追加到链的末尾，*add*的第一个参数是区域链，第二个参数是[R, C]。如果在区域链的末尾没有[R, C]的邻点，则用过程*make_new_chain* 创建一个新的区域链，它仅包含一个元素[R, C]，该过程的第一个参数是加了新链的链集合，这个新链的唯一元素是位置[R, C]，位置[R, C]是过程的第二个参数，第三个参数是与链关联的标记LABEL。

当每一行R都扫描后，把边界已知的当前区域链合并成单个的边界链，作为输出，然后释放与这些区域关联的散列表入口和列表元素。图10-14显示标记图像以及由边界查找算法得到的输出。

	1	2	3	4	5	6	7
1	0	0	0	0	0	0	0
2	0	0	0	0	2	2	0
3	0	1	1	1	2	2	0
4	0	1	1	1	2	2	0
5	0	1	1	1	2	2	0
6	0	0	0	0	2	2	0
7	0	0	0	0	0	0	0

a）具有两个区域的标记图像

区域	长度	列表
1	8	(3, 2)(3, 3)(3, 4)(4, 4)(5, 4)(5, 3)(5, 2)(4, 2)
2	10	(2, 5)(2, 6)(3, 6)(4, 6)(5, 6)(6, 6)(6, 5)(5, 5)(4, 5)(3, 5)

b）标记图像的边界运算输出

图10-14 边界查找算法对标记图像的运算结果

算法10.4 寻找标记图像S的区域边界

S[R, C]是输入标记图像。
NLINES是图像的行数。
NPIXELS是图像每行像素的个数。
NEWCHAIN是一个标志，当像素开始一个新链时该值为真，当一个新像素被加到现存链上时该值为假。

```
procedure border(S);
{
  for R:= 1 to NLINES
     {
      for C:= 1 to NPIXELS
        {
        LABEL:= S[R, C];
        if new-region(LABEL) then add(CURRENT, LABEL);
        NEIGHB:= neighbors(R, C, LABEL);
        T:= pixeltype(R, C, NEIGHB);
        if T == 'border'
        then for each pixel N in NEIGHB
           {
```

```
                CHAINSET:= chainlist(LABEL);
                NEWCHAIN:= true;
                for each chain X in CHAINSET while NEWCHAIN
                  if N==rear(X)
                  then {add(X, [R. C]); NEWCHAIN:= false}
                if NEWCHAIN
                then make_new_chain(CHAINSET, [R. C], LABEL);
                }
              }
            for each region REG in CURRENT
              if complete(REG)
              then {connect_chains(REG); output(REG); free(REG)}
          }
      }
```

习题10.6 边界跟踪算法的局限

边界跟踪算法对要跟踪的区域做了一定的限制。在什么情况下，它无法正确识别区域的边界？

10.3.2 Canny边缘检测和连接

Canny边缘检测算子和连接算子能够从图像中抽取边缘线段。在第5章中与其他边缘检测算子一起，我们简要介绍过Canny算子。Canny算子很常用，近期对边缘算子的比较工作说明了它应用的普遍性。Canny算子的应用例子在第5章曾提到过。图10-15是从第2章的大图像中抽取的两个汽车零件的图像。可以看出边缘检测和边界跟踪算法存在的众所周知的问题：实际目标的轮廓线段，与光照或反射造成的边界轮廓段交错在一起。这样的轮廓，很难用通用的目标识别系统进行自下而上的分析，但是对于特定的目标模型，对这种表征进行自上而下的匹配则可以成功进行，在后面的章节中我们会看到这一点。因此，图像边缘表征的质量，与它们在整个机器视觉系统中的应用情况有关。

296 ↓ 297

298

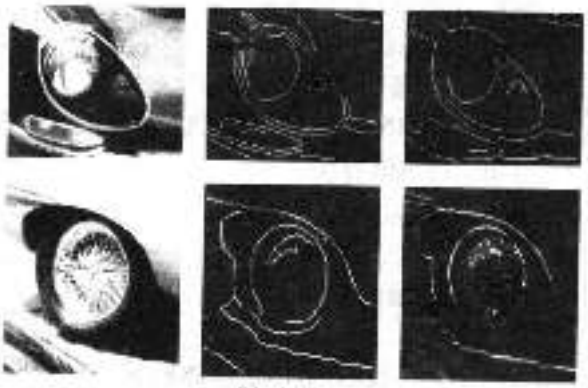


图 10-15

（左上）黑色轿车前灯的图像
（中上）$\sigma=1$的Canny算子运算结果
（右上）$\sigma=4$的Canny算子运算结果
（左下）车轮图像
（左中）$\sigma=1$的Canny算子运算结果
（右下）Robert算子的运算结果。上行中左上角，由于存在镀面反射，使边缘检测算子无法很好地检测到前灯轮廓处的边沿。在下行中，轮胎与档泥板相连，注意车的影子也与轮胎相连，结果轮胎和幅条部未能很好地检测到

算法10.5中的Canny边缘检测算法产生细化的图像轮廓，仅由一个平滑参数σ控制。图像首先用散差为σ的高斯滤波器做平滑处理。在平滑后图像的每个像素处计算梯度幅值和方向。梯度方向用来细化边缘，如

果像素响应不高于梯度方向上它的两邻点的像素响应，则抑制该像素响应，从而使边缘得到细化，这种方法称为非最大抑制（nonmaximum suppression）。当需要进行边界细化时，这种方法可和任意边缘算子共同使用。要与像素[x, y]进行比较的两个8-邻点，其寻找方法是将算出的梯度方向取整，在中心像素的两边各得到一个邻点。梯度幅值被细化后，就开始跟踪具有高幅值的轮廓。在最后的综合阶段，按顺序跟踪连续的轮廓段。选择轮廓跟踪初始点时，只选择梯度幅值满足高阈值的边缘像素。但是，一旦开始跟踪，轮廓也可能通过梯度幅值满足低阈值的像素点，低阈值通常是高起动阈值的一半。

当边界段本身是闭合的，有时就能检测出图像区域。图10-16和10-17就是这样的实例。将边界像素的集合分成直线或圆圈后，可对这些分割结果进一步分析。例如，矩形建筑物的边界可能产生四条直线段。识别直线段的方法，可采用霍夫变换或直接拟合直线的参数模型。 299

算法10.5 Canny边缘检测：计算输入图像的细化连通边缘

I[x, y]：输入亮度图像；σ高斯平滑处理的散差。
E[x, y]：输出二值图像。
IS[x, y]：要平滑的亮度图像。
Mag[x, y]：梯度幅值；**Dir[x, y]**：梯度方向。
$\mathbf{T_{low}}$是低亮度阈值；$\mathbf{T_{high}}$是高亮度阈值。

```
procedure Canny(I[], σ) ;
{
    IS[] = image I[] smoothed by convolution with Gaussian G_σ(x, y);
    use Roberts operator to compute Mag [x, y] and Dir[x, y] from IS[];
    Suppress_Nonmaxima (Mag[], Dir[], T_low, T_high) ;
    Edge_Detect (Mag[], T_low, T_high, E[]);
}
procedure Suppress_Nonmaxima (Mag[], Dir[]) ;
{
    define + Del[4] = (1, 0), (1, 1), (0, 1) (−1, 1);
    define − Del[4] = (−1, 0), (−1−,1), (0, −1) (1, −1);
      for x := 0 to MaxX-1;
      for y := 0 to MaxY-1;
       {
       direction := ( Dir[x, y] +π/8 ) modulo π/4;
       if (Mag[x, y] ≤ Mag[(x, y) + Del[direction]]) then Mag[x, y] := 0;
       if (Mag[x, y] ≤ Mag[(x, y) + −Del[direction]]) then Mag[x, y] := 0;
       }
}
procedure Edge_Detect(Mag[], T_low, T_high, E[] );
{
    for x := 0 to MaxX - 1;
    for y := 0 to MaxY - 1;
          {
```

```
            if (Mag[x, y] ≥ (T_high) then Follow_Edge(x, y, Mag[],T_low, T_high, E[]);
        }
}
procedure Follow_Edge (x, y, Mag[], T_low, T_high, E[] );
{
    E[x, y] := 1;
    while Mag[u, v] > T_low for some 8-neighbor [u, v]of[x, y]
      {
        E[u, v] := 1;
        [x, y] := [u, v];
      }
}
```



图10-16 识别图像的符号区域一般比较容易，因为这样的区域对比度高。这些图是应用Canny算子的结果

（左边）用墨水写在纸上的字（图像由John Weng提供）

（右边）砖墙上风化的字



图10-17 北京毛主席纪念堂图片，以及用Canny算子$\sigma = 1$和$\sigma = 2$抽取的轮廓图。其中几个目标的检测效果很好，但也检测出一些阴影

习题10.7

考虑Canny边缘检测算法的轮廓跟踪过程。通过追踪具有高梯度幅值的像素，从而得到图像轮廓。其中只选择与梯度方向垂直的两邻点作为下一个跟踪目标点，这样做合适吗？为什么？举例验证你的答案。

习题10.8 利用Canny算子进行边缘检测

做下面实验。寻找Canny边缘检测的程序或者具有Canny算子的图像处理工具，找一些具

有平行边缘的扁平物体如刀片，一些圆形物体如钻头柄。从不同的方向拍取它们的图像。如果可能，进行高分辨率扫描。对这些图像进行Canny边缘检测，对边缘的质量进行分析，包括平行边缘之间距离的可重复性。对于刀片的“锐利边缘”，和钻头的“柔和边缘”，检测结果有什么不同?

10.3.3 相邻连贯的边缘生成曲线

10.3.1节的边界跟踪算法要求输入表示区域集合的标记图像。当遍历图像时，算法沿着每块区域的边界逐行跟踪，由于假设每条边界对应一个闭合区域，因此不存在把边界分成两段或多段的像素点。如果输入是做了标记的边缘图像，即边缘像素值为1而非边缘像素值为0，则跟踪边缘线段的问题就更加复杂。这里的边缘像素不一定是封闭区域边界上的点，并且线段由连通的边缘像素组成，而这些线段从端点、角点或连接点到端点、角点或连接点结束，中间没有其他连接点或角点。图10-18显示的就是这样的一幅标记边缘图像。图像中的像素[3,3]是三条边缘线段的连接点。像素[5,3]是一个角点，如果要求线段在角点处结束，那么它也可视为线段端点。算法在跟踪这些线段时必须考虑下面的任务要求:

(1) 开始一条新线段。

(2) 给线段加入一个内点像素。

(3) 结束一条线段。

(4) 检测连接点。

(5) 检测角点。

和边界跟踪相同，需要采用有效的数据结构来管理过程中每一步的信息。采用的数据结构与边界查找算法中采用的非常相似，不过现在不是“过去”、“当前”和“未来”区域，而是“过去”、“当前”和“未来”线段。线段是表示图像中直线或曲线的边缘点列表。当前线段保存在内存中，通过散列表访问。完成的线段被存入磁盘，同时释放它们在散列表中占的空间。主要差别表现在连接点和线段的检测方法上，线段从上面或左边开始，从下面或右边结束。定义一个扩展的邻域算子，称为*pixeltype*，它决定像素是否是孤立点、新线段的起始点、旧线段的内点、旧线段的终点、连接点或者角点。如果像素是内点或者旧线段的终点，那么也要返回旧线段的ID号。如果像素是连接点或角点，则返回进入线段的ID列表(INLIST)和离开线段的像素列表(OUTLIST)。标记图像上的边缘跟踪过程参见算法10.6。图10-19是对图10-18的标记图像进行边缘跟踪的结果。

	1	2	3	4	5
1	**1**	0	0	0	**1**
2	0	**1**	0	**1**	0
3	0	0	**1**	0	0
4	0	0	**1**	0	0
5	0	0	**1**	**1**	**1**

图10-18 标记边缘图像，三条线段相交于连接点[3, 3]，像素点[5, 3]可能是角点

线段ID号	长度	列表
1	3	(1, 1)(2, 2)(3, 3)
2	3	(1, 5)(2, 4)(3, 3)
3	3	(3, 3)(4, 3)(5, 3)
4	3	(5, 3)(5, 4)(5, 5)

图10-19 对图10-18进行边缘跟踪的结果。假设点[5, 3]被判断为角点。如果角点不是线段终点，则线段3的长度为5，其像素列表为: [3, 3][4, 3] [5, 3] [5, 4][5, 5]

其中省略了在连接点处对进入线段和离开线段ID号的跟踪细节。这部分算法非常简单，假设与连接点邻近的每个像素都属于不同的线段，在这种情况下，如果线段宽度大于一个像素，算法将检测到很多小线段，其实它们并不是新的线段。对边缘图像应用连通收缩算子，就可以避免这种情况。另一种方法是，让*pixeltype*算子更聪明一些。它可以观察更大的邻域，利用启发式规则确

300
↓
302

定这是当前线段较粗的部分，还是新线段的开始。一般根据实际应用建立这些启发式规则。

算法10.6 寻找二值边缘图像S中的线段

S[R, C]是输入标记图像。
NLINES是图像的行数。
NPIXELS是每行像素的个数。
IDNEW是最新线段的ID。
INLIST是由pixeltype返回的进入线段的ID列表。
OUTLIST是由pixeltype返回的离开线段的ID列表。

```
procedure edge_track(S);
{
IDNEW := 0;
for R:= 1 to NLINES
  for C:= 1 to NPIXELS
    if S[R,C] ≠ background pixel
    {
    NAME := address (R, C); NEIGHB:= neighbors (R, C);
    T:= pixeltype(R, C, NEIGHB, ID, INLIST, OUTLIST);
    case
      T = isolated point : next;
      T = start point of new segment: {
        IDNEW := IDNEW + 1;
        make_new_segment(IDNEW, NAME); } ;
      T = interior point of old segment : add(ID, NAME);
      T = end point of old segment : {
        add(ID, NAME);
        output(ID); free(ID) } ;
      T = junction or corner point:
        for each ID in INLIST {
          add(ID, NAME);
          output(ID); free(ID); } ;
        for each pixel in OUTLIST{
          IDNEW := IDNEW + 1;
          make_new_segment(IDNEW, NAME); } ;
    }
}
```

习题10.9 决定像素的类型

给出算子pixeltype的代码，利用像素的3 × 3邻域将像素分成这几类：孤立点、起点或终点、内点、连接点和角点。

10.3.4 用霍夫变换检测直线和圆弧

霍夫变换（Hough transform）是检测灰度（或彩色）图像中直线和曲线的一种方法。给定所求的曲线族，产生图像中出现的属于该族的曲线集合。本节讨论霍夫变换技术，并用它检测图像中的直线段和圆弧段。

1. 霍夫变换技术

霍夫变换算法需要一个累加数组，数组的维数与所求曲线族方程中未知参数的个数对应。例如，检测直线段$y = mx + b$，对每个线段要求两个参数：m和b。该直线族累加数组的两个维数，对应m的量化值和b的量化值。累加数组累计直线$y = mx + b$在箱格**A**[**M**, **B**]范围存在的证据，其中**M**和**B**分别是m和b的量化值。

利用累加数组**A**，霍夫变换检查图像中的每个像素及其邻域。先决定是否有足够的证据证明该像素是边缘点，如果是，则计算通过该像素的某种曲线的参数。对于直线段$y = mx + b$，如果像素的边缘强度测度（比如梯度）足够高的话，则估计通过该像素的直线的m和b。一旦估计出给定像素的参数，再将参数量化到对应值**M**和**B**，累加数组**A**[**M**, **B**]加上一个增量。有的方法是加1，有的方法是加上被处理像素的梯度大小。处理完所有像素后，查找累加数组的峰值。峰值对应图像中最有可能的直线参数。

累加数组中包含无限的直线（或曲线）参数，并未说明实际线段的起点和终点。为得到该信息，添加称为**PTLIST**的并行结构。**PTLIST**[**M**, **B**]包含对累加器**A**[**M**, **B**]的结果有贡献的所有像素位置的列表。从这些列表可以确定实际线段。

上面描述的是一般霍夫方法，未涉及实现的细节。下面详细讨论直线检测和圆检测的霍夫算法。

2. 直线段检测

直线方程$y = mx + b$对垂直线不起作用。更好的模型是方程$d = x\cos\theta + y\sin\theta$，其中$d$是从直线到原点的垂直距离，$\theta$是垂线与$x$轴的夹角。我们就采用这种方程形式，但要转化到行$r$和列$c$坐标。由于列坐标$c$与$x$对应，行坐标$r$与$-y$对应，则方程变为：

$$d = c\cos\theta - r\sin\theta \qquad (10\text{-}17)$$

其中d是从直线到图像原点（假设位于左上角）的垂直距离，θ是垂线与c（列）轴的夹角。图10-20显示了直线段的参数。假设从原点到线段的垂线，与线段交于点[50, 50]，$\theta = 315°$。那么我们有

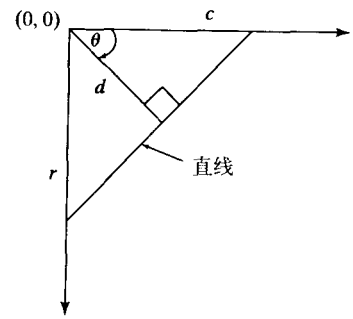


图10-20 直线方程$d = -r\sin\theta + c\cos\theta$的参数$d$和$\theta$

$$d = 50\cos(315) - 50\sin(315) = 50(0.707) - 50(-0.707) \approx 70$$

累加器**A**的下标，对应d和θ的量化值。O'Gorman和Clowes（1976）在他们的实验中，对木偶的灰度图像取d的量化间隔为3s，θ的量化间隔为10°。以这种方式量化后的累加数组如图10-21所示。填充累加器**A**的O'Gorman和Clowes算法和并行数组**PTLIST**的算法，参见后面的过程*accumulate_lines*。

算法在（行，列）空间表示。函数*row_gradient*和*column_gradient*分别是估计行梯度分量和列梯度分量的邻域函数，函数*gradient*根据行、列梯度分量得到梯度幅值。函数*atan2*是标准

303
~
304

科学计算库函数，该函数根据梯度的行列分量返回位于正确象限的角度。假设*atan2*的返回值处于0°和359°之间。很多函数返回的角度以弧度表示，这还需要转化为角度。如
305 果距离d得出负值（例如对于$\theta = 135°$），那么它的绝对值表示到直线的距离。这个过程的作用原理参见图10-22。注意采用3 × 3的梯度算子，直线是两个像素宽。同时要注意计数不是在两个标准累加器内进行，而是在其他累加器中进行。

过程*accumulate_lines*采用的是O'Gorman和Clowes霍夫变换方法。累加器和列表数组得到填充，当然不存在抽取直线段的标准方法。特殊过程*find_lines*参见算法10.8，显示出直线段抽取过程中出现的一些问题。

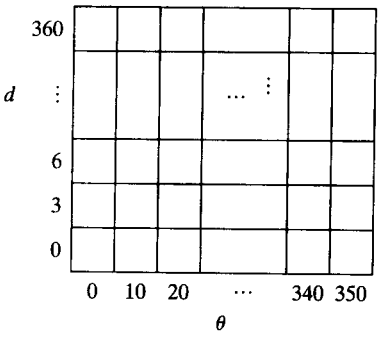


图10-21 检测256 × 256图像中直线段的累加数组

算法10.7 霍夫变换检测直线：将灰度图像S中的直线段加到累加器A中

S[R, C]是输入灰度图像。
NLINES是图像的行数。
NPIXELS是每行像素的个数。
A[DQ,THETAQ]是累加数组。
DQ是从直线到原点的量化距离。
THETAQ是直线垂直方向的量化角度。

```
procedure accumulate_lines(S, A);
{
A := 0;
PTLIST := NIL;
for R := 1 to NLINES
  for C := 1 to NPIXELS
      {
      DR := row_gradient (S,R, C) ;
      DC := col_gradient (S,R, C) ;
      GMAG := gradient (DR, DC) ;
      if GMAG > gradient_threshold
         {
         THETA := atan2 (DR,DC) ;
         THETAQ := quantize_angle(THETA);
         D := abs (C*cos (THETAQ) - R*sin(THETAQ));
         DQ := quantize_distance(D);
         A[DQ, THETAQ] := A[DQ, THETAQ] + GMAG;
         PTLIST(DQ, THETAQ) := append(PTLIST(DQ, THETAQ),[R, C])
         }
      }
306 }
```

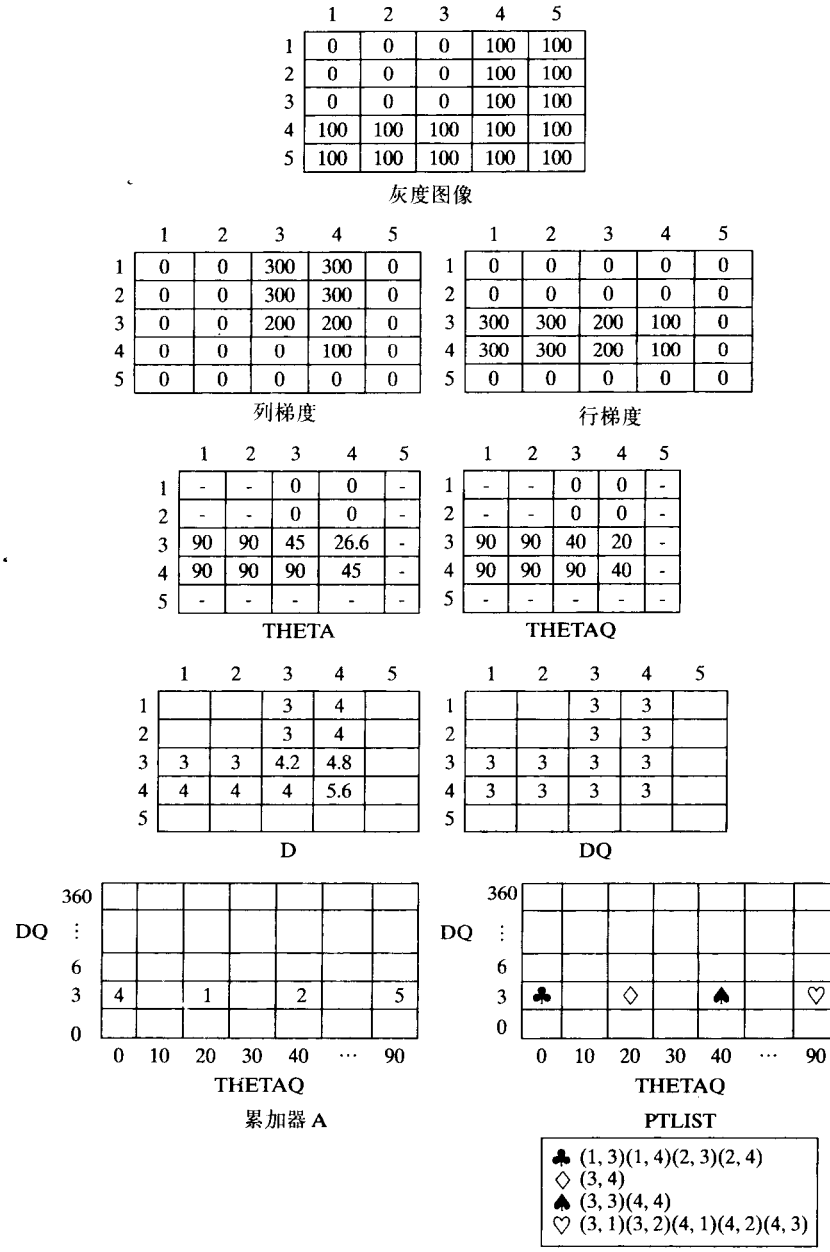


图10-22 过程*accumulate*的执行结果，对简单的灰度图像用Prewitt模板进行运算。对于这个简单例子，正确检测到的特征和错误检测到的特征差不多。对于具有较长线段的实际图像，正确检测到的特征将更多一些

307

算法10.8 用O'Gorman/Clowes方法查找独立直线段的点的列表

A[DQ, THETAQ]是从accumulate_lines得到的累加数组。

DQ是从直线到原点的量化距离。

THETAQ是直线垂直方向的量化角度。

procedure find_lines;

```
{
V:= pick_greatest_bin (A, DQ, THETAQ) ;
while V > value-threshold
  {
  list_of_points := reorder(PTLIST[DQ, THETAQ]);
  for each point [R, C] in list-of-points
   for each neighbor [R', C'] of [R, C] not in list_of_points
    {
    DPRIME := D[R', C'];
    THETAPRIME := THETA[R', C'];
    GRADPRIME := GRADIENT[R', C'];
    if GRADPRIME > gradient-threshold
      and abs (THETAPRIME - THETAQ) < 10
    then {
         merge(PTLIST[DQ, THETAQ], PTLIST[DPRIME,
           THETAPRIME]);
         set_to_zero[A, DPRIME, THETAPRIME];
         }
    }
  final_list_of_points := PTLIST[DQ, THETAQ];
  create_segments (final_list_of_points);
  set_to_zero[A, DQ, THETAQ];
  V := pick_greatest_bin[A, DQ, THETAQ];
  }
}
```

函数*pick_greatest_bin*返回最大累加器的值，并将最后两个参数DQ和THETAQ设置为该箱格的量化d值和θ值。函数*reorder*对箱格内的点列表进行排序：$\theta<45$或$\theta>135$时根据列坐标排序，$45\leqslant\theta\leqslant135$时根据行坐标排序。希望数组D和THETA中保存的是累加过程中算出的量
308 化D值和THETA值。同样希望数组GRADIENT中保存的是算出的梯度幅值。这些可作为中间图像存起来。过程*merge*将像素邻点所在的点列表与该像素所在的点列表合并起来，保持空间顺序不变。过程*set_to_zero*对累加器清零，使其不被重用。最后，过程*create_segments*搜索最后的有序点集，寻找大于一个像素的间距。它创建并保存在间距处终止的线段集合。为了更加准确，利用最小二乘过程将系列点拟合成直线段。需要提及的重要一点是，霍夫过程能够抽取出明显的断线或虚线特征，例如一排石子或者一条被下落树枝分割的道路。

习题10.10

这个习题与Kasturi等人（1990）的工作有关，用霍夫变换识别文本行。应用已有的程序或工具，并编写需要的新程序进行下面的实验：（a）打字或打印出几行不同方向的文本，并将图像二值化，加入一些别的目标，如斑点或曲线。（b）进行连通成分标记，并输出所有目

标的中心，其中目标的边界框正好框住字符。(c) 将所有中心输入到霍夫直线检测过程，讨论文本行检测的效果如何。

3. 圆检测

霍夫变换技术可扩展到检测圆和其他参数曲线。圆的标准方程有三个参数。如果点[R, C]位于圆上，那么点[R, C]到圆心的梯度如图10-23所示。这样如果给定点[R, C]，选择了半径d，计算出从[R, C]到圆心的向量方向，就可以找到圆心的坐标。半径d、圆心的行坐标r_c和圆心的列坐标c_c是霍夫算法要检测的圆的三个参数。在行-列坐标系中，圆用下面的方程表示：

$$r = r_c + d\sin\theta \tag{10-18}$$

$$c = c_c - d\cos\theta \tag{10-19}$$

采用这些方程，圆检测的累加算法即后面的*accumulate_ circles*算法。 309

对这个过程进行简单修改，把梯度幅值考虑进去，如直线段检测过程那样。将其应用于技术文档图像，结果如图10-24所示。

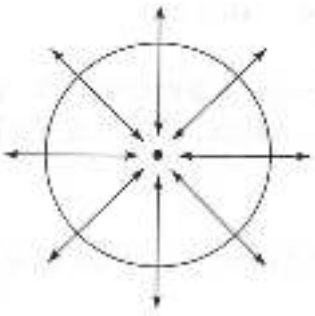


图10-23 直圆边界点的梯度方向。根据指向圆内的梯度，可以求出圆心的位置

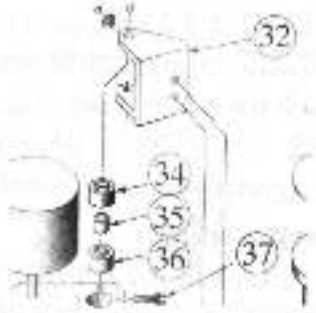


图10-24 对技术图利用霍夫变换检测出的圆，为了显示清楚，在被检测到的圆外套了一个略大的外圆

算法10.9 霍夫变换检测圆：将灰度图像S中的圆累积到累加器A中

S[R, C]是输入灰度图像。
NLINES是图像中的行数。
NPIXELS是每行像素的个数。
A[R, C, RAD]是累加数组。
R是圆心的行索引。
C是圆心的列索引。
RAD是圆的半径。

```
procedure accumulate_circles(S,A);
{
A:= 0;
PTLIST:= 0;
for R:= 1 to NLINES
  for C:= 1 to NPIXELS
    for each possible value RAD of radius
```

```
        }
        THETA := compute_theta(S,R,C,RAD);
        R0:= R - RAD*cos(THETA);
        C0 := C + RAD*sin(THETA);
        A[R0, C0, RAD] := A[R0, C0, RAD]+1;
        PTLIST[R0, C0, RAD] :=append(PTLIST[R0, C0, RAD], [R, C])
        }
    }
```

4. 任意曲线检测

霍夫变换可推广到具有解析形式$f(x, a) = 0$的任意曲线，其中x表示图像点，a是参数向量，过程如下：

（1）初始化累加数组A[a]为0。

（2）对每个边缘像素x确定a，使得$f(x, a) = 0$，并设A[a] := A[a]+1。

（3）A的局部最大值对应图像中的曲线。

如果在a中有m个参数，每个参数具有M个离散值，那么时间复杂度是$O(M^{m-1})$。霍夫变换方法已经被进一步推广到由一系列边界点确定的任意形状（Ballard, 1981）。这就是著名的广义霍夫变换（generalized Hough transform）。

310

5. Burns直线检测器

一些混合方法利用了霍夫变换原理。Burns直线检测器（Burns等人, 1986）是为了检测室外复杂场景中的直线。Burns方法总结如下：

（1）计算每个像素的梯度幅值和方向。

（2）对于具有足够高梯度幅值的点，用两个标记表示梯度方向两种不同的量化措施。（例如，对于8个箱格情况，如果第一种量化措施是0至44, 45至90, 91至134等等，那么第二种量化措施是-22至22, 23至67, 68至112等），结果产生两个符号图像。

（3）对于每幅符号图像，检测连通成分，计算每个成分的线段长度。

- 每个像素是两个成分的成员，成分来自两幅符号图像。
- 每个像素对较长的成分进行表决。
- 每个成分收到对其表决的像素数。
- 选择收到大多数支持的成分（直线段）。

Burns直线检测器用到了两种有效算法：霍夫变换和连通成分算法。为了去除量化影响，在O'Gorman和Clowes方法中，采用两套独立的量化措施搜索相邻的箱格。实际应用中，它存在一个问题，这个问题也影响所有基于像素小邻域来估计角度的直线检测方法。这个问题在于数字直线并不直。对角线实际上由一系列水平和垂直阶梯组合而成。如果角度检测方法采用的邻域太小，将找到许多细小的水平和垂直线段，而不是较长的对角线。所以在实际中，Burns直线检测法以及任何其他基于角度的直线检测法，都会将直线分成小段，而人类则把这些直线看成是一整条连通线。

311

习题10.11 Burns和霍夫算法的比较

实现检测直线的霍夫变换和Burns算子，并比较它们在包含大量直线的实际图像上的效果。

习题10.12 直线检测

实现下面方法，检测灰度图像I中的直线。重要集类将与I中的重要线段对应。

```
for all image pixels I[R, C]
  {
  compute the gradicant G_mag and G_dir
  if G_mag > threshold
  then output [G_mag, G_dir] to set H
  }
detect clusters in the set H ;
```

10.4 线段拟合模型

拟合数据的数学模型不仅能够揭示重要的数据结构，也为进一步分析提供合适的表达方法。直线模型可表示建筑物的边缘，平面模型可表示建筑物的表面，对于圆、圆柱和许多其他形状都存在合适的数学模型。

下面的最小二乘方法（method of least squares）能够确定拟合数据的最佳数学模型的参数。这些数据可以用前面描述的区域分割或边界分割方法得到，例如，前面提过可将所有像素点[r, c]用直线模型拟合，在霍夫累加数组中，这些像素对某种直线假设A[THETAQ, DQ]进行表决。候选的模型种类有无限多，为了应用最小二乘法，必须通过某种方法确定合适的模型形式。一旦确定了模型形式及其参数，就可确定该模型对数据的拟合结果是否是可接受的，拟合效果好就意味着检测出了具有某种形状的目标，或者为进一步分析提供一种更紧凑的数据表示方法。

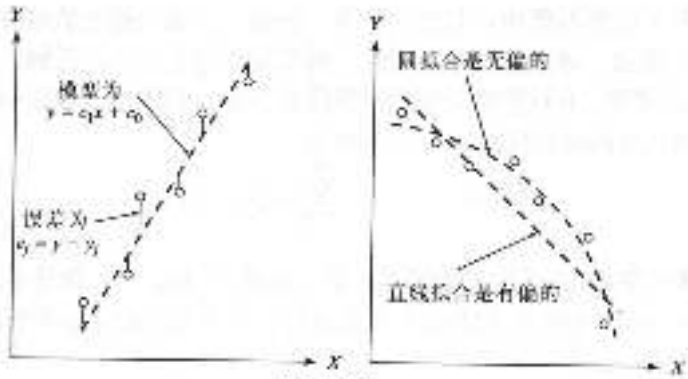


图 10-25

（左）用模型$y=f(x)$拟合6个数据点

（右）可能的直线模型和圆模型。余类说明直线拟合是有偏的，圆拟合是无偏的

1. 直线拟合

通过简单实例解释最小二乘理论。直线模型是带有两个参数的函数$y=f(x)=c_1x+c_0$。如果我们想测试一组观测点$\{(x_j, y_j),\ j=1, n\}$，看看它们是否位于该直线上。首先要确定线性函数的最佳参数c_1和c_0，然后检查这些观测点距离函数有多近。可用不同指标度量观测点与模型的近似程度。图10-25显示用一条直线来拟合6个数据点。可以移动直线，得到另一条不同的直线，拟合结果仍然很好。根据定义74，最小二乘指标（least-squares criteria）定义了最佳拟合直线。 312

定义74 最小二乘误差指标通过下列公式，衡量模型$y=f(x)$对n个观测点$\{(x_j, y_j),\ j=1, n\}$的拟合效果：

$$LSE = \sum_{j=1}^{n} (f(x_j) - y_j)^2$$

最佳模型$y = f(x)$指能够使该指标最小化的参数模型。

定义75 方均根误差（RMSE），指模型与观测点之间差异的平均值：

$$RMSE = \left[\sum_{j=1}^{n} (f(x_j) - y_j)^2)/n \right]^{1/2}$$

注意对于直线拟合，这个差异不是直线到观测点的欧几里得距离，而是如图10-25所示与y轴平行的距离。

定义76 最大误差指标通过下列公式，衡量模型$y = f(x)$对n个观测点$\{(x_j, y_j), j = 1, n\}$的拟合效果。

$$MAXE = \max(\{|(f(x_j) - y_j)|\}_{j=1,n})$$

313 注意这个指标只与最差拟合点有关，而RMS误差与所有拟合点有关。

表10-1 用$y = 3x-7$生成数据并加上噪声，利用最小二乘法得到拟合模型$y = 2.971x-6.962$

Data Pts(x_j, y_j)	(0.0, −6.8)	(1.0, −4.1)	(2.0, −1.1)	(3.0, 1.8)	(4.0, 5.1)	(5.0, 7.9)
Residuals $y-y_j$:	−0.162	0.110	0.081	0.152	−0.176	−0.005

2. 参数的封闭解

最小二乘指标得到普遍使用，有两个原因。首先，当噪声模型是高斯噪声时，必然会选择最小二乘指标；其次，求最佳模型参数时，容易推导出封闭形式的解。我们首先推导最佳拟合直线的参数封闭解，其他模型采用类似的推导过程。直线模型的最小二乘误差可以显式表示如下。其中公式中的观测数据x_j、y_j视为常量。

$$LSE = \varepsilon(c_1, c_0) = \sum_{j=1}^{n} (c_1 x_j + c_0 - y_j)^2 \tag{10-20}$$

误差函数ε是带两个参数c_1和c_0的光滑非负函数，它在点(c_1, c_0)处具有全局最小值，其中$\partial\varepsilon/\partial c_1 = 0$，$\partial\varepsilon/\partial c_0 = 0$。对公式（10-21）进行求导，并利用和的导数等于导数的和这个事实，得到下面的结果。

$$\partial\varepsilon/\partial c_1 = \sum_{j=1}^{n} 2(c_1 x_j + c_0 - y_j)x_j = 0 \tag{10-21}$$

$$= 2\left(\sum_{j=1}^{n} x_j^2\right) c_1 + 2\left(\sum_{j=1}^{n} x_j\right) c_0 - 2\sum_{j=1}^{n} x_j y_j \tag{10-22}$$

$$\partial\varepsilon/\partial c_0 = \sum_{j=1}^{n} 2(c_1 x_j + c_0 - y_j) = 0 \tag{10-23}$$

$$= 2\left(\sum_{j=1}^{n} x_j\right) c_1 + 2\sum_{j=1}^{n} c_0 - 2\sum_{j=1}^{n} y_j \tag{10-24}$$

这些方程可表示成矩阵形式。求解这些方程就得到最佳直线参数。对于任意多项式拟合的一般情况，将产生一组表达形式类似的方程，称为规范化方程（normal equations）。

$$\begin{bmatrix} \sum_{j=1}^{n} x_j^2 & \sum_{j=1}^{n} x_j \\ \sum_{j=1}^{n} x_j & \sum_{j=1}^{n} 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_0 \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{n} x_j y_j \\ \sum_{j=1}^{n} y_j \end{bmatrix} \tag{10-25}$$

习题10.13 用一条直线拟合3个点

利用公式（10-25），计算通过点[0, −7]、[2, −1]和[4, 5]的最佳直线参数c_1和c_0。 314

习题10.14 规范化方程

（a）对观测数据(x_j, y_j)，$j = 1,n$，用三次多项式$c_3x^3 + c_2x^2 + c_1x + c_0$拟合，推导包含4个参数的矩阵方程形式。（b）根据矩阵元素的模式，预测四次多项式拟合的矩阵形式。

3. 误差的经验解释

在机器视觉问题中，误差和个别误差的经验解释一般比较直接。例如，如果模型用拟合所有观测数据所产生的误差是一到两个像素，我们就会接受这个拟合结果。对于受控2D成像环境，其中主要包含直边目标，要研究的内容就是，看看检测到的边缘点与理想直线有多大的偏离程度。如果个别点离拟合直线很远（这些点称为局外点（outliers）），则意味着特征检测出现错误，目标上有缺陷，或者存在另一个目标或模型。在这些情况下，合适的做法是从观测数据中删除这些局外点，重新拟合，这样得到的模型就免受局外点的影响。所有的原始点仍可用新模型进行解释。如果用拟合模型进行曲线分割，一般要删除端点，因为它们实际上属于另一种形状的目标或部件。

4. 误差的统计解释*

可用正规统计假设来解释误差。一般假设是，y_j的观测值仅仅是模型值$f(x_j)$加上服从正态分布$N(0, \sigma)$的高斯噪声，其中σ可通过分析测量误差得到，可利用上面的经验方法。假设个别观测j与观测k之间的噪声是相互独立的。变量$S_{sq} = \sum_{j=1}^{n}((f(x_j) - y_j)^2)/\sigma^2)$服从$\chi^2$分布，这样它的似然度可通过公式或查表确定。直线拟合的自由度是$n-2$，因为从n次观测中要估计2个参数。如果χ^2分布的95%低于观测到的S_{sq}，那么就应该拒绝模型拟合数据的假设。也可用其他的置信水平。χ^2检验不仅适用于接受/拒绝一个假设，而且适用于从一组竞争模型中选择最可能的模型。例如，抛物线模型可能会与直线模型发生竞争。注意在这种情况，抛物线模型$y = c_2x^2 + c_1x + c_0$有3个参数，这样χ^2分布将有$n-3$的自由度。

直观上，观测j的误差与观测$j-1$或$j+1$的误差相互独立这一假设不是太合适。例如，一个错误的产生可能会引起点的整个邻域不再服从理想模型。独立性假设可以根据正负号变化（run-of-signs）进行检验，根据正负号变化可检测误差中的系统偏差，而系统偏差意味着采用另一种形状模型将产生更好的拟合效果。如果噪声确实是随机的，那么误差的正负号也是随机的，从而造成误差的上下波动。图10-25（右）显示有偏的线型拟合和无偏的圆形拟合情况。误差符号说明了直线拟合是有偏的。关于评估拟合质量的统计假设检验，参见本章末的参考文献。 315

习题10.15 拟合3D点的平面方程

（a）对5个表面点（20, 10, 130）、（25, 20, 130）、（30, 15, 145）、（25, 10, 140）、（30, 20,

140），进行最小二乘平面拟合，求解模型$z=f(x,y)=ax+by+c$的3个参数a、b、c；（b）对5个点的3个坐标都加上1个随机变化，重复问题（a）。通过抛硬币来确定变化量：如果硬币正面朝上则加1，反面朝上则减1。

习题10.16 Prewitt算子是最优的

说明对亮度函数的3×3邻域用最小二乘平面进行拟合，可以得到第5章的Prewitt梯度算子，为计算$f[x,y]$处的梯度，拟合9个点：$(x+\Delta x, y+\Delta y, f[x+\Delta x, y+\Delta y])$，其中$\Delta x$和$\Delta y$可取$-1$、0、$+1$。对于亮度表面的最佳拟合平面模型$z=ax+by+c$，证明利用两个Prewitt模板就可实际算出$a$和$b$。

5. 拟合中的问题

考虑拟合中的几类问题是非常重要的。

局外点 由于每个观测值都影响RMS误差，大量的局外点会使拟合失去价值。最初的拟合结果可能偏离理想模型太远，从而无法识别并去掉真正的局外点。这时可采用稳健统计方法，参见本章末列出的Boyer等人的（1994）文献。

误差定义 误差的数学定义，是y轴方向的偏差，而不是真正的几何距离。这样最小二乘拟合所得到的曲线或曲面，未必能够最接近几何空间中的数据。图10-25右图中最右边的一点就说明了这个问题，在几何上该点离圆非常近，但沿y轴的函数偏差却很大。当用复杂曲面拟合3D点时，这种效果更加明显。虽然几何距离通常比函数偏差更有意义，但有时并不容易计算。对于直线拟合情况，当直线接近竖直时，采用第3章的最佳轴计算方法要比这里的最小二乘方法效果更好。最佳轴计算公式以点和线间的几何距离最小为基础。

非线性优化 有时无法得到模型参数的封闭解。但误差指标仍可进行优化，利用**参数空间搜索**技术寻找最优参数。爬山法、基于梯度的搜索甚至穷尽搜索都可用于优化。参见Chen和Medioni以及Sullivan，Sandford和Ponce（1994）的工作，其中涉及到非线性优化及前面提到的问题。

高维数 当数据维数或模型参数个数较多时，对拟合的经验解释和统计解释都是困难的。另外，如果采用搜索技术来寻找参数的话，甚至难以知道这些参数是否是最优的，或者只是
316 误差指标的局部最小值。

拟合条件 有时拟合模型必须满足附加的约束条件。例如，我们可能需要寻找通过观测点的最佳直线，而且它必须和另一条直线垂直。约束最优化方法参见参考文献。

6. 基于拟合的曲线分段

上面的模型拟合方法及理论，需要假设模型形式和一组观测数据。通过边界跟踪，可得到长骨形边界点，对这些边界点可按下面方法进行分割。首先，检测边界序列中的高曲率点或尖端点，用这些点对曲线进行分割；然后，用断点之间的曲线段检验假设的模型。结果产生一组曲线段，以及描述每段形状的数学模型和参数。对边界曲线分段的另一种方法是模型拟合。在第一阶段，对k个序列点的子序列进行模型拟合，对每个可接受的拟合结果，将其χ^2值存入一个集合中；第二个阶段，通过不断往子序列中增加其他端点，来扩展可接受的拟合结果。拟合线段持续增长，直到加入的新端点使拟合结果的χ^2值下降为止，结果得到一组可能重叠的子序列，每个子序列都有一个模型以及拟合模型的χ^2值。然后把这个集合传递给更高层的过程，高层过程根据检测到的部件构建出目标模型。这个过程在思想上和10.1.2中的区域增长很相似，它对直线段进行增长，把边缘像素处的方向作为关键特征，而在区域增长中

把灰度值属性作为关键特征。

10.5 识别更高层结构

图像分析经常需要对区域或线段进行综合考虑。例如，四边形区域和边缘直线段结合，可能意味着图中具有建筑物；边缘线段的交点是建筑物的角；蓝色区域中的绿色区域可能是一个小岛。区段组合的方法是无穷的，下面我们只讨论两种常见的边缘段组合方法，它们构成更丰富的结构信息。这两种组合结果是条带（ribbon）和角点（corner）。

10.5.1 条带检测

一种非常通用的图像区域类型是条带。条带通常是2D或3D的细长目标的图像。例如，印刷电路上的线路、房屋门、桌上的笔，或者穿过田野的道路。在这些例子中，条带两边近似平行，但不一定是直的。虽然下面的讨论局限于直边，但条带有更一般的形状，例如酒瓶或装饰灯柱，其侧面轮廓是某种复杂的曲线。关于条带轴对称。电线、绳索，曲折的溪流或道路，在图像中都呈现条带。绳索或灯柱的影子也如此。第14章讨论的称为广义圆柱体（generalized cylinders）的3D目标部件，其视图也呈现条带。图10-16的左边是由四个条带表 317
示的图符，其中两个明显有弯曲。对通用条带的抽取我们留作以后进行研究，现在集中讨论直边的条带。

定义77 条带是关于其上轴大致对称的细长区域，条带边缘与背景的对比差异一般具有对称性，但也有例外。

如图10-26所示，对霍夫变换稍加扩展，就可对边缘方向和位置以及穿过边缘的梯度方向进行编码。第5章和本章前面讨论过，对于梯度幅值较大的像素点$[r, c]$，其梯度方向θ可以利用算子如Sobel算子算出，方向范围为$[0, 2\pi]$。从图像原点到该像素的向量是$[r, c]$，将该向量投影到方向θ的单位向量上，得到带正负号的距离d。

$$d = [r, c] \circ [-\sin\theta, \cos\theta] = -r\sin\theta + c\cos\theta \tag{10-26}$$

正值d与像素$[r, c]$的一般极坐标表示相同。当从原点到边缘的方向与梯度方向相反时，就会得到负值d，例如对棋盘上的线条聚类将产生两个类别。图10-26说明了这种思想。考虑图中的边P_2P_3。沿着这条边的像素应该都具有大约30°的梯度方向。从原点到P_2P_3的垂直线具有同样的方向。因此沿着P_2P_3的像素在霍夫参数空间近似变换为$[d_2, 30°]$。沿着直线段P_1P_2的像素具有210°的梯度方向，这与从原点到P_1P_2的垂线方向相反，因此沿着线段P_1P_2的像素近似变换为$[-d_1, 210°]$。

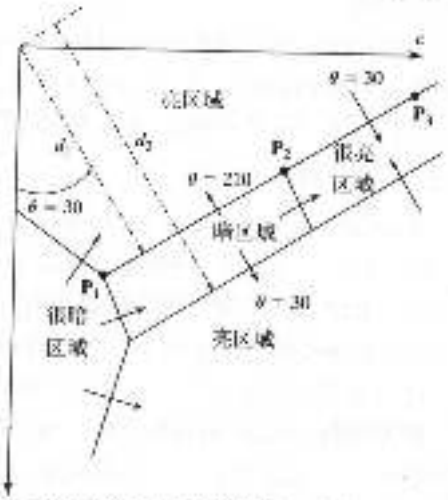


图10-26 霍夫变换可对边缘的位置、方向以及梯度方向编码。沿着同一条图像直线，从暗区域到亮区域的变换与从亮区域到暗区域的变换得到相反的梯度方向

318

习题10.17

图10-27显示以图像原点为中心在亮背景上的深色环。画出类似图10-26所示的霍夫变换参数空间。

直带检测 利用霍夫参数以及从算法*accumulate_ lines*得到的点列表，可以检测更复杂的图像结构。方向相差180°的两条边缘表明区域可能是个条带。另外，如果这些点列互相距离很近，那么则说明存在梯度方向相反的更大的线性特征，如图10-17中的支柱。

图10-28显示白房子的部分图像，图中有一个落水管。图片是在强烈光线下拍摄的，其中有很明显的阴影。利用梯度算子，通过*accumulate_lines*搜集边缘线段上的像素，很明显看见在深色背景前有一条亮带，对应落水管部位（两条边是AB和ED）。落水管的阴影s也相对亮背景产生一条深色带。

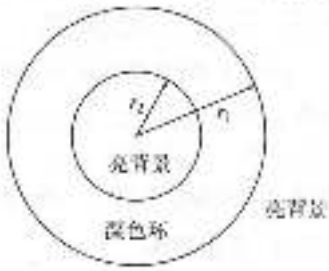


图10-27 深色环以原点为中心，背景为亮色，深色区域介于小圆和大圆之间



图 10-28

（左）带落水管及明显阴影的房子图像
（中）由3×3Prewitt算子检测结果，取梯度幅值大于10%的像素点
（右）抽取出的条带和角点

习题10.18

编写程序，用霍夫变换检测条带。（a）用Sobel算子抽取所有像素点的梯度幅值和方向，然后只对高幅值的像素进行变换。（b）检测[d, θ]空间中的聚类。（c）检测聚类对（[d_1, θ_1]和[d_2, θ_2]），其中θ_1和θ_2相隔π。（d）删除那些不是大致轴对称的聚类对。

10.5.2 角点检测

重要的区域角点，可通过检测满足下面关系的边缘线段对E_1和E_2得到。

（1）拟合边缘点集E_1和E_2的直线，在实际图像坐标空间相交于点[u, v]。

（2）点[u, v]与集合E_1和E_2的端点都接近。

（3）E_1和E_2的梯度方向关于它们的对称轴对称。

这个定义只是建立了“L”型的角点模型，条件（2）排除了“T”、“X”和“Y”型的角点。计算出的交点[u, v]具有亚像素精度。图10-29显示了角点结构的几何特征。一开始识别边缘线段时，可用霍夫变换、边界跟踪及直线拟合或者任何其他合适的算法来进行。对满足以上条件的每对（[d_1, θ_1]，[d_2, θ_2]），将四元组（[d_1, θ_1]，[d_2, θ_2]，[u, v]，α）加入候选角点的集合。角度α为角点处形成的角度。这组角点特征可用于建立更高层次的描述，或者直接用于第11章的图像匹配或变形运算。

从图10-2的积木图像可以很容易抽取出几个角点，但是，由于受观察角度的影响，这些角点很多是由物体之间相互遮挡引起的，而不代表实际3D物体两折痕的交点。其中小拱桥顶部有4个很明显的实际角点。图10-28中三角形ABC的顶点都是由光线和一定观察角度造成的。最后我们做出如下结论：虽然在特殊问题领域经常用到边缘线段，但在一般问题上使用边缘线段，存在很大的歧义性。通常进行高层结构解释时，需要用到应用领域的特殊知识。

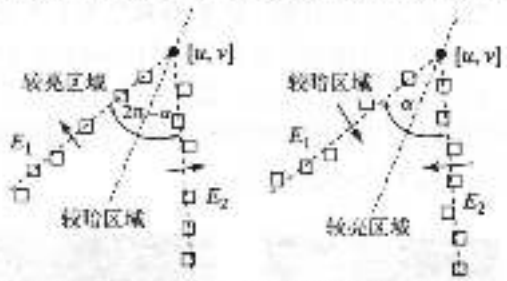


图10-29 检测角点，角点用具有适当关系的边缘线段对表示

320

习题10.19

如何改变条带检测算法，使它（a）只检测接近垂直的条带，（b）检测宽不超过W的条带。

10.6 运动一致性分割

我们已经看到，在确定场景内容和行为方面运动有着非常重要的作用，第9章讨论了场景变化检测和利用视频进行运动跟踪的方法。

10.6.1 时空边界

运动目标的轮廓可以利用空间和时间上的差异进行识别，前面只用了某些特征的空间差异，例如单幅图像的亮度或纹理。如果得到场景的两幅图像$\mathbf{I}[x, y, t]$和$\mathbf{I}[x, y, t + \Delta t]$，就可以计算空间梯度和时间梯度，并将二者结合起来。可以定义一个时空梯度幅值（spatio-temporal gradient magnitude），等于空间梯度幅值和时间梯度幅值的乘积，如公式（10-27）所示。一旦算出图像的STG[]，所有讨论过的轮廓抽取方法都可以用。抽取的轮廓将是运动目标的边界而不是静态目标的边界。

$$STG[x, y, t] = Mag[x, y, t](|I[x, y, t] - I[x, y, t + \Delta t]|) \tag{10-27}$$

10.6.2 运动轨迹聚类

在图像序列的两帧之间计算运动向量，可用第9章介绍的特殊兴趣点或兴趣区域进行计算。根据图像位置、速度和方向对运动向量聚类可以实现区域分割，如图10-30所示，对平移目标聚类应能得到很好的效果，因为目标上的点应该具有相同的速度。通过更复杂的分析，还可检测同时旋转和平移的目标。

图10-31显示手语应用情况，其中双手运动的目的是为了交流。研究目标是将美国手语信息输入到机器中。图中只给出了一个序列的几帧图像，显示了手语者持续大约2s的手势变化情况。图10-31所示的结果，在帧内采用颜色分割，帧间采用运动分割产生的结果。运算步骤参见算法10.10，关于产生图10-31的算法细节，可以参考Yang和Abuja（1999）发表的论文。算法的前几步适用于不同的图像序列，对每幅图像作颜

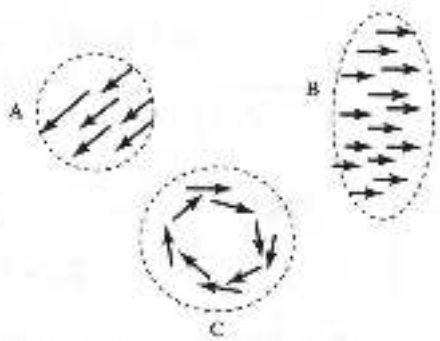
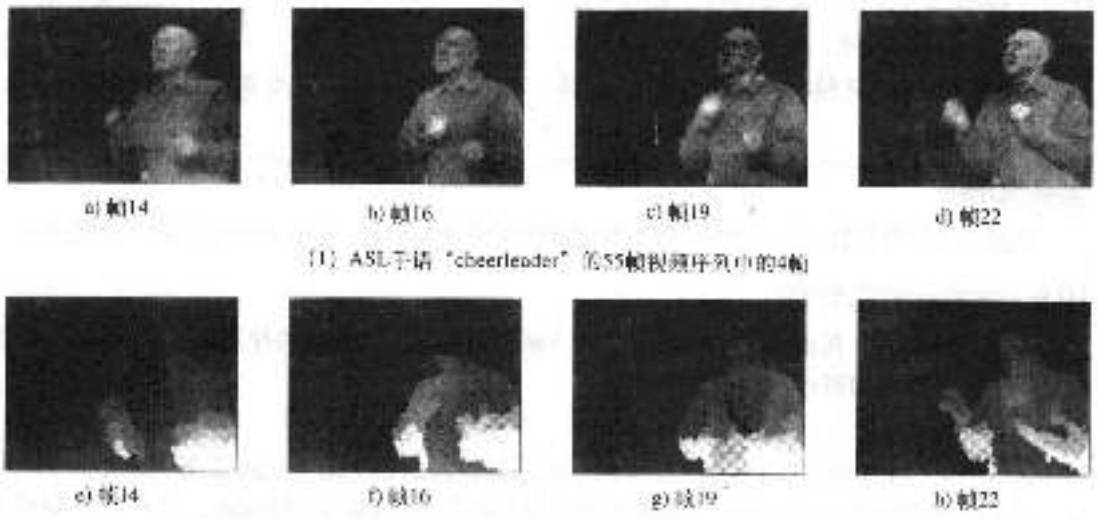


图10-30 利用位置、速度和方向的一致性，对运动场中的向量进行聚类，平移目标（A, B）比旋转目标（C）更易检测

321

色分割，在不同帧之间进行分割区域匹配。匹配结果用于计算前后图像对的密集运动场，然后对运动场进行分割，以推导单个像素的运动轨迹。把运动场分割成包含统一运动的区域，这时我们才利用相关的领域知识，识别人手和人脸。第6章提到的皮肤颜色模型用来识别皮肤区域，认为其中最大的一块皮肤区域是人脸。针对所有的图像帧跟踪两手掌区域的中心，这两个轨迹可用来识别所做的手势。Yang和Ahuja（1999）对40种美国手语的多个样例做了实验，结果证实识别率超过90%。



a) 帧14　b) 帧16　c) 帧19　d) 帧22

(I) ASL手语"cheerleader"的55帧视频序列中的4帧

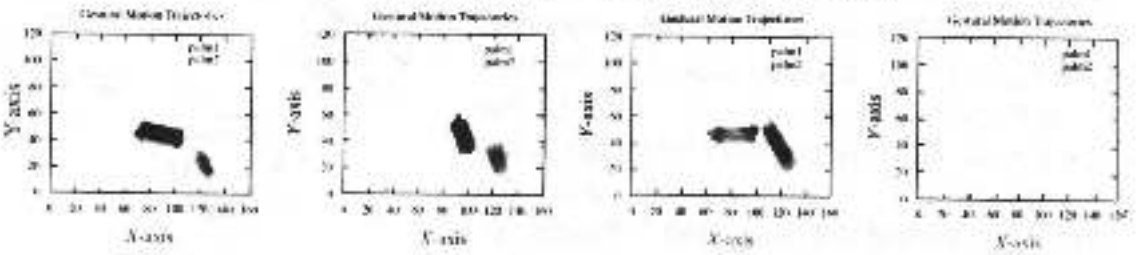
e) 帧14　f) 帧16　g) 帧19　h) 帧22

(II) 图像序列cheerleader的运动分割
(同一运动区域的像素用同样的灰度级别显示，不同区域用不同的灰度级别显示)



i) 帧14　j) 帧16　k) 帧19　l) 帧22

(III) 利用颜色和大小从图像序列cheerleader中抽取的人头和手掌区域



m) 帧14~16　n) 帧16~19　o) 帧19~22　p) 帧22~25

(IV) 从ASL手语"cheerleader"分割图中抽取手势运动轨迹
(由于显示所有像素的轨迹，结果形成一个团儿)

图10-31 从图像序列抽取运动轨迹（图片由Ming-Hsuan Yang和Narendra Ahuja提供）

算法10.10 利用颜色和运动跟踪ASL手势的算法。(Motivated by Yang and Ahuja (1999))

输入ASL手语视频序列。

输出两个手掌的运动轨迹。

1. 利用颜色对序列的每一帧$\mathbf{I}_t$进行区域分割。
2. 根据颜色和邻域匹配每对图像($\mathbf{I}_t$, $\mathbf{I}_{t-1}$)的各区域。
3. 计算$\mathbf{I}_t$区域与$\mathbf{I}_{t-1}$对应区域相匹配的仿射变换。
4. 利用匹配区域的变换帮助计算单个像素的运动向量。
5. 利用运动一致性和图像位置对上面得到的运动场进行分割。
6. 利用皮肤颜色模型识别两手掌区域和人脸区域。
7. 合并前面分割得到的邻近皮肤区域。
8. 用椭圆逼近手掌和人脸。
9. 跟踪整个序列的椭圆中心，建立运动轨迹。
10. (利用双手轨迹识别手势。)

习题10.20

已知参数$[d_1, \theta_1]$、$[d_2, \theta_2]$定义的两条直线。(a) 推导交点$[x, y]$公式，(b) 推导对称轴$[d_3, \theta_3]$公式。

习题10.21

得到含运动目标的前后两帧场景图像，利用公式(10-27)计算时空图像。(最好两帧图像取自运动JPEG视频序列，或者用平台扫描仪数字化深色剪纸图，轻轻移动剪纸得到第二幅图像。)

习题10.22

针对ASL应用情况，说明如何修改算法10.10，使其更加简单快速。

习题10.23

假设有两个运动轨迹$\mathbf{P}_j$, $j = 1, \cdots, N$和$\mathbf{Q}_k$, $k = 1, \cdots M$，其中$\mathbf{P}_j$和$\mathbf{Q}_k$是时间顺序一致的2D点。设计算法匹配这两条轨迹，当两轨迹相同时输出1.0；当二者非常不同时，输出0.0。注意M和N可能不相等。

10.7 参考文献

分割是计算机视觉领域尚未得到解决的古老难题之一。Haralick和Shapiro (1985) 发表的论文中，对分割方面的早期工作进行了很好的综述，其中大多数工作针对的是灰度图像。第一个针对自然彩色图像的实用分割方法，是Ohlander, Price和Reddy (1978) 提出的。到了最近几年，这个领域才重新又变得富有成果。Shi和Malik的规范化切痕工作，开始于1997年，他们因开创新的研究工作而享有殊荣，该研究就是对大图像集合中的任何彩色图像进行分割。在线条图分析中，Freeman首先在60年代提出他的链码方法。他在1974年的论文中讨论了链码的使用。霍夫变换只是作为一个专利发表出来，Duda和Hart对它进行了推广和扩展，

O'Gorman和Clowes1976年关于直线段的论文，以及Kimme、Ballard和 Sklansky（1975）关于圆检测的工作，都证明了霍夫变换的实用性。Burns直线检测出现于十年之后，对霍夫变换在技术上进行了改进，使其更加稳健、更加可靠。Samet 1990关于空间数据结构的著作，是四叉树方面的优秀参考文献。

Bowyer等人（1994）提出，图像分割可利用稳健统计学将数据与模型拟合。任何预期的
322 模型都可用来拟合图像。稳健拟合可以去除大量的局外点，得到与某个模型相拟合的图像线
~ 段。当所有预期的模型都能拟合上，就可以说图像得到了分割，分割出的部分由拟合点组成。
324

1. Ballard, D. H., 1981, Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recog.*, v. 13(2):111–122.
2. Boyer, K., K. Mirza, and G. Ganguly. 1994. The robust sequential estimator: a general approach and its application to surface organization in range data, *IEEE Trans. Pattern Analysis and Machine Intelligence,* v. 16(10) (Oct. 1994), 987–1001.
3. Burns, J. R., A. R. Hanson, and E. M. Riseman. 1986. Extracting straight lines. *IEEE Trans. Pattern Analysis and Machine Intelligence*, v. PAMI-8:425–455.
4. Chen, Y., and G. Medioni. 1994. Surface description of complex object from multiple range images. *Proc. IEEE Conf. Comput. Vision and Pattern Recog.*, Seattle, WA (June 1994), 513–518.
5. Duda, R. O., and P. E. Hart. 1972. Use of the Hough transform to detect lines and curves in pictures. *Communications of the ACM,* v. 15:11–15.
6. Freeman, H. 1974. Computer processing of line-drawing images. *Computing Surveys,* v. 6:57–97.
7. Haralick, R. M., and L. G. Shapiro. 1985. Image segmentation techniques. *Comput. Vision, Graphics, and Image Proc.*, v. 29(1) (January 1985), 100–132.
8. Kasturi, R., S. Bow, W. El-Masri, J. Shah, J. Gattiker, and U. Mokate. 1990. A system for interpretation of line drawings. *IEEE Trans. Pattern Analysis and Machine Intelligence,* v. PAMI-12:978–992.
9. Kimme, C., D. Ballard, and J. Sklansky. 1975. Finding circles by an array of accumulators. *Communications of the ACM,* v. 18:120–122.
10. O'Gorman, F., and M. B. Clowes. 1976. Finding picture edges through collinearity of feature points. *IEEE Trans. Comput.*, v. C-25:449–454.
11. Ohlander, R., K. Price, and D. R. Reddy. 1978. Picture segmentation using a recursive region splitting method. *Comput. Graphics and Image Proc.*, v. 8:313–333.
12. Ohta, Y., T. Kanade, and T. Sakai. 1980. Color information for region segmentation. *Comput. Graphics and Image Proc.*, v. 13:222–241.
13. Rao, K. 1988. *Shape Description from Sparse and Imperfect Data.* Ph.D. thesis, Univ. of Southern California.
14. Samet, H. 1990. *Design and Analysis of Spatial Data Structures.* Addison-Wesley, Reading, MA.
15. Shi, J., and J. Malik. 1997. Normalized cuts and image segmentation. *IEEE Conf. Comput. Vision and Pattern Recog.*, 731–737.
16. Sullivan, S., L. Sandford, and J. Ponce. 1994. Using geometric distance for 3D object modeling and recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, v. 16(12) (Dec. 1994), 1183–1196.
17. Yang, M.-H., and N. Ahuja. 1999. Recognizing hand gesture using motion trajectories. *Proc. IEEE Conf. Comput. Vision and Pattern Recog. 1999,* Ft. Collins, CO (23–25 June 1999), 466–472.

325

第11章 2D匹配

这一章研究如何在图像与地图、图像与模型、图像与图像之间建立对应关系，以及如何利用这些对应关系。其中的匹配算法只讨论二维空间情况。在第14章中，将把匹配算法推广到3D-2D匹配以及3D-3D匹配。在很多实际应用中，2D匹配就足够了，而不需要进行完全3D分析。

考虑这样一个问题：一个小镇为了制定发展计划或者为了征税的目的，要对全镇土地的使用进行清查。于是在一个晴朗的日子里，一架飞机受命拍摄全镇土地的航测图像。然后把这些图像与相应地区的最新地图进行比较，参考航测图像对旧地图进行修改从而得到更新的地图。此外，为了标明建筑物、道路、油井等的位置，以及指明不同田地里的作物类型，也要对其他数据库进行更新。这个工作完全可以通过手工完成，但是目前普遍使用计算机来实现。第二个例子来自医疗领域。医疗上经常需要对病人心肺中的血液流动情况进行检查。首先拍摄一幅病人的X光图像，然后在病人血流中注射特殊的染色剂，再拍一幅X光图像。如果不是因为身体其他组织如骨头等引起的噪声，第二幅图像就能够揭示血液的流动情况。如果用第二幅图像减去第一幅图像，就可以减小噪声和人为干扰，而将重点集中在染色部分的变化上。但是，在进行上述运算之前，对第一幅图像要进行几何变换（geometrically transformed）或变形（warped）处理，以补偿身体微小运动的影响，这些运动是由于身体位置变化、心脏运动、呼吸等引起的。

11.1 2D数据配准

本章用到了一个简单通用的数学模型，该模型也可以用于其他情况。公式（11-1）和图11-1显示出模型**M**上的点和图像**I**上的点之间的可逆映射关系。事实上，**M**和**I**都可以是任意的2D坐标空间，可以代表一张地图、一个模型或一幅图像。

326

$$\begin{aligned} M[x, y] &\cong I[g(x, y), h(x, y)] \\ I[r, c] &\cong M[g^{-1}(r, c), h^{-1}(r, c)] \end{aligned} \tag{11-1}$$

定义78 从一个2D坐标空间到另一个2D坐标空间的映射称为**2D变换**（2D transformation）。

公式（11-1）所定义的变换，有时被称为空间变换、几何变换或变形。（有人用变形这个词专指非线性变换）。函数g和h在模型点$[x, y]$和图像点$[r, c]$之间建立起对应关系，这样模型中的特征点就可以在图像中找到它的对应位置。假设映射是可逆的，则可以利用逆映射进行反方向计算。在税务登记问题中，可以通过这样的映射函数把地图上的特征边界转换到航测图像中。然后就可以

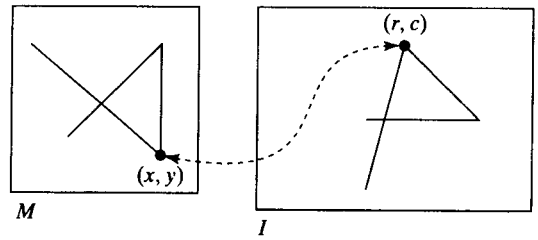


图11-1 2D空间M和I之间的映射。M可以代表一个模型而I可以代表一幅图像，但一般来讲，二者均可以是任意的2D空间

对代表特殊特征的图像区域进行分析，以规划新的建筑设施或种植新的作物类型。（当前，这个分析工作很可能是由人工借助交互式图形工作站完成的。）在医疗问题中，放射学家就能够利用该函数分析差分图像$h[r_2, c_2]=f[r_2, c_2]-g[r_2, c_2]$。这时映射函数所起的作用就是把两幅图像中的点对应起来。

定义79 关于相同场景、近似视点的两幅图像，对图像点进行几何变换，使得两幅图像中的对应特征点在变换后具有相同的坐标，这个过程就是称为**图像配准**（Image registration）。

另一个常见而且重要的应用是，基于另一幅图像上的采样点建立新的图像。这实际上不是匹配运算。如图11-2所示的例子，我们也许想从图像I_1中剪切出子图像I_2。尽管新图像I_2的内容只是原图像I_1的一个子集，但I_2的像素数可以与I_1的像素数一样多，甚至可以更多。



图11-2 （左图）第8章中用过的标志牌场景图像，（右图）对原图采样变换后剪切出的新图像

在实际应用中这个理论存在几个问题。函数g和h的形式是什么？他们是否线性、是否连续等等。一个空间中的直线映射到另一个空间中是直线还是曲线？在这两个空间中，同一点对之间的距离是否相同？更重要的是，我们如何应用不同的函数特性得到需要的映射？模型或图像的2D空间是连续的还是离散的？如果至少其中之一是数字图像，那么量化效应将会对精度和显示质量产生影响。（在图11-2的右边图像中就存在量化效应。）

327

习题11.1

如何对图11-2右侧的图像进行增强，以减弱量化效应或者阶梯效应的影响？

11.2 点的表示

本章我们专门讨论2D空间的点运算。在第13章中，将把有关定义及结论推广到3D空间。其中大多数推广都是容易理解的，当然不是全部。对同学们来说，在学习3D空间更复杂的运算之前，能够掌握基本概念和基本表达方法是非常重要的。一个2D点有两个坐标，通常用行向量$\mathbf{P}=[x, y]$或列向量$\mathbf{P}=[x, y]^{\mathrm{T}}$来表示。我们采用列向量表示，即与多数工程图书中的表达方式保持一致：当对点$\mathbf{P}$做$\mathbf{T}$变换时，表达形式上将$\mathbf{T}$写在左边，而将$\mathbf{P}$写在右边。为了方便，书中经常用行向量表示一个点，即省略了转置符号^{T}，各坐标之间用逗号分开。当用列向量表示点时，各坐标上下排列，就不需要再用逗号分开了。

$$\mathbf{P}=[x, y]^{\mathrm{T}}=\begin{bmatrix}x\\ y\end{bmatrix}$$

有时我们需要根据特征点的类型对一个点做标记。例如，一个点可能是一个孔的中心点，一个多边形的顶点，或者是算出来的两线段延长线的交点。在本章稍后讨论的自动匹配算法中，有效地利用了点的类型。

11.2.1 参考坐标系

328

点的坐标总是相对于某个坐标系。通常在进行环境分析时要用到几个坐标系，如在第2章

末所讨论的内容。当涉及多个坐标系时，用上角标来表示点坐标所相对的坐标系。

定义80 如果$\mathbf{P}_j$是一个特征点而$\mathbf{C}$是一个参考坐标系，那么我们用$^{\mathbf{C}}\mathbf{P}_j$来表示该点在坐标系$\mathbf{C}$中的**坐标**。

11.2.2 齐次坐标

很快我们就会明白，无论是公式表达还是计算机运算，利用点的齐次坐标（homogeneous coordinate）都是很方便的，尤其是进行仿射变换时。

定义81 2D点$\mathbf{P} = [x, y]'$的**齐次坐标**是$[sx, sy, s]'$，其中s是比例系数，一般为1.0。

最后，需要注意的是坐标系的习惯表示以及图像显示坐标系的特点。本章中绘出的图表坐标系，与数学课本上的习惯保持一致，即第一个坐标（x或u或r）自原点向右延伸，第二个坐标（y或v或c）自原点向上延伸。但是图像显示程序在显示一幅n行m列的图像时，第一行（行$r = 0$）在顶部而最后一行（行$r = n-1$）在底部，因此r自顶向下延伸而c自左向右延伸。在代数上这不会带来问题，但有时会使我们感觉不习惯，因为显示出来的图像需要在心里逆时针旋转90°以便和数学中的传统方向一致。

11.3 仿射映射函数

有一大类空间变换可以用一个矩阵乘以点的齐次坐标来表示。这里只做简要性介绍，但涉及的内容相当广泛，更详细的介绍可以查阅参考文献中列出的计算机图形学或机器人学方面的教材。向量空间的特点可以参考第5章的内容。

11.3.1 缩放

缩放是常见的一种图像变换。同比例缩放以同样的比例系数改变所有的坐标，或是等量改变所有目标的尺寸。在图11-3中把对2D点$\mathbf{P} = [1, 2]$进行2倍放大，得到新的点$\mathbf{P}' = [2, 4]$。对三角形的三个顶点进行2倍放大，就会使三角形大小变为原来的2倍。缩放是线性变换（linear transformation），这意味着在2D欧几里得空间，对点的缩放可以通过两个基向量的缩放来表示。例如，$[1, 2] = 1[1, 0] + 2[0, 1]$，以及$2[1, 2] = 2(1[1, 0] + 2[0, 1]) = 2[1, 0]+4[0, 1]=[2, 4]$。公式（11-2）表明，对2D点的缩放可以通过乘上一个简单的矩阵来表示，这个矩阵对角线上的数值就是缩放系数。第二个例子是更一般的情况，即x和y单位向量上的缩放系数不同，见公式（11-3）。回忆第2章中讲过的5种坐标系，以mm为单位的实际图像点的坐标，转换到以行、列为单位的像素图像点的坐标，就属于这类缩放变换。对正方形像素的摄像机来讲，$c_x = c_y = c$，但对于电视标准的摄像机来讲，c_x与c_y之比是4/3。

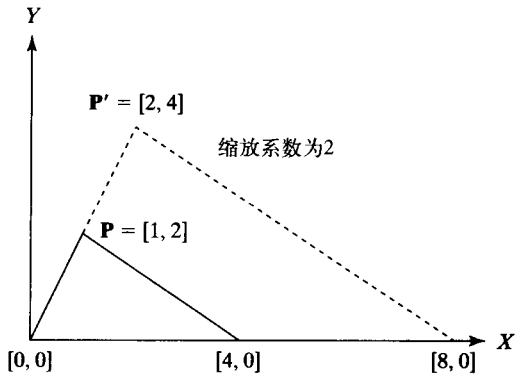


图11-3 对2D向量上点的坐标进行2倍放大

329

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} c & 0 \\ 0 & c \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} cx \\ cy \end{bmatrix} = c \begin{bmatrix} x \\ y \end{bmatrix} \tag{11-2}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} c_x & 0 \\ 0 & c_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c_x x \\ c_y y \end{bmatrix} \tag{11-3}$$

习题11.2 对非正方形摄像机像素进行缩放变换

假设一个正方形的CCD芯片边长0.5英寸，在有效区域内含480 × 640像素。给出一个缩放变换矩阵，把像素坐标[r, c]转换成英寸坐标[x, y]。其中像素图像中心[0,0]与英寸图像中心[0.0]对应。根据你的变换矩阵，第100行200列的像素中心的整数坐标是什么？

11.3.2 旋转

另一种常见的运算是2D空间中点的旋转。图11-4的左边显示的是将2D点$\mathbf{P} = [x, y]$绕原点逆时针转过θ角后得到一个新的点$\mathbf{P}' = [x', y']$。公式（11-4）表明，通过乘上一个简单的矩阵可以方便地表示出2D点绕原点的旋转。与任何线性变换一样，可以把矩阵的各列看成是对基向量变换的结果（图11-4中的右边）。其他任何向量的变换都可以表示为基向量的线性组合。

$$\begin{aligned} R_\theta([x, y]) &= R_\theta(x[1, 0] + y[0, 1]) \\ &= xR_\theta([1, 0]) + yR_\theta([0, 1]) = x[\cos\theta, \sin\theta] + y[-\sin\theta, \cos\theta] \\ &= [x\cos\theta - y\sin\theta,\ x\sin\theta + y\cos\theta] \end{aligned}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x\cos\theta - y\sin\theta \\ x\sin\theta + y\cos\theta \end{bmatrix} \tag{11-4}$$

330

2D旋转可以围绕2D平面上的任意点，而并非一定是参考坐标系的原点。具体见后面的习题。

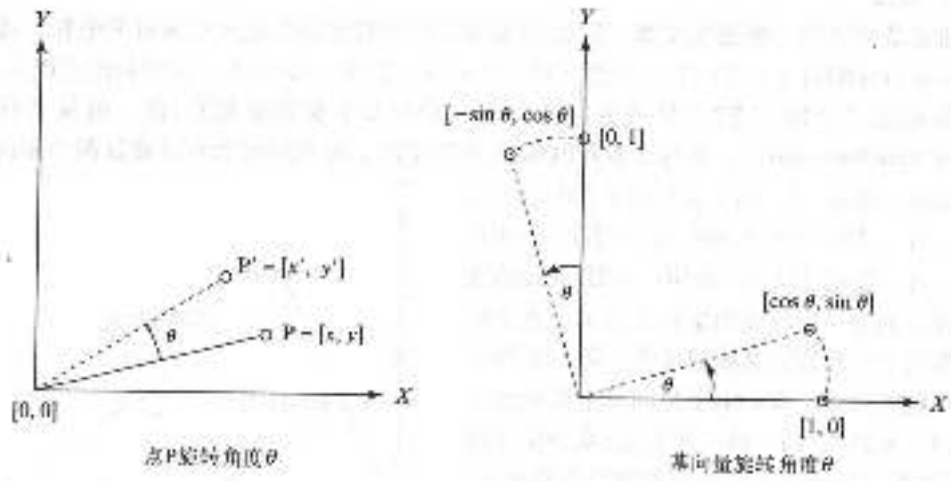


图11-4 2D点的旋转通过基向量的旋转表示

习题11.3

（a）在XY坐标系中画出3个点[0, 0]，[2, 2]和[0, 2]。（b）利用公式（11-2）对3个点进行0.5倍缩放并画出结果。（c）在另一张图上，画出3个点按公式（11-4）绕原点旋转90°的结果。（d）设缩放矩阵是S，旋转矩阵是R，设SR是矩阵S左乘矩阵R的结果。分别用SR和RS对3个点进行变换，二者一样吗？

11.3.3 正交和标准正交变换*

定义82 如果一组向量中的所有向量两两正交，也就是说它们的标量积为零，则称这组向量是**正交**（orthogonal）的。

定义83 如果一组向量是正交集，并且所有向量都具有单位长度，则称这组向量是**标准正交**（orthonormal）的。

旋转变换不改变基向量的长度及其正交性。无论是定性理解还是代数推导都可以得出这个结论，于是直接就可以得出：旋转变换后两点之间的距离与变换前两点之间的距离是相同的。刚体变换（rigid transformation）也有类似的性质，它主要由旋转和平移组成。刚体变换通常针对刚性物体或者用于坐标系变换。非1.0倍的同比例缩放使向量长度发生变化，但两向量间的夹角保持不变。目标上具有不随其位置或者摄像机位姿而变化的一些图像特征，在寻找这些不变特征时，向量长度和夹角问题是要考虑的重要问题。 331

11.3.4 平移

点的坐标常常需要移动一个常量，这相当于改变坐标系的原点。例如对一幅像素图像的行-列坐标进行平移，变换成地图的纬度-经度坐标。因为平移不能把原点[0, 0]仍然映射成原点，所以不能用缩放和旋转变换所用的简单2 × 2矩阵模型，也就是说平移不是线性变换。我们要把变换矩阵扩展到3 × 3维以进行平移和其他运算，相应地要在点向量上增加另一个坐标以得到齐次坐标。一般这个附加坐标值取为1.0，但有时使用其他的值可能会更方便些。

$$\mathbf{P} = [x, y] \simeq [wx, wy, w] = [x, y, 1] \quad for\ w = 1$$

公式（11-5）中所示的矩阵乘法，可以用作对点[x, y]进行平移**D**的模型，即[x′, y′] = **D**([x, y]) = [x + x_0, y + y_0]。

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + x_0 \\ y + y_0 \\ 1 \end{bmatrix} \tag{11-5}$$

习题11.4 绕一点的旋转

求使平面绕点[5, 8]旋转π/2角度的3 × 3矩阵。

提示：首先推导把点[5, 8]移动到新坐标系原点的变换矩阵$D_{-5,-8}$。我们要求的矩阵由$D_{5,8}R_{\pi/2}D_{-5,-8}$组合而成。对3个点[5, 8]、[6, 8]、[5, 9]进行变换，验证所求的矩阵是正确的。

习题11.5 关于坐标轴的反射

关于y轴的反射（reflection）变换是把基向量[1, 0]映射到[−1, 0]，把基向量[0, 1]映射到[0, 1]。（a）构造表示该反射变换的矩阵。（b）对3个点[1, 1]、[1, 0]、[2, 1]进行变换，验证所求的矩阵是正确的。

11.3.5 旋转、缩放和平移

图11-5显示的是一种常见的情况：正方形像素摄像机垂直向下正对工作台平面**W**[**x**, **y**]，拍到一幅图像**I**[**r**, **c**]。需要一个公式把以行和列为单位的像素坐标[r, c]转换到以mm为单位的坐标[x, y]。这可以通过公式（11-6）实现，把旋转**R**、缩放**S**、平移**D**组合起来，表示为$\mathbf{P}_i = \mathbf{D}_{x_0,y_0}\mathbf{S}_s\mathbf{R}_\theta^z\mathbf{P}_p$。有四个参数决定行-列坐标到工作台x − y坐标的映射：旋转的角度θ、把像素变

332 换到mm的比例系数s，以及两个位移量x_0和y_0。由两个控制点（control point）$\mathbf{P}_1$和$\mathbf{P}_2$的坐标可以算出这四个参数。这些点由工作空间中标记清晰而且容易测量的特征点构成，并且在图像中要能够很容易地观察到，例如“+”号，在土地规划应用中，常常把道路的交叉点、建筑物的拐角、河流的急转弯处等用作控制点。要强调的重点是，同一特征点如$\mathbf{P}_1$，可以用两个（或更多）的不同坐标向量表示，一个是与I有关的行-列坐标，另一个是与W有关的mm单位x - y坐标。把这些表示方法分别记为$^{I}\mathbf{P}_1$和$^{W}\mathbf{P}_1$。例如，在图11-5中，有$^{I}\mathbf{P}_1 = [100, 60]$和$^{W}\mathbf{P}_1 = [200, 100]$。

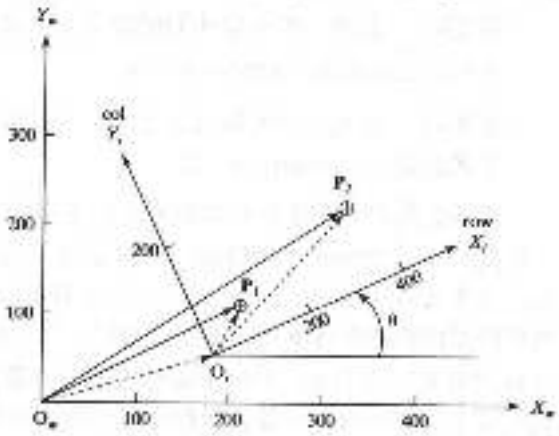


图11-5 正方形像素摄像机垂直向下正对工作台拍取的图像，要对特征点的图像坐标进行旋转、缩放和平移，才能得到工作台空间的坐标

定义84 控制点（control point）是指可以清晰分辨并易于测量的点，通过它们建立不同坐标空间之间的对应关系。

给出点$\mathbf{P}_i$在两个坐标系中的坐标，由矩阵公式（11-6）可以得到两个方程，方程中含4个未知参数。

$$\begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \tag{11-6}$$

$$x_w = x_i\, s\cos\theta - y_i\, s\sin\theta + x_0 \tag{11-7}$$

$$y_w = x_i\, s\sin\theta + y_i\, s\cos\theta + y_0 \tag{11-8}$$

利用点$\mathbf{P}_2$的坐标可得到另外两个方程。通过解这些方程可以求出变换公式中的4个参数。

333 其中θ独立于其他参数，可以按以下方法很容易解出来：首先，向量$\mathbf{P}_1\mathbf{P}_2$在I中的方向可以由$\theta_i = \arctan(^{I}y_2 - {}^{I}y_1)/(^{I}x_2 - {}^{I}x_1)$确定；然后，在W中向量的方向可由$\theta_w = \arctan(^{W}y_2 - {}^{W}y_1)/(^{W}x_2 - {}^{W}x_1)$确定。旋转角就是这两个角度之差$\theta = \theta_w - \theta_i$。确定了$\theta$之后，方程中的所有正弦和余弦项都可以求出，于是产生3个方程，其中含3个未知量，由这3个方程可以很容易解出s和x_0、y_0。读者通过习题11.6完成这个求解过程。

习题11.6 把图像坐标化为工作台坐标

环境如图11-5所示，（如视觉系统需要把物体的位置通知给搬运机器人。）以矩阵的形式给出图像坐标$[x_i, y_i, 1]$到工作台坐标$[x_w, y_w, 1]$的变换关系，利用控制点$^{I}\mathbf{P}_1 = [100, 60]$、$^{W}\mathbf{P}_1 = [200, 100]$、$^{I}\mathbf{P}_2 = [380, 120]$、$^{W}\mathbf{P}_2 = [300, 200]$计算4个参数。

11.3.6 仿射变形实例

通过选择3个点，可以很容易地从数字图像中抽取出平行四边形区域来。第一个点决定要创建输出图像的原点，第二和第三个点决定平行四边形边的极点。输出图像是根据输入图像采样点建立的任意大小的矩形像素阵列。图11-6是基于该思想的程序执行的结果。为了生成中间的那幅图像，由选取的3个点确定的两轴不是正交的，因此在输出图像中出现了切变。这个切变可以通过以下方式去除，即从中间那幅图中抽取第二幅图，使新的采样轴与倾斜的轴

对齐。图11-7是另一个例子，从20美元的钞票上抽取出安德鲁·杰克逊的扭曲脸部（参见图11-7）。在这两个例子中，尽管都只抽取出了输入图像的一部分，但输出图像包含的像素数量与输入图像相同。

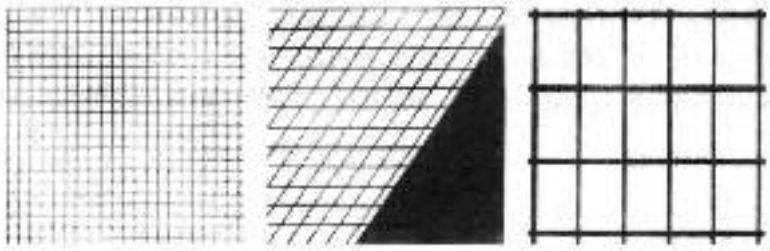


图 11-6

（左图）128×128的网格数字图像

（中图）经仿射变形抽取的128×128范围图像，仿射变形由其图像中的3个点确定

（右图）对中间图像进行部分矫正后的128×128图像

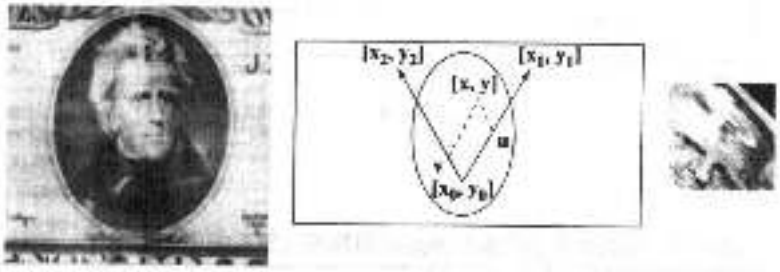


图11-7 利用带切变的仿射映射，从20美元钞票上抽取的安德鲁·杰克逊扭曲的脸部

生成图11-7脸部图像的程序，利用用户选取的3个点对平行四边形区域进行变换。输出图像是$n \times m$或512×512像素，像素坐标表示为[r，c]。对于输出图像中的每个像素[r，c]，在像素[x，y]处对输入图像的值进行采样，像素[x，y]通过变换公式（11-9）计算得到。公式中的第一种形式是基于基向量的直观表达形式，公式中的第二种形式是与第一种形式等价的标准表达形式。 334

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + \frac{r}{n}\left(\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} - \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}\right) + \frac{c}{m}\left(\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} - \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}\right)$$

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} (x_1 - x_0)/n & (x_2 - x_0)/m & x_0 \\ (y_1 - y_0)/n & (y_2 - y_0)/m & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r \\ c \\ 1 \end{bmatrix} \tag{11-9}$$

概念上讲，点[x，y]是根据沿新轴方向的新单位向量定义的，这个新轴由用户的选择点确定，计算出的坐标[x，y]必须经过取整处理才能得到整数值的像素坐标，与数字图像^{1}I的像素位置对应。如果x或y中的任何一个超出了范围，则对应输出点就被设置为黑，这种情况下^{2}I[r，c] = 0；否则^{2}I[r，c] = ^{1}I[x，y]。在杰克逊脸部的右上方可以看到一个黑色的三角形，这是由于采样平行四边形超出了20美元钞票输入图像的范围所造成。

11.3.7 目标识别与定位实例

这个例子是计算变换矩阵，对图11-8中左边所示的目标模型与右边的目标图像进行匹配。

假设自动特征抽取算法只找到目标内的三个特征孔。空间变换将模型中的点[x, y]映射到图像中的点[u, v]。假设成像环境是受控的，已经通过缩放对图像坐标进行了变换并生成图示的$u-v$坐标。现在仅需要两个图像点就可以推出旋转和平移矩阵，该旋转和平移将使模型上的点与图像上的点对应起来。表11-1和表11-2中所示的是模型和图像中点的位置以及这些点之间的距离。假设对应的点对是（A, H_2）和（B, H_3），利用这一对对应点来推导变换关系。注意这些对应点对与已知的点间距离关系是一致的。我们将在11.5节讨论做出这些假定的算法。

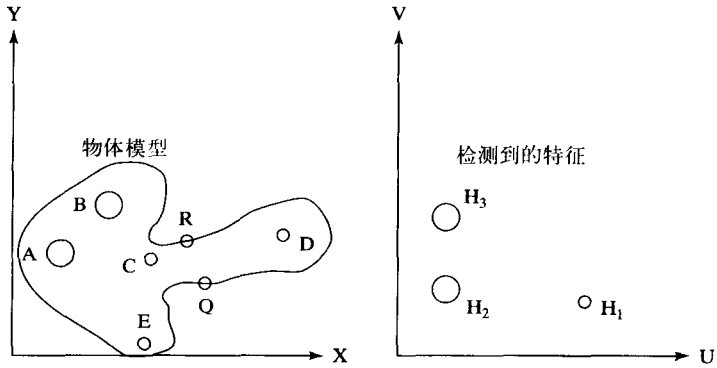


图 11-8

（左图）物体模型

（右图）在图像上检测到的三个孔

表11-1 模型中点的位置和各点间的距离（取孔的中心坐标）

点	坐标	到A的距离	到B的距离	到C的距离	到D的距离	到E的距离
A	(8, 17)	0	12	15	37	21
B	(16, 26)	12	0	12	30	26
C	(23, 16)	15	12	0	22	15
D	(45, 20)	37	30	22	0	30
E	(22, 1)	21	26	15	30	0

表11-2 图像中点的位置和各点间的距离（取孔的中心坐标）

点	坐标	到H_1的距离	到H_2的距离	到H_3的距离
H_1	(31, 9)	0	21	26
H_2	(10, 12)	21	0	12
H_3	(10, 24)	26	12	0

模型中由A到B的向量方向是$\theta_1 = \arctan(9.0/8.0) = 0.844$，图像中与之对应的$H_2$到$H_3$的向量方向是$\theta_2 = \arctan(12.0/0.0) = \pi/2 = 1.571$。因此，旋转角$\theta = 0.727$弧度。利用公式（11-6），将
335 匹配点的坐标即模型中点A和图像中点H_2的坐标代入公式，将得到公式（11-10），其中u_0、v_0是图像面上未知的平移成分。注意$\sin\theta$和$\cos\theta$的值实际上是已知的，因为θ已经算出来了。

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 10 \\ 12 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & u_0 \\ \sin\theta & \cos\theta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 \\ 17 \\ 1 \end{bmatrix} \tag{11-10}$$

从矩阵中所含的两个线性方程可以很容易算出$u_0 = 15.3$和$v_0 = -5.95$，利用匹配点B和H_1进行检验，可以得出类似的结果。每对不同的点都将得出略有出入的变换。为了更精确的算出变换关系，可以采用覆盖2D空间的许多个点来进行运算，这个方法将在后面进行讨论。完成了空间变换的计算，现在就可以计算模型上的任意点在图像空间的位置，包括抓取点$R = [29, 19]$和$Q = [32, 12]$。模型上的点R变换到图像上的点$'R = [24.4, 27.4]$。用$Q = [32, 12]$作为变换的输入，输出另一个抓取点在图像中的位置$'Q = [31.2, 24.2]$。 336

$$\begin{bmatrix} u_R \\ v_R \\ 1 \end{bmatrix} = \begin{bmatrix} 24.4 \\ 27.4 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 15.3 \\ \sin\theta & \cos\theta & -5.95 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 29 \\ 19 \\ 1 \end{bmatrix} \tag{11-11}$$

如果已知从图像坐标到支撑物体的工作台坐标的变换关系，机器人就能够抓取摄像机下的真实物体。当然，考虑到诸如成像畸变、特征检测不准以及计算错误等造成的微小影响，机器人手爪张开的宽度应比由$'R'Q$变换得到的长度略宽一些。尽管图像只有离散空间的采样点，抓取行为却是针对真实的连续物体，而且坐标也是有意义的实数。图像数据本身只在整数网格点上有定义，如果我们的目的是通过检验明亮的图像像素来验证孔$'C$和$'D$的存在，那么对模型点的变换结果应进行取整处理，这样才能与图像像素位置吻合。不然的话，就要检验包含变换后实际坐标的整个数字邻域。通过这个例子，我们看到了比对方法在2D目标识别方面的潜力，比对是指将目标模型与图像中的重要特征点进行对比。

定义85 利用旋转、缩放和平移（RST）把模型特征变换成图像特征，通过匹配进行目标识别的方法称为**比对识别**（recognition-by-alignment）。

习题11.7 变换顺序能够互换吗？

假设我们有3个表示原始变换的矩阵：$\boldsymbol{R}_\theta$表示绕原点的旋转，$\boldsymbol{S}_{s_x,s_y}$表示缩放，$\boldsymbol{D}_{x_0,y_0}$表示平移。（a）缩放和平移是否可以互换？即$\boldsymbol{S}_{s_x,s_y}\boldsymbol{D}_{x_0,y_0} = \boldsymbol{D}_{x_0,y_0}\boldsymbol{S}_{s_x,s_y}$成立吗？（b）旋转和缩放是否可以互换？即$\boldsymbol{R}_\theta\boldsymbol{S}_{s_x,s_y} = \boldsymbol{S}_{s_x,s_y}\boldsymbol{R}_\theta$成立吗？（c）旋转和平移可以互换吗？通过代数推导和定性思考得出结论，并进行解释。

习题11.8

构造出关于直线$y = 3$的反射变换矩阵，首先进行平移$y_0 = -3$，随后是关于x轴的反射。通过求3个点[1，1]，[1，0]，[2，1]的变换结果，验证得出的变换矩阵是否正确，并绘出输入点和输出点。 337

习题11.9 验证矩阵$\boldsymbol{D}_{x_0,y_0}$与$\boldsymbol{D}_{-x_0,-y_0}$的乘积是一个$3 \times 3$的单位矩阵，解释结果为何是这样。

11.3.8 一般仿射变换*

我们已经讲了仿射变换中旋转、缩放和平移3种基本变换。第四种基本变换是切变。图11-9显示切变的结果。在u-v坐标系中，所有的点向量沿v轴方向移动，移动幅度与它们到v轴的距离成正比。关于v轴的切变，点$[u, v]$将变换到$[u, e_x u + v]$；关于u轴的切变，点$[u, v]$将变换到$[u + e_y v, v]$。公式（11-12）和公式（11-13）给出了矩阵方程。回想一下，切变矩阵的列向量正好是基向量变换后的图像。

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ e_u & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{11-12}$$

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & e_v & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{11-13}$$

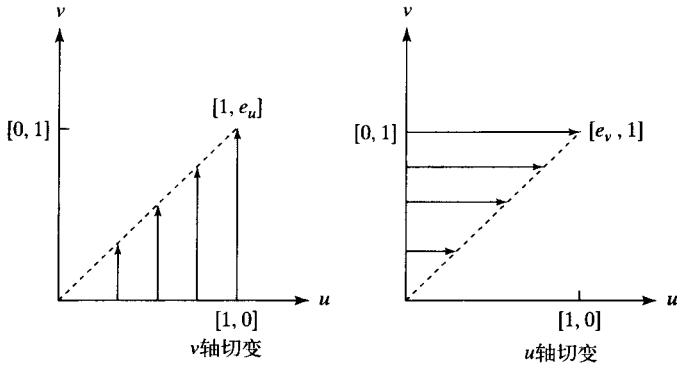


图 11-9

反射是仿射变换的第五种基本变换。关于u轴的反射变换分别把基向量[1, 0]、[0, 1]映射到[1, 0]、[0, −1]，而关于v轴的反射变换分别把基向量[1, 0]、[0, 1]映射到[−1, 0]、[0, 1]。用2 × 2或3 × 3的矩阵就可以很清楚地表示出来。任意仿射变换都可以通过旋转、缩放、平移、切变和反射组合而成。这些基本变换的逆是存在的，并且与基本变换具有相同的形式。因此，如公式（11-14）所示，一般的仿射变换矩阵含有6个参数。已知3对不共线的对应点坐标，可
338 以得到该类型的3个矩阵方程，通过解这3个方程，就可以求出这6个参数。在图11-6中我们已经看到对倾斜网格进行切变运算的情况。

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{11-14}$$

11.4 最佳2D仿射变换*

如公式（11-15）所示，一般的2D-2D仿射变换需要求出6个参数，求这6个参数时只用了3组相匹配的点对（$[x_j, y_j]$，$[u_j, v_j]_{j=1,\cdots,3}$）。

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{11-15}$$

这些点中任意一个的坐标存在误差，都必然会造成求得的参数存在误差。解决这个问题更好的方法之一是，采用更多的匹配控制点以得到6个参数的最小二乘估计。类似第10章直线拟合的方法，我们可以定义一个误差指标函数。

$$\begin{aligned} \varepsilon(a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}) = \sum_{j=1}^{n} & ((a_{11}x_j + a_{12}y_j + a_{13} - u_j)^2 \\ & + (a_{21}x_j + a_{22}y_j + a_{23} - v_j)^2) \end{aligned} \tag{11-16}$$

误差函数分别对6个变量a_{ij}求偏导$\partial e/\partial a_{ij}$，使之为0，就得到6个方程，用矩阵的形式表示如下：

$$\begin{bmatrix} \Sigma x_j^2 & \Sigma x_j y_j & \Sigma x_j & 0 & 0 & 0 \\ \Sigma x_j y_j & \Sigma y_j^2 & \Sigma y_j & 0 & 0 & 0 \\ \Sigma x_j & \Sigma y_j & \Sigma 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Sigma x_j^2 & \Sigma x_j y_j & \Sigma x_j \\ 0 & 0 & 0 & \Sigma x_j y_j & \Sigma y_j^2 & \Sigma y_j \\ 0 & 0 & 0 & \Sigma x_j & \Sigma y_j & \Sigma 1 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \end{bmatrix} = \begin{bmatrix} \Sigma u_j x_j \\ \Sigma u_j y_j \\ \Sigma u_j \\ \Sigma v_j x_j \\ \Sigma v_j y_j \\ \Sigma v_j \end{bmatrix} \tag{11-17}$$

习题11.10

用三对匹配控制点（[0, 0]，[0, 0]），（[1, 0]，[0, 2]）、（[0, 1]，[−2, 0]）求解方程（11-17），你的计算结果与通过基向量变换的结果一样吗？ 339

习题11.11

用三对匹配控制点（[0,0]，[1,2]），（[1,0]，[3,2]），（[0,1]，[1,4]）求解方程（11-17），你的计算结果与通过基向量变换的结果一样吗？

实现图像与地图或者图像与图像之间的对应，采用很多控制点是一种很常用的方法。图11-10中显示的基本上是同一场景的两幅图像。在图的下面给出了11对匹配控制点。在两幅图像（或地图）中，控制点都是目标的角点，这些点都是具有唯一性的可识别点。本例中的控 340
制点是通过显示程序然后利用鼠标选取的。余差列表表明，利用求得的变换矩阵进行计算，右侧图像中的u，v坐标值与变换得到的值相差不到两个像素，大多数余差是小于一个像素。使用自动特征检测方法，在亚像素精度上确定特征点的位置，可以得到更好的结果。如果通过计算机鼠标和人眼来确定控制点，控制点坐标常常会产生一个像素的误差。利用求得的仿射变换，左侧图像中的目标就可以在右侧图像寻找到。这样我们就不难理解，为了更新征税地图上的目标物，征税地图与航测图像的对应关系是怎样建立起来的。

习题11.12

从图11-10中选取3对匹配控制点，例如（[288,210,1]，[31,160,1]），验证仿射变换矩阵把第一幅图映射成第二幅图。

11.5 仿射映射法2D目标识别

本节研究从模型点映射到图像点的2D目标识别方法。在仿射映射部分我们已经介绍了通过比对进行识别的方法。通用方法采用的是一般特征点，而每个应用领域都存在一些特殊特征，可以给这些特征附加特殊标记。在零件分类应用中我们可能选取角点或孔的中心，而在土地测量应用中可能选取交叉点和高曲率土地与水域的边界点。

图11-11是一个总的模型-匹配范例。图11-11a是飞机零件的边界模型，在匹配中可能用到的特征点用小黑点做了标记。图11-11b是真实的飞机零件图，与模型的方位基本一致。图11-11c是真实零件旋转45°后的图像。图11-11d也是真实零件的图像，但由于摄像机角度的关系导致图像存在明显的扭曲。本节要讲的识别算法，是确定一幅给定的图像中，如图11-11b，11-11c和11-11d，是否包含如图11-11a中的目标模型，并且确定摄像机与目标之间的相对位姿（pose）

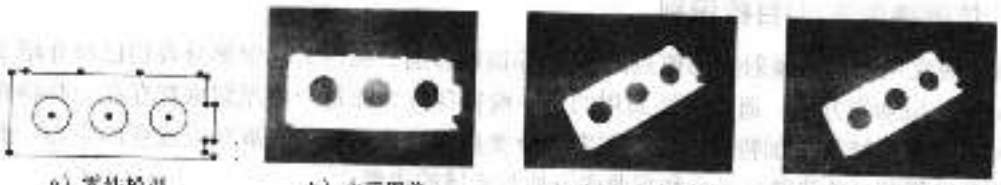
（包括位置和姿态）。





```
======  最优2D仿射拟合程序  ======

匹配控制点对:
288 210  31 160    232 288  95 205    195 372 161 229    269 314 112 159
203 424 199 209    230 136 130 196    284 401 180 124    327 428 198  69
284 299 100 146    337 231  45 101    369 223  38  64

变换矩阵:

[ -0.0414 ,  0.773 , -119
  -1.120  , -0.213 ,  526 ]

22个方程的余差（以像素为单位）:

0.18  -0.68  -1.22  0.47  -0.77  0.06   0.34  -0.51   1.89  0.04  0.96
1.51  -1.04  -0.81  0.05   0.27  0.13  -1.12   0.39  -1.04 -0.12  1.81

======  拟合程序完成  ======
```

图11-10 同一场景的图像，以及从左图到右图的最佳仿射映射。该映射采用11个控制点得到。左图中的坐标用[x，y]表示，x向下延伸，y向右延伸；右图中的坐标用[u，v]表示，u向下延伸，v向右延伸。图像下面的11组坐标是匹配控制点的x、y、u、v。你能对两幅图像的特征进行匹配吗？（图像由Oliver Faugeras教授供）



a）零件模型　b）水平图像　c）旋转45°的图像　d）旋转且扭曲的图像

图11-11 飞机零件的2D模型和3幅匹配图

11.5.1 局部特征焦点法

局部特征焦点法利用目标的局部特征和它们之间的2D空间关系进行目标识别。首先要建立一套目标模型，每个模型对应一个要识别的目标物体。每个模型要包含一组焦点特征

(focus feature)，也就是目标物体上容易被检测到的主要特征，当然这些特征不能被其他物体遮挡。对每个焦点特征，它的邻近特征也包含在模型中。可用这组邻近特征验证是否找到正确的焦点特征，以及帮助确定目标物体的位姿。

在匹配阶段，从包含一个或多个目标的图像上抽取特征。匹配算法首先查找焦点特征。 341
在找到属于给定模型的焦点特征后，再查找焦点特征附近的图像特征，这些特征要与模型中焦点特征的邻近特征尽可能多地匹配上。一旦找到了这样的一组图像特征，并且这些图像特征与目标模型的特征之间的对应关系已经确定，那么算法就做出图像中包含该目标的假设，然后通过验证技术确定这个假设的正确性。

验证过程必须确定，图像中是否有足够的证据能够证明场景中确实存在假设存在的目标。对于多面体，常用目标边界作为合适的证据。利用相对应的特征确定从模型点到图像点的仿射变换，然后用这个变换把边界线段变换到图像空间中。只要不存在遮挡现象，变换后的线段应该大体上与图像中的线段对齐。由于图像噪声和特征抽取及匹配所产生的误差，变换后的线段不可能与图像中的线段完全重合，但可以找到包含变换后线段的一个矩形区域，作为与图像线段相匹配的证据。如果找到了足够的证据，那么就认为该模型线段通过验证，并进行标记。如果足够多的模型线段通过验证，那么就认为图像中确实存在该目标，并且位于经变换运算所得到的位置处。

局部特征焦点算法，把已知的模型**F**和一幅图像进行匹配，具体算法如下所述。模型中有一组焦点特征{$\mathbf{F}_1$，$\mathbf{F}_2$，…，$\mathbf{F}_M$}。对于每个焦点特征$\mathbf{F}_m$，都对应一组邻近特征$\mathbf{S}(\mathbf{F}_m)$，这些邻近特征用于验证焦点特征。在图像上检测到一组图像特征{$\mathbf{G}_1$，$\mathbf{G}_2$，…，$\mathbf{G}_I$}。对于每个图像 342
特征$\mathbf{G}_i$，都有一组邻近的图像特征$\mathbf{S}(\mathbf{G}_i)$。

图11-12是局部特征焦点算法的示意图，包括两个模型E和F，以及一幅图像。检测到的特征是圆孔和尖角。假设模型F中的局部特征F1与图像中的特征G1对应，并发现模型中的邻近特征F2、F3和F4分别与图像中的邻近特征G2、G3和G4存在很好的对应。验证过程将显示，模型F确实在图像中存在。考虑另一个模型E，已经做出假设：特征E1及邻近特征E2、E3和E4分别与图像中的特征G5、G6、G7和G8对应，但是在进行验证时，模型E的边界与图像中的线段不能很好的对齐，那么这个假设就要放弃掉。

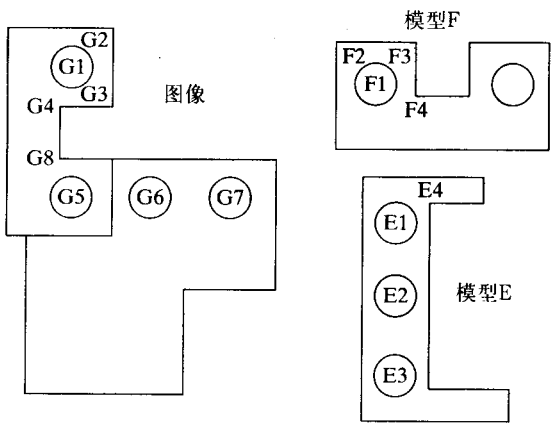


图11-12 局部特征焦点法。图像中显示模型F摆在另一个目标上

343

算法11.1 用局部特征焦点法寻找从模型特征到图像特征的变换

$\mathbf{G}_i$, i = 1,…,I，检测到的图像特征集。

$\mathbf{F}_m$, m = 1,…,M，模型中的焦点特征集。

S(f)，特征f的近邻特征集。

procedure local_feature_focus(**G**, **F**)；

{

```
for each focus feature F_m
  for each image feature G_i of the same type as F_m
    {
      Find the maximal subgraph S_m of S(F_m)that
        matches a subgraph S_i of S(G_i);
      Compute the transformation T that maps the points of
        each feature of S_m to the corresponding feature of S_i;
      Apply T to the boundary segments of the model;
      if enough of the transformed boundary segments find
        evidence in the image then return(T);
    };
}
```

11.5.2 位姿聚类

我们已经看到，利用旋转、缩放和平移变换，根据两个匹配控制点能够得到模型特征与图像特征间的对应关系。一旦在图像与模型之间找到两个匹配控制点，就可以通过公式（11-6）得出结果。由于匹配时可能存在歧义性，使得自动获得匹配控制点并不容易。位姿聚类方法对所有可能的控制点对都算出一个RST队列，然后进行检查以找到相似参数的聚类。如果在模型与图像间确实存在很多匹配特征点，那么在参数空间中就应该存在一个聚类。位姿聚类算法简单表示如下。

算法11.2 通过位姿聚类寻找从模型特征到图像特征的变换

$\mathbf{P}_i$，$\mathbf{i}=\mathbf{1},\cdots,\mathbf{D}$，检测到的图像特征集。

$\mathbf{L}_j$，$\mathbf{j}=\mathbf{1},\cdots,\mathbf{M}$，存储的模型特征集。

```
procedure pose_clustering (P, L);
{
for each pair of image feature points (P_i, P_j)
  for each pair of model feature points (L_m, L_n)of same type
    {
      compute parameters α of RST mapping
        pair (L_m, L_n)onto (P_i, P_j);
      contribute α to the cluster space;
    };
examine space of all candidates α for clusters;
verify every large cluster by mapping all
  model feature points and checking the image;
return(verified {α_k});
}
```

定义86 设T是一个空间变换，这个变换把模型M与图像I中的目标O对应起来。目标

O的**位姿**（pose）是指由T的参数α所定义的位置和方向。 344

利用所有的特征点对，将造成过多的冗余。在匹配航测图像与地图的应用中，可以使用检测到的公路网交叉点或区域如田地角交叉点。把交叉的角度作为匹配中使用的类型，比如常见的交叉类型有“L”、“Y”、“T”、“箭头”和“X”，如图11-13所示。假设我们仅使用组合类型LX或TY。在图11-14所示的例子中，有5个模型对和4个图像对。尽管可能有4 × 5 = 20种配对方式，但两端类型一致的配对方式只有10种。表11-3中是根据这10对匹配特征算出的变换参数。这10个变换中，除了表中最后一列以*标注的3个变换外，其他变换的参数之间都明显不一致。这3组参数构成一类，其参数平均值为$\theta = 0.68$，$s = 2.01$，$u_0 = 233$，$v_0 = -41$。为了实现正确的匹配，希望差异更小一些，但这个差异是由于特征点定位的微小误差和成像过程中的非线性畸变引起的。如果RST匹配中的参数值不够精确，可以用它们来验证匹配点，然后把这些匹配点作为控制点寻找匹配精度更高的非线性映射或仿射映射（带更多的参数）。 345

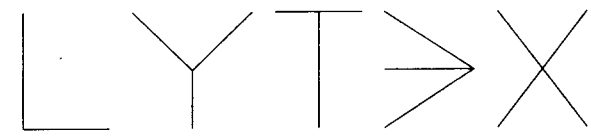


图11-13 匹配中常用的线段连接

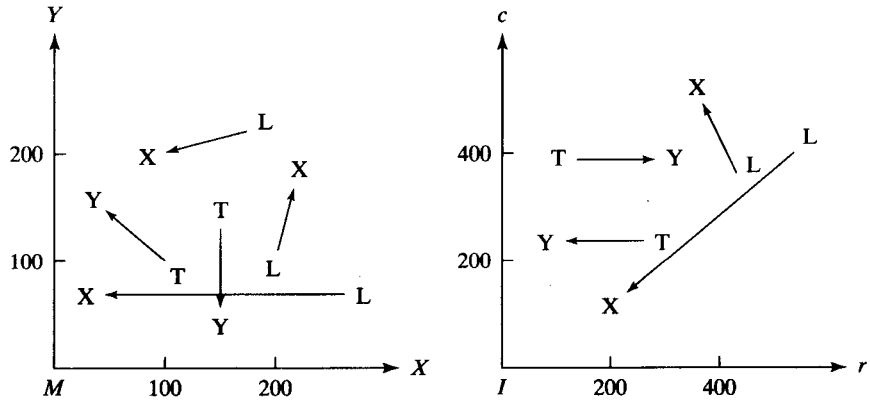


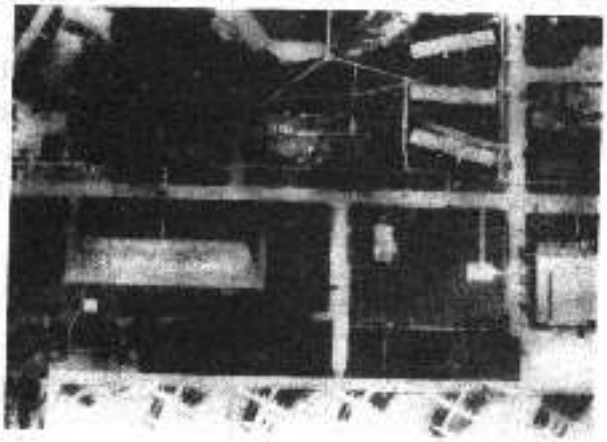
图11-14 位姿检测示例，使用5个模型特征点对和4个图像特征点对

表11-3 由图11-14中的10种位姿计算构成的聚类空间

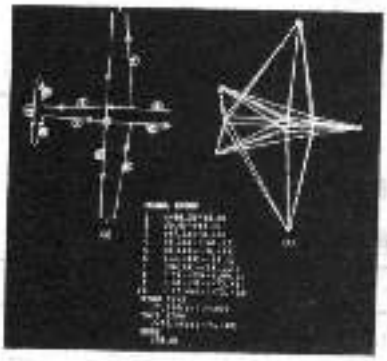
模 型 对	图 像 对	θ	s	u_0	v_0	
L(170, 220), X(100, 200)	L(545, 400), X(200, 120)	0.403	6.10	118	−1240	
L(170, 220), X(100, 200)	L(420, 370), X(360, 500)	5.14	2.05	−97	514	
T(100, 100), Y(40, 150)	T(260, 240), Y(100, 245)	0.663	2.05	225	−48	*
T(100, 100), Y(40, 150)	T(140, 380), Y(300, 380)	3.87	2.05	166	669	
L(200, 100), X(220, 170)	L(545, 400), X(200, 120)	2.53	6.10	1895	200	
L(200, 100), X(220, 170)	L(420, 370), X(360, 500)	0.711	1.97	250	−36	*
L(260, 70), X(40, 70)	L(545, 400), X(200, 120)	0.682	2.02	226	−41	*
L(260, 70), X(40, 70)	L(420, 370), X(360, 500)	5.14	0.651	308	505	
T(150, 125), Y(150, 50)	T(260, 240), Y(100, 245)	4.68	2.13	3	568	
T(150, 125), Y(150, 50)	T(140, 380), Y(300, 380)	1.57	2.13	407	60	

位姿聚类可使用低级特征，但如果对特征进行类型过滤，精度和效率都会得到提高。可用简单的$O(n^2)$算法进行聚类：对每一个参数集α，根据某种距离测度统计与它接近的其他参数集α_i的个数。这样在聚类空间中，对n个参数集中的每一个，都需进行$n-1$次距离计算。一种更快但不够灵活的方法是装箱算法，装箱是文献中介绍的一种传统方法，在第10章关于霍夫变换部分讨论过。每个生成的参数集都对参数空间的箱格有贡献，然后为了进行统计要对所有的箱格进行检查，当类似α的集合跨过邻近的箱格时，就可能丢失一个聚类。

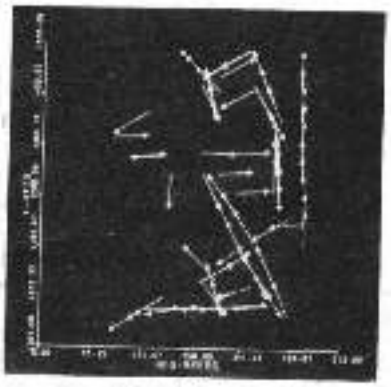
聚类方法已经被用来检测航测图像中是否存在特殊的飞机模型，如图11-15所示，采用第5章和第10章中的方法，从图像中抽取边缘和曲率特征。将这些特征构成的不同覆盖窗口与图b中显示的模型相匹配。图c显示某个窗口内检测到的边缘，其中使用相同的变换参数使很多特征与模型特征相对应。



a)



b)



c)

图11-15 用位姿聚类检测某架特殊飞机（经IEEE许可再版）

a）机场的航测图像

b）依据实际边缘，以及由角点和曲线顶点构成的抽象边缘构成的飞机模型

c）图像窗口包含检测到的特征，这些特征经相同的变换得到，并且与多个模型特征相匹配。

11.5.3 几何散列

局部特征焦点法和位姿聚类算法都是将单一模型与一幅图像匹配。如果存在几个不同的目标模型，那么这两种方法就要对每个模型分别进行运算，而每次只能针对一个模型，因此当存在很多不同目标时，这两种方法就不太合适。几何散列主要针对大型模型数据库。几何

散列需要进行大量的离线预处理而且占用大量空间，这是为了能够快速进行在线目标识别和位姿确定。

假设已知

1. 一个大型模型数据库

2. 一个未知目标，其特征从图像中抽出，该目标是某个模型的仿射变换结果

并希望确定究竟是哪一个模型，采用的变换是什么。 346

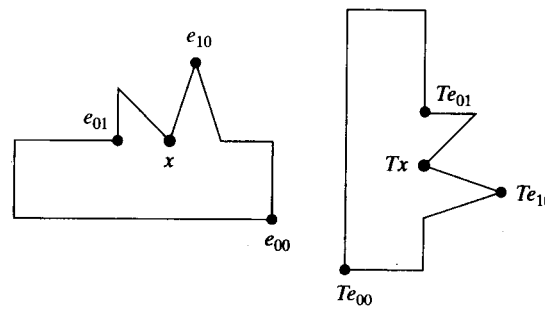
把模型**M**看作特征点的有序集合。可以用**M**的任意三个不共线点的子集$E = \{e_{00}, e_{01}, e_{02}\}$来构造一个仿射基集，这个仿射基集定义了**M**上的一个坐标系，如图11-16a所示。一旦选定这个坐标系，就可以用仿射坐标(ξ, η)的形式表示任意的点x, $x \in \mathbf{M}$，其中

$$x = \xi(e_{10} - e_{00}) + \eta(e_{01} - e_{00}) + e_{00} \quad 347$$

此外，对点x进行仿射变换T，得到

$$Tx = \xi(Te_{10} - Te_{00}) + \eta(Te_{01} - Te_{00}) + Te_{00}$$

这样Tx关于$(Te_{00}, Te_{01}, Te_{10})$的仿射坐标，与$x$关于$(e_{00}, e_{01}, e_{10})$的仿射坐标相同，都是$(\xi, \eta)$。如图11-16b所示。



a）原始目标　　b）变换后的目标

图11-16 关于仿射基集的点的仿射变换

离线预处理（offline preprocessing）离线预处理阶段建立一个散列表，表中包含数据库中所有的模型。这个散列表建立之后，仿射坐标对(ξ, η)指明散列表中一个箱格，散列表中存储了模型-基对（**M**，**E**）的列表清单，其中模型**M**上的某点x具有关于基**E**的仿射坐标(ξ, η)。离线预处理算法在算法11.3中给出。

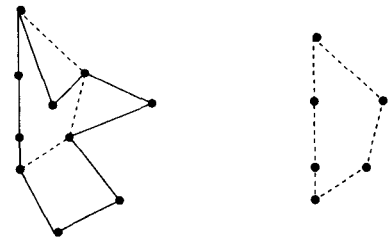
在线识别（Online Recognition）在线识别阶段使用预处理阶段建立的散列表。识别阶段也使用一个用模型-基对做索引的累加数组A。对每对（M，E）都将箱格初始化为零，箱格用来对存在使（M，E）属于图像的变换T的假设进行投票表决。仅对那些得票较高的模型-基对计算出实际变换，并作为后面投票表决验证阶段的一部分。在线识别和位姿估计算法在下面给出。

假设有s个模型，每个模型大概有n个点，那么预处理阶段的复杂度是$O(sn^4)$，这是由于要处理s个模型，每个模型要处理三元组的复杂度为$O(n^3)$，处理模型中其他点的复杂度为$O(n)$。在匹配中，工作量取决于在图像中找到的特征点质量如何，其中有多少被遮挡住，以及检测出多少错的或额外的特征点。最好的情况是，第一次选择的三元组，就是来自同一模型的三个实际特征点，那么这个模型的得票数就很高，验证过程成功，工作就完成了。对于这个最好的情况，假设散列表的平均列长度是一个很小的常数，散列时间也基本是个常数，那么匹配阶段的复杂度大概是$O(n)$。在最坏情况下，比如模型根本不在数据库中，每个三元组都被试过，那么复杂度是$O(n^4)$。在实际中，对所有三元组都试过的情况很少发生，而只试一组就成功的情况也同样少见。下面是会带来误差的几个方面。

348

1. 特征点坐标有误差
2. 丢失或添加特征点
3. 遮挡，多个目标
4. 不稳定的基
5. 对点的子集做不合理的仿射变换

特别地，算法有可能虚构一个基于点的子集的变换，这些点通过了验证测试，但结果实际上是错误的。图11-17举例说明了这一点。位姿聚类和局部特征焦点法同样会产生类似的情况。



a）图像点　　b）虚构的目标

图11-17 几何散列算法，错误地认为图像中存在一个已知模型。本例中，60%的特征点（左）将验证通过图像中存在一个目标（右）的假设，尽管这个目标实际上并不存在

算法11.3 几何散列的离线预处理

D是模型数据库。
H是初值为空的散列表。

```
procedure GH_Preprocessing (D, H);
{
for each model M
  {
  Extract the feature point set F_M of M;
  for each noncollinear triple E of points from F_M
    for each other point x of F_M
      {
      Calculate (ξ, η) for x with respect to E;
      Store (M, E) in hash table H at index (ξ, η);
      };
  };
}
```

349

算法11.4 使用散列表寻找正确模型和把图像特征映射到模型特征的变换

H是由预处理阶段建立的散列表。
A是用（**M**，**E**）做索引的累加数组。
I是要分析的图像。

```
procedure GH_Recognition(H, A, I);
{
Initialize accumulator array A to all zeroes;
Extract feature points from image I;
for each basis triple F
  {
  for each other point v
```

```
        {
        Calculate (ξ, η) for v with respect to F;
        Retrieve the list L of model-basis pairs from the
            hash table H at index(ξ, η);
        for each pair (M, E) of L
            A[M, E] = A[M, E] + 1;
        }
      Find the peaks in accumulator array A;
      for each peak(M, E)
        {
        Calculate T such that F = TE;
        if enough of the transformed model points of M find
          evidence on the image then return (T);
        }
      }
  }
```

11.6 相关匹配法2D目标识别

我们已经讨论了3种方法，它们将观测到的图像点与模型点进行匹配。这3种方法是局部特征焦点法、位姿聚类法和几何散列法。本节我们讨论三个简单的带有一般性的目标识别范例。3个范例都把识别看成是从模型结构到图像结构的映射，即依据模型特征寻找图像特征的一致性标记，识别就是把一个模型足够多的特征映射成有效图像特征。3个范例的不同之处在于如何建立映射。 350

匹配范例中用到的4个重要概念是部件（part）、标记（label）、分配（assignment）和关系（relation）。

- **部件**是场景中的目标或结构，如区域、边缘、孔、角点或团儿。
- **标记**是在某个层次上为识别部件而标识的符号。
- **分配**是从部件到标记的一个映射。如果P_1表示一个区域，L_1是表示湖的符号，L_2是表示田地的符号，一个分配可能是（P_1，L_2），也可能是具有歧义性的（P_1，$\{L_1, L_2\}$）。（P_1，NIL）分配对表示P_1在现有的标记集合中没有对应的解释。场景的解释是指所有分配对构成的集合。
- **关系**是正式的数学概念。我们能够找到和算出场景中各目标之间的关系，并且把这种关系存储起来。例如，$R_4(P_1, P_2)$就可以表示区域P_1与区域P_2是相邻的关系。

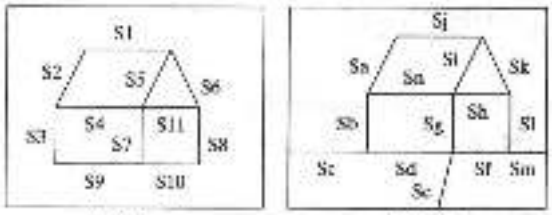
有了这4个概念，我们就可以对一致性标记进行定义了。

定义87 已知部件集合P，这些部件的标记集合L，部件集P上的关系R_P，以及标记集L上的关系R_L，那么**一致性标记**（consistent labeling）f就是满足下列条件的从部件到标记的一个分配。

$$若(p_i, p_{i'}) \in R_P, \quad 则(f(p_i), f(p_{i'})) \in R_L$$

例如要寻找两图像间的匹配关系，对每幅图像我们都有从中抽取的一组线段，以及相连

线段对之间的连接关系。设P是第一幅图像中的线段集合，R_P表示相连线段对的集合，$R_P \subseteq P \times P$。类似设L是第二幅图像中的线段集合，R_L是相连线段对的集合，$R_L \subseteq L \times L$。图11-18表示出两幅图像以及集合P、R_P、L和R_L。注意R_P和R_L都是对称关系，如果(Si, Sj)属于一个关系，则(Sj, Si)也属于这个关系。例子中只列出了满足$i<j$的组对(Si, Sj)，其镜像组对(Sj, Si)隐含存在。



图像1　　图像2

$P = \{S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11\}$.
$L = \{Sa, Sb, Sc, Sd, Se, Sf, Sg, Sh, Si, Sj, Sk, Sl, Sm\}$.

$R_P = \{(S1, S2), (S1, S5), (S1, S6), (S2, S3), (S2, S4), (S3, S4), (S3, S9), (S4, S5), (S4, S7), (S4, S11), (S5, S6), (S5, S7), (S5, S11), (S6, S8), (S6, S11), (S7, S9), (S7, S10), (S7, S11), (S8, S10), (S8, S11), (S9, S10)\}$.

$R_L = \{(Sa, Sh), (Sa, Sj), (Sa, Sn), (Sb, Sc), (Sb, Sd), (Sb, Sn), (Sc, Sd), (Sd, Se), (Sd, Sf), (Sd, Sg), (Sc, Sf), (Se, Sg), (Sf, Sg), (Sf, Sl), (Sf, Sm), (Sg, Sh), (Sg, Si), (Sg, Sn), (Sh, Si), (Si, Sk), (Sh, Sn), (Si, Sj), (Si, Sk), (Si, Sn), (Sj, Sk), (Sk, Sl), (Sl, Sm)\}$.

图11-18　一致性标记问题举例

在这个例子中，一致性标记就是映射f，表示如下：

$$f(S1) = Sj \quad f(S7) = Sg$$
$$f(S2) = Sa \quad f(S8) = Sl$$
$$f(S3) = Sb \quad f(S9) = Sd$$
$$f(S4) = Sn \quad f(S10) = Sf$$
$$f(S5) = Si \quad f(S11) = Sh$$
$$f(S6) = Sk$$

另一个例子，回到图11-8及相关表所示的目标识别问题。匹配范例采用两点间的距离关系，每一对孔通过它们之间的距离关联起来。对于旋转和平移来说，距离是不变量，但对于
351 缩放来说距离是变化的。我们用$l2(A, B)$和$l2(B, C)$表示模型中点A和点B、点B和点C之间相距12。但是$l2(C, D)$与距离表中的情况不一致，允许出现失真和检测误差的化，我们认为$l2(C, D)$是有效的，尽管实际上C和D之间相距为$12 \pm \Delta$，Δ表示微小偏移量。

习题11.13　一致性标记问题

证明上面给出的标记f是一致性标记。因为是对称关系，必定满足如下改进的约束条件：

$$若(p_i, p_{i'}) \in R_P, \quad 则(f(p_i), f(p_{i'})) \in R_L \quad 或 \quad (f(p_{i'}), f(p_i)) \in R_L$$

11.6.1 解释树

定义88　**解释树**（interpretation tree, IT）是一种树状结构，表示对部件的所有可能的标记分配。解释树上的每条通路遇到终止时，要么是完全一致分配，要么是关系失败的部分分配。

图11-19所显示的，是图11-8中图像数据的解释树的一部分。树有3层，为图像中可见的3个孔H_1、H_2、H_3分配标记。第一层上没有出现不一致的情况，因为没有进行检测的距离约束。而在第二层，仅用一个距离进行检测就使大多数标记终止。例如部分分配{(H_1，A)，(H_2，A)} 352
是不一致的，因为0(A，A) 与关系21(H_1，H_2) 不一致。由于空间关系在图11-19中只显示出少量通路。其中方框表示的标记通路是一个完全一致分配，椭圆表示的通路也是一致的，但因为包含一个NIL标记而无法使用更多的检测约束条件。该分配与方框表示的完全分配中的前两对标记正好互相颠倒，只有一个检测距离是一致的。由于存在对称性，解释树具有多条成功的通路。尽管解释树包含的通路数可能是指数级的，但由于存在关系约束，大多数通路将终止于第3层。使用NIL标记主要是为了检测场景中的人为特征和其他目标的特征。

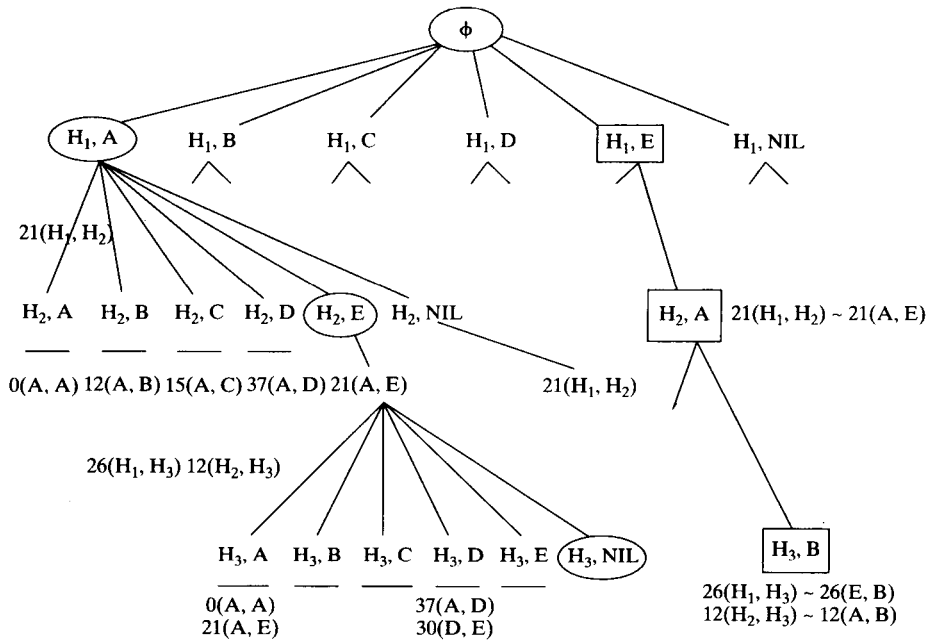


图11-19 针对图11-8（右）中的部件，搜索其一致性标记的解释树

利用递归回溯方法很容易设计出解释树，其中以深度优先的方式生成通路。对于任意程序实例，用NIL初始化的参数f包含一致性部分分配。一旦某个部件的新标记与这个部分分配一致，算法就进入解释树的下一层，并为未标记部件假定另一个标记。如果检测到不一致情况，算法就回退进行另一个选择。如编码所示，算法返回第一条完全通路，如果那个标记明显属于L的话，通路中就包含NIL标记。对算法进行改进，可以返回最不可能有NIL对的完全通路，或者返回所有完全通路。 353

递归解释树搜索算法的定义具有一般性，可以处理任意N元关系R_P和R_L，并非只限于二元关系。R_P和R_L可以是单独一个关系，比如第一个例子中的连接关系；也可以是不同关系的组合，如连接、平行和距离关系。

算法11.5 通过解释树搜索寻找从模型特征到图像特征满足模型关系的映射

P是检测到的图像特征的集合。

L是存储的模型特征的集合。

```
RP是图像特征关系。
RL是模型特征关系。
f是要返回的一致性标记，初始化为NIL。
    procedure Interpretation_Tree_Search(P, L, RP, RL, f);
    {
    p := first(P);
    for each l in L
      {
      f' = f ∪ {(p, l)}; \\添加部分标记到解释
      OK = true;
      for each N-tuple (p1, ⋯ , pN)in RP containing component p
          and whose other components are all in domain(f)
        \\检测关系
        if (f' (p1), ⋯, f'(pN)is not in RL then
          {
          OK: = false;
          break;
          }
       if OK then
        {
        P' = rest(P);
        if isempty(P') then output(f');
        else Interpretation_Tree_Search(P', L, RP, RL, f');
        }
      }
     }
```

11.6.2 离散松弛

354 松弛法只使用局部约束，而不是使用所有可能的约束，如解释树一条通路上的所有匹配约束。经过N次迭代，关于一个部件邻域的局部约束，可以在通路上穿过目标传播到相距N条边的另一个部件。尽管在一次迭代中使用的约束，比解释树搜索时用到的那些约束强度要弱，但这些约束可以并行使用，从而可以加快和简化处理过程。

开始时，只要类型正确可用任何标记来对一个部件进行标记，假设为该部件分配了所有可能的一组标记。离散松弛要检验特定部件与所有其他部件间的关系，通过这样来减少特定部件的可能标记。在字符识别问题中，如果知道下一个字母不是“U”，那么就可以推断当前字母不是“Q”。另一个应用领域中，如果知道某个图像区域不是水域，那么其中的物体也就不会是轮船。离散松弛法是David Waltz推出的，他使用离散松弛法来约束为线条图边缘分配的标记。（Winston的著作中讨论了Waltz滤波。）Waltz使用的是串行算法，这里我们提出一个并行算法。

一开始根据部件P_i的类型，分配所有的标记L_j的集合给每个部件，然后检验所有的关系，

看看是否有的标记是不可能的，把不一致标记从集合中去掉。对每个部件的标记集进行并行过滤处理。如果有标记从集合中被过滤出来，那么就执行下一个过滤过程。如果没有标记发生变化，那么过滤就完成了。结果也许没有留下可能的解释，也许有好几个解释。接下来的例子具有指导意义。为了简化问题，假设没有检测到不属于模型的额外特征。和前面一样，假设某些特征可能被遗漏了。

现在匹配表11-1和表11-2中的数据。过滤过程开始时，对3个孔H_1、H_2、H_3的每一个可能的所有5个标记进行处理。为了更加有趣、更加实用，*允许距离匹配中有±1的公差*。表11-4显示的是，第一次过滤后中间结果的3个标记集合。表格中的每一项都给出了删除或保留标记的原因。从H_1的标记集中删除A，因为没有H_3的标记能解释关系$26(H_1, H_3)$。H_2的标记集中保留A，因为有标记$E \in L(H_1)$能解释关系$21(H_2, H_1)$，标记$B \in L(H_3)$能解释关系$12(H_2, H_3)$。H_2的标记集中保留C，因为有$d(H_2, H_1) = 21 \approx 22 = d(C, D)$。 355

在第一次过滤结尾，如表11-5所示，H_2只有两个可能的标记，H_1和H_3各一个，分别为E和B。在第一次过滤结尾去掉的标记集，将在第二次过滤的并行处理中使用，在第二次过滤中用异步并行命令进一步过滤每个标记集。

第二次过滤从$L(H_2)$中删除标记C，因为使用D作为H_1的标记不能解释$21(H_1, H_2)$。第三次过滤之后，附加的过滤不能改变任何标记集，所以过程收敛。在这个例子中，标记集都是单独表示一个分配和一个解释。具体算法参见算法11.6。尽管松弛标记法简单、快速，但与解释树搜索相比，因为约束只能成对使用，松弛标记有时会在解释中带有更多的歧义性。松弛标记法可用作解释树搜索的预处理内容，它能够充分减少树搜索中的分支情况。

表11-4 松弛标记法的第一次过滤中间结果

A	B	C	D	E	
H_1	no N ∋ $d(A, N) = 26$	no N ∋ $d(B, N) = 21$	no N ∋ $d(C, N) = 26$	no N ∋ $d(D, N) = 26$	$21(H_1, H_2)$ $A \in L(H_2)$ $26(H_1, H_3)$ $B \in L(H_3)$
H_2	$21(H_2, H_1)$ $E \in L(H_1)$ $12(H_2, H_3)$ $B \in L(H_3)$	no N ∋ $d(B, N) = 21$	$21(H_2, H_1)$ $D \in L(H_1)$ $12(H_2, H_3)$ $B \in L(H_3)$		
H_3	no N ∋ $d(A, N) = 26$	$12(H_3, H_2)$ $A \in L(H_2)$ $26(H_3, H_1)$ $E \in L(H_1)$			

表11-5 松弛标记法第一次过滤完成

	A	B	C	D	E
H_1	no	no	no	no	possible
H_2	possible	no	possible	no	no
H_3	no	possible	no	no	no

表11-6 松弛标记法第二次过滤完成

	A	B	C	D	E
H_1	no	no	no	no	possible
H_2	possible	no	no	no	no
H_3	no	possible	no	no	no

表11-7 松弛标记法第三次过滤完成

	A	B	C	D	E
H_1	no	no	no	no	possible
H_2	possible	no	no	no	no
H_3	no	possible	no	no	no

算法11.6 离散松弛标记法：对检测到的图像特征，从可能的标记中去掉不兼容标记

P_i, i=1, …, D是检测到的图像特征集合。

$S(P_i)$, i=1, …, D是最初的兼容标记集合。

R是确定兼容性的一个关系。

```
procedure Relaxation_Labeling(P,S,R);
{
repeat
  for each (P_i, S(P_i))
  {
    for each label L_k ∈ S(P_i)
      for each relation R (P_i, P_j) over the image parts
        if ∃ L_m ∈ S(P_j) with R(L_k, L_m) in model
          then keep L_k in S(P_i)
          else delete L_k from S(P_i)
  }
until no change in any set S(P_i)
return(S);
}
```

习题11.14

给出表11-5所示的通过第一次过滤后，在每个标记集中删除或保留每个标记的详细理由。

11.6.3 连续松弛*

在严格一致性标记过程中，如树搜索和离散松弛，部件p的标记l在任何处理阶段要么是可能的，要么是不可能的。只要发现一个部件-标记对(p, l)与某个实例对不相容，就认为标记l对部件p的标记是非法的。一个标记要么可能要么不可能，正是这个特性使前面的算法成为离散算法。相对的，我们可以把部件-标记对(p, l)与一个实数结合，实数表示分配标记l给部件p的概率或可能性，这种算法称为连续算法。本节我们讨论二元对称关系的连续松弛标

记算法。

连续松弛标记问题是一个6元组$CLRP = (P, L, R_P, R_L, PR, C)$。和前面一样，$P$是部件集合，$L$是部件的标记集合，$R_P$是部件关系，$R_L$是标记关系。$L_i$是部件$i$的容许标记集合，$L$通常是所有部件$i$的$L_i$的并集。假设$|P| = n$，$PR$是$n$个函数的集合$PR = \{pr_1, \cdots, pr_n\}$，其中$pr_i(l)$是标记$l$对部件$i$有效的先验概率。C是含$n^2$个相容系数的集合$C = \{c_{ij}\}$，$i = 1, \cdots, n$；$j = 1, \cdots, n$。$c_{ij}$可以看成是部件$j$对部件$i$的标记所施加的影响。因此，如果把约束关系$R_P$看成一个图，那么$c_{ij}$就是部件$i$和部件$j$间连接边的权值。

356 ~ 357

不直接使用R_P和R_L，而是将二者结合构造n^2个函数的集合$R = \{r_{ij}\}$，$i = 1, \cdots, n$；$j = 1, \cdots, n$，其中$r_{ij}(l, l')$表示部件i使用标记l与部分j使用标记l'的相容性。离散情况中，$r_{ij}(l, l')$可以是1，表示$((i, l)(j, l'))$相容；也可以是0，表示这个组合不相容。连续情况中，$r_{ij}(l, l')$可以是0到1间的任意值，表示部件i和j间关系与标记l和l'间关系的相容程度。相容性信息可以从R_P和R_L中得来，R_P和R_L本身可能是简单的二元关系，也可能是属性二元关系。在属性二元关系中，与一对部件（或一对标记）相关联的属性，表示部件对之间具有所需关系的似然性。连续松弛标记问题的解，与一致性标记问题一样，是一个映射$f: P \to L$，这个映射为每个部件分配一个标记。与离散情况不同的是，关于映射f必须满足什么条件没有具体的定义，而f的定义就隐含在程序中，这个程序就称为连续松弛（continuous relaxation）。

离散松弛算法迭代地从部件i的标记集L_i中移除可能的标记，同样连续松弛也迭代更新与每个部件-标记对有关的概率。初始概率由先验概率的函数集PR确定。算法在第0步以初始概率开始，于是对于每个部件i和标记l，我们定义第0步的概率为：

$$pr_i^0(l) = pr_i(l) \tag{11-18}$$

在松弛的第k步迭代，用上一步的集合和相容性信息计算新的概率集合$\{pr_i^k(l)\}$。为了定义$pr_i^k(l)$，我们首先定义$q_i^k(l)$为：

$$q_i^k(l) = \sum_{(j)(i,j)\in R_P} c_{ij} \left[\sum_{l' \in L_j} r_{ij}(l, l') pr_j^k(l') \right] \tag{11-19}$$

函数$q_i^k(l)$表示当前概率对部件i的标记的影响，其中当前概率与受部件i约束的其他部件的标记有关。那么更新pr_i^k的计算公式可以写成：

$$pr_i^{k+1}(l) = \frac{pr_i^k(l)\left[1 + q_i^k(l)\right]}{\sum_{l' \in L_i} pr_i^k(l')\left(1 + q_i^k(l')\right)} \tag{11-20}$$

上式的分子可以写成当前概率$pr_i^k(l)$与$pr_i^k(l)q_i^k(l)$的和，后一项是当前概率与其他相关部件影响的乘积，其他相关部件的影响以自身标记的当前概率为基础。分母是分子项对部件i的所有标记求和，起规范化作用。

358

习题11.15 连续松弛

图11-20显示由线段组成的模型和图像。当两线段终点重合或彼此紧邻时，认为二者有*closadj*关系。（a）构造模型部件的属性关系$R_P = \{(p_i, p_j, d)|p_i\ closadj\ p_j\}$，以及图像标记的属性关系$R_L = \{(l_i, l_j)|l_i\ closadj\ l_j\}$。（b）如果$(p_i, p_j) \in R_P$，定义相容系数$c_{ij} = 1$；否则$c_{ij} = 0$。你自己选定一种方式，结合$R_P$和$R_L$定义$R$。如果$p_i$与$l_j$互相平行，设$pr_i(l_j)$为1；如果二者互相垂直，设$pr_i(l_j)$为0；如果一条为斜线，另一条为水平线或者竖直线，则设$pr_i(l_j)$为0.5。为模

型部件和图像标记定义pr。(c) 执行几次连续松弛迭代，寻找一个从模型部件到图像标记的可能标注方法。

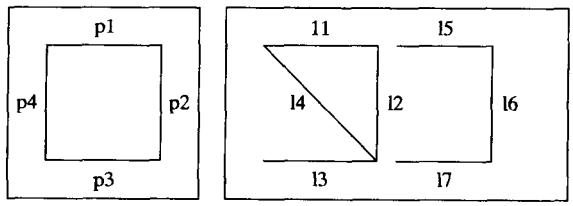


图11-20 连续松弛习题的模型和图像

11.6.4 相关距离匹配

在很多实际应用中，完全一致性标记是不现实的。由于特征抽取误差、噪声干扰和遮挡现象，图像会丢失或者增加部分内容，不能保持应有的关系，这时可以使用连续松弛法，但连续松弛法不能保证找到最优解。在问题中，如果寻找最优解非常重要，那么可以通过搜索找到从P到L的最佳映射f，也就是保留最多的关系，或让NIL标记的数量最少化。最早由Haralick和Shapiro(1981)定义的*相关距离*（relational distance）概念，允许我们在维数不同、关系数量任意的一般情况下定义最佳映射。在这之前首先要对图像或者目标的*相关描述*（relational description）进行定义。

定义89 相关描述D_P是一个关系序列$D_X = \{R_1, \cdots, R_I\}$，其中对于每个$i = 1, \cdots, I$，都存在一个正整数$n_i$，使得对于集合$P$有 $R_i \subseteq P^{ni}$。P是要描述的实体部件的集合，关系R_i指明部件间的各种关系。

相关描述是一种数据结构，可以描述二维形状模型、三维目标模型和图像中的区域等等。

设$D_A = \{R_1, \cdots, R_I\}$是部件集$A$的相关描述，$D_B = \{S_1, \cdots, S_I\}$是部件集$B$的相关描述。假设$|A| = |B|$，如果不相等，就在较小的集合中添加虚构部件使等式成立。这个假设是为了保证
359 相关距离是标准测度。

设f是由A到B的任意一一映射。对任意 $R \subseteq A^N$，其中N是一个正整数，关系R和函数f的合成运算（composition）$R \circ f$如下:

$$R \circ f = \{(b_1, \cdots, b_N) \in B^N \mid \text{存在} \quad (a_1, \cdots, a_N) \in R \text{ 及 } f(a_n) = b_n, n = 1, \cdots, N\} \tag{11-21}$$

合成算子把R的N个组元一一映射到B^N的N个组元。

函数f把集合A中的部件映射成集合B中的部件。f关于D_A和D_B的第i对对应关系（R_i和S_i）的*结构误差*（structural error）如下:

$$E_S^i(f) = |R_i \circ f - S_i| + |S_i \circ f^{-1} - R_i| \tag{11-22}$$

结构误差表示，R_i中有多少组元不能用f映射到S_i中，以及S_i中有多少组元不能用f^{-1}映射到R_i中。结构误差的表达式中只考虑了一一对应关系。

f关于D_A和D_B的*总误差*（total error），是每对对应关系结构误差的和，也就是

$$E(f) = \sum_{i=1}^{I} E_S^i(f) \tag{11-23}$$

总误差定量给出了两相关描述D_A和D_B间关于映射f的差异。

这样，D_A和D_B间的相关距离$GD(D_A, D_B)$由下式给出:

$$GD(D_A, D_B) = \min_{\substack{1\text{-}1 \\ f:A \to B \\ \text{onto}}} E(f) \tag{11-24}$$

也就是说，相关距离是从A经f一一映射到B的最小总误差。使总误差最小的映射f称为从D_A到

D_B的最佳映射。如果有多于一个的最佳映射，可以用纯关系范例之外的附加信息来选择最好的映射。当相关描述包含某些特定的对称性时，就会出现多于一个的最佳映射。

举几个例子来说明相关距离。图11-21显示2个有向图，每个有向图都有4个节点。从$A = \{1, 2, 3, 4\}$到$B = \{a, b, c, d\}$的一个最佳映射是$\{f(1) = a, f(2) = b, f(3) = c, f(4) = d\}$。关于这个映射我们有

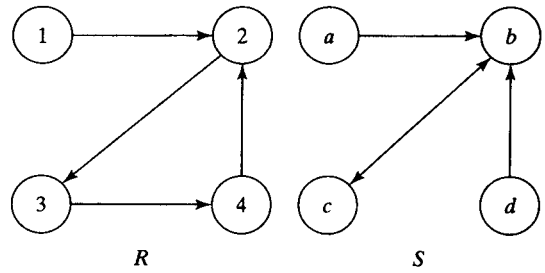


图11-21 两个相关距离为3的有向图

360

$$
\begin{aligned}
|R \circ f - S| &= |\{(1,2)(2,3)(3,4)(4,2)\} \circ f - \{(a,b)(b,c)(c,b)(d,b)\}| \\
&= |\{(a,b)(b,c)(c,d)(d,b)\} - \{(a,b)(b,c)(c,b)(d,b)\}| \\
&= |\{(c,d)\}| \\
&= 1 \\
|S \circ f^{-1} - R| &= |\{(a,b)(b,c)(c,b)(d,b)\} \circ f^{-1} - \{(1,2)(2,3)(3,4)(4,2)\} \\
&= |\{(1,2)(2,3)(3,2)(4,2)\} - \{(1,2)(2,3)(3,4)(4,2)\}| \\
&= |\{(3,2)\}| \\
&= 1 \\
E(f) &= |R \circ f - S| + |S \circ f^{-1} - R| \\
&= 1 + 1 \\
&= 2
\end{aligned}
$$

因为f是最佳映射，所以相关距离也是2。

图11-22给出一套目标模型M_1、M_2、M_3和M_4，它们的基本部件是图像区域。图中有两个关系：连接和平行。基本部件之间都是二元关系。考虑前两个模型M_1和M_2。最佳映射f映射基本部件1到1′、2到2′和3到3′。在该映射下，连接关系是同构关系。模型M_1中的平行关系（2，3），在模型M_2中的2′和3′之间不再保持平行关系，所以M_1和M_2之间的相关距离刚好是1。现在考虑模型M_1和M_3。最佳映射映射1到1″、2到2″和3到3″、虚拟基本部件到4″。在该映射下，平行关系是同构关系，但M_3比M_2中多出一个连接关系。相关距离也是1。

最后考虑模型M_3和M_4。最佳映射

361

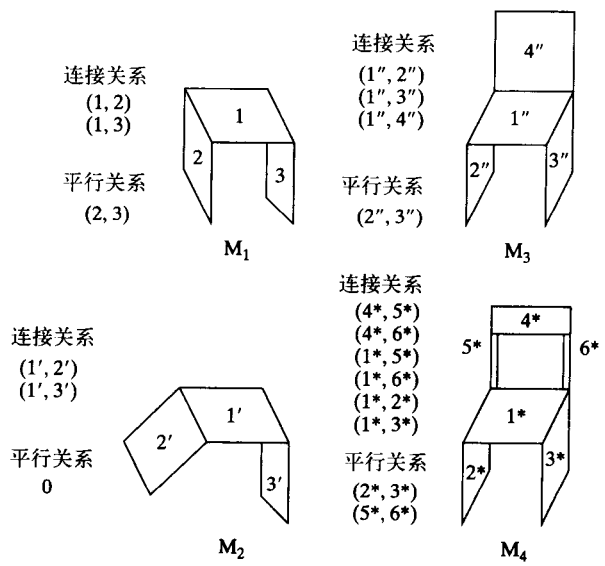


图11-22 四个目标模型。M_1到M_2以及M_1到M_3的相关距离是1。M_3到M_4的相关距离是6

映射1″到1*、2″到2*、3″到3*、4″到4*，5_d到5*和6_d到6*。（5_d和6_d是虚拟基本部件。）关于这个映射我们有

$$\begin{aligned}
|R_1 \circ f - S_1| &= |\{(1'', 2'')(1'', 3'')(1'', 4'')\} \circ f \\
&\quad - \{(4^*, 5^*)(4^*, 6^*)(1^*, 5^*)(1^*, 6^*)(1^*, 2^*)(1^*, 3^*)\}| \\
&= |\{(1^*, 2^*)(1^*, 3^*)(1^*, 4^*)\} \\
&\quad - \{(4^*, 5^*)(4^*, 6^*)(1^*, 5^*)(1^*, 6^*)(1^*, 2^*)(1^*, 3^*)\}| \\
&= |\{(1^*, 4^*)\}| \\
&= 1
\end{aligned}$$

$$\begin{aligned}
|S_1 \circ f^{-1} - R_1| &= |\{(4^*, 5^*)(4^*, 6^*)(1^*, 5^*)(1^*, 6^*)(1^*, 2^*)(1^*, 3^*)\} \circ f^{-1} \\
&\quad - \{(1'', 2'')(1'', 3'')(1'', 4'')\}| \\
&= |\{(4'', 5_d)(4'', 6_d)(1'', 5_d)(1'', 6_d)(1'', 2'')(1'', 3'')\} \\
&\quad - \{(1'', 2'')(1'', 3'')(1'', 4'')\}| \\
&= |\{(4'', 5_d)(4'', 6_d)(1'', 5_d)(1'', 6_d)\}| \\
&= 4 \\
|R_2 \circ f - S_2| &= |\{(2'', 3'')\} \circ f - \{2^*, 3^*)(5^*, 6^*)\}| \\
&= |\{(2^*, 3^*)\} - \{(2^*, 3^*)(5^*, 6^*)\}| \\
&= |\emptyset| \\
&= 0 \\
|S_2 \circ f^{-1} - R_2| &= |\{(2^*, 3^*)(5^*, 6^*)\} \circ f^{-1} - \{(2'', 3'')\}| \\
&= |\{(2'', 3'')(5_d, 6_d)\} - \{(2'', 3'')\}| \\
&= |\{(5_d, 6_d)\}| \\
&= 1 \\
E_S^1(f) &= 1 + 4 = 5 \\
E_S^2(f) &= 0 + 1 = 1 \\
E(f) &= 6
\end{aligned}$$

362

习题11.16 相关距离树搜索

修改解释树搜索算法，寻找两个结构描述间的相关距离，并确定最佳映射。

习题11.17 单向相关距离

公式（11-24）定义的相关距离，使用了双向映射误差，这在比较两个孤立目标时很有效。当将模型与图像进行匹配时，希望只用单向映射误差，检验图像中有多少模型关系，而不做反向的工作。为了进行模型-图像匹配，请重新定义单向相关距离。

习题11.18 相关距离中的NIL映射

公式（11-24）定义的相关距离，没有明确处理NIL标记。如果部件f有一个NIL标记，那

么任何关系 (i, j) 都会引起错误，因为 $(f(i), \text{NIL})$ 不会出现。修改相关距离的定义，把NIL标记作为错误只计数一次，并且不再因NIL标记引起的关系丢失而进行惩罚。

习题11.19 属性相关距离

公式（11-24）定义的相关距离，没有明确处理属性关系。在属性关系中，除部件序列外，每个组元还包含一个或多个关系属性。如线段的连接关系也许还有连接线段间夹角属性。形式上，部件集P和属性集A上的一个属性n元关系R是一个集合，$R \subset P_n \times A_m$，其中m是非负整数，表示关系的属性数。依据属性关系，请修改相关距离的定义。

11.6.5 相关索引

有时即使采用松弛过滤法，树搜索也显得太慢，尤其是当比较图像与大型模型数据库时。对于用标记关系进行的结构描述，可以用一个更简单的表决方案近似相关距离。直观上，假设观察到两个同心圆和具有一条公共边的两个90°直角。希望快速找到具有这些结构的所有模型，并希望在更多细节上能与这些模型相匹配。为了做到这一点，我们可以建立一个索引，通过索引查找具有分图结构的模型。首先查找包含两个同心圆特征的所有模型，并且给每个模型投上一票，然后查找包含两相连90°角的所有模型，凡是两次都被查到的模型将得到两票。如果在识别前，从每个模型中抽取重要的二元关系，并把这些关系记录在查找表中，离线建立一个索引，那么这些查找就可以快速完成。

设$DB = \{M_1, M_2, \cdots, M_T\}$是含T个目标模型的数据库，每个目标模型$M_i$由特征部件$P_i$以及标记关系$R_i$集合而成。为了解释起来更加简单，假设每个部件只有一个标记，而不是属性向量；假设关系都是二元关系，同样每个组元都只有一个标记。在这种情况下，模型可以用2-图（2-graph）的集合来表示。每个2-图有两个节点和两条有向边线组成。每个节点代表一个部件，每条边线代表一个有向二元关系。节点的值是部件的标记，而不只是唯一的标示符。同样边线的值是关系的标记。比如一个节点可能代表椭圆而另一个则可能代表一对平行线。从平行线节点到椭圆节点的连线表示关系“包含”，相反方向上的连线表示关系“被包含”。 363

相关索引在预处理阶段建立大型散列表。用2-图字符串为散列表建立索引。完成之后可以查寻表中的任何2-图，并快速检索包含特殊2-图的所有模型的列表清单。在我们的实例中，可以检索到两平行线间有一椭圆的所有模型。在根据图像识别目标时，要抽取特征和计算所有表示图像的2-图。数据库中的每个模型都有一个累加器，一开始都清零。然后用图像中的2-图搜索散列表，检索出关系模型的列表清单，并对每一个模型投票。离散算法的每次投票就是加一操作，概率算法则是加上一个概率值。在对所有2-图进行投票操作后，得票最多的模型就是验证候选模型。

11.7 非线性变形

非线性变形函数也是很重要的。有时我们要对图像中的非线性畸变进行矫正，如鱼眼镜头的径向畸变。有时我们又希望对图像进行艺术性变形处理。图11-23显示一种非线性变形，它把一个规则网格映射到一个圆柱上，其效果等同于把一幅平面图像卷到一个圆柱上，并在远处进行观察。图11-24是把同样的

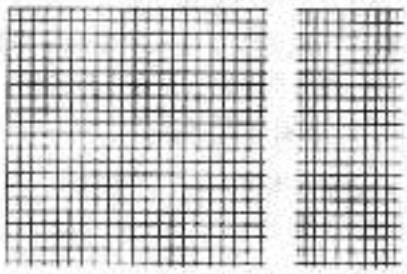


图 11-23

（左）规则网格

（右）卷在圆柱体上的变形网格

变形应用到一张20美元钞票上。直观上需要选择与圆柱中心对应的某个图像轴，然后通过公式计算，在输入图像的基础上产生一幅变了形的输出图像，效果就像是卷在一个圆柱上。图11-24显示出两个变形结果。最右边的图像变形采用的圆柱半径要比中间变形采用的半径小。



图 11-24
(左) 20美元钞票的中间部分图像
(中) 安德鲁·杰克逊的头像，卷在周长640像素的圆柱上
(右) 只中图，只是圆柱周长为400像素

图11-25显示的是如何推导一个圆柱变形。为变形选定一个轴（由x_0决定）和一个宽度W。W对应圆柱体周长的1/4。输入图像长度为d的部分卷在圆柱上，然后投影到输出图像。事实上，d对应长度$x - x_0$，其中x_0是圆柱轴的x坐标。变形不改变输入图像点的y坐标，所以有$v = y$。由图可得到下列关系式。首先$W = (\pi/2)r$，即W等于1/4的圆柱周长。d与W的关系是$d/W = \phi/(\pi/2)$，而$\sin\phi = d'/r$，由上述几个等式可得$d = x - x_0 = (2W/\pi)\arcsin((\pi/2W)(u - u_0))$。当然，$d' = u - u_0 = u - x_0$。

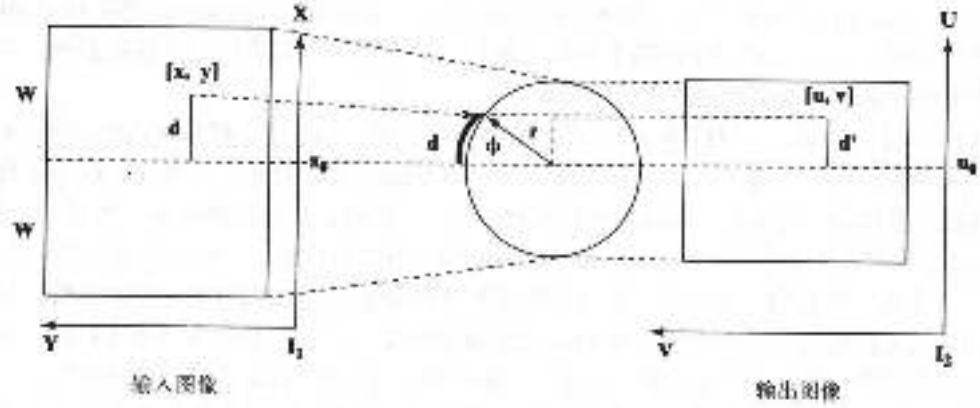


图11-25 把左侧的输入图像卷到中间圆柱上就产生了右侧的输出图像。
输入图像中的距离d在输出中图像中变为d'

已知输出图像的坐标$[u, v]$，变形参数x_0和W，可以用公式计算输入图像的坐标$[x, y]$。这看起来是逆向的，为什么不从输入图像变换到输出图像呢？无论是否这样做，都不能保证输出图像的每个像素都是唯一确定的。对于数字图像，我们希望输出图像的每个像素都只计算一次，而且其像素值是根据输入图像算出的，如算法11.7所示。另外这样很容易使输出图像的像素数多出或者少于输入图像的像素数。解决方法就是在生成输出图像时，向输入图像进行逆映射，然后再对输入图像进行采样。

算法11.7 圆柱变形运算

^{1}I[x, y]是输入图像，
x_0是轴线位置，
W是宽度，
^{2}I[u, v]是输出图像，

procedure Cylindrical_Warp(^{1}I[x, y])

```
{
r = 2W/π;
for u := 0, Nrows−1
  for v := 0, Ncols−1
  {
    ²I[u, v] = 0; \\背景
    if (|u−u0| ≤ r)
     {
       x = x0 + r arcsin((u− x0)/r);
       y = v;
       ²I[u, v] = ¹I[round(x), round(y)];
     }
  }
return (²I[u, v]);
}
```

习题11.20

（a）求一个变换，把输入图像的圆形域映射到半球上，然后再把这个半球投影成一幅图像。原图像中的圆形域通过中心（x_c，y_c）和半径r_0确定。（b）编写计算机程序实现这个映射。

11.7.1 径向畸变矫正

多数镜头都存在径向畸变，对人类感官来说影响不大，但进行光度测量时如果不矫正的话会产生很大误差。物理学上已经推出，图像点的径向畸变与该点到光轴的距离成正比。图11-26显示两种常见的畸变情况，以及矫正后图像。如果光轴接近图像中心穿过，那么对图像各点进 366
行平移就能够实现矫正。位移的大小与像素到中心的距离平方成正比。这个矫正不是一个线性变换，因为图像上各点的位移量是不同的。有时用径向距离的更高偶次幂进行矫正，如公式（11-25）的数学模型所示。设$[x_c, y_c]$为图像中心，光轴经$[x_c, y_c]$穿过图像。假设只用径向距离的前两项偶次幂计算径向畸变，则对图像点的矫正计算如下。其中常数c_2和c_4的最佳取值，可以通过分析已知控制点的径向位移得到，也可以在标定过程中通过最小二乘拟合得到。

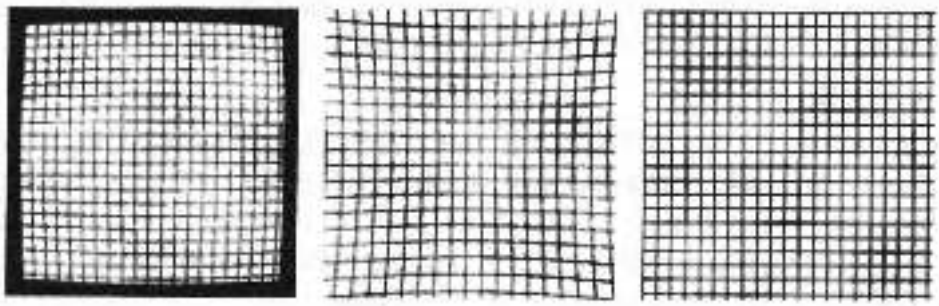


图11-26 两类径向畸变：左边是桶形畸变，中间是枕形畸变，通过变形运算可以对两种畸变进行矫正并生成右边的图像

$$
\begin{aligned}
R &= \sqrt{(x-x_c)^2+(y-y_c)^2)} \\
D_r &= (c_2R^2+c_4R^4) \\
x &= x_c+(x-x_c)D_r \\
y &= y_c+(y-y_c)D_r
\end{aligned}
\tag{11-25}
$$

11.7.2 多项式映射

不严重的全局性失真可以利用多项式映射的方法进行矫正，如公式（11-26）所示的二元二次多项式就足够了。为适应不同的几何因素，要对12个系数进行估计。为了估计这些系数，至少需要6个控制点在映射前后的坐标。实际操作时要采用更多的控制点。每个控制点生成两个方程。如果只取公式（11-26）的前三项，则这个映射就是仿射映射。

$$
\begin{aligned}
u &= a_{00}+a_{10}x+a_{01}y+a_{11}xy+a_{20}x^2+a_{02}y^2 \\
v &= b_{00}+b_{10}x+b_{01}y+b_{11}xy+b_{20}x^2+b_{02}y^2
\end{aligned}
\tag{11-26}
$$

习题11.21

367 如果公式（11-25）中的$c_4=0$，证明径向畸变模型可通过公式（11-26）的多项式映射得到。

11.8 总结

在2D匹配的主题下，本章讨论了很多概念。一个主题内容是通过变换进行2D映射。变换是比较简单的图像处理运算，可用于从图像中抽取一个区域，在同一坐标系中对两幅图像进行配准，去除2D图像中的失真或者使2D图像发生形变。人们开发出了进行这些变换的代数运算工具，对各种方法和应用情况进行了讨论。在第13章中对这些内容进一步扩展，研究3D场景和3D模型中点的映射关系。本章的第二个主题内容是，通过与2D模型进行对应，从而对2D图像进行解释。比对识别（recognition-by-alignment）是一般的范例。通过找到一个模型和一个RST变换来解释图像，其中的RST变换把已知的模型结构映射到图像结构。文中给出了几种不同的算法，包括位姿聚类算法、解释树搜索算法和局部特征焦点算法。也给出了离散松弛算法和相关匹配算法。尽管这两种方法是在有约束的几何关系下引出的，而实际上它们可用于一般情况。当拓扑关系本身比度量关系更稳健时，相关匹配就应该比刚性对比更稳健。由镜头失真、视轴倾斜和量化效应等引起的图像失真，会导致度量关系失效。而拓扑关系如端点相同、连接、相邻以及包含等，通常不受这类失真的影响。基于图像或模型零件上的拓扑关系成功匹配，可用于大量的匹配点，然后再用这些匹配点建立一个多参数映射函数，以补偿度量上的失真。实际应用中可直接采用本章介绍的计算方法。第14章中将把这些方法扩展到3D情况。

11.9 参考文献

Van Wie和Stein（1977）所讨论的系统，能够自动将卫星图像与地图进行配准。拍摄图像的时刻近似映射是已知的。该映射通过模板对控制点进行精确搜索，然后再利用搜索到的控制点对映射关系进行修改。Wolberg（1990）的著作全面介绍了图像变形，包括对输入图像采样和通过平滑来减轻混叠失真的详细讨论。基于位姿聚类的2D匹配参考的是Stockman等人（1982）发表的论文，其中包括飞机检测的例子。Stockman（1987）针对3D情况进行了更一般的讨论。Grimson和Lozano-Perez（1984）论述了如何把距离约束用于模型点与观测数据点之间的匹配。最小二乘拟合在其他文献中已经得到了深入讨论，这是一个很有深度的话题。噪声情况下估计变换参数，常常采用最小二乘技术。这时采用的控制点个数要比最小控制点个数多

得多。Wolberg（1990）的著作中，讨论了几种最小二乘变形方法，而Daniel和Wood（1971）的著作主要讨论一般的最小二乘拟合问题。有时找不到适用于整幅图像的有效几何变换，这时 368
可以把图像分为几个区域，每个区域都有自己的控制点，对每个区域进行变形处理。相邻区域的变形必须沿边界平滑地进行过渡。关于这种方法Gostasby（1988）提出一种灵活处理方式。

一致性标记问题的理论和算法可以参考Haralick和Shapiro（1979、1980）发表的论文。Rosenfeld、Hummel和Zucker（1976）定义了离散松弛和连续松弛的概念。Hummel和Zucker（1983）进一步分析了连续松弛。结构匹配法参考Shapiro和Haralick（1981）发表的论文。基于2-图的相关索引请参考Costa和Shapiro（1995）的论文。把结构零件的不变属性用做模型索引的内容请参考Chen和Stockman（1996）发表的论文。

1. Chen, J. L., and G. Stockman. 1996. Indexing to 3D model aspects using 2D contour features. *Proc. Int. Conf. Comput. Vision and Pattern Recog. (CVPR),* San Francisco, CA (June 18–20), expanded paper to appear in the journal *CVIU.*
2. Clowes, M. 1971. On seeing things. *Artifical Intelligence,* v. 2:79–116.
3. Costa, M. S., and L. G. Shapiro. 1995. Scene analysis using appearance-based models and relational indexing. *IEEE Symposium on Comput. Vision* (Nov. 1995), 103–108.
4. Daniel, C., and F. Wood. 1971. *Fitting Equations to Data.* John Wiley & Sons, Inc., New York.
5. Goshtasby, A. 1988. Image registration by local approximation methods. *Image and Vision Computing,* v. 6(4):255–261.
6. Grimson, W., and T. Lozano-Perez. 1984. Model-based recognition and localization from sparse range or tactile data. *Int. J. Robotics Research,* v. 3(3):3–35.
7. Haralick, R, and L. Shapiro. 1979. The consistent labeling problem I. *IEEE Trans.,* v. PAMI-1:173–184.
8. Haralick, R., and L. Shapiro. 1980. The consistent labeling problem II. *IEEE Trans.,* v. PAMI-2:193–203.
9. Hummel, R., and S. Zucker. 1983. On the foundations of relaxation labeling processes. *IEEE Trans.,* v. PAMI-5:267–287.
10. Lamden, Y., and H. Wolfson. 1988. Geometric hashing: a general and efficient model-based recognition scheme. *Proc. 2nd Int. Conf. Comput. Vision,* Tarpon Springs, FL (Nov. 1988), 238–249.
11. Rogers, D., and J. Adams. 1990. *Mathematical Elements for Computer Graphics,* 2nd ed. McGraw-Hill, New York.
12. Rosenfeld, A., R. Hummel, and S. Zucker. 1976. Scene labeling by relaxation operators. *IEEE Trans. Systems, Man, and Cybern.,* v. SMC-6:420–453.
13. Shapiro, L. G., and R. M. Haralick. 1981. Structural descriptions and inexact matching. *IEEE Trans. Pattern Recog. and Machine Intelligence,* v. PAMI-3(5):504–519.
14. Stockman, G., S. Kopstein, and S. Benett. 1982. Matching images to models for registration and object detection via clustering. *IEEE Trans. PAMI,* v. PAMI-4(3): 229–241. 369
15. Stockman, G. 1987. Object recognition and localization via pose clustering. *Comput. Vision, Graphics and Image Proc.,* v. 40:361–387.
16. Van Wie, P., and M. Stein. 1977. A LANDSAT digital image rectification system. *IEEE Trans. Geosci. Electron.,* v. GE-15 (July 1977).
17. Winston, P. 1977. *Artificial Intelligence.* Addison-Wesley.
18. Wolberg, G. 1990. *Digital Image Warping.* IEEE Computer Society Press, Los Alamitos, CA. 370

第12章　2D图像中的3D信息

本章研究2D图像中出现的一些现象，这些现象揭示了图像中隐含的3D结构特征。人类能够根据视觉输入感知和分析3D世界的结构信息。人类的这种能力非常神奇，运用起来毫不费力，但人类关于自己的视觉感知机制仍然知之甚少。首先我们要强调以下三点：第一，虽然这里的讨论要用到推理分析，但人类能够很容易感知出结构信息而不需要有意识地进行推理。关于人类视觉的很多方面我们理解得还不是很清楚；第二，尽管我们能够建立几种视觉线索的模型，但对复杂场景的解释需要同时使用多个线索，这是一个竞争和协作的过程；第三，我们的兴趣点不是为了解释人类的视觉行为，而是为了解决有限范围内的应用问题。这有限的范围允许我们利用简单的一组线索进行研究。

本章首先对用到的方法做简单说明。下一节讨论本征图像（intrinsic image），这是一种中间的2D表示，存储了3D场景重要的局部特征。然后研究纹理特征、运动特征以及形状特征，这些特征使我们能够从2D图像推断出场景的3D特征。本章重点讨论对原始信息的识别问题，而不把建立数学模型做为讨论的重点，但在本章的最后要介绍一下数学建模。这些模型可用于透视成像、通过体视计算深度以及通过薄透镜公式描述视场与分辨率和图像模糊的关系。其他数学建模留待第13章介绍。

12.1　本征图像

可以认为3D场景是由目标面元（表面元素）组成的，这些面元受光源照亮，在2D图像中投影为一个区域。3D面元间的边界或者面元上的照明发生变换，都会在2D图像中出现反差边或者是轮廓（contour）。对如图12-1和12-2所示的简单场景，所有的面元及其光照都可以通过场景描述表示出来。有的科学家相信，人类低层视觉系统的主要功能是构造场景的表示，并把它作为进一步处理的基础。这是一个很有趣的问题，但这个问题我们不去管它，而是继续讨论我们关心的问题。我们用这样的表示来描述场景、描述图像和进行机器分析，而不考虑它是否符合人类视觉系统的计算结果。

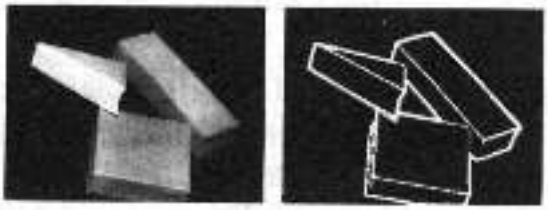


图12-1　（图像由Deborah Trytten提供）
（左）三个积木块的亮度图像
（右）5×5 Prewitt边沿检测结果

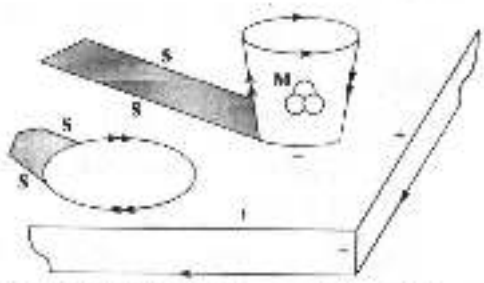


图12-2　带注解标记的2D图像，反映了2D反差与3D现象（如表面方向和光照情况）之间的关系。表面标注用“+”或“−”表示，单箭头“>”表示右边表面形成的刃边，双箭头“>>”表示右边表面的光滑弧边。阴影边界用“S”表示，反射边界用“M”表示

图12-2显示一只鸡蛋和一个空纸杯放在桌子的一角。目前的视点情况是，鸡蛋和杯子都挡住了桌面。区域边缘上的箭头指明哪个面元遮挡了另一个面元，箭头方向指明哪个是遮挡表面。习惯上，如果我们沿边缘前进时遮挡表面位于边缘的右侧，则前进方向就是箭头的方向。单箭头“>”表示刃边（blade），就像刀刃那样。当沿刃边前进时遮挡表面的方向改变不大。当越过边缘时，被遮挡表面的方向与遮挡表面的方向无关。图12-1的右图中，所有的目标边界都是刃边。图12-2中，靠下的桌子边缘形成一个刃边，因为桌边是很窄的平面片，它挡住了未知的背景。纸杯的上部边缘是一个刃边，因为该表面挡住了背景，并且沿边缘前进时表面方
372 向相同。更有趣的是杯子前表面上部的那条边，是一个刃边，因为表面挡住了杯子的内部。

双箭头“>>”表示翼边（limb）。当观察表面光滑的3D目标时就会形成翼边，就像人体四肢那样。在2D图像中沿翼边边界前进时，相应3D面元的方向发生变化，并且方向与视线垂直。表面本身是自遮挡（self-occulding）的，意思是随着3D面元向目标后面移动并逐渐从2D视图中消失，面元方向进行连续平滑的变化。刃边是3D目标的真正边缘，而翼边则不是。图像中，鸡蛋的全部边界都是翼边，杯子也有两条独立的翼边。艺术家们知道，当逆着光线渐渐靠近翼边时，表面会渐渐变暗。通常称刃边和翼边为跳跃边缘（jump edge），即从遮挡表面到后面的被遮挡目标之间有一个不定深度的跳变。在图12-10中可以看到更复杂的场景，其中包含很多与图12-2中类型相同的边缘线条。如灯和灯柱有翼边，左边建筑物的最右侧边缘是刃边。

表面突然变化或者两个表面相连接时就会形成折痕（crease）。图12-2中，在桌边及杯与桌连接处形成折痕。桌边处的表面是向外凸的，用“+”表示；杯与桌相连处的表面是向内凹的，用“-”表示。注意，机器视觉系统从传感器数据开始自底向上进行分析，它并不知道场景中包含有杯子和桌子。人类也不知道杯子是否粘在桌子上，或者干脆杯和桌就是一个整体，但我们的经验倾向于这种自上向下的解释过程！虽然不是经常发生，但折痕常常引起2D图像中的光强或者反差发生明显变化，这是因为一个面常常比另一个面更直接地对着光线。

习题12.1

在面前的桌上放一个杯子，闭上一只眼睛看它。用一支铅笔接触杯子侧面，用铅笔表示表面法线的方向，检验铅笔是否与你的视线垂直。

习题12.2

图12-1中的三角块，在边缘图像中形成六个轮廓线段，这六个线段的标记各是什么？

习题12.3

参考第1章中的图1-7，其中含三个机器零件。（图像中的大部分轮廓线用白色突出表示。）画出所有的轮廓并对它们进行标记。有足够的标记来表示所有的轮廓线段吗？是否用到了我
373 们定义的所有标记？

其他两种图像轮廓不是由3D表面形状引起的。表面反照率的不同会引起表面出现反光痕迹“M”。例如图12-2中杯子上的痕迹，当杯子材料比较亮时痕迹就为深色。光照边界“I”或阴影“S”是由到达表面的光照变化引起的，是由其他目标的阴影形成的。

下面定义的概念用来描述表面结构。有一点要明白，我们所表示的3D场景结构与某个2D视图有相似的视觉效果。这些3D结构常常会在亮度图像中产生可检测的轮廓线。

定义90 折痕是指表面发生突变的地方或者是两个不同表面的交接处，折痕两边，

表面上的点是连续的，但表面法线方向不是连续的。折痕的表面几何可通过视点邻域看出来。当然，要求折痕在该视点下是可见的。

定义91 一个连续表面遮挡住后面的另一个表面，当沿表面边界前进时，表面法线方向的变化是平滑连续的，并与视线方向相对。这时就形成**刃边**。图像中的刃边轮廓是光滑的曲线。

定义92 一个连续表面遮挡住后面的另一个表面，当沿表面轮廓前进时，表面法线方向平滑变化，并且与视线方向垂直。因此表面也会遮挡住它自己，这时就形成**翼边**。边界图像是光滑的曲线。

定义93 **反光痕迹**是由于表面材料的反射变化引起的。例如表面涂了不同颜料或者由不同材料拼接而成时，就会出现反光痕迹。

定义94 由于照明发生变化或者另一目标产生阴影，使表面光照发生突变，就会产生**光照边界**。

定义95 **跳跃边缘**指翼边或者刃边。当越过遮挡目标表面和被遮挡背景表面之间的边缘或轮廓时，深度是不连续的。

习题12.4 标记立方体图像的线段。

按一般方位画一个立方体，显示出3个面、9条线段和7个角。(a) 假设立方体飘浮在空气中，从{+, −, >, >>}中给9条线段各分配一个标记，所分配的标记要对产生线段的3D结构做出恰当的解释。(b) 假设立方体置于平坦桌面上，在此条件下重复 (a) 的过程。(c) 假设立方体实际上是挂在墙上的恒温器，在此条件下重复 (a) 的过程。 374

习题12.5 标记常见物体的图像

标记图12-3中的线段。物体是桌子上带商标X的未开启的苏打水罐，以及打开的空盒子。

第5章讨论了检测亮度图像中反差点的方法。第10章讨论了轮廓跟踪和轮廓表示。不幸的是，几种不同的3D现象会引起2D图像中的相同效果。例如对于亮度图像中的一条2D轮廓线，如何确定它究竟是由实际目标引起的，还是由另一个目标的影子引起的？考虑晴天拍摄的小树林图像。(或者参见第5章后面骆驼在海滩上的图像，骆驼的四肢图像就出现了这种情况。) 对于定义草地上的树影 ("S")，通过边缘检测可能比用树干形成的翼边 (>>) 效果更好。在图像解释中，如何区分阴影与树的图像，或者区分阴影和人行道的图像？

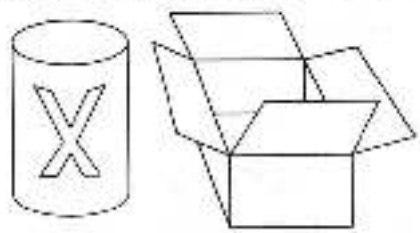


图 12-3

(左) 带商标X的未开启的苏打水罐是封闭柱形桶子，其上有橙色大字母 "X"

(右) 一个空盒子，上面的四边都已开了，所以可以看到部分未被遮挡的盒底

习题12.6

联想第5章学过的内容，解释为什么在图像中检测树干的影子比检测树干本身更容易。

有的研究人员提议开发能生成本征图像的感知系统。本征图像的每个像素应该包含四个本征场景值：

- 场景面元的深度（range，depth）
- 场景面元的方向（orientation，surface normal）
- 场景面元接收的照度（illumination）
- 场景面元的反照率（albedo，surface reflection）

人类擅长于根据图像像素的周围情况对像素进行解释。本征图像的自动构建仍然是一个研究课题，但同以前相比研究热度已经有所下降。很多图像分析任务并不需要本征图像。第13章将介绍构建本征图像或部分本征图像的几种方法。与图12-2对应的本征图像如图12-4所示。图中的数字只说明包含鸡蛋末端的一小条本征图像的信息。沿着桌面深度值逐渐变大，只是在桌子边缘处的变化较快，在遮挡桌面的鸡蛋表面上有一个跳变。桌面的方向或者说桌面的法线方向在上表面都是相同的，而且在桌子的边缘有突变。蛋表面的方向在各点之间平滑变化。反照率数值说明，桌面（值为5）是比蛋面（值为9）更暗的材料。照度值记载桌面像素值的变化，阴影区的像素值为1，而非阴影区的像素值不是1。假设光源来自右上方，朝对光线的鸡蛋表面的像素值为3，显得比那些直接面向光线的部分暗一些，这是因为在单位面积上接收到的光能较少的缘故。

375

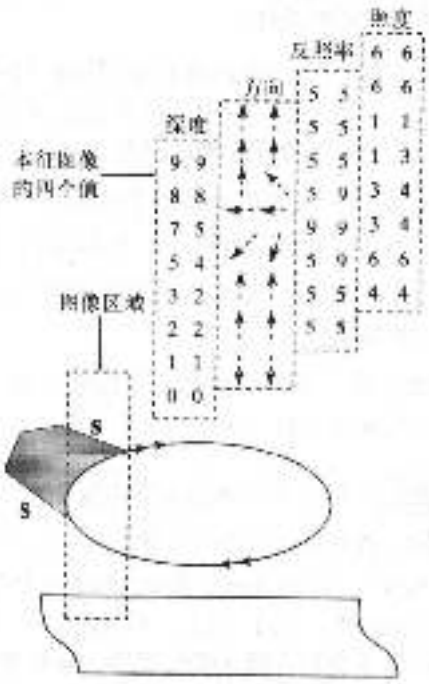


图12-4 与图12-2对应，含部分鸡蛋的一小条本征图像，每个像素包含四个值，分别代表表面深度、方向、照度和反照率。详细说明参见正文

习题12.7 室外场景图像的线段标记

参考第2章在魁北克城拍摄的图片，画出图像中可见的主要轮廓，并用标记集{I/S，M，+，−，>，>>}标记这些轮廓。

376

12.2 线条图标记

图像中的轮廓结构，与3D目标结构有很大关系。本节我们将从微观上来说明这一点，对目标和观察条件要进行一定的限制，**假设所有的3D目标都是三面角目标，即所有的面元都是平面，所有的角都由三个面相交形成**。图12-5中的模块就是这样的目标。用术语面、折痕和角描述3D结构，用术语区域、边和连接描述3D结构的2D图像。假设3D模块的2D图像是包含区域、边连接的线条图。进一步地，假设创建2D图像的微小视点变化，不会引起这幅线条图拓扑结构的变化，也就是说不会有面、边和连接的出现或消失。常常称处于这种情况的目标是处于**常规位置**。

尽管对模块微观世界做出这么多不现实的限定，但已经证明这里介绍的方法在很多实际环境中是有用的。我们将用第11章中的算法进行匹配和解释。模块领域的研究内容具有一定的历史价值，并且支持对新方法的进一步研究。

通过前面的讨论，我们已经知道如何用{+，−，>}标记图像的边缘，根据我们对3D结构的理解，用这些标记表示出折痕或刃边，没有用到翼边，因为模块世界中没有翼边。大约30年前，有人发现形成连接的线段标记组合是强约束的，一共只有16种可能的组合，如图12-6所示，图12-5显示，对应相同2D线条图的两种不同的3D解释，其中的连接类型发生了变化。

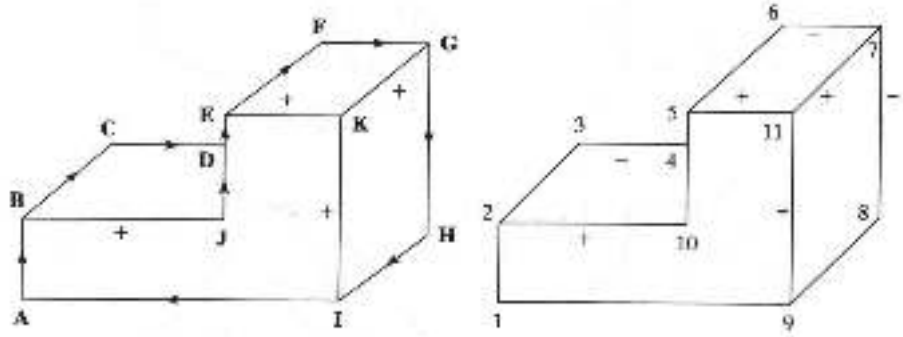


图12-5 同一线条图的两种不同解释。右图中省略的刃边标记与左图相同

（左）模块飘浮在空白中

（右）模块贴在后面的墙上

根据连接边的数目和边间的角度不同，共有四类连接，如图12-6所示，自上到下，分别称为L连接、箭头连接、叉连接和T连接。图12-5是具有四类连接的例子。用J标志的连接是图12-6中顶部最左边的L型连接，用C标志的连接是自顶部右端数起，第二个L连接的例子，G是图12-6第二行最右边的箭头连接，图中只有一个T连接，以D标志。注意，如图12-6所示，T连接的遮挡边（横边）对被遮挡边没有施加约束，四种可能都应该予以保留。图12-5左边模块的四个箭头连接（B，E，G，I）有相同的（凸）结构，但右边模块中有另一类型（凹）的箭头连接（7），表示由模块和墙壁相交构成的凸面。

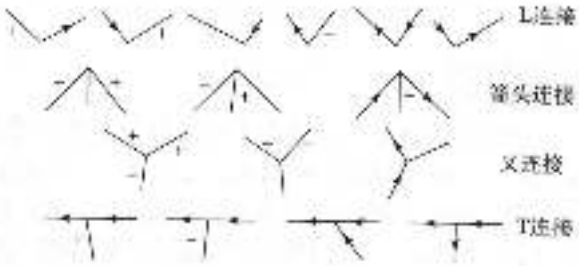


图12-6 三面角模块世界（所有的3D角由三平面相交形成，目标处于常规观察位置）的图像，仅有16种可能的拓扑连接。连接类型自上而下依次是，L连接、箭头连接、叉连接、T连接

377

在继续讨论之前，读者应该确信全部16种连接实际上都可以从3D模块的投影推出。更困难的是证明不可能有其他的连接。这个难题已经有人证明过了，读者只需在做习题12.8和习题12.9时证实找不到其他的连接即可。

习题12.8

根据你的观察，对图12-1的左侧场景进行解释，并对右侧的线段进行标记。

习题12.9

尝试把12-7中所有模块的所有边标记为折痕或刃边。所有连接都应来自图12-6中列出的类型。（a）哪些线条图有一致性标记？（b）哪些线条图看起来与实际目标对应却不能被标记，

该标记为什么失败？（c）哪些线条图看起来对应不存在的目标？对所有的线条图都能进行一
378 致性标记吗？

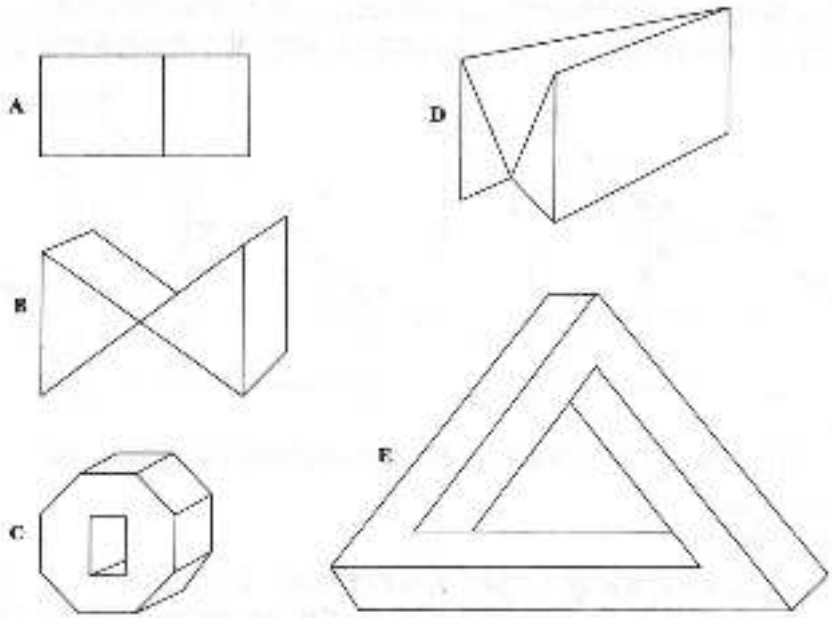


图12-7 在有限的模块世界中，线条图可以有3D解释，也可以没有3D解释？哪些有解释？哪些没有？为什么？

习题12.10

画出实际场景的线条图并进行标记，这个场景至少有两种不同的物体，都包含全部四种连接类型。建立你自己的场景，可以采用本节的几个图形结构。

第11章中介绍的两种算法，可用来自动标记线条图：一个算法是顺序回溯，另一个算法是并行松弛标记。我们首先把要解决的问题形式化：**已知2D线条图，具有一组边P_i（观察到的目标），给每条边分配标记L_i（模型目标）以解释边的3D情况，使连接标记的类型属于图12-6列出的16种类型。**符号P和L的使用与第11章中的情况一致。算法细节请参考第11章的内容。为了强调几点，后面给出粗略的算法步骤。除非提供其他附加信息，否则这两种算法通常会产生多种解释。流行的做法是，把线条图中所有凸表面上的边标记为"+"，使凸表面位
379 于右侧。

算法12.1 用回溯法标记模块边缘，并对场景图中的所有边进行一致性解释

输入：表示边集E和连接集V的图。

输出：边集E到标记集$L=\{+,-,>,<\}$的映射。

- 任意假定一个边排列顺序：$E=\{P_1, P_2, \cdots, P_n\}$。
- 前进阶段i，用标记集$L=\{+,-,>,<\}$中下一个未用过的标记对边P_i进行标记。
- 检验新标记与所有其他边的一致性，其他边通过V中的某个连接与该边相邻。
- 如果新分配的标记产生的连接不属于16种类型，那么回退；否则进入下一个前进阶段。

如果可能的话，应该首先对标记约束最多的边进行赋值。甚至外界信息（如立体视觉）已经指明这条边与3D折痕对应。根据角和相关边的数目确定每个连接的类型，需要做一些预处理工作。其他改进的方法中，把16种类型的解释分配给连接标记，去掉那些存在矛盾的连接标记，即该连接与邻近连接对公共边的解释互相矛盾。图12-8是关于四面塔状物线条图的解释树。搜索空间相当小，说明了三面角模块世界的强约束机制。

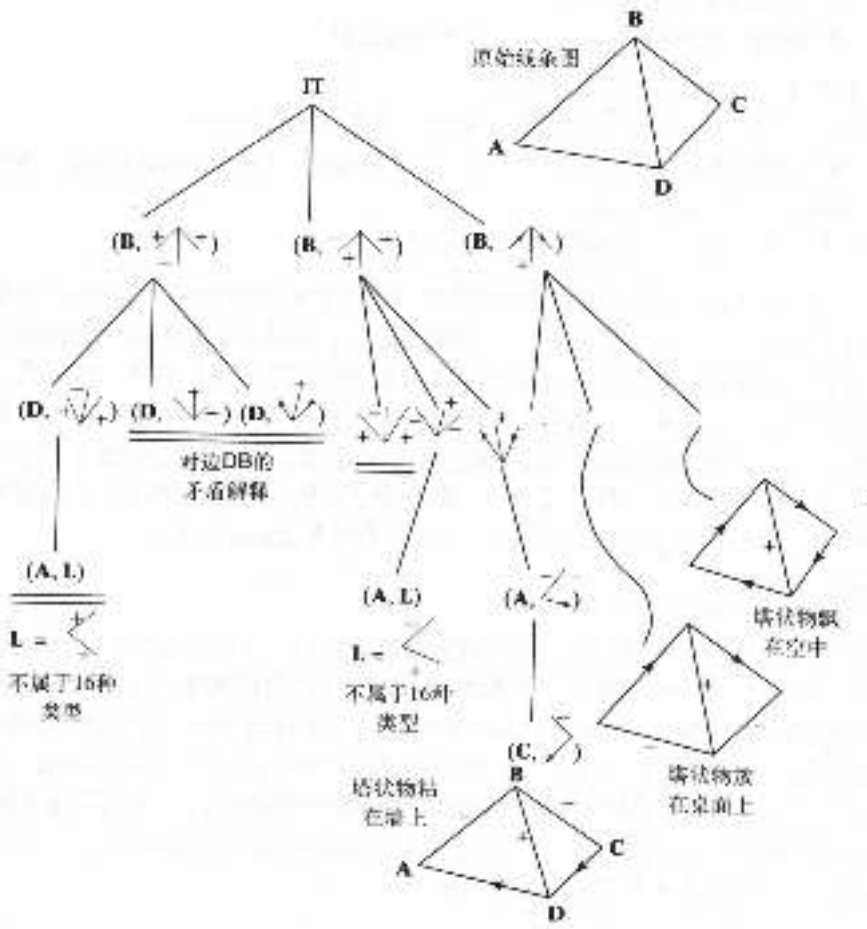


图12-8 右上角塔状物线条图的解释树。在树的每一层，用图12-6中的16种连接标记，对四个连接进行标记。在树的第一层，给连接B分配解释标记，接下来给连接D、A和C分配解释标记。右下角，三条完全通路产生三种解释

习题12.11

完成图12-8的解释树中省略的右侧部分，提供所有的边和节点。

习题12.12

构造5层解释树，给图12-8所示塔状物所有的边分配一致性标记。首先，用第11章中的一致性标记形式进行问题表示，用5个观察到的边和4个可能的边标记，来定义P、L、R_P、R_L。然后画出解释树。树中是否有三条完全通路与图12-8中的三条完全通路对应？

松弛法线段标记

第11章讲过，可用离散松弛算法来约束对线条图部件的解释。在此为线条图的各边分配标记。当然也可用类似的程序为连接分配标记。

算法12.2 用离散松弛法标记模块边缘，并对场景图各边进行一致性解释

输入：表示边集E和连接集V的图。

输出：边集E到标记集$L=\{+,-,>,<\}$子集的映射。

- 初始化，给每条边P_i分配标记$\{+,-,>,<\}$。
- 在每一步，通过对所有边做如下处理，过滤出可能的标记：
  给与P_i相连的边赋以可能的标记，如果标记L_i不能构成合法的连接，那么从P_i的标记集中去掉标记L_i。
- 当标记集合大小不再减小时，就停止迭代。

算法12.2是对大量不同类似算法的简单概括。因为算法简单并可以以任何顺序执行，甚至每步都可以并行执行，范例中建立了一个有趣的模型，模拟沿着人类视网膜神经网络的信息流动方向所发生的现象。有人研究过图像亮度约束条件，以及多分辨率工作方式。模块世界的研究工作趣味浓厚，后来的工作也卓有成效。但这只是玩具形式，对多数实际场景来说没有用处，因为（a）多数3D目标并不满足我们所做的假设，（b）实际2D图像表示与要求的线条图相去甚远。已经提出了一些改进方法，例如为了能够表示曲面目标，对标记和连接类型进行了扩充，并对线条图误差进行了调整，这些将在参考文献中提及。

习题12.13 内克（Necker）现象

习题12.12对处于常规位置的塔状物图像边缘进行标记，本题在此基础上稍做改变。图12-9是立方体的线框图，没有任何遮挡，图像中12条折痕边都是可见的。（a）凝视最左边的一幅，通过你的观察能对这幅线条图进行3D解释吗？经过几分钟的凝视后，这个解释有变化吗？（b）标记中间的图像，使连接G成为前角。删去连接H及附带的三条边，于是表示出一个不透明的立方体。（c）重复（b）的工作，令H为前角，删除连接到G的各边。注意在我们定义的模块世界中，3D线框目标不是合法的目标。但是我们可以使用相同的推理方式，去解释立方体任何角的邻域。这些角确实属于16种连接的类型。

380
382

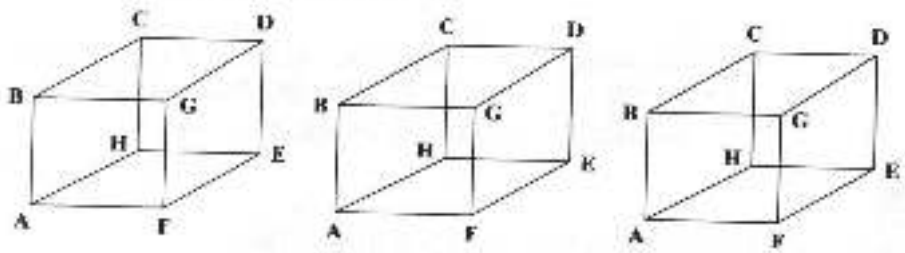


图12-9 参见习题12.13。内克立方体有多种解释。盯着看其中的一幅图，一般都有不同的解释方式。中间的两个叉型连接，可以解释成前角，也可以解释成后角

习题12.14

把第11章的解释树程序应用到图12-7所示的线条图，写出完全通路中的正确标记。

12.3 2D图像中的3D线索

图像是实际世界的2D投影，但是喜欢艺术或电影的人都知道，2D图像能够唤起丰富的3D情感。2D图像中存在很多线索，可用于3D解释。

在图12-10中可以看到一些深度线索。两个熟睡的人挡住了长椅，长椅挡住了灯柱，灯柱挡住了复杂的栏杆，栏杆挡住了树，树挡住了有尖顶的建筑物，建筑物又挡住了天空。可以从右侧灯柱的影子和较明亮的灯柱右表面看出，太阳从图的右边远处照过来。同样，右侧看 383
不见的栏杆在地上投下复杂的影子，在瓦片铺就的院子中产生虚假的外观。地上的纹理表明地是平面，纹理逐渐缩短表明地面逐渐远离观察者。通过左侧建筑物墙壁的边沿走向，人们可以很明显地看出墙壁的方向。栏杆图像从右向左的走势信息强烈暗示我们：在3D环境中，栏杆深度在向后延伸。类似的，长椅从左向右的走势也说明其深度向后延伸。灯柱和人的图像比尖顶大很多，说明尖顶离得很远。

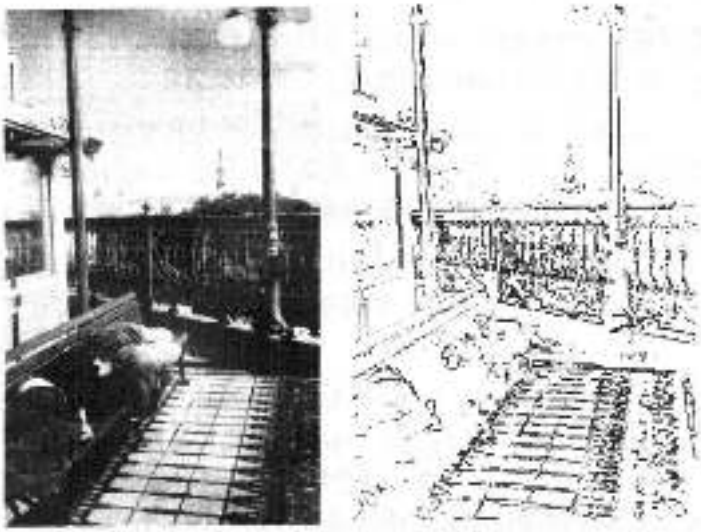


图12-10 在大湖区圣罗伦斯河的悬崖上拍摄的彭北克城
（左）图像中有很多深度线索
（右）Roberts边缘检测、阈值化后保留10%的像素

定义96 当一个目标遮挡另一个目标时就出现**穿插**（interposition）现象，这时遮挡目标到观察者的距离比被遮挡目标的距离要近。

习题12.15

把第11章中的松弛标记程序应用到图12-7的线条图中。如果有任何一条边的标记集变为NULL，那么就没有一致性解释。如果有任何一条边在最终标记集中的标记多于一个，那么该算法存在歧义性问题。在这种情况下，可以对线条图使用多个标记，然后验证哪些是可实际实现的标记。

习题12.16

找出图12-3盒子中线段的所有T连接。是否每个连接都真的表示一个表面被另一个表面遮挡？

如上面所讨论的，在对图12-10进行解释时，目标穿插现象给出了非常明显的线索。毫无疑问，长椅比被它遮挡的灯柱更近，而灯柱比栏杆更近。个别目标的识别可能有助于利用这些线索，但这不是必需的。图像轮廓中形成的T连接给出了很明显的局部线索，参见图12-11。注意在图12-10右侧的边缘图像中，建筑物边缘是它和长椅上沿形成的T连接的竖边，栏杆则是栏杆和灯柱右侧边形成的T连接的竖边。一对相对的T连接是更明显的线索，因为它表明一个连续的目标在另一个目标的后面穿过。这个边缘图像是很复杂的，因为它表示室外场景。对于较简单的情况，请参考后面的习题。**可以利用已识别的目标或表面的穿插现象，计算目标间的相对深度。**



图12-11 T连接。表示一个目标被另一个目标遮挡。T连接的横边对应遮挡目标，其竖边对应被遮挡的目标。两个相对的T连接与一个T连接相比，前者提供了更明显的遮挡证据

384

定义97 **透视缩放**（perspective scaling）是指，目标的距离与它在图像中的大小成反比。缩放这个术语专门用来比较与图像面平行的目标大小。

识别图12-10中的尖顶时，我们知道它们离得很远，因为其图像尺寸很小。当从右向左看时，栏杆的竖直部件变小，同样，当从很高的建筑物上观看下面的街道时，距离地面越高，人和汽车就显得越小。**目标在图像中的大小可用来计算该目标的3D深度。**

定义98 在与目标轴成锐角的方向观察目标时，图像中的目标会出现**透视缩短**（foreshortening）现象。这提供了另一个明显的线索，反映了2D视图与3D目标之间的关系。

观察图12-10中的长椅及上面的人，它们形成的图像长度，与长椅近距离水平横放所形成的图像长度相比，前者要显得短一些。同样，当场景中栏杆逐渐远离时，栏杆的竖直部件在图像中逐渐靠近。如果视线与栏杆面垂直，这个缩短现象是不会出现的，纹理梯度也是相关的3D线索。纹理成分容易受到透视缩放和透视缩短的影响。发生的纹理变化给观察者提供了纹理表面的距离和方向信息。当仰望砖结构建筑物，沿着平铺的地板或铁轨方向观看，从玉米地或体育场的人群上面看过去，这个效果是很明显的。图12-12可以说明这一点。当我们的朋友身穿有着规则纹理图案的衣服时，纹理梯度还告诉我们关于他们体形的信息。图12-13显示出现纹理梯度的简单情况。随着3D距离的增加，纹理或者虚线在图中向着图像中心逐渐靠近。图12-14显示的是，用规则的栅格光纹照射场景中的目标，就在目标表面上形成了纹理。该结构光不仅使我们得到表面形状的信息，而且能用来自动计算表面的法线方向甚至深度，下一章我们就会看到这一点。可以用图像中纹理的变化计算由该纹理产生的3D表面的方向。

385



图12-12 玉米地图像，显示出复杂的纹理（玉米和玉米的行）和纹理梯度。图像中自下到上纹理变得更密，因为每平方厘米的图像包含了更多的玉米叶（John Gerrish提供）

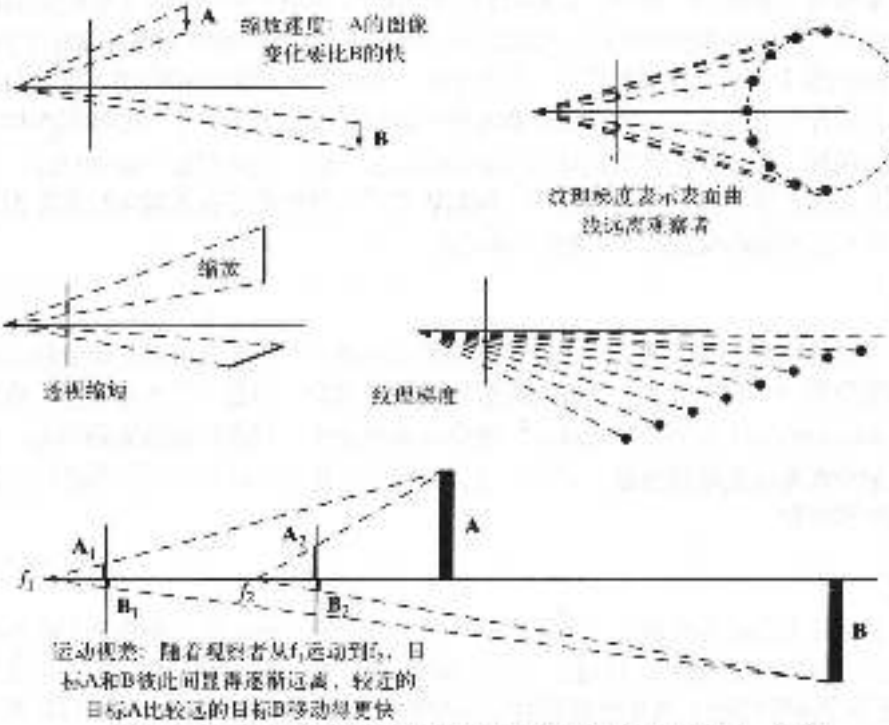


图12-13 缩放、透视缩短、纹理梯度和运动视差的效果示意图。图中，靠前的图像面用一条垂线段表示，目标位于其右面

定义99 纹理梯度（texture gradient）是图像纹理（测量的或感知的）沿图像中某个方向的变化，它常常能够反映3D目标的距离或表面方向的变化，其中纹理是指所研究目标表面上具有的纹理。

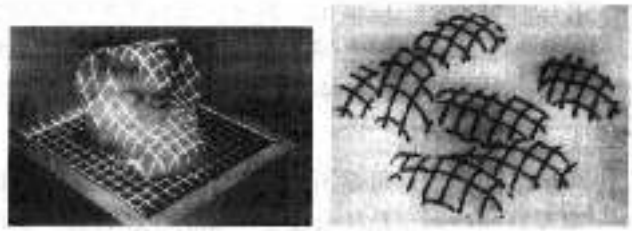


图12-14 结构光照射目标形成的纹理，揭示了目标的3D表面形状（图像由Gongzhu Hu提供）

（左）光相投射到置于平地的光滑雕塑上

（右）去掉平面有从后的目标上的条纹，这是什么物体？

3D规则纹理表面在图像中会产生纹理梯度，反过来则不一定正确。当然，艺术家通过在2D纸上创造纹理梯度来产生3D的表面效果。

定义100 运动中的观察者能够通过**运动视差**（motion parallax）得到目标的深度信息，在这种情况下，即使是静止目标彼此间也会出现相对运动的现象：近处目标的图像要比远处目标的图像运动得更快一些。

尽管运动视差是由观察者运动引起的，但如果观察者静止而目标运动的话，也会出现类似的效果。图12-13把刚讨论过的几种效果通过透视投影表现出来。当我们沿街道行走时（假设闭着一只眼睛），身边经过的目标例如垃圾箱或大门，它们的图像在视网膜上的运动远比前

方同类目标的运动速度快。开车时，迎面而来的车辆在一定距离外图像是稳定的，最终它们会从我们的车窗外飞驰而去。同样，经过我们身边的汽车图像，其变化速度要比远处汽车的图像变化快得多。由于透视投影的数学原理相同，运动视差与透视缩放和透视缩短有关。

在一幅2D图像中，除了我们上面讨论的现象外，还有更多其他的3D线索，比如与较近的目标相比，远处的目标会带更多的青色，或者由于目标和观察者之间空气的散射作用，图像可能显得不够明快。通过变化的焦距可以得到深度信息，这一点将在第13章中讨论。另外我们还没有论及现实世界中的其他约束条件，例如还没有假设地平面或者确定特殊垂直方向的引力世界，而人类视觉系统是离不开这两个条件的。

习题12.17

闭上一只眼睛观察一支铅笔，保持它与两眼间的连线平行，然后把它从眼前移动到一臂远处，铅笔图像的尺寸发生变化，是缩放现象还是缩短现象？还是二者兼而有之？抓住铅笔中心，保持眼睛和铅笔中心间的距离固定，绕中心旋转铅笔，观察铅笔图像的变化。图像尺寸的变化是缩放现象还是缩短现象？还是二者兼而有之？把与图像尺寸有关的近似三角公式表示为旋转角的函数。

386
|
387

习题12.18

让一根手指垂直贴近你的鼻子，轮流睁开双眼，两秒钟一换。会观察到手指有明显的运动（实际是不动的），把手指往后移动，重复前面的过程，把手指移到一臂远处，再重复前面的过程。（把指尖对准门把手或其他更远的目标效果会更好。）描述手指位移量与到鼻子的距离之间的关系。

12.4 其他3D现象

第10章讨论了一些理论，通过聚集图像特征以得到较大的3D结构，如格式塔（Gestalt）原理。这些原理在从图像得到3D解释方面是卓有成效的。当然有时会出现错误，也就是对一些情况做出了不正确的解释。下面简要讨论从2D图像特征到3D结构解释的其他重要现象。

12.4.1 从X恢复形状

20世纪80年代，根据不同图像特征计算表面形状的研究工作突飞猛进。研究中常常使用单一图像特征，而不是结合使用不同的图像特性。有的数学模型将在第13章进行讨论，而下面详细介绍所用的3D现象。在此我们仅介绍用作3D形状线索的特征X。

1. 从明暗恢复形状

艺术课上会讲到明暗手法的使用，明暗处理是在2D图像上产生3D效果的重要手法。光滑的目标，例如苹果，由于光线的入射角与反射角相等，如果从反射角位置观察苹果，就会感到强烈的反光。同时，当目标的表面法线方向逐渐与照明方向垂直时，光滑目标的表面就会逐渐变暗。平面表面的外观在图像上是均匀的，因为图像亮度与平面法线方向和照明方向的夹角成正比，可以通过公式从图像亮度计算表面法线的方向，但多数计算方法需要标定数据。利用标定数据建立图像亮度与法线方向之间的关系，还有的方法需要用到多台摄像机。一般算法都要用到与几个参数有关的模型公式，有关参数如反射能量、入射能量、照明方向、反射方向、面元方向及面元的反射系数。需要这么多的参数，我们只能期望在高度受控的环境中，能够根据明暗信息很好地恢复出形状。图12-15是具有均匀网格的圆柱体图像，照明来自单一方向。图

12-16显示两光滑目标的图像，带有很好的明暗信息，使我们能够看出它们的形状。

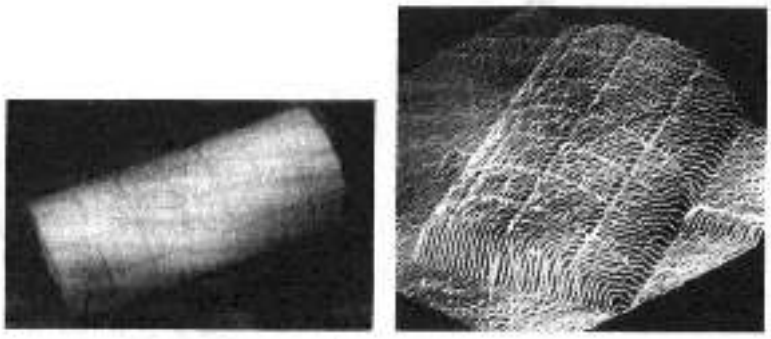


图 12-15

（左）被图案圆柱体的图像，把网格纸缠到被罐外形成圆柱体的表面

（右）亮度函数的3D图，这时的视点稍有变化，注意观察用亮度值表示出的圆柱体形状

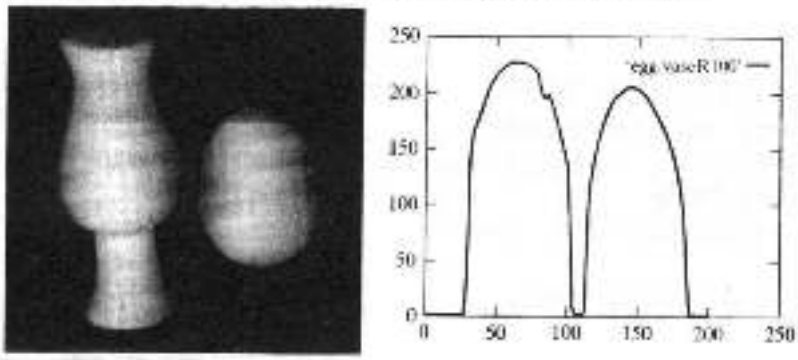


图12-16 光滑目标的亮度图像，目标是花瓶和鸡蛋，右图是穿过一行高亮像素行的亮度曲线，注意观察亮度如何与目标形状密切相关（Deborah Trytten提供）

2. 从纹理恢复形状

假设纹理存在于单个3D表面，并且纹理模式有一定规律，就可以用2D纹理梯度的概念计算3D表面的方向。前面已经讲过纹理梯度的概念，图12-18显示以某个角度观察3D表面上的规则纹理，从而在2D图像中形成了纹理梯度。要特别定义两个角度，建立表面方向与观察方向之间的关系。 388

定义101 表面法线在图像中的投影方向角称为表面的**倾斜角**（tilt）。表面法线与视线的夹角称为表面的**俯仰角**（slant）。参见图12-18。

假设有人直立地站着，眼睛看着前面平坦的麦田。如果头是竖直的，那么田地的倾斜角是90°。如果看得足够远，那么俯仰角接近90°；如果只是看到脚下，那么俯仰角近似0°。如果头向左倾45°，那么田地的倾斜角变为45°，如果头向右倾45°，则田地倾斜角为135°。图12-19主要包含两个平面，即地面上的人行道和带台阶的墙。人行道倾斜角90°，俯仰角大约75°。（道路向上拱起15°。）带台阶的墙倾斜角约170°，俯仰角约70°。倾斜角和俯仰角的概念可以用到任意表面，而并不只是那些接近地面的表面。例如建筑物的内墙或外墙，盒子或卡车的正面等。事实上这些概念也可以用于曲面元，但由于表面法线方向的变化，使得计算图像中的纹理梯度更加困难。 389

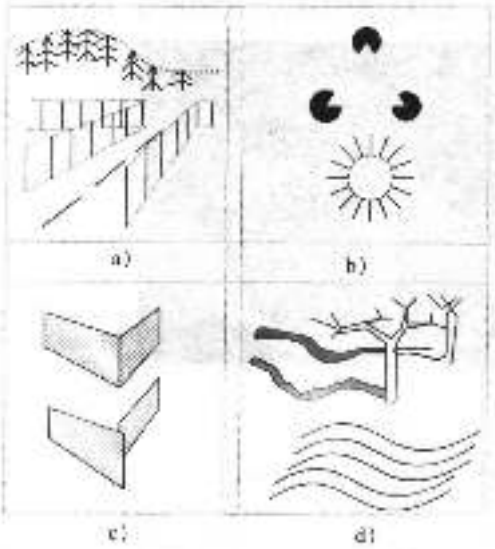


图12-17 从2D图像特征推出其他3D线索

a）相似特征聚集会形成虚拟直线和虚拟曲线

b）虚拟边界能够误导人类，使我们感到中间穿插了与背景亮度不同的目标

c）2D中的对齐常常意味着3D中的对齐，但有时不是这样

d）2D图像中的曲线包含3D表面的形状信息

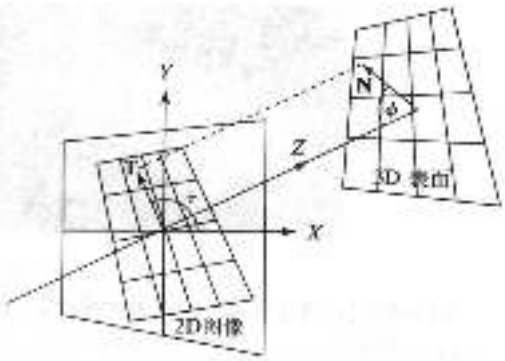


图12-18 根据表面法线N相对视觉坐标系的方向，确定表面的倾斜角和俯仰角。倾斜角τ是N投影到图像中的方向（T），俯仰角φ是N与视线的夹角

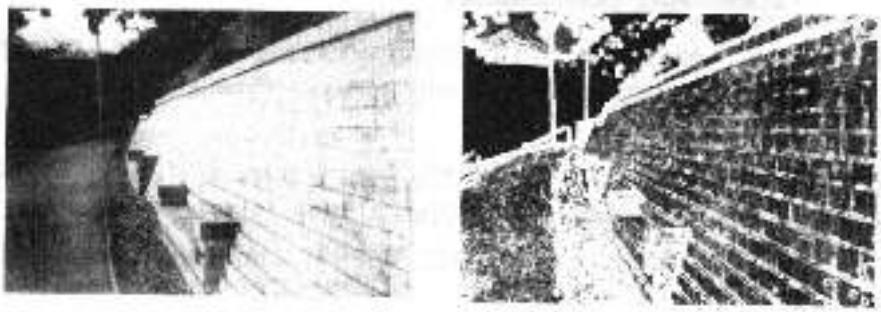


图 12-19

（左）包含很多纹理的图像

（右）5×5 Prewitt边缘检测结果。人行道的倾斜角是90°，俯仰角大约75°，墙的倾斜角大约170°，俯仰角大约70°

习题12.19

（a）对于图12-5中的目标，给出四个表面的倾斜角和俯仰角。（b）对于图12-1中的目标，做相同的工作。

3. 从边界恢复形状

人类可以通过图像中2D边界的形状推断3D目标的形状，对于图像中的椭圆，直接的3D解释是圆盘或球。如果圆面上明暗信息和纹理都是均匀的，那么就认为是圆盘；如果明暗或纹

理向边界逐渐变化，那么就认为是球。卡通画和其他线条图经常不加明暗效果和纹理，但人类仍然可以从中推断出3D形状。

被光滑曲线围绕的区域，其内部点对应的表面法线方向可以算出。考虑简单的圆周情况，光滑的假设意味着在3D中，目标轮廓上的表面法线垂直于视线，同时又垂直于图像中的圆周，这就允许我们给图中的边界点分配唯一的法线方向。法线方向与边界点的走向相反。这些边界点是圆周直径的端点。然后就可以沿着整个直径插入平滑变化的表面法线方向，确保中间像素的法线方向指向观察者。要做到这一点还需要一个附加条件，因为椭球表面与球表面会有所不同，半球壳的内表面与外表面也不同。附加条件能够限制只产生一个表面，这个表面也可能是错误的。可以通过明暗信息约束表面方向的分布。鸡蛋和球体的明暗效果是不同的，但球的内外侧也许没有这种明暗差别。 390 - 391

习题12-20

找一幅人或动物造型的卡通画。(a) 图上有表现3D目标形状的明暗效果、阴影或纹理吗？如果没有，假设有光源位于前右上方，请添加一些效果。(b) 在纸上画出目标的边界，把边界内20个左右的点添加表面法线方向，以表示出目标形状，就像本征图像中表示的一样。

12.4.2 消隐点

透视投影使平行线发生有趣的变形。几个世界以来，艺术家和画家一直在利用这个知识进行创作。图12-20显示两个广为人知的现象。第一个现象，向光轴倾斜的3D线在2D图像中消失于一个点，称这个点为消隐点（vanishing point）。第二个现象，如图所示的一组平行线有相同的消隐点。利用透视投影的代数模型，可以很容易对这个现象进行解释。平行于同一平面的不同方向的平行线，其消隐点构成消隐线（vanishing line）。特别地，地面上不同方向的平行线的消隐点构成了地平线（horizon line）。图12-20中，地面是由矩形块铺成的表面，点V_1和V_3构成地平线。注意三条平行线（公路）消失在点V_2，这个点与矩形纹理构成的消隐点处于同一地平线上。利用这些透视特性，可以根据未标定摄像机拍摄的图像，推断出摄像机模型。最近，根据这些原理已经开发出一些系统，它们能够利用几个视点的场景视频建立场景的3D模型。 392

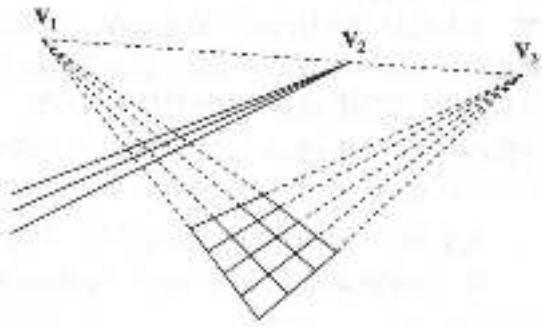


图12-20 透视投影下，与光轴倾斜的3D线，在2D图像中消失在某个点。平行线在图像中相交于相同的消隐点

12.4.3 根据焦距变化求深度

单个摄像机如人类的眼睛一样，可用来计算与像素点对应的表面点的深度。睫状肌改变形状起到调焦作用，使眼睛能看清目标。传感器通过调焦使目标或目标边缘进入注视范围内，以此得到目标的深度信息。人类已经根据这个原理制造出摄像设备，其中包含自动聚焦控制功能。为了叙述简单，可以设想摄像机的焦距在某个范围内平稳变化。对应每个f值，对得到的图像进行边缘检测。对于每个像素，保存产生清晰边缘的f值，并利用f值确定该像素对应的3D表面点的深度。很多图像点不是由3D中的反差邻域产生的，因此不会产生可用的清晰边缘值。短焦距镜头，如f<8mm，具有很好的景深（depth of field），这意味着目标与摄像机的距

离可以有较大的变化范围，在这个范围内都能够产生较好的聚焦效果。短焦距下不利于确定到焦点的准确距离，而此时采用长焦距比较有利。在12.7节我们会看到，如何根据物理学中的透镜方程得出这个结论。

利用图像亮度和阴影 前面已经提到过，结构光可以在均匀表面上产生特征。同样，阴影也可以起到类似的作用。人类和机器都能够根据表面上的图案来推断表面的存在和形状。考虑图12-17d中的曲线。根据曲线的形状，可以认为3D表面呈波浪形。在被雪覆盖的地形中，阴影对滑雪者是有帮助的。对他们来说，判断地势时即使有六英寸的小错误，也会很容易失去平衡。类似的情形显示在图12-14中，图中的投影光带图案指明了土豆的椭圆形状。

12.4.4 运动现象

我们已经讨论过运动视差。当运动着的视觉传感器跟踪拍摄3D目标时，随着传感器接近目标，目标的2D图像点显得膨胀了。（如果目标逃离速度比跟踪速度快，则目标的图像点将是收缩而不是膨胀。）称跟踪的中心点为**膨胀中心**（focus of expansion）。如果目标朝向传感器运动，也会出现类似现象。这种目标图像的快速膨胀现象称为**渐近**（looming）现象。第9章的光流理论可以解释这种现象。图像流与目标或追踪者的距离和速度之间的关系已经有了定量的描述方式。

12.4.5 边界和虚拟线

如图12-17所示，边界和曲线可能是虚拟的（virtual）。参见左上角的图，围栏柱的两端、树尖及公路路标在图像中形成虚拟曲线。右上角显示两个著名的心理学测试例图：上面的例图，人们会感到比较亮的三角面挡在三个深色圆之上；下面的例图，则会让人感到比较亮的圆面挡住了从中心发出的光线。如果人类视觉系统就是认为存在穿插的目标，它一定否认该目标刚好与被遮挡的背景有相同的反射系数。对人类视觉系统来说，非常容易出现这种错觉，即使去掉图12-17中的虚线仍是如此。机器视觉系统则不会出现这样的错误，即不会感到图中的中心区域要比背景更亮，这是因为它能够得到具体的像素亮度值。

393

定义102 图像中类似的点或目标，沿着某条直线或曲线进行聚集，在图像中就会出现一条**虚拟直线**（virtual line）或**虚拟曲线**（virtual curve）。

习题12.21

仔细制作两张白色卡片，其中包含图12-17b中的两个虚幻图。把这两张卡片拿给5个人观看，看看他们是否认为中心区域更明亮一些。你不能直接这样问他们，而应该问一般性的问题，让他们描述看到了什么，例如问他们"你感觉图中有什么？""请说出它们的形状和颜色"。然后对结果进行总结。

12.4.6 非偶然对齐

空间中的目标之间或目标与观察者之间的对齐现象存在偶然性，但人类视觉系统不愿承认这一点。相反，我们常常认为2D图像中的对齐是由于3D对齐引起的，例如当我们看到图12-17c上面的两个四边形时，会认为3D中有两个矩形表面在边缘处相交，认为这条边是透视缩短后形成的折痕。图12-17c下面的两个四边形是另一个视点的图像。上面的叉连接和箭头连接变成了下面的T连接，这时就感到是一个表面遮挡了另一个表面。虚拟曲线感知效果是基于相同原理的另一种表现形式。事实上，图12-17中的四幅图都是基于相同的原理。就像Irving Rock在1983年的论文中提到的，人类视觉系统倾向于接受关于图像数据解释的最简单

的假设。(这个观点能解释很多实验现象，结果使视觉过程类似于推理过程。但是这个观点似乎和有的实验数据矛盾，使视觉编程非常困难。)

下面是图像解释中要用到的启发式规则，其中没有一条可以在所有情况下都能给出正确的解释，很容易找到反例。这里用到的术语边，除了它的2D含义外，也指3D中的折痕、标志或阴影。

- 图像中的一条直边，对应3D中的一条直边。
- 2D图像中连接点的边，对应3D中角的边。(更一般的，2D中的重合对应3D中的重合。)
- 2D曲线上的类似目标，对应3D曲线上的类似目标。
- 2D多边形区域，对应3D多边形面。
- 2D光滑曲线边界，对应3D光滑目标。
- 2D对称区域，对应3D对称目标。

394

12.5 透视成像模型

现在推导透视成像的代数模型。建立摄像机坐标系C中的点与实际图像坐标系R中的点之间的关系，其推导过程相当简单。首先考虑如图12-21所示的1D情况。对于从飞机上直接向下拍摄平坦地平面这类问题，图12-21所示的情况就是一个比较合适的模型。传感器拍摄到点B，该点投影到图像生成点E。传感器坐标系的中心是点O，OB长度在光轴OA上的分量为z_c。点B在图像中的像点到图像中心的距离是x_i。f是焦距。利用相似三角形，可得到公式(12-1)。公式说明实际2D图像的坐标(或尺寸)等于3D坐标(或尺寸)乘以焦距与距离之比。只要所有的3D点位于到传感器距离相同的同一个平面上，那么2D图像就是对3D目标的缩小版。这个模型可应用于实际，如显微镜分析、航测图像分析或扫描文档分析。

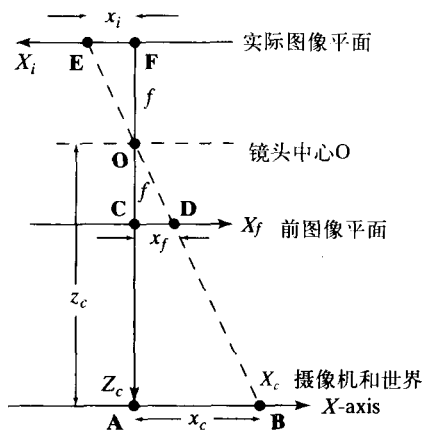


图12-21 实际图像平面和前图像平面的简单透视模型。利用相似三角形中的关系$x_i/f = x_c/z_c$，目标的实际尺寸x_c与图像尺寸x_i建立起联系

$$x_i/f = x_c/z_c \quad \text{or} \quad x_i = (f/z_c)\,x_c \tag{12-1}$$

用前图像平面(front image plane)比用实际图像平面更方便，因为在前图像平面上的目标与实际目标的方向一致。前图像平面是一个抽象图像平面，它与实际目标位于光心的同一侧，到光心的距离为f。在前图像平面上的目标，与在实际图像平面上的目标有相同的比例，而方向与实际目标相同。前图像平面上的点C和D与实际图像平面上的点F和E对应。透视成像公式对前图像平面上的点成立。从现在开始，我们用的都是前图像平面。

3D到2D的透视投影情况参见图12-22以及公式模型(12-2)。x和y的计算公式，推导过程与1D情况类似，也是利用相似三角形推出的。注意从3D到2D的投影，是一个多对一的映射。395
从图像点到3D空间光线上的所有3D点都对应同样的2D图像点，这样在成像过程中就会丢失很多3D信息。公式(12-2)提供一个代数模型，利用这个公式，计算机算法构建从图像点(x_i, y_i)进入3D的光线上的所有3D点的集合。本书关于3D工作的讨论中，离不开这个重要的数学公式。在结束这个话题前，要强调的是，公式(12-2)的简单形式，仅仅是把3D摄像机系中

的点与2D实际图像系中的点联系起来。涉及物体坐标系或实际世界坐标系中的点时，需要用到代数变换。这一点将在第13章中讨论。如果摄像机以恒定距离$z_c = c_1$观察平面目标，那么图像只是目标平面的简单缩放。设$c_2 = f/c_1$，就得到简单的关系$x_i = c_2 x_c$和$y_i = c_2 y_c$。这样就对图像坐标进行了简化，可知图像是实际目标的缩放版本。

$$x_i/f = x_c/z_c \text{ or } x_i = (f/z_c)\,x_c$$
$$y_i/f = y_c/z_c \text{ or } y_i = (f/z_c)\,y_c \tag{12-2}$$

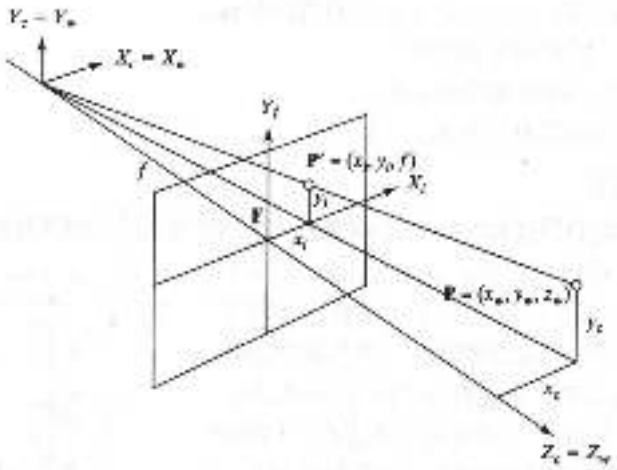


图12-22 透视投影到2D图像的一般模型

习题12.22 同比例缩放的特点

摄像机垂直向下正对一张桌子，使图像平面与桌面平行（类似一个照片放大装置），参见
396 图12-21。证明1in长的钉子放在桌面上的任何位置，只要处于视场中，其图像（线段）都有相同的长度。

习题12.23 视觉引导的拖拉机

参见图12-23，假设用前视摄像机引导农用拖拉机前进，进行除草和施肥。如图所示，摄

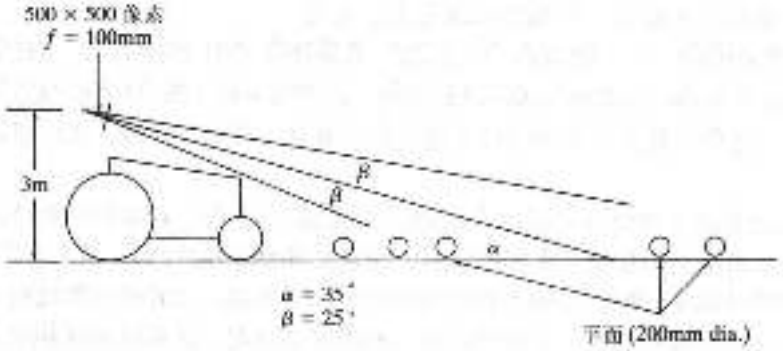


图12-23 视觉引导农用拖拉机的摄像示意图。参见习题12.23

像机焦距100mm，摄像机距离地平面3000mm。视场角是50°，光轴与地平面成35°角。(a) 在拖拉机前面，沿着地面的视场长度是多少？(b) 假设作物实际间隔500mm，当它们处于视场极限位置时，它们在图像中间隔多少？(图像间隔因靠近或远离作物而不同。)(c) 如果作物大体上是球形的，直径200mm，图像大小是500 × 500像素，那么作物在图像中的直径是多少像素？(同样，答案因靠近或远离作物而不同)(d) 相邻的作物在图像中是重合在一起还是它们之间有间隔？

12.6 通过立体视觉求深度

如图12-24所示，利用立体视觉传感器确定3D点在空间中的位置，只需要具备简单的几何知识和代数知识。小心放置两台摄像机，使它们的X轴重合，Y轴和Z轴分别相互平行。Y轴垂直于纸面，所以在实际推导中并不使用。右侧摄像机的原点或投影中心的偏移量为b，b是立体视觉系统的基线(baseline)。目标点**P**在左图像中对应点为$\mathbf{P}_l$，在右图像中对应点为$\mathbf{P}_r$。通过几何分析，可以确定点**P**位于光线$\mathbf{LP}_l$和$\mathbf{RP}_r$的交点处。

根据相似三角形，得出公式(12-3)：

$$\begin{aligned} z/f &= x/x_l \\ z/f &= (x-b)/x_r \\ z/f &= y/y_l = y/y_r \end{aligned} \tag{12-3}$$

397

从图中可以看出，坐标y_l和y_r是相同的。对公式(12-3)做一些变换，就可以得到点P的两个未知坐标x和z。

$$\begin{aligned} z &= fb/(x_l - x_r) = fb/d \\ x &= x_l z/f = b + x_r z/f \\ y &= y_l z/f = y_r z/f \end{aligned} \tag{12-4}$$

在求解点P的深度时，我们引入了*视差*(disparity)的概念，也就是公式(12-4)中的d，它是左右图中图像坐标x_l和x_r之差。求解这些方程就可以得到点**P**在3D空间中的三个坐标。公式(12-4)说明，到点**P**的距离随着视差的减小而增加，随着视差的增加而减小。视差趋近零时，距离趋近无穷。这种简单的立体成像系统，在两个y图像坐标间没有视差。

定义103 当同一个3D点投影到不同的两摄像机图像上时，对应点在图像上的位置差就称为**视差**。

在图12-24中，要定位的3D空间点**P**是一个简单点，在确定图像匹配点$\mathbf{P}_l$和$\mathbf{P}_r$时不会出现问题。对包含很多表面点的实际3D场景，确定对应点就非常困难的，因为通常并不清楚左图像中的哪个点与右图像中的哪个点对应。假设有一对如图12-12所示的玉米田图像，在图像的各行有很多相似的边缘点。一般需要立体摄像机做精确对应，只有这样才可以保证搜索对应点时是在两幅图像的相同行中进行。尽管已经知道并使用了很多约束，问题仍然存在。很明显的一种情况是，点**P**在两幅图像中都看不到。玉米田的稠密纹理造成要处理的特征点过多，398
特征点过少的相反情况也很常见。特征点过少发生在无纹理的光滑目标，如大理石雕像或被白雪遮挡的小山。工业应用中，可以利用结构光人为地加上特征点，如图12-14所示。后面进行更详细的讨论。

尽管存在上述困难，不断的研究和改进还是实现了几个商业化的立体视觉系统。有的采

用不只两台摄像机。有的系统能够以接近摄影机帧频的速度，产生深度图像。第16章讨论在ATM机上通过立体视觉系统进行身份识别。

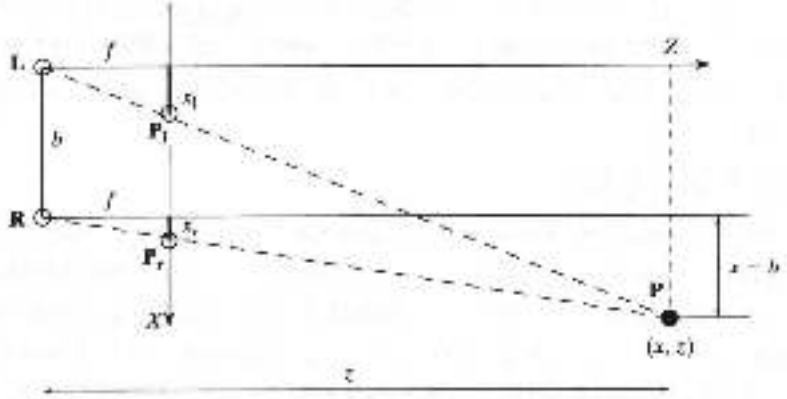


图12-24 简单立体视觉系统的几何模型。传感器坐标系建立在左眼L（或摄像机）上，基线是b。所有的量都是相对L测量的，只有x_r是相对于R测量的

习题12.24

进行下面的简单立体视觉实验。（a）看一本书，它位于鼻子前面约30cm处，轮流睁开一只眼睛，每次两秒钟。你观察到特征点如标题字母在左右图像中的视差了吗？（b）让书位于一臂远的地方，重复实验。视差变大或变小了吗？（c）小心地转动书。你能找到一个角度，在这个角度下，右眼可以看到书的封面而左眼看不到吗？

习题12.25 立体计算的误差

假设立体摄像机基线$b = 10$cm，焦距$f = 2$cm，观察点$\mathbf{P} = (10\text{cm}, 1000\text{cm})$的成像情况，参见图12-24。注意点P位于右摄像机的光轴上。假设由于各种误差，图像坐标x_l比实际值小1%，而图像坐标x_r是准确的。由公式（12-4）算出的深度z的误差多少？以cm为单位。

立体显示 人机交互中为了将3D形状显示给用户，由计算机图形系统生成立体显示效果。图形问题是计算机视觉的逆问题，所有的3D表面点(x, y, z)都是已知的，系统要做的是建立左右图像。根据公式（12-4）可得到公式（12-5），利用目标点坐标(x, y, z)，基线b和焦距f，就可以计算出图像坐标(x_l, y_l)和(x_r, y_r)。因此，已知目标的计算机模型，图形系统就能产生两幅图像。这两幅图像以下列一种方式传递给用户：（a）利用特殊头戴式显示器，将一幅图像送到左眼，另一幅送到右眼；或（b）利用补色，将两幅图像交替显示在CRT上，用户双眼戴不同的滤光镜观看屏幕。如果不需要运动的话，还有廉价的第三种方法，即双眼同时观看打印在单色纸上的并排立体图像对，就会融合出立体景象。（例如，双眼盯看文献Tamimoto（1998）中图12-25的立体对。）

$$x_l = xf/z$$
$$x_r = f(x-b)/z \tag{12-5}$$
$$y_l = y_r = yf/z$$

399

第15章详细讨论了立体显示如何用在虚拟现实系统中。这种系统使用户能自由参与到3D虚拟现实的场景中。同时还可用这种系统，将3D MRI体数据结构呈献在放射专家的眼前。

建立对应关系

立体视觉系统最难的部分不是深度计算，而是确定在深度计算中使用的对应关系。如果对应关系不正确，那么将产生不正确的深度，虽然可能只是一小点偏离，但也可能是完全的错误。本节中，我们主要讨论寻找对应关系的方法和一些有帮助的约束条件。

1. 交叉相关

寻找两幅图像像素间的对应关系，最早用的是第5章中介绍的交叉相关技术。对于已知图像$\mathbf{I}_1$（立体图像对中的第一幅图像）中的点$\mathbf{P}_1$，假设在图像$\mathbf{I}_2$（立体图像对中的的第二幅图像）中存在某个固定区域，在该区域中一定可以找到与$\mathbf{P}_1$对应的点$\mathbf{P}_2$。区域的大小由拍摄这些图像的摄像机设备信息决定。在工业视觉任务中，可以很容易地根据摄像机参数得到这个信息，而摄像机参数又可通过标定过程得到（参见第13章）。在遥感遥测和其他任务中，可能要通过训练图像和地面实际情况来估计这个信息。不论哪种情况，对于图像$\mathbf{I}_1$的像素$\mathbf{P}_1$，搜索$\mathbf{I}_2$上的选定区域，对$\mathbf{P}_1$和$\mathbf{P}_2$的邻域进行交叉相关运算。把交叉相关响应最大的像素，作为$\mathbf{P}_1$的最佳匹配点，并用该像素寻找对应3D点的深度。交叉相关技术已经成功用于寻找卫星和航测图像的对应关系。图12-25显示交叉相关技术。图像$\mathbf{I}_1$中的黑点是需要寻找对应关系的点。图像$\mathbf{I}_2$中的正方形区域是要搜索匹配的区域。

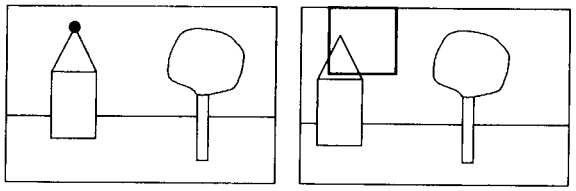


图12-25 利用交叉相关技术寻找立体图像对的对应关系

2. 图符匹配和相关约束

寻找对应关系的第二种常用方法是，在一幅图像中寻找与另一幅图像特征相匹配的特征。典型特征有连接类型、线段或区域。匹配可采用第11章中定义的一致性标记形式。部件集$\mathbf{P}$是第一幅图像$\mathbf{I}_1$中的特征集合。标记集L是第二幅图像$\mathbf{I}_2$的特征集合。如果特征类型多于一种，那么部件的标记类型必须与部件类型相同。（注意一般要避免使用T连接，因为T连接一般是由边与边之间的遮挡引起的，而不是由3D目标的结构引起的。）此外，$\mathbf{P}$上的空间关系R_P要与L上的空间关系R_L相同。如图12-26所示，如果要匹配的特征是连接点，那么对应的连接点应该有相同的类型（一个L连接映射到另一个L连接）。如果在第一幅图像中两个连接由一条线段相连（例如L连接和箭头连接），那么在第二幅图像中，对应的连接之间也由一条线段相连。如果要匹配的特征是线段，那么匹配可利用平行、共线等关系。对于区域匹配，可以使用区域邻接关系。

400

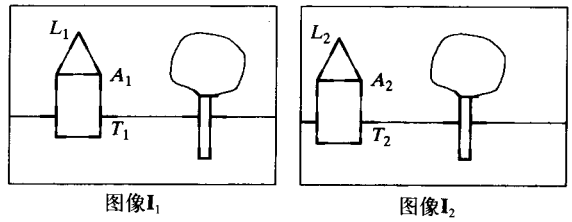


图12-26 利用图符匹配寻找立体图像对的对应关系。图中的L连接和箭头连接是可能的匹配点。一般应避免使用T连接，因为它们通常是遮挡的结果，而不是3D结构的实际特征

这就带来了一个问题，这个问题在任何立体匹配中都会出现。并非第一幅图像中的所有特征都可以在第二幅中检测到。由于视点原因，一些特征可能正好看不到。某些特征在一幅图像中出现，但在另一幅图像中可能被遮挡住。有的特征可能检测不到或者丢失了，而有的无关特征却能检测到。所以不能期望图符匹配过程能够得到完美的一致性标记，而应允许出

现一定的误差，可以寻找一种最小误差映射，或者利用连续松弛法得到近似解。

求出从第一幅图像特征到第二幅图像特征的映射后，任务还没有完成。连接点的对应关系产生的是一个稀疏深度映射，也就是仅在很小的点集上深度是已知的。线段的对应关系可以产生端点或中点间的对应关系。对于区域间的对应关系，还要做其他工作，以确定区域中的哪些点是相对应的。通过在已知的数值之间线性插值，使稀疏深度映射变得稠密。可以想像，这样做会带来很大的误差，也许这就是为什么在实际中，尤其是当图像不是工业场景而是自然场景时，仍然广泛使用交叉相关的原因。

3. 外极线约束

如果已知摄像机的相对方向，则可以大大简化立体匹配过程。对于一幅图像上的已知点，在另一幅图像中寻找它的对应点，这时要进行二维空间搜索。如果知道摄像机的相对方向，就可以利用图像对的外极线几何（epipolar geometry）使搜索在一维空间进行。图12-27显示的是简单情况下的外极线几何情况。两图像面位于同一平面并且与基线平行。已知图像$\mathbf{I}_1$中的点$\mathbf{P}_1 = (x_1, y_1)$，则图像$\mathbf{I}_2$中的对应点$\mathbf{P}_2 = (x_2, y_2)$与$\mathbf{P}_1$位于相同的扫描线上，也就是$y_1 = y_2$。
401 我们称这对图像为标准图像对。

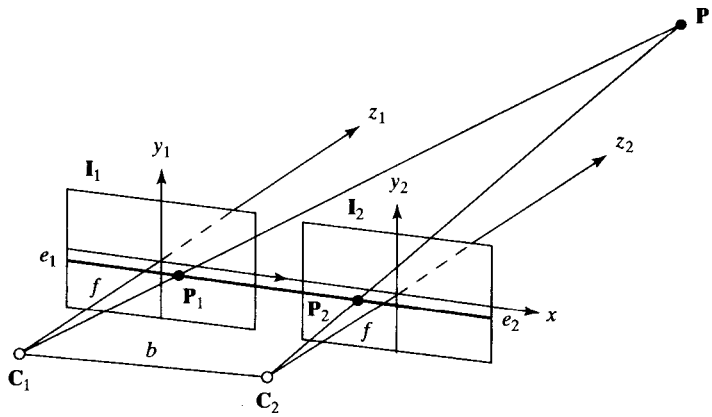


图12-27 标准图像对的外极线几何。3D点$\mathbf{P}$在图像$\mathbf{I}_1$中的投影为$\mathbf{P}_1$，在图像$\mathbf{I}_2$中的投影为$\mathbf{P}_2$，两幅图像位于同一平面，与两摄像机间的基线平行。光轴垂直于基线并互相平行

虽然规定这个标准结构使几何处理变得很简单，但把摄像机这样布置有时是不行的，而且这个结构产生的视差不大，不能据此得出精确的深度信息。一般的立体视觉结构中，摄像机具有随意的位置和姿态，二者要能够观察到目标的主要部分。图12-28显示一般情况下的外
402 极线几何。

定义104 包含3D点$\mathbf{P}$、两个光心（或摄像机）$\mathbf{C}_1$和$\mathbf{C}_2$、以及$\mathbf{P}$在两幅图像中的投影点$\mathbf{P}_1$和$\mathbf{P}_2$的平面称为**外极面**（epipolar plane）。

定义105 外极面与两幅图像平面$\mathbf{I}_1$和$\mathbf{I}_2$的交线e_1和e_2称为**外极线**（epipolar line）。

在图像$\mathbf{I}_1$中，已知外极线e_1上的点$\mathbf{P}_1$和摄像机的相对姿态（参见第13章），就可以找到图像$\mathbf{I}_2$中对应的外极线e_2，在e_2上必然存在对应点$\mathbf{P}_2$。如果在图像$\mathbf{I}_1$中，另一个点$\mathbf{P}_1'$位于与$\mathbf{P}_1$不同的外极面上，那么它也将位于不同的外极线上。

定义106 立体图像对的**外极点**（epipole）就是所有外极线的交点。

点$\mathbf{E}_1$和$\mathbf{E}_2$分别是图像$\mathbf{I}_1$和$\mathbf{I}_2$上的外极点。

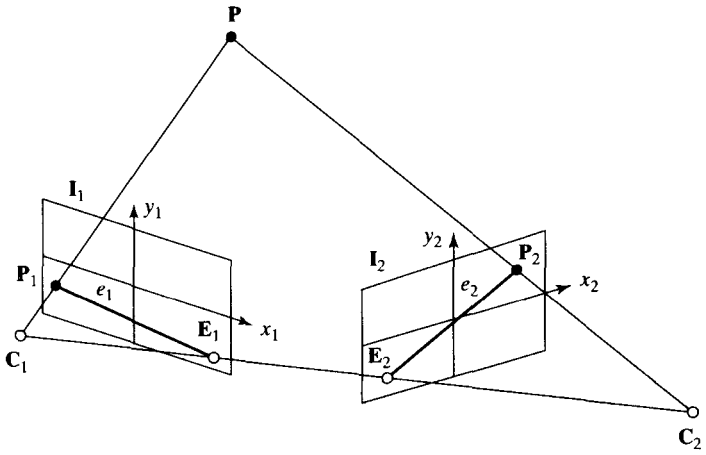


图12-28 一般图像对的外极线几何。3D点$\mathbf{P}$在图像$\mathbf{I}_1$中的投影为$\mathbf{P}_1$，在图像$\mathbf{I}_2$中的投影为$\mathbf{P}_2$，两图像面不在一个平面上。$\mathbf{P}_1$在图像$\mathbf{I}_1$中的外极线是e_1，对应点$\mathbf{P}_2$在图像$\mathbf{I}_2$中的外极线是e_2，$\mathbf{E}_1$是图像$\mathbf{I}_1$的外极点，而$\mathbf{E}_2$是图像$\mathbf{I}_2$的外极点

4. 顺序约束

已知场景中的两个点和它们在两幅图像中的投影点。顺序约束指的是，如果这两点位于场景中的连续表面上，那么在每幅图像中，它们以相同的顺序位于外极线上。这个约束比外极线约束更有意义，因为在进行匹配时，我们并不知道两个图像点对应的3D点是否位于相同的3D表面。该约束有助于寻找可能的匹配，但如果严格应用这个约束，则可能引起对应关系的错误。

5. 误差与场景覆盖

在设计立体视觉系统时，要在场景覆盖与计算深度的误差间求得平衡。如果基线很短，确定图像点$\mathbf{P}_1$和$\mathbf{P}_2$的位置时误差就较小，但在计算3D点$\mathbf{P}$的深度时误差就较大，可以从示意图中推出这个结论。增大基线可以改进搜索精度，但是随着摄像机彼此远离，图像点之间的对应关系可能会丢失，因为遮挡的可能性更大了。建议两摄像机光轴间最好是成45°角。

12.7 薄透镜方程*

薄透镜的工作原理参见图12-29所示。来自目标点$\mathbf{P}$并与光轴平行的光线穿过透镜和焦点$\mathbf{F}_i$到达像点$\mathbf{p}'$。从$\mathbf{P}$出发的其他光线也到达$\mathbf{p}'$，因为透镜具有光线收集器的作用。穿过光心的光线沿直线到达$\mathbf{p}'$。从$\mathbf{p}'$出发并与光轴平行的光线穿过透镜和第二个焦点$\mathbf{F}_j$。

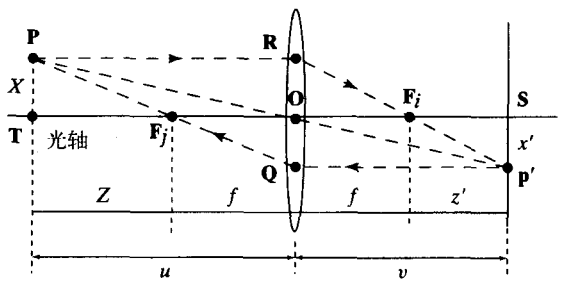


图12-29 薄透镜原理。来自目标点$\mathbf{P}$并与光轴平行的光线穿过透镜和焦点$\mathbf{F}_i$到达像点$\mathbf{p}'$。从$\mathbf{p}'$出发并与光轴平行的光线穿过透镜和第二个焦点$\mathbf{F}_j$

根据图12-29的几何原理，可以推导出薄透镜方程。因为距离X与从$\mathbf{R}$到$\mathbf{O}$的距离相同，由相似三角形$\mathbf{ROF}_i$和$\mathbf{Sp}'\mathbf{F}_i$可得下列公式。

403

$$\frac{X}{f}=\frac{x'}{z'} \tag{12-6}$$

利用相似三角形**POT**和**p'OS**得到第二个公式

$$\frac{X}{f+Z}=\frac{x'}{f+z'} \tag{12-7}$$

把公式（12-6）中X的值代入公式（12-7）得到

$$f^2=Zz' \tag{12-8}$$

用$u-f$代替Z，用$v-f$代替z'得到

$$uv=f(u+v) \tag{12-9}$$

最后在两边除以（uvf），得到最常用的透镜方程，这个形式建立了焦距与物距u和像距v之间的关系。

$$\frac{1}{f}=\frac{1}{u}+\frac{1}{v} \tag{12-10}$$

1. 焦距和景深

如图12-29所示，假设对点**P**的像点进行了调焦，如果成像平面前后移动，像点将变得模糊，如图12-30所示。对v成立的透镜公式，对新像距v'则不成立。同样，如果成像平面不动而点**P**移动，改变了物距u，透镜方程也不成立。在这两种情况中，成像平面上得到的不是清晰的点，而是由点扩展成的直径为b的圆。我们现在要建立该圆大小与摄像机分辨率和景深的关系。

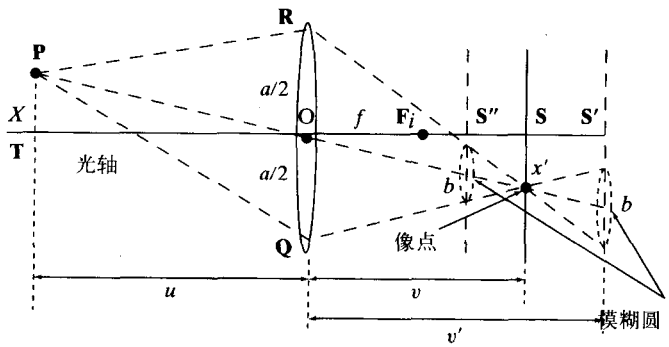


图12-30 如果点的深度或图像平面的位置与透镜方程不符合，点**P**的像点将变得模糊。如果**S**是**P**产生清晰像点的图像位置，那么当图像平面移动到**S'**或**S''**时，**P**的像点将变得模糊，成为直径为b的圆

假设模糊圆的直径b为1个像素时是可以接受的。从这个假设出发，计算在这个模糊限度内，点**P**到摄像机的最近和最远距离。设物距u是要测量的正常深度，v是根据透镜方程得到的理想像距，a是透镜的孔径，f是焦距。

404 具体参见图12-30。在上述条件下能够得到清晰的像点，现在研究要保证模糊直径在b以内，u可被改变多少。

对于图12-30中v'的极端情况，由相似三角形得到

$$\begin{aligned} v'&=\frac{a+b}{a}v \quad \text{对于} \quad v'>v \\ v'&=\frac{a-b}{a}v \quad \text{对于} \quad v'<v \end{aligned} \tag{12-11}$$

注意，根据透镜方程，对于$v'>v$，到摄像机的距离u'比u短；对$v'<v$，u'则比较大。计算最近点u_n，它将产生如图所示直径为b的模糊圆，利用反映u、v和f关系的透镜方程，以及公式（12-11）中$v'>v$时的公式，可以算出最近点u_n。

$$u_n = \frac{fv'}{v'-f} = \frac{f\frac{(a+b)}{a}v}{\frac{(a+b)}{a}v - f} = \frac{f\frac{(a+b)}{a}\frac{uf}{(u-f)}}{\frac{(a+b)}{a}\frac{uf}{(u-f)} - f} = \frac{uf(a+b)}{af+bu} = \frac{u(a+b)}{a+\frac{bu}{f}} \tag{12-12}$$

同样，利用$v'<v$时的公式，重复上面的步骤，可以得到最远平面位置u_r。

$$u_r = \frac{uf(a-b)}{af-bu} = \frac{u(a-b)}{a-\frac{bu}{f}} \tag{12-13}$$

定义107 对于给定的成像参数和模糊限度b，最远平面和最近平面之间的距离就是景深。

405

因为一般情况是$u>f$，从公式（12-12）的最后可以看出$u_n<u$。保持其他条件不变，如果焦距f变短，将使最近点u_n更靠近摄像机。同样可以解释$u_r>u$，并且缩短f将使最远点离摄像机更远。因此，焦距较短的透镜比焦距较长的透镜有更大的景深。（不幸的是，焦距较短的透镜一般径向畸变也较大。）

2. 分辨率与模糊

理想的光学CCD摄像机，如果具有n行像素，在最好的情况下可以分辨出$n/2$条直线，其中相邻的直线间保证有一个像素的间隔。512 × 512的CCD阵列可以检测到256条暗线，这些线被一像素宽的亮像素行分隔开。（如果必要的话，沿垂直光轴的方向轻微移动摄像机，直到图像上的直线图案与像素行对齐。）如果模糊圆的直径大于一个像素，则所有直线会融合到一起形成灰色图像。上面给出的公式使我们可以根据给定的检测问题设计出相应的成像仪器。一旦知道要检测什么特征，以及实际应用要做哪些权衡，就可以确定检测器阵列和镜头。

定义108 摄像机的**分辨力**（resolving power）定义为$R_p = 1/(2\Delta)$，单位为line/in.（或mm），其中Δ是以in.（mm）为单位的像素间距。

例如有正方形CCD阵列，边长10mm，500 × 500像素，那么分辨力是$1/(2 \times 2 \times 10^{-2}$mm/line)，或者25 line/mm。假设黑白胶卷由相隔5×10^{-3}mm的卤化银分子组成，那么分辨力是100 line/mm，或2500 line/in.。人眼中感知颜色的锥状体，紧密排列在一起，形成中央凹，大约间隔$\Delta = 10^{-4}$in.。换算成视网膜上的分辨力，是5×10^3 line/in.。假设人眼直径是20mm = 0.8in.，则可以算出图12-31中的对边角为$\theta \approx \sin(\theta) = 2\Delta/0.8\text{in.} = 2.5 \times 10^{-4}$弧度。这大约是一弧分，意味着人类能够检测到2米

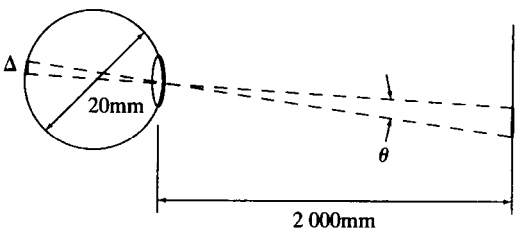


图12-31 人类视网膜上的微小目标图像

外墙上0.5mm宽的铅笔线条。

12.8 总结性讨论

本章研究了2D图像结构和3D表面及目标间的关系，人类协同利用这些关系，感知世界并
406 在其中自由穿行。尽管只对深度和立体视觉与焦距变化的关系进行了详细讨论，但也归纳了很多现象的定量模型，这些现象包括从明暗恢复形状和从纹理恢复形状等。对艺术家而言，这些模型是重要的工具，尤其是运用计算机图形学，通过2D画布或者2D图形表达出3D结构时，就更是如此。在第13章中，将讨论如何利用这些方法，根据2D图像自动识别目标的3D结构。我们要提醒读者，其中有些方法可靠性或者精度太差，不能单独使用，除非是在某些受控环境中使用。用这些算法为户外导航机器人提供实时视觉仍是一个困难的问题，这也是当前比较活跃的研究领域之一。

习题12.26

在文艺著作中找到广场或者雅典卫城的图画，复制一份，标出艺术家作画时用到的消隐点和消隐线。

12.9 参考文献

视觉心理学家J.J. Gibson（1950）的早期著作是研究视觉信息线索的经典之作。很多80年代的计算机视觉研究工作都可以追根于Gibson的著作。很多实验受到David Marr（1982）方法的影响，他的信息处理范例认为，人们首先要与所用的信息隔离，以进行决策或者理解，然后探究数学模型，最后寻找可能的实现。Marr还相信，人类的视觉系统实际上对场景表面能构造出相当完整的描述，这种观点现在已经不盛行了。视觉心理学家Irvin Rock（1983）的著作，回顾了多年来的实验并得出结论：视觉感知需要智能操作，并且需要进行推理。这本书既可作为人类视觉特征的资源手册，也可作为方法论进行研究。Barrow和Tenanbaum在1978年引入了本征图像的概念。他们的提议基本上等同于2-1/2D简图，2-1/2D简图由Marr提出并出现在他1982出版的著作中。本书中关于本征图像的讨论，主要是参考第3章中提到的Charnink和McDermott的著作（1985）。

Huffman（1971）和Clowes（1971）都发现了模块世界的连接约束机制。Waltz于1975年对这一工作进行了推广，能够处理阴影和非三面角。连接类型可以多到数千种，对于人类的自觉推理来说，这个数字太大了，但对计算机不会带来任何困难。Waltz提出一种有效的算法，能够去掉可能的线段标记，通常称为Waltz滤波。Winston于1977年编著的人工智能教材，是模块世界关于形状解释方面内容详细的一本好书，其中还包含如何得到连接类型的内容。

407 本书给出的并行松弛方法，参考了Rosenfeld等人（1976）以及其他研究人员的大量类似工作，部分工作参考了Waltz的研究结果。通过附加几何约束可以防止出现这样的解释：即线条图不可能是由积木世界目标实际成像后形成的，关于这方面的内容请参考Sugihara（1986）。Malik（1987）的论文中，讨论了如何扩展线段和连接标记类型，以处理一大类曲面目标。Stockman等人（1990）的工作，讨论了如何利用稀疏深度样本来重建和解释场景的不完整线条图，其中的场景远比三面角积木世界更具一般性。Haralick和Shapiro（1992/93）的两卷集著作包含了透视变换的大量内容。

在一篇广为人知的Marr和Poggio（1979）的论文中，在信息处理方面讨论了人类的立体视觉系统，并提出了一种类似松弛法的视觉处理方法。Tanimoto（1998）的论文，讨论了如何将立体图用于数学研究，特别包含了一些彩色立体图，人类可通过这些立体图来感知3D形

状。为方便人类观察而建立立体场景，这是当前研究的热点。例如，Peleg和Ben-Ezra（1999）采用单台移动摄像机建立了历史景点的立体场景。自动调焦装置大量存在于商业摄像机市场。很显然，可以采用快速廉价的装置得到场景表面的深度。Krotkov（1987）、Nayar等人（1992）以及Subbarao和Tyan（1998）的工作提供了这一领域的知识背景。

1. Barrow, H., and J. Tenenbaum. 1978. Recovering intrinsic scene characteristics from images. In *Computer Vision Systems,* A. Hansom and E. Riseman, eds. Academic Press, New York.
2. Charniak, E., and D. McDermott. 1985. *Artifical Intelligence.* Addison-Wesley, Reading, MA.
3. Clowes, M. 1971. On seeing things. *Artificial Intelligence,* v. 2:79–116.
4. Gibson, J. J. 1950. *The Perception of the Visual World.* Houghton-Mifflin, Boston.
5. Haralick, R., and L. Shapiro. 1992/3. *Computer and Robot Vision, Volumes I and II.* Addison-Wesley, Reading, MA.
6. Huffman, D. 1971. Impossible objects as nonsense sentences. In *Machine Intelligence,* v. 6, B. Meltzer and D. Michie, eds. Elsevier, New York, 295–323.
7. Kender, J. 1980. *Shape from Texture,* Ph.D. dissertation. Dept. of Computer Science, Carnegie Mellon Univ., Pittsburgh, PA.
8. Koenderink, J. 1984. What does the occluding contour tell us about solid shape? *Perception,* v. 13.
9. Krotkov, E. 1987. Focusing, *Int. J. Comput. Vision,* v. 1:223–237.
10. Malik, J. 1987. Interpreting line drawings of curved objects. *Int. J. Comput. Vision,* v. 1(1).
11. Marr, D., and T. Poggio. 1979. A computational theory of human stereo vision. *Proc. Royal Society,* v. B 207:207–301.
12. Marr, D. 1982. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* W. H. Freeman and Co., New York.
13. Nayar, S. 1992. Shape from focus system. *Proc. Comput. Vision and Pattern Recog.,* Champaign, Illinois (June 1992), 302–308.
14. Peleg, S., and M. Ben-Ezra. 1999. Stereo panorama with a single camera. *Proc. Comput. Vision and Pattern Recog.,* Fort Collins, CO (23–25 June 1999), v. I:395–401.
15. Rock, I. 1983. *The Logic of Perception.* A Bradford Book, MIT Press, Cambridge, MA.
16. Rosenfeld, A., R. Hummel, and S. Zucker. 1976. Scene labeling by relaxation processes. *IEEE Trans.* SMC, v. 6.
17. Stockman, G., G. Lee, and S. W. Chen. 1990. Reconstructing line drawings from wings: the polygonal case. *Proc. of Int. Conf. Comput. Vision 3,* Osaka, Japan.
18. Subbarao, M., and J-K. Tyan. 1998. Selecting the optimal focus measure for autofocusing and depth-from-focus. *IEEE-T-PAMI,* v. 20(8):864–870.
19. Sugihara, K. 1986. *Machine Interpretation of Line Drawings.* MIT Press, Cambridge, MA.
20. Tanimoto, S. 1998. Connecting middle school mathematics to computer vision and pattern recognition. *Int. J. Pattern Recog. and Artificial Intelligence,* v. 12(8):1053–1070.
21. Waltz, D. 1975. Understanding line drawings of scenes with shadows. In *The Psychology of Computer Vision,* P. Winston, ed. McGraw-Hill, New York, 19–91.
22. Winston, P. 1977. *Artificial Intelligence.* Addison-Wesley, Reading, MA.

408

409

第13章　3D感知与目标位姿计算

本章主要关心2D图像结构与3D目标结构之间的定量关系。上一章主要讨论了图像和现实之间的定性关系，本章我们将研究如何进行视觉测量与计算，这些测量与计算在3D目标识别与检测以及机器人操作与导航中都要用到。

举个例子，请参考图13-1，为了设计出更好的驾驶室环境，需要对驾驶员的开车姿势进行测量。图13-2显示的是另一种应用场合，为了让机器人能够抓起零件，视觉系统要先识别出3D零件并确定零件的位姿。这时，相机系统和机械臂要在3D世界坐标系下进行信息交互。





图13-1　驾驶室安装了4个摄像机，这副幅图是其中两个摄像机拍摄的。多摄像机测量系统用来计算身体上一些点的3D位置（图中用椭圆标记的地方），根据这些点的3D位置就可算出人体姿势（图片由密歇根州立大学人类工程学实验室罗伯特·瑞纳德提供）

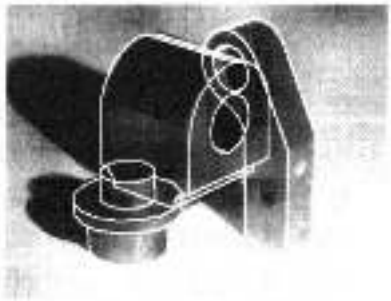


图13-2　线框图覆盖在3D目标的图像上面。计算机视觉用于识别目标并确定目标位置。识别系统把2D零件图像与3D零件模型进行匹配，并根据模型计算要得到这幅图像所需的3D几何变换。然后再把每个零件的标识与位姿信息反馈到机器人控制器（图像由Mauro Costa提供）

本章讨论3D感知中的一些工程学和数学问题。先通过几何分析对问题进行简单说明，然后推导出数学模型。数学上主要是关于3D变换的代数运算。另外还要介绍3D模型的作用，不同传感器的配置以及传感器的标定过程。

13.1　一般体视结构

图13-3是常见的立体视觉系统，两个摄像机同时观察同一个工作区，在计算机图形学中，常常采用右手坐标系统，z轴的负方向由摄像机向外，这样距离摄像机较远的点，其深度坐标的负值就较大。在本章的多数模型中，我们采用正深度坐标，但有时候使用另一套坐标系统，主要是为了和文献出处保持一致。图13-3是常见的立体视觉结构，不需要12章中对两台摄像机安装位置提出的特殊要求。两台摄像机观察工作台上相同的工件区，这时工作台就是一个完整的3D世界，并且有自己的世界坐标系$\mathbf{W}$。可以直观地看到，工作区中3D点$^{W}\mathbf{P} = [^{W}P_x\ ^{W}P_y\ ^{W}P_z]$的位置，可通过两条投影线$^{W}\mathbf{P}^{1}\mathbf{O}$和$^{W}\mathbf{P}^{2}\mathbf{O}$的交点确定。在13.3.3节中，给出计算交点的数学推导过程，计算方法很简单，但测量误差会使问题变得复杂。

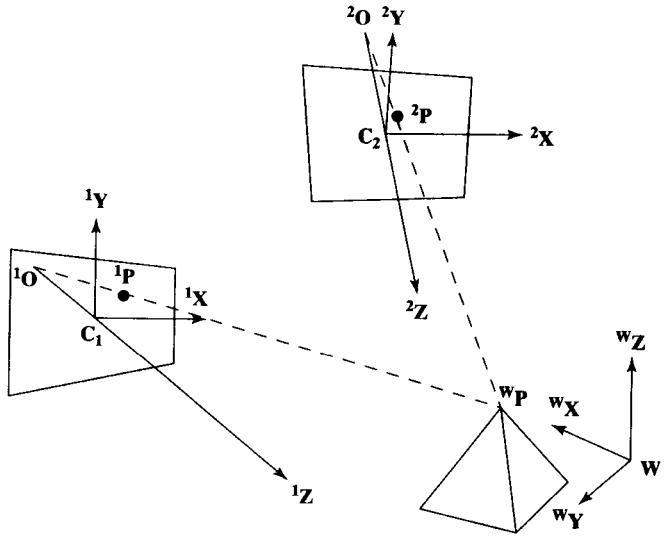


图13-3 两摄像机C_1和C_2观测相同的3D工作区。工件上的点**P**在第一幅图中的投影为^1**P**，在第二幅图中的投影为^2**P**

为了进行图13-3所示的立体视觉计算，需要已知下列条件：

- 首先要知道摄像机C_1在工作区**W**中的位姿，以及摄像机的一些内部参数，如焦距。这些信息用*摄像机矩阵*（camera matrix）来表示，对每一个图像点^1**P**通过该矩阵确定了3D空间中的一条光线。利用第13.3节和13.7节介绍的摄像机标定过程可以得到这些信息。
- 同样要知道摄像机C_2在工作区**W**中的位姿以及它的内部参数，也就是需要它的摄像机矩阵。
- 要找出3D点与两个2D图像点（^{w}**P**, ^{1}**P**, ^{2}**P**）之间的对应关系。
- 要有公式来计算两条投影线^w**P**^{1}**O**和^w**P**^{2}**O**的交点^w**P**。

在讨论这些条件之前，对于图13-3所示的视觉系统，我们先介绍配置上的三种重要情况。

- **图13-3中包括两台摄像机，它们在世界坐标系中的位置要进行标定。**通过计算两对应图像点的投影线的交点，得到3D点的坐标。
- **其中一台摄像机可用投影仪代替。**投影仪通过一束光照亮一个或更多的表面点，或者投射特殊图案如交叉十字线，如图13-4所示。后面我们将会看到，投影仪的标定方式和摄像机的标定情况非常相似。发出的光线与到摄像机的投影线有相同的代数表达方式。当一个表面上没有明显的特征，需要对表面上的点进行测量时，使用投影仪就有很多优点。
- **目标的模型知识可以取代一台摄像机。**假设图13-3中的锥形物高度已知，即^wP_z已知，也就是说点**P**被限制在平面$z = {}^wP_z$上。通过计算来自摄像机C_1的投影线与该平面的交点，就很容易地算出其他两个坐标。很多情况下模型信息会带来足够的约束条件，这时只用一台摄像机就够了。

13.2 3D仿射变换

2D空间的仿射变换已经在第11章讲过，这一章把它扩展到3D空间。这些变换不仅对3D机器视觉来说非常重要，而且对于机器人学和虚拟现实也非常重要。基本变换是平移、旋转、缩放和剪切。这些基本变换可以明确表示出来。而有一些变换就很难明确表示。为了方便我

们仍然采用齐次坐标，把3D点$[P_x, P_y, P_z]$表示为$[sP_x, sP_y, sP_z, s]$，s是非零的比例系数。（像前面一样，点的坐标竖直排列，但在不引起歧义的情况下我们会省去转置符号。）本章在表示一个点时，经常要用到脚标，因为命名的坐标系比11章更多。关于变换，要增加从3D空间到2D空间的透视、正交、弱透视投影等内容。

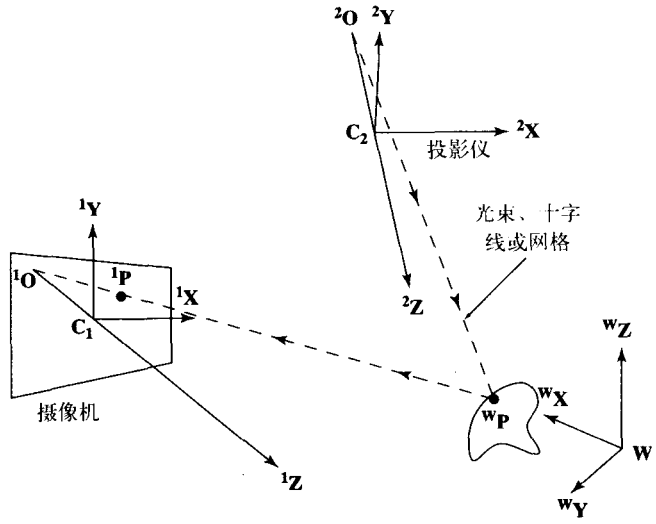


图13-4 在一般体视系统中，用投影仪代替一台摄像机。与图13-3具有相同的几何和代数约束。但投影仪能够为无特征的表面带来表面特征

13.2.1 坐标系

为了定量地确定点在空间中的位置，需要定义*坐标系*（coordinate frame或coordinate system）。图13-5是一个场景四个不同的相对坐标系，锥形物顶点**P**有四种不同的坐标表示方式。首先，点**P**在CAD模型中表示为 ${}^{M}\mathbf{P} = [{}^{M}P_x, {}^{M}P_y, {}^{M}P_z] = [b/2, b/2, \frac{\sqrt{2}}{2}b]$，其中$b$是底边长度，这个CAD模型的位姿就是图中所示的情况。其次，在工作台坐标系中锥形物顶点**P**的坐标为： 413

$$ {}^{W}\mathbf{P} = [{}^{W}P_x, {}^{W}P_y, {}^{W}P_z] = \mathbf{TR}\left[\frac{b}{2}, \frac{b}{2}, \frac{\sqrt{2}}{2}b\right], \tag{13-1} $$

其中**TR**是坐标系**M**相对坐标系**W**的旋转与平移的组合变换。最后，如果两个传感器**C**和传感器**D**（或者是人）从工作台的相对方向观察锥形物，点**P**和点**Q**之间的左右关系就是相反的，它们的坐标值是不同的，即${}^{C}\mathbf{P} = [{}^{C}P_x, {}^{C}P_y, {}^{C}P_z] \neq {}^{D}\mathbf{P} = [{}^{D}P_x, {}^{D}P_y, {}^{D}P_z]$。

为了使传感器之间、传感器与场景中3D物体之间、传感器与机械手之间联系起来，我们用数学方法推导各坐标系之间的关系。用同样的方法能够建立空间中物体的运动模型。为了方便，我们用符号来表示坐标点对应的坐标系以及坐标变换的方向。用${}^{W}_{M}\mathbf{T}$表示把模型坐标点${}^{M}\mathbf{P}$变换成工作台坐标点${}^{W}\mathbf{P}$的变换，具体如下：

$$ {}^{W}\mathbf{P} = {}^{W}_{M}\mathbf{T}\,{}^{M}\mathbf{P} \tag{13-2} $$ 414

这种表示方法是Craig在1986年的机器人学教材中采用的，在推导目标运动或者进行目标匹配时，这种方法特别有用。在坐标系一目了然的情况下，我们使用简单的表示方法。下面我们继续学习变换。

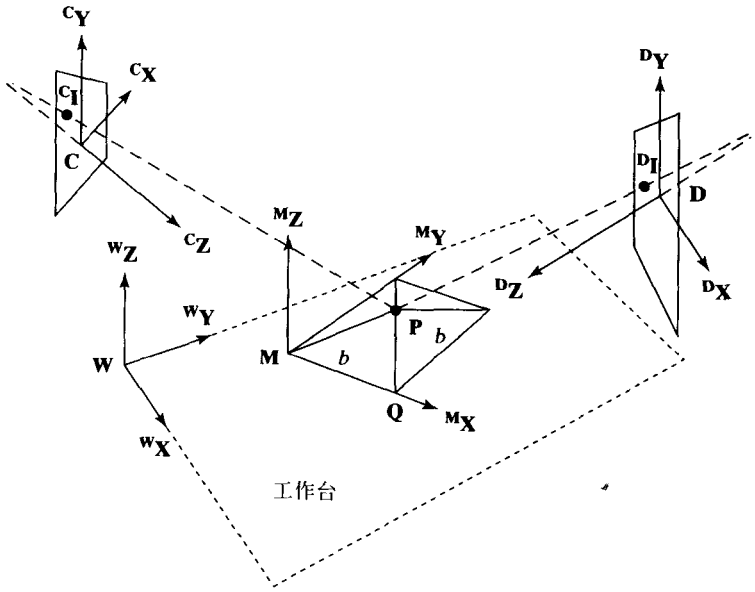


图13-5 点**P**的坐标可表示在四个不同的坐标系中。(1) 模型坐标系**M**，(2) 世界或工作台坐标系**W**，(3) 传感器**C**和传感器**D**的坐标系。随着坐标系的不同点**P**的坐标也不同，例如在传感器坐标系**C**下，点**P**在点**Q**的左边，而在传感器坐标系**D**下，点**P**在点**Q**的右边

13.2.2 平移

对坐标系1中的点${}^1\mathbf{P}$的三个坐标x_0、y_0和z_0，加上一个平移向量就得到坐标系2中的点${}^2\mathbf{P}$。在图13-5的例子中，为了将模型坐标点与它在工作台上的位姿联系起来，需要做平移（与旋转）变换。

$$
{}^2\mathbf{P} = \mathbf{T}(x_0, y_0, z_0)\,{}^1\mathbf{P}
$$

$$
{}^2\mathbf{P} = \begin{bmatrix} {}^2P_x \\ {}^2P_y \\ {}^2P_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & x_0 \\ 0 & 1 & 0 & y_0 \\ 0 & 0 & 1 & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^1P_x \\ {}^1P_y \\ {}^1P_z \\ 1 \end{bmatrix} \tag{13-3}
$$

13.2.3 缩放

3D缩放矩阵能够对每一个坐标采用不同的比例系数。有时所有的比例系数是相同的，如度量单位改变时，或者把模型初始化为一定尺寸时采用同比例缩放。

$$
{}^2\mathbf{P} = \mathbf{S}\,{}^1\mathbf{P} = \mathbf{S}(s_x, s_y, s_z)\,{}^1\mathbf{P}
$$

$$
\begin{bmatrix} {}^2P_x \\ {}^2P_y \\ {}^2P_z \\ 1 \end{bmatrix} = \begin{bmatrix} s_x\,{}^2P_x \\ s_y\,{}^2P_y \\ s_z\,{}^2P_z \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^1P_x \\ {}^1P_y \\ {}^1P_z \\ 1 \end{bmatrix} \tag{13-4}
$$

13.2.4 旋转

通过矩阵来表示绕坐标轴的基本旋转特别容易。我们需要做的就是写出矩阵的列向量，也就是旋转变换下单位向量的变换值。(回想一下，任何3D线性变换，都完全可以通过三个基

向量的一系列变换表示。）绕z轴的变换实际上与11章的2D变换一样，只不过这时公式中包含着3D点的z坐标。图13-6显示基本旋转下基向量的变换情况。 415

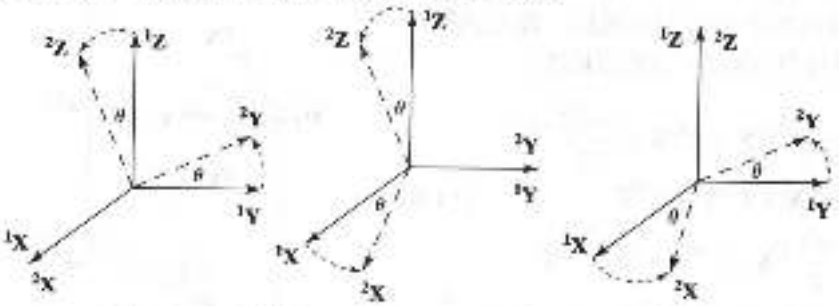


图13-6 分别绕X轴（左）、Y轴（中）和Z轴（右）旋转θ角

绕X轴旋转θ角：

$$
{}^{2}\mathbf{P} = \mathbf{R}({}^{1}X, \theta)\,{}^{1}\mathbf{P}
$$

$$
\begin{bmatrix} {}^{2}P_x \\ {}^{2}P_y \\ {}^{2}P_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{1}P_x \\ {}^{1}P_y \\ {}^{1}P_z \\ 1 \end{bmatrix} \tag{13-5}
$$

绕Y轴旋转θ角：

$$
{}^{2}\mathbf{P} = \mathbf{R}({}^{1}Y, \theta)\,{}^{1}\mathbf{P}
$$

$$
\begin{bmatrix} {}^{2}P_x \\ {}^{2}P_y \\ {}^{2}P_z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{1}P_x \\ {}^{1}P_y \\ {}^{1}P_z \\ 1 \end{bmatrix} \tag{13-6}
$$

绕Z轴旋转θ角：

$$
{}^{2}\mathbf{P} = \mathbf{R}({}^{1}Z, \theta)\,{}^{1}\mathbf{P}
$$

$$
\begin{bmatrix} {}^{2}P_x \\ {}^{2}P_y \\ {}^{2}P_z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{1}P_x \\ {}^{1}P_y \\ {}^{1}P_z \\ 1 \end{bmatrix} \tag{13-7}
$$

习题13.1

证明三个基本旋转矩阵的各列是标准正交向量，再证明行向量也是标准正交向量。 416

习题13.2

求旋转矩阵，以原点到点[1, 1, 0]'之间的直线为轴，逆时针旋转π/4角度。

习题13.3

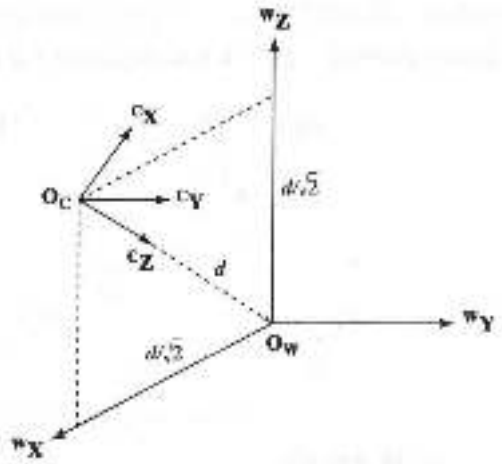
给定旋转角θ弧度以及坐标轴的方向余弦$[c_x, c_y, c_z]'$，如何建立旋转矩阵？

例题：世界坐标系**W**经变换得到摄像机坐标系**C**（见下页右图），推导旋转与平移的组合

变换矩阵。

为了得到旋转矩阵**R**，根据**C**中的基向量坐标，我们写出**W**中的基向量坐标，反过来**W**中的坐标点就可以变换成**C**中的坐标点。

$$
\begin{aligned}
{}^{W}\mathbf{X} &= \frac{-\sqrt{2}}{2}\,{}^{C}\mathbf{X} + 0\,{}^{C}\mathbf{Y} + \frac{-\sqrt{2}}{2}\,{}^{C}\mathbf{Z} \\
{}^{W}\mathbf{Y} &= 0\,{}^{C}\mathbf{X} + 1\,{}^{C}\mathbf{Y} + 0\,{}^{C}\mathbf{Z} \\
{}^{W}\mathbf{Z} &= \frac{\sqrt{2}}{2}\,{}^{C}\mathbf{X} + 0\,{}^{C}\mathbf{Y} + \frac{-\sqrt{2}}{2}\,{}^{C}\mathbf{Z}
\end{aligned}
\tag{13-8}
$$



这三个向量是旋转矩阵的三个列，表示坐标系**C**相对于坐标系**W**的方向。一H旋转摄像机，世界坐标系中的点就沿z轴平移d，以使世界坐标系的原点在**C**坐标系中的坐标为[0,0,d]'。最终的坐标转换为：

417

$$
{}^{C}_{W}\mathbf{TR} = \begin{bmatrix} \frac{-\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{-\sqrt{2}}{2} & 0 & \frac{-\sqrt{2}}{2} & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13-9}
$$

检验 ${}^{C}_{W}\mathbf{TR}\,{}^{W}\mathbf{O}_{W} = {}^{C}_{W}\mathbf{TR}[0,0,0,1]' = [0,0,d,1]' = {}^{C}\mathbf{O}_{W}$，以及 ${}^{C}_{W}\mathbf{TR}\,{}^{W}\mathbf{O}_{c} = {}^{C}_{W}\mathbf{TR}\left[d\frac{\sqrt{2}}{2},0,d\frac{\sqrt{2}}{2},1\right]' = [0,0,0,1]' = {}^{C}\mathbf{O}_{c}$是成立的。

习题13.4

考虑上一个例子的环境，将一个立方体放在世界坐标的原点$\mathbf{O}_W$处。将它的角点K_i变换成摄像机坐标系下的坐标。通过计算$\|K_i - K_j\|$，证明摄像机坐标系下有四条边具有单位长度。

13.2.5 任意旋转

任何旋转都可以表示成公式（13-10）的形式。系数r_{ij}组成的矩阵是一个标准正交矩阵，所有的行和列都是相互正交的单位向量。前面所有的基本旋转矩阵都具有这样的特性。3D空间的任何刚体旋转都可以表示成围绕轴**A**旋转一定的角度θ。**A**不一定是坐标轴，它可以是3D空间中的任意轴。为了弄明白这一点，假设要把基向量${}^{1}\mathbf{X}$变换为不同坐标系下的向量${}^{2}\mathbf{X}$。旋转轴**A**由${}^{1}\mathbf{X}$和${}^{2}\mathbf{X}$的叉积得到，如果${}^{1}\mathbf{X}$是旋转不变的，那么它本身就是旋转轴。

$$
{}^{2}\mathbf{P} = \mathbf{R}(A,\theta)\,{}^{1}\mathbf{P}
$$

$$
\begin{bmatrix} {}^{2}P_x \\ {}^{2}P_y \\ {}^{2}P_z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{1}P_x \\ {}^{1}P_y \\ {}^{1}P_z \\ 1 \end{bmatrix} \tag{13-10}
$$

因此，刚体从时刻t_1到时刻t_2的运动结果，就可以用一个平移向量和一个旋转矩阵表示，而与它在这段时间内的真正轨迹无关。一个齐次矩阵能够同时包含平移和旋转，其中有6个参数：3个旋转参数和3个平移参数。

$$\begin{bmatrix} {}^2P_x \\ {}^2P_y \\ {}^2P_z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^1P_x \\ {}^1P_y \\ {}^1P_z \\ 1 \end{bmatrix} \tag{13-11}$$

418

习题13.5

参考图13-7，求齐次变换矩阵，它把位于原点的立方体的所有角点都映射成另一个立方体的对应角点。假设角点${}^1\mathbf{O}$映射到${}^2\mathbf{O}$，${}^1\mathbf{P}$映射到${}^2\mathbf{P}$，并且是刚性变换。

习题13.6 旋转矩阵的逆

证明旋转矩阵总是可逆的，求公式（13-10）中旋转矩阵$\mathbf{R}(A, \theta)$的逆。

13.2.6 基于变换的比对

这里讨论如何比对模型三角形与拍摄到的三角形。这个例子只有一定的说服力，它通过一步步的变换演算得出结果。更重要的是，它通过模型三角形的3顶点与拍摄的三角形的3顶点对齐，提供比对任何刚体模型的基本方法。以学过的基本变换为基础，通过代数运算就可以得出结果。图13-8和图13-9借助几何图形形象地演示了这个变换过程。

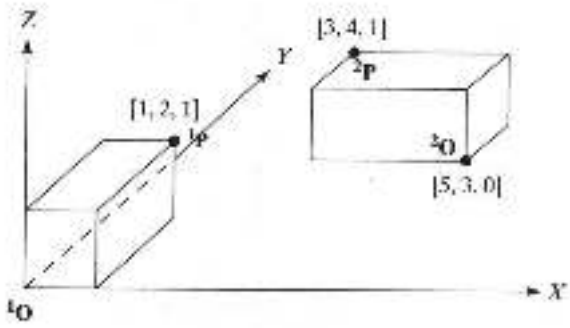


图13-7 同一立方体的两个视图

问题是如何得到变换${}^{\mathrm{M}}_{\mathrm{W}}\mathbf{T}$，使三角模型的顶点A、B和C映射成工作区中与之相合的三角形顶点D、E和F。为了做到这一点，我们同时对这两个三角形进行变换，使AB边与${}^{\mathrm{W}}$X轴重合，并使整个三角落在W坐标系中的X－Y平面内。这时A与D、B与E、C与F的坐标一定是相同的。对变换结果进行整理，就能得出我们要求的变换${}^{\mathrm{M}}_{\mathrm{W}}\mathbf{T}$，它把坐标点${}^{\mathrm{M}}\mathbf{P}_i$映射成对应的坐标点${}^{\mathrm{W}}\mathbf{P}_i$。对这个过程进行归纳形成算法13.1，最后易见，每一步都是可行的，而且都是可逆的，但实现算法需要认真编程，仔细设计数据结构。这是一个比较重要的方法，理论上任何两个相合的刚体，都能通过比对三个对应点而使两个刚体对齐。事实上，当比对远离△ABC的那些点时，测量和计算误差有可能带来很大的偏差，因此常常要用到更多的点，采用最优化的计算方法。

419

420

算法13.1 计算刚体变换${}^{\mathrm{M}}_{\mathrm{W}}\mathbf{T}$，使模型点A、B、C与实际点D、E、F对齐

1. 输入3D模型的三个点A、B、C和对应的3D实际点D、E、F。
2. 求平移变换${}^{\mathrm{M}}_{\mathrm{W}}\mathbf{T}_1$，移动三个模型点，使点A与世界坐标系原点重合。求平移变换${}^{\mathrm{W}}_{\mathrm{W}}\mathbf{T}_2$，移动三个实际点，使点D与世界坐标系原点重合。这时在W坐标系中只对齐了点A和点D。
3. 求旋转变换${}^{\mathrm{M}}_{\mathrm{W}}\mathbf{R}_1$，使AB边与X轴重合。求旋转变换${}^{\mathrm{W}}_{\mathrm{W}}\mathbf{R}_2$，使DE边与X轴重合。这时在W坐标系中对齐了AB边和DE边。
4. 求绕X轴的旋转变换${}^{\mathrm{M}}_{\mathrm{W}}\mathbf{R}_3$，使点C落进X-Y平面。求绕X轴的旋转变换${}^{\mathrm{W}}_{\mathrm{W}}\mathbf{R}_4$，使点F落进X-Y平面。现在在W坐标系中三个点都已经对齐。

5. 现在模型三角形和实际三角形重合在一起，用公式表示如下：

$$ {}^{W}_{W}\mathbf{R}_3\ {}^{W}_{W}\mathbf{R}_1\ {}^{W}_{M}\mathbf{T}_1\ {}^{M}\mathbf{P}_i = {}^{W}_{W}\mathbf{R}_4\ {}^{W}_{W}\mathbf{R}_2\ {}^{W}_{W}\mathbf{T}_2\ {}^{W}\mathbf{P}_i \tag{13-12} $$

$$ {}^{W}\mathbf{P}_i = (\mathbf{T}_2^{-1}\,\mathbf{R}_2^{-1}\,\mathbf{R}_4^{-1}\,\mathbf{R}_3\,\mathbf{R}_1\,\mathbf{T}_1)\,{}^{M}\mathbf{P}_i \tag{13-13} $$

6. 返回 $ {}^{W}_{M}\mathbf{T} = (\mathbf{T}_2^{-1}\,\mathbf{R}_2^{-1}\,\mathbf{R}_4^{-1}\,\mathbf{R}_3\,\mathbf{R}_1\,\mathbf{T}_1) $

421

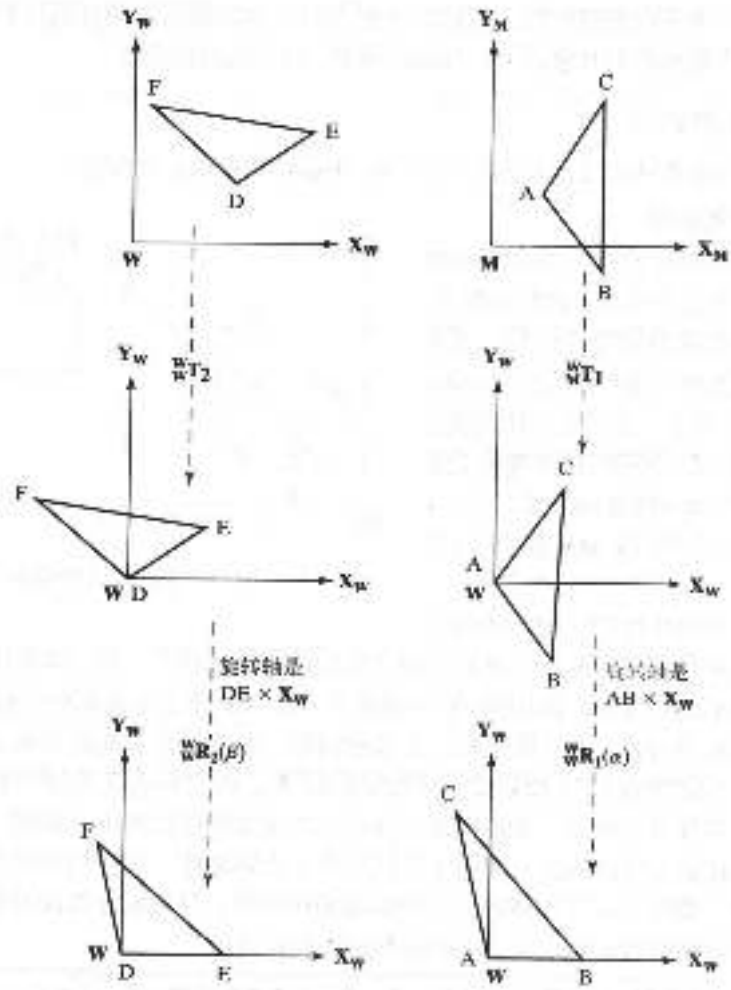


图13-8 （Part I）三角形比对。△ABC是模型三角形，而△DEF是扫描的三角形。首先，对两三角形进行平移，使A和D与原点重合，然后，通过旋转使线段AB和DE与X轴重合

习题13.7 平移矩阵的逆

证明平移矩阵总是可逆的。公式（13-3）中平移矩阵$\mathbf{T}(t_x, t_y, t_z)$的逆是什么？

习题13.8

可以看出，如果$\|A-B\| \neq \|D-E\|$，则算法13.1就会失败，请加入合适的测试方法和误差返

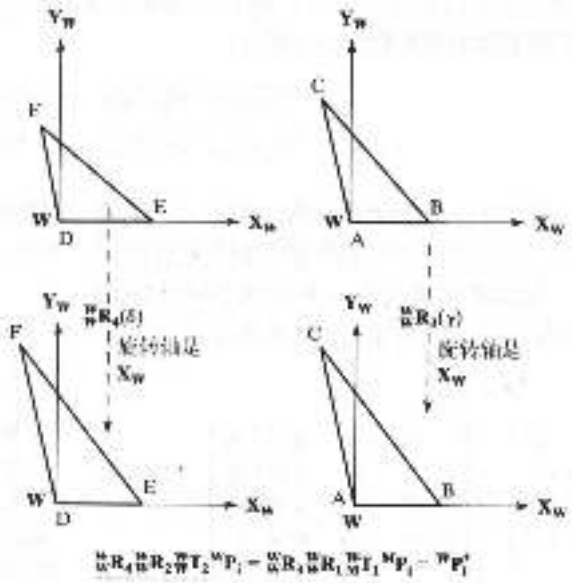
回，解决这种三角形不相合的情况。

习题13.9 编写三角形比对程序*

对算法13.1，编写程序并进行测试，比对模型三角形与世界坐标系下的相合三角形，对每个基本运算都用单独的函数实现。

习题13.10

（a）证明算法13.1返回的变换矩阵，把模型点A映射到实际点D，把模型点B映射到实际点E，把模型点C映射到实际点F。（b）证明在变换过程中，模型上的其他点与A、B、C之间的距离保持不变。（c）证明，如果通过刚体变换，具有n个顶点的刚性多面体模型能够与实际目标对齐，那么利用算法13.1只比对两个三角形就可以得到这个转换。



422

图13-9 （Part II）三角形比对（接图13-8）。对两个三角形做旋转，旋转轴是X轴，三角形最终落在X-Y平面内，AC、DF与X-Y平面的夹角决定了旋转角的大小

13.3 摄像机模型

这一节我们将会看到，公式（13-14）中的摄像机模型C，是较合适的透视成像代数模型，还会看到对于固定摄像装置如何确定矩阵中的元素，然后计算机利用这些矩阵元素进行3D测量计算。

$$^I\mathbf{P} = {}^I_W\mathbf{C}\,{}^W\mathbf{P} \tag{13-14}$$

$$\begin{bmatrix} s\,{}^IP_r \\ s\,{}^IP_c \\ s \end{bmatrix} = {}^I_W\mathbf{C}\begin{bmatrix} {}^WP_x \\ {}^WP_y \\ {}^WP_z \\ 1 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & 1 \end{bmatrix}\begin{bmatrix} {}^WP_x \\ {}^WP_y \\ {}^WP_z \\ 1 \end{bmatrix}$$

$${}^IP_r = \frac{[c_{11}\ c_{12}\ c_{13}\ c_{14}] \circ [{}^WP_x\ {}^WP_y\ {}^WP_z\ 1]}{[c_{31}\ c_{32}\ c_{33}\ 1] \circ [{}^WP_x\ {}^WP_y\ {}^WP_z\ 1]}$$

$${}^IP_c = \frac{[c_{21}\ c_{22}\ c_{23}\ c_{24}] \circ [{}^WP_x\ {}^WP_y\ {}^WP_z\ 1]}{[c_{31}\ c_{32}\ c_{33}\ 1] \circ [{}^WP_x\ {}^WP_y\ {}^WP_z\ 1]}$$

下一步证明这个3×4的摄像机矩阵${}^I_W\mathbf{C}_{3\times4}$表示透视成像变换，它把实际的3D点${}^W\mathbf{P} = [{}^WP_x\ {}^WP_y\ {}^WP_z]^t$投影成图像点${}^I\mathbf{P} = [{}^IP_r\ {}^IP_c]^t$。这个矩阵模型有足够的参数，可以作为世界坐标系W和摄像机坐标系C之间的坐标变换模型，以及透视变换和实际图像坐标到图像阵列行列坐标的缩放变换模型。该矩阵方程采用齐次坐标形式。从公式（13-14）可以看出比例系数s的点积形式。下面讨论如何求取摄像机矩阵${}^I_W\mathbf{C}$的参数。

13.3.1 透视变换矩阵

第12章中已经给出过透视变换的代数表达式，把结果改写为公式（13-15）的形式。这些公式是在世界坐标和摄像机坐标单位一致的情况下推出的。另外，图像坐标${}^IP_r$、${}^IP_c$的单位

与3D空间坐标的单位一样，而不是用像素坐标单位。（上标F表示浮点数，而不是焦距，在透视变换矩阵中用参数f表示焦距。）

$$
\begin{aligned}
{}^{F}P_x/f &= {}^{C}P_x/{}^{W}P_z \quad or \quad {}^{F}P_x = \left(f/{}^{C}P_z\right){}^{C}P_x \\
{}^{F}P_y/f &= {}^{C}P_y/{}^{W}P_z \quad or \quad {}^{F}P_y = \left(f/{}^{C}P_z\right){}^{C}P_y
\end{aligned}
\tag{13-15}
$$

图13-10表示纯透视变换，仅有一个参数f，即焦距。公式（13-16）中的矩阵${}^{F}_{C}\prod(f)$是4×4
423 形式，这样可以和其他的变换矩阵进行合成。不过，第三行的${}^{F}P_z = f$不是必需的，最终将被忽略，也常常不写出来。注意这个矩阵的秩是3而不是4，所以它的逆是不存在的。

$$
{}^{F}\mathbf{P} = {}^{F}_{C}\mathbf{\Pi}(f)\,{}^{C}\mathbf{P}
$$

$$
\begin{bmatrix} s\,{}^{F}P_x \\ s\,{}^{F}P_y \\ s\,{}^{F}P_z \\ s \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \begin{bmatrix} {}^{C}P_x \\ {}^{C}P_y \\ {}^{C}P_z \\ 1 \end{bmatrix} \tag{13-16}
$$

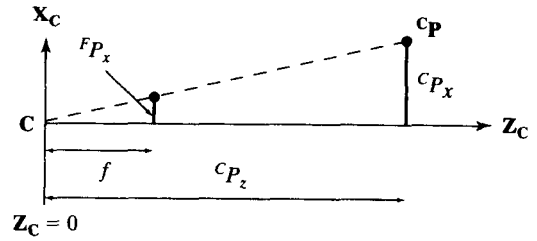


图13-10 摄像机坐标系的原点在投影中心，恒有${}^{F}P_z = f$

另一种透视变换是把摄像机坐标系的原点放在图像中心，使${}^{F}P_z = 0$，如图13-11所示。投影矩阵如公式（13-17）所示。（该公式的优点在于，当f→∞时就得到正投影。）

$$
{}^{F}\mathbf{P} = {}^{F}_{C}\mathbf{\Pi}(f)\,{}^{C}\mathbf{P}
$$

$$
\begin{bmatrix} s\,{}^{F}P_x \\ s\,{}^{F}P_y \\ s\,{}^{F}P_z \\ s \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1/f & 1 \end{bmatrix} \begin{bmatrix} {}^{C}P_x \\ {}^{C}P_y \\ {}^{C}P_z \\ 1 \end{bmatrix} \tag{13-17}
$$

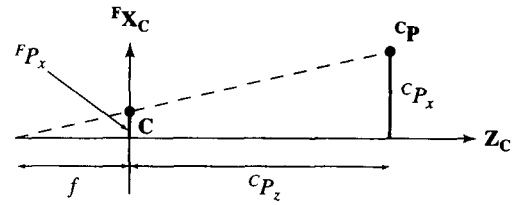


图13-11 摄像机坐标系的原点在图像中心，恒有${}^{F}P_z = 0$

像图13-3所示的一般情况，世界坐标系**W**与摄像机坐标系**C**不一致。需要经过旋转和平移把世界坐标点${}^{W}\mathbf{P}$变换成摄像机坐标点${}^{C}\mathbf{P}$。需要知道三个旋转参数和三个平移参数，但它们以复杂的方式结合在一起而构成变换矩阵的元素，从前面几节的内容也能够看到这一点。

$$
\begin{aligned}
{}^{C}\mathbf{P} &= \mathbf{T}(t_x, t_y, t_z)\mathbf{R}(\alpha, \beta, \gamma)\,{}^{W}\mathbf{P} \\
{}^{C}\mathbf{P} &= {}^{C}_{W}\mathbf{TR}(\alpha, \beta, \gamma, t_x, t_y, t_z)\,{}^{W}\mathbf{P}
\end{aligned}
$$

$$
\begin{bmatrix} {}^{c}P_x \\ {}^{c}P_y \\ {}^{c}P_z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{w}P_x \\ {}^{w}P_y \\ {}^{w}P_z \\ 1 \end{bmatrix} \tag{13-18}
$$

将上面的变换组合起来，从而得到从**W**到**C**，然后经透视变换由${}^{C}\mathbf{P}$得到图像平面上的${}^{F}\mathbf{P}$这
424 个过程的变换模型。忽略矩阵中的第三行，因为${}^{F}P_z$只是个常量。${}^{F}\mathbf{P}$位于实际图像平面，通过缩放变换可以得到行列像素坐标${}^{I}\mathbf{P}$。请复习线性代数中有关矩阵乘法方面的知识。

$$
\begin{aligned}
{}^{F}\mathbf{P} &= {}^{F}_{C}\mathbf{\Pi}(f)\,{}^{C}\mathbf{P} \\
&= {}^{F}_{C}\mathbf{\Pi}(f)\left({}^{C}_{W}\mathbf{TR}(\alpha, \beta, \gamma, t_x, t_y, t_z)\,{}^{W}\mathbf{P}\right) \\
&= \left({}^{F}_{C}\mathbf{\Pi}(f){}^{C}_{W}\mathbf{TR}(\alpha, \beta, \gamma, t_x, t_y, t_z)\right)\,{}^{W}\mathbf{P}
\end{aligned}
$$

$$\begin{bmatrix} s & {}^{F}P_x \\ s & {}^{F}P_y \\ s & \end{bmatrix} = \begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & 1 \end{bmatrix} \begin{bmatrix} {}^{W}P_x \\ {}^{W}P_y \\ {}^{W}P_z \\ 1 \end{bmatrix} \tag{13-19}$$

公式（13-19）中的矩阵用d_{ij}表示元素，而不是用c_{ij}，因为它不是我们想要的摄像机矩阵。到目前为止所有的推导采用的都是实际长度单位，如毫米或英寸，并不包括到行列像素点的缩放变换。把毫米到行列像素转换的变换系数加入公式（13-19），很容易得到完全摄像机矩阵$\mathbf{C}$。假设实值单位的像素横坐标是d_x，纵坐标是d_y，实值坐标$[{}^{F}P_x, {}^{F}P_y]$的参考坐标系的原点[0.0, 0.0]位于图像的左下角，下一步我们想用整值坐标$[r, c]$代替实值坐标，$[r, c]$表示图像阵列的像素行列坐标，整值坐标的参考坐标系原点[0, 0]位于图像的左上角。从实值到整值像素的变换，包括纵轴方向的改变，表示如下：

$$\mathbf{I}_{\mathbf{P}} = \begin{bmatrix} s & r \\ s & c \\ s & \end{bmatrix} = {}^{\mathbf{I}}_{\mathbf{F}}\mathbf{S} \begin{bmatrix} s & {}^{F}P_x \\ s & {}^{F}P_y \\ s & \end{bmatrix} \tag{13-20}$$

其中${}^{\mathbf{I}}_{\mathbf{F}}\mathbf{S}$定义为：

$${}^{\mathbf{I}}_{\mathbf{F}}\mathbf{S} = \begin{bmatrix} 0 & -\frac{1}{d_y} & 0 \\ \frac{1}{d_x} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{13-21}$$

完全摄像机矩阵将实际3D坐标点转换成图像像素点，最后结果为：

$$\mathbf{I}_{\mathbf{P}} = \left({}^{\mathbf{I}}_{\mathbf{F}}\mathbf{S}\,{}^{\mathbf{F}}_{\mathbf{C}}\Pi(f)\,{}^{\mathbf{C}}_{\mathbf{W}}\mathbf{TR}(\alpha, \beta, \gamma, t_x, t_y, t_z)\right){}^{\mathbf{W}}\mathbf{P}$$

$$\begin{bmatrix} s & {}^{I}P_r \\ s & {}^{I}P_c \\ s & \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & 1 \end{bmatrix} \begin{bmatrix} {}^{W}P_x \\ {}^{W}P_y \\ {}^{W}P_z \\ 1 \end{bmatrix} \tag{13-22}$$

也就是公式（13-14）中的完全摄像机矩阵。 425

利用摄像传感器建立3D点的视觉模型，对这个过程我们简单做个总结。首先安放摄像机，使它的坐标系与世界坐标系重合；接下来转动摄像机（${}^{\mathbf{C}}_{\mathbf{W}}\mathbf{R}$），使它相对$\mathbf{W}$系满足最后的方向要求；然后平移摄像机（${}^{\mathbf{C}}_{\mathbf{W}}\mathbf{T}$），使它从合适的位置观察工作区，这时利用透视投影模型（${}^{\mathbf{F}}_{\mathbf{C}}\prod(f)$），所有的3D点都能够投影到摄像机的图像平面；最后对实值图像坐标$[{}^{F}P_x, {}^{F}P_y]$进行缩放变换，并改变纵轴的方向就可以得到像素坐标$[{}^{I}P_r, {}^{I}P_c]$。所有步骤我们用的都是数学推导方法。利用距离和角度测量数据得到足够精确的摄像机矩阵$\mathbf{C}$，然后再以足够的精度进行上述变换，实际上这个过程执行起来是有难度的。一般是通过对摄像机进行标定来获得摄像机矩阵。上面的推导证实了摄像机矩阵的形式是对的，利用第13.4节要描述的控制点拟合方法，可以得到摄像机矩阵参数的实际值，在讨论标定之前，我们先看看摄像机矩阵的重要用途。

习题13.11

根据这一节讨论，很容易看出摄像机矩阵具有如下形式：

$$\begin{bmatrix} s\ {}^{I}P_x \\ s\ {}^{I}P_y \\ s \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{bmatrix} \begin{bmatrix} {}^{W}P_x \\ {}^{W}P_y \\ {}^{W}P_z \\ 1 \end{bmatrix} \tag{13-23}$$

证明通过把摄像机矩阵乘以系数$1/c_{34}$，就可以由12个参数变成公式（13-22）中的11个参数的形式。验证11参数的形式同样能够实现从3D场景点到2D图像点的映射。

13.3.2 正投影与弱透视投影

${}^{C}\mathbf{P}$的正投影不考虑实际点的z坐标，等价于沿与光轴平行的方向把每个实际点投影到图像平面。图13-12对正投影和透视投影做了比较。正投影可以看作是焦距f无穷远的透视投影，如公式（13-24）所示。在计算机图形学中，常常通过正投影表达目标截面的真正尺度。正投影也在计算视觉理论的研究中得到应用。正投影比透视投影更简单，因此在许多情况下用它来近似透视投影进行理论检验。

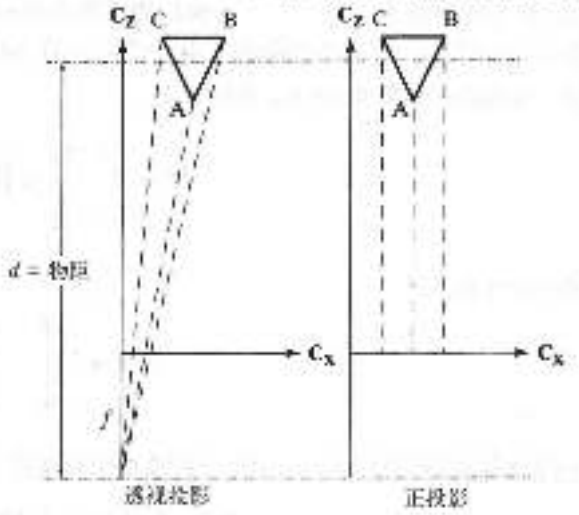


图13-12 透视投影与正投影

$$\mathbf{{}^{F}P} = {}^{F}_{C}\mathbf{\Pi}(\infty)\,\mathbf{{}^{C}P}$$

$$\begin{bmatrix} {}^{F}P_x \\ {}^{F}P_y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} {}^{C}P_x \\ {}^{C}P_y \\ {}^{C}P_z \\ 1 \end{bmatrix} \tag{13-24}$$

426

通常，透视变换能够用正投影和实际图像平面内的同比例缩放来近似。不考虑z坐标以及采用同比例缩放的投影，称为弱透视（weak perspective），比例系数$s=f/d$，是摄像机焦距与物距之比。该比例系数非常有用，参见图13-12。

$$\mathbf{{}^{F}P} = {}^{F}_{C}\mathbf{\Pi}(s)\,\mathbf{{}^{C}P}$$

$$\begin{bmatrix} {}^{F}P_x \\ {}^{F}P_y \end{bmatrix} = \begin{bmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \end{bmatrix} \begin{bmatrix} {}^{C}P_x \\ {}^{C}P_y \\ {}^{C}P_z \\ 1 \end{bmatrix} \tag{13-25}$$

根据经验，当物距是物体大小的20倍时，这种弱透视近似是可以接受的。近似效果也与物体离光轴的距离有关，这个距离越近越好。当图13-12中的三角形物体偏离光轴的右侧很远，则点A和B的透视像点将会簇拥在一起，直到B点被A点挡住，然而对于正投影，与点A和点B对应的像点之间将保持原来的距离。大多数机器人和工业视觉系统都尽量将物体放在视场的中心，对于航空成像应用也是如此。在这些情况下，弱透视是比较合适的模型。

习题13.12

假设$f\to\infty$，根据公式（13-16）推导出公式（13-24）。

427

用弱透视代替实际透视，常常使数学推导与算法变得更加容易。对于识别算法中的匹配，

经常是用近似模型就足够了。另外对于利用实际透视模型的复杂迭代算法，封闭形式的弱透视解能够提供一个比较好的初始点。Huttenlocher和Ullman（1988）在这个问题上已经发表了一些基础性的文章。表13-1对实际透视和弱透视进行了比较。在表格中所示的范围内，由其中的数据可以看出弱透视是实际透视很好的近似模型。

表13-1 弱透视与实际透视

$^{C}P_z$	f = 5mm	s = 5/1000	f = 20mm	s = 20/1000	f = 50	s = 50/1000
0	0.000	0.000	0.000	0.000	0.000	0.000
10	0.051	0.050	0.204	0.200	0.510	0.500
20	0.102	0.100	0.408	0.400	1.020	1.000
50	0.255	0.250	1.020	1.000	2.551	2.500
100	0.510	0.500	2.041	2.000	5.102	5.000
200	1.020	1.000	4.082	4.000	10.204	10.000
500	2.551	2.500	10.204	10.000	25.510	25.000
1000	5.102	5.000	20.408	20.000	51.020	50.000

用$s = f/1000$弱透视$^{I}_{C}\prod_{weak}(s)$，对3D点$[P_x, 0, 980]$进行变换，将此变换与透视变换$^{I}_{C}\prod_{per}(f)$做比较，计算比较值$^{I}P_x$。透视变换的焦距是5mm，20mm和50mm。物距z坐标值1000mm，弱透视的比例系数设为$f/1000$。

利用公式（13-25）的弱透视模型，得到用8个参数定义的弱透视变换模型：

$$^{F}\mathbf{P} = {}^{F}_{C}\mathbf{\Pi}_{weak}\ {}^{C}_{W}\mathbf{TR}\ {}^{W}\mathbf{P}$$

$$\begin{bmatrix} {}^{I}P_x \\ {}^{I}P_y \end{bmatrix} = \begin{bmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{W}P_x \\ {}^{W}P_y \\ {}^{W}P_z \\ 1 \end{bmatrix} \tag{13-26}$$

$$= \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \end{bmatrix} \begin{bmatrix} {}^{W}P_x \\ {}^{W}P_y \\ {}^{W}P_z \\ 1 \end{bmatrix} \tag{13-27}$$

习题13.13

公式（13-27）中，弱透视变换的8个参数中只有7个独立参数，这7个独立参数是什么？

13.3.3 基于多摄像机的3D点计算

下面我们讨论如何根据两个像点$[r_1, c_1]$和$[r_2, c_2]$算出未知的3D点$[x, y, z]$，两个像点由标定好的两台摄像机摄取。因为现在点的坐标系统现在是明确的，所以在点的表示符中省去了上角标。图13-3显示了视觉系统的环境，公式（13-14）给出了每台摄像机的模型。由成像公式可以得到4个线性方程，其中包含3个未知数x、y和z。 428

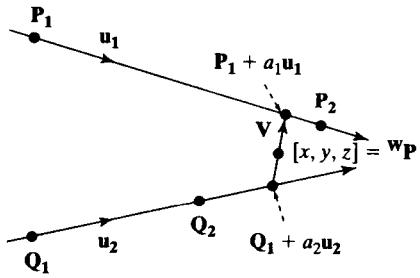
$$\begin{bmatrix} sr_1 \\ sc_1 \\ s \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} tr_2 \\ tc_2 \\ t \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{13-28}$$

从公式（13-28）中去掉齐次坐标s和t，就得到下面含3个未知数的4个线性方程。

$$\begin{aligned} r_1 &= (b_{11} - b_{31}r_1)x + (b_{12} - b_{32}r_1)y + (b_{13} - b_{33}r_1)z + b_{14} \\ c_1 &= (b_{21} - b_{31}c_1)x + (b_{22} - b_{32}c_1)y + (b_{23} - b_{33}c_1)z + b_{24} \\ r_2 &= (c_{11} - c_{31}r_2)x + (c_{12} - c_{32}r_2)y + (c_{13} - c_{33}r_2)z + c_{14} \\ c_2 &= (c_{21} - c_{31}c_2)x + (c_{22} - c_{32}c_2)y + (c_{23} - c_{33}c_2)z + c_{24} \end{aligned} \tag{13-29}$$

4个方程中的任意3个联立都可以得出未知点$[x, y, z]$，但求出的坐标值会产生微小的差异。4个方程同时联立在一起是矛盾的，因为摄像机模型和图影点的近似误差，两台摄像机的投影线并没有在数学3D空间相交于一点。最好的解决方案是计算这两条空间斜交投影线之间的最短距离，也就是计算它们公垂线段的长度。如果公垂线比较短，我们就取公垂线的中点作为两条投影线的交点，即图13-13中的点$[x, y, z]$。如果公垂线太长，那么就断定在进行像点$[r_1, c_1]$和$[r_2, c_2]$对应计算时出现了问题。



$\mathbf{P}_1$和$\mathbf{P}_2$是一条直线上的两个点，而$\mathbf{Q}_1$和$\mathbf{Q}_2$是另外一条直线上的两个点。$\mathbf{u}_1$和$\mathbf{u}_2$是沿两条直线的单位向量。向量$\mathbf{V} = \mathbf{P}_1 + a_1\,\mathbf{u}_1 - (\mathbf{Q}_1 + a_2\,\mathbf{u}_2)$就是连接两条直线的最短距离向量，其中$a_1$和$a_2$是两个要确定的比例系数。为了使$\mathbf{V}$的长度最小，利用求导方法可以确定$a_1$和$a_2$。而利用$\mathbf{V}$一定正交于$\mathbf{u}_1$和$\mathbf{u}_2$这个约束条件，可以更容易算出$a_1$和$a_2$

图13-13 两条空间斜交线之间的最短距离，就是它们之间公垂线线段的长度

429 利用两空间斜交线与公垂线正交这一约束条件，可以得到如下2个含未知数a_1和a_2的线性方程:

$$\begin{aligned} ((\mathbf{P}_1 + a_1\mathbf{u}_1) - (\mathbf{Q}_1 + a_2\mathbf{u}_2)) \circ \mathbf{u}_1 &= 0 \\ ((\mathbf{P}_1 + a_1\mathbf{u}_1) - (\mathbf{Q}_1 + a_2\mathbf{u}_2)) \circ \mathbf{u}_2 &= 0 \end{aligned} \tag{13-30}$$

$$\begin{aligned} 1a_1 - (\mathbf{u}_1 \circ \mathbf{u}_2)a_2 &= (\mathbf{Q}_1 - \mathbf{P}_1) \circ \mathbf{u}_1 \\ (\mathbf{u}_1 \circ \mathbf{u}_2)a_1 - 1a_2 &= (\mathbf{Q}_1 - \mathbf{P}_1) \circ \mathbf{u}_2 \end{aligned} \tag{13-31}$$

利用消元法或者行列式法可以很容易解出a_1和a_2。

$$\begin{aligned} a_1 &= \frac{(\mathbf{Q}_1 - \mathbf{P}_1) \circ \mathbf{u}_1 - ((\mathbf{Q}_1 - \mathbf{P}_1) \circ \mathbf{u}_2) \circ (\mathbf{u}_1 \circ \mathbf{u}_2)}{1 - (\mathbf{u}_1 \circ \mathbf{u}_2)^2} \\ a_2 &= \frac{((\mathbf{Q}_1 - \mathbf{P}_1) \circ \mathbf{u}_1)(\mathbf{u}_1 \circ \mathbf{u}_2) - (\mathbf{Q}_1 - \mathbf{P}_1) \circ \mathbf{u}_2}{1 - (\mathbf{u}_1 \circ \mathbf{u}_2)^2} \end{aligned} \tag{13-32}$$

如果$\|s\mathbf{V}\|$小于某个阈值，我们就认为两条直线相交于点$[x, y, z]^t = (1/2)\,[(\mathbf{P}_1 + a_1\,\mathbf{u}_1) + (\mathbf{Q}_1 +$

$a_2\ \mathbf{u}_2$)]。回头看看我们会发现，所有的计算都依赖于两点（$\mathbf{P}_1$和$\mathbf{P}_2$）确定一直线。通常投影线由摄像机光心和图像点决定。如果光心未知，通过选择某个值$z = z_0$，然后解公式（13-19）中的两个方程求出坐标x和y，就得到第一个摄像机投影线上的一个点。如果这条线同z轴接近平行的话，那么应该选取$x = x_0$。用同样的方式可以求出所需的4个点。

习题13.14

用你最擅长的程序语言，设计一个函数并进行测试，计算两条空间斜交线之间的距离以及公垂线段的中点。该函数应该将4个3D点作为输入，并且要利用本节的数学公式。

习题13.15

用Cramer法则求解公式（13-31）中的a_1和a_2时，要求系数矩阵的行列式不为零。证明这种情况一定发生在两台摄像机同时观察同一个点时。

我们可以用一台摄像机和一台投影仪来进行3D表面测量。几何上和数学上都等同于两台摄像机的情况。这样做最大的好处就是，投影仪能够在光滑的表面上人为产生纹理，以使定义特征点并进行对应计算。在讨论完通过标定得到摄像机和投影仪模型之后，接着再介绍结构光的使用。

13.4 最佳仿射标定矩阵

摄像机标定问题，就是建立像素点在给定摄像机的图像阵列中的位置与3D场景中要成像的实体点之间的关系。总的来说，这个过程在图像分析的各个方面都要用到，包括计算目标的3D位置和姿态以及测量目标的尺寸。在第13.3.3节的立体三角计算中也要用到。

在第13.3节已经讲过，公式（13-14）中的11参数摄像机矩阵是比较合适的数学模型。下面我们介绍如何用最小二乘拟合的方法求出这11个参数的值。摄像机的视场和焦距不变，标定物放在场景中合适的地方，标定物上面的测量点坐标已知，具体参考图13-14。得到n组数据$<{}^{I}\mathbf{P}_i, {}^{W}\mathbf{P}_i>$，其中图像点${}^{I}\mathbf{P}_i = [{}^{I}P_{i1}, {}^{I}P_{i2}]$，对应被观测的3D点是${}^{W}\mathbf{P}_i = [{}^{W}P_{i1}, {}^{W}P_{i2}, {}^{W}P_{i3}]$。点数$n$至少是6个，最好是25或者更多。

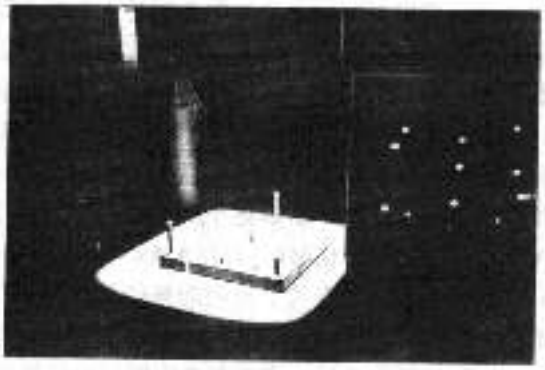


图13-14 左边标定物上有高度不等的9个锥子（可以旋转三次，这样就能得到25个标定点），右边显示的是标定物的图像

13.4.1 标定物

用标定物（calibration jig）主要是方便找到明确的3D点的位置。图13-14、13-18和13-22用到了三个不同的标定物。需要仔细确定标定物在$\mathbf{W}$坐标系中的位置，或者定义的世界坐标系能够使3D特征点的坐标$[{}^{W}P_{i1}, {}^{W}P_{i2}, {}^{W}P_{i3}]$容易确定。然后通过摄像机摄取图像，得到对应这些特征点的2D坐标$[{}^{I}P_{i1}, {}^{I}P_{i2}]$。其他类型的标定物，有的是用金属线和小球构成的刚体架，有的是带有特殊标记的刚体板。

13.4.2 最小二乘问题

从成像模型中消去齐次系数s就能够得到公式（13-33），这样对应每条投影线，就有两个

线性方程，而每个标定点对应一条投影线。为了简化表示方法，同时又不引起符号上的混乱，
431 我们用$[x_j, y_j, z_j]$代替${}^{W}\mathbf{P}_j = [{}^{W}P_x, {}^{W}P_y, {}^{W}P_z]$来表示实际点，用$[u_j, v_j]$代替${}^{I}\mathbf{P}_j = [{}^{I}P_r, {}^{I}P_c]$表示图像点。对应每一个标定点，可以得到下面两个方程：

$$\begin{aligned} u_j &= (c_{11} - c_{31}u_j)x_j + (c_{12} - c_{32}u_j)y_j + (c_{13} - c_{33}u_j)z_j + c_{14} \\ v_j &= (c_{21} - c_{31}v_j)x_j + (c_{22} - c_{32}v_j)y_j + (c_{23} - c_{33}v_j)z_j + c_{24} \end{aligned} \tag{13-33}$$

对上面的方程进行整理，把已知项和未知项分开并用向量表示。有了标定数据，左边的各项是已知项，右边的所有c_{km}项是需要求的未知项。

$$\begin{bmatrix} x_j, y_j, z_j, 1, 0, 0, 0, 0, -x_j u_j, -y_j u_j, -z_j u_j \\ 0, 0, 0, 0, x_j, y_j, z_j, 1, -x_j v_j, -y_j v_j, -z_j v_j \end{bmatrix} \begin{bmatrix} c_{11} \\ c_{12} \\ c_{13} \\ c_{14} \\ c_{21} \\ c_{22} \\ c_{23} \\ c_{24} \\ c_{31} \\ c_{32} \\ c_{33} \end{bmatrix} = \begin{bmatrix} u_j \\ v_j \end{bmatrix} \tag{13-34}$$

由于对应每条投影线能得出两个方程，从n组标定点就能得到$2n$个线性方程，用矩阵方式表示时，x是未知的列向量，**b**是图像坐标列向量。

$$\mathbf{A}_{2n\times 11}\mathbf{x}_{11\times 1} \approx \mathbf{b}_{2n\times 1} \tag{13-35}$$

可以看出只有11个未知数，但是方程的个数有12个或者更多，这是一个超定系统。不存在
432 满足所有方程的参数向量**x**，因此用最小二乘法得到的结果是较合适的解。我们希望所求的参数能够满足如下条件，即实测图像坐标与经摄像机矩阵预测的坐标之间的差平方和最小。图13-16显示2个标定点产生4个坐标差。与第11章一样称这些坐标差为*余差*（residual）。图13-17是对最小二乘解的抽象表示。我们想求出用矩阵**A**各列的线性组合表示的一组参数c_{km}，这个线性组合与**b**最接近。求解这个问题的关键是要看到余差向量$\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{x}$与**A**的列空间正交，即$\mathbf{A}^t\mathbf{r} = \mathbf{0}$。用$\mathbf{b} - \mathbf{A}\mathbf{x}$代替**r**，得到$\mathbf{A}^t\mathbf{A}\mathbf{x} = \mathbf{A}^t\mathbf{b}$。$\mathbf{A}^t\mathbf{A}$是对称正定矩阵，因此它的逆存在，于是解出$\mathbf{x} = (\mathbf{A}^t\mathbf{A})^{-1}\mathbf{A}^t\mathbf{b}$。有几个通用的数值算法库可用来解决这个问题。（对于MATLAB，使用简单的命令x = A\b就行。一旦求出最小二乘解x，余差向量r就可以通过r = b - Ax求出。）请参考Heath 1997年的文献，或者参考你所用的线性代数算法库的用户手册。

图13-18是利用图13-15中的标定物进行摄像机标定的结果。标定物的角点用字母“A”～
433 “P”标注，它们的实际坐标[X, Y, Z]是已知的，列在图13-18中图像坐标[U, V]的右侧。在现在的视点位置下，角点“B”、“C”、“M”是被遮挡住的，所以没有与它们对应的图像点坐标。通过其余13个对应点的拟合，求出摄像机矩阵：摄像机矩阵位于图13-18的下部，余差列在图13-18的右侧。在利用摄像机矩阵求出的26个坐标之中，有16个坐标值的误差在一个像素之内，另10个坐标值的误差大于一个像素，不过只有2个坐标值的误差超过两个像素。这个例子说明了仿射摄像机模型的有效性，但也显示出由于角点位置和短焦距透镜所引起的误差。

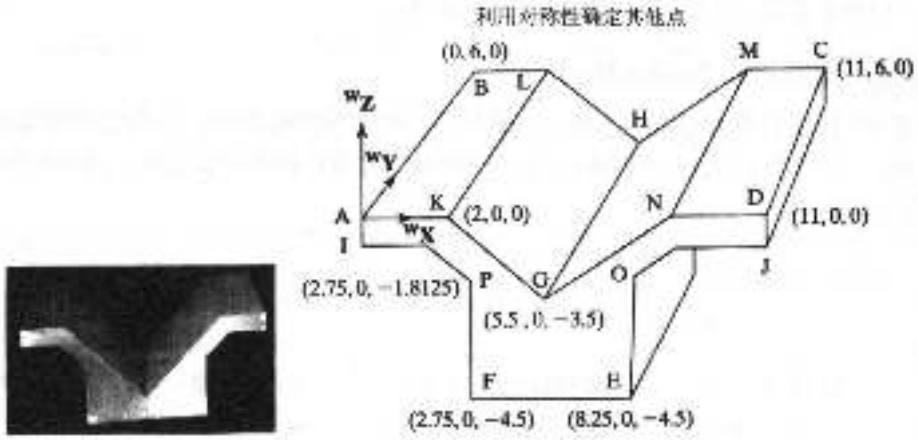


图13-15 具有多个角点的精确标定物。标定物长11in.，宽6in.，高4.5in.。在图13-18中给出了所有3D角点的坐标

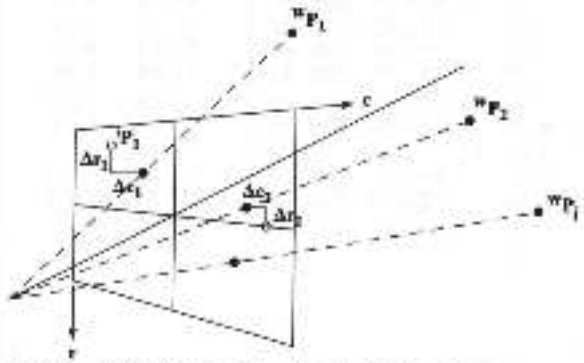


图13-16 图像平面的余差，等于实测图像点（空心点）的坐标与通过公式（13-14）中的摄像机矩阵算出的点（实心点）坐标之间的差

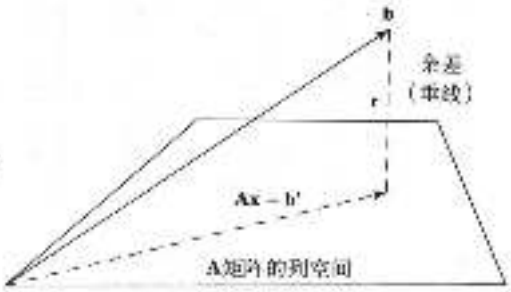


图13-17 系统Ax = b的最小二乘解。平面表示矩阵$A_{5\times11}$的11维列空间。全部线性组合A_x都在该空间内，但$B_{5\times1}$不在该空间内。最小二乘法计算b'，b'就是b到这11维空间的投影，b'是该空间中与b最接近的点

习题13-16 计算摄像机模型

（a）查找最小二乘拟合的软件。输入图13-18中的对应点，计算摄像机矩阵，并与图13-18中的矩阵做比较。（b）去掉余差最大的3个点，再求一次摄像机矩阵，有没有新的余差大于2个像素？（c）定义1×1×1立方体的3D坐标，把它放在图13-15中的标定物上部的一个平面上。利用已有的摄像机矩阵，求立方体的8个角点所对应的图像坐标，同时对接触向上标定物的四个角点也进行类似的计算。画出得到的图像点，并画出相连的边线。得出的图像看起来是立方体吗？

434

习题13-17 亚像素精度

参考图13-14，利用第3章的方法，可以以亚像素的精度计算销子的中心。怎么计算？能

对[P_w P_i]用非整数坐标做为标定数据吗？怎样进行？

习题13.18 最佳弱透视摄像机模型

对图13-18中左侧的数据进行拟合，求最佳弱透视摄像机矩阵。认真复习简化成像方程的推导过程，推导出新的系统方程Ax = b。得到最佳摄像机矩阵参数之后，将余差与图13-18中右侧的余差做比较。

```
#
# IMAGE: qlview1.rax
#
#
#                 输入数据                                    |          输出数据
#                                                             |
点    2D图像点(U,V)          3D空间点(X,Y,Z)                  |   2D拟合数据          X和Y的余差
A     95.00   336.00    0.00    0.00    0.00                  |   94.53   337.89      0.47   -1.89
B                       0.00    6.00    0.00                  |
C                      11.00    6.00    0.00                  |
D    592.00   358.00   11.00    0.00    0.00                  |  592.21   358.36     -0.21   -0.36
E    472.00   168.30    8.25    0.00   -4.50                  |  470.14   168.30      1.86   -0.30
F    232.00   155.30    2.75    0.00   -4.50                  |  232.30   154.43     -0.30    0.87
G    350.00   205.00    5.50    0.00   -3.50                  |  349.17   202.47      0.83    2.53
H    362.00   323.00    5.00    0.00   -2.50                  |  363.44   324.32     -1.44   -1.32
I     97.00   305.00    0.00    0.00    0.75                  |   97.90   304.96     -0.90    0.04
J    592.00   336.00   11.00    0.00   -0.75                  |  591.78   334.94      0.22    1.06
K    184.00   344.00    2.00    0.00    0.00                  |  184.46   343.40     -0.46    0.60
L    263.00   431.00    2.00    6.00    0.00                  |  261.52   429.65      1.48    1.35
M                       0.00    6.00    0.00                  |
N    501.00   363.00    9.00    0.00    0.00                  |  501.16   362.78     -0.16    0.22
O    467.00   279.00    8.25    0.00   -1.81                  |  468.35   281.09     -1.35   -2.09
P    224.00   266.00    2.75    0.00   -1.81                  |  224.06   266.43     -0.06   -0.43

#                    摄像机矩阵

    44.84       29.80       5.504      94.53

    2.518       42.34       40.79      337.9

    -0.0006832  0.06489     -0.01027   1.000
```

图13-18 利用图13-15所示的标定物进行摄像机标定的结果

习题13.19 摄像机标定

表13-2中显示的是，图13-15中标定物的16个3D角点所对应的图像点。实际上，像点坐标来自两幅图像。采用仿射标定方法，根据表中的2-6列数据，用5组标定数据对计算摄像机矩阵。

435

表13-2 使用两台摄像机得到的标定物图像的3D特征点

Point	^{w}x	^{w}y	^{w}z	^{1}u	^{1}v	^{2}u	^{2}v
A	0.0	0.0	0.0	167	65	224	168
B	0.0	6.0	0.0	96	127	196	42
C	11.0	6.0	0.0	99	545	96	431
D	11.0	0.0	0.0	171	517	154	577
E	8.25	0.0	−4.5	352	406	366	488
F	2.75	0.0	−4.5	347	186	440	291
G	5.5	0.0	−3.5	311	294	358	387
H	5.5	6.0	−3.5	226	337	NA	NA

（续）

Point	^{w}x	^{w}y	^{w}z	^{1}u	^{1}v	^{2}u	^{2}v
I	0.0	0.0	0.75	198	65	303	169
J	11.0	0.0	−0.75	203	518	186	577
K	2.0	0.0	0.0	170	143	248	248
L	2.0	6.0	0.0	96	198	176	116
M	9.0	6.0	0.0	97	465	114	363
N	9.0	0.0	0.0	173	432	176	307
O	8.25	0.0	1.81	245	403	259	482
P	2.75	0.0	−1.81	242	181	318	283

3D实际坐标^{w}x，^{w}y，^{w}z的单位为英寸。图像1的坐标是^{1}u，^{1}v，单位是行和列。图像2的坐标是^{2}u，^{2}v。

习题13.20 立体视觉计算

（a）用表13-2中的数据，计算两个标定矩阵，一个利用第2-6列的数据，另一个利用第2-4，7-8列的数据。（b）利用第13.3.3节的方法，计算点A的3D坐标。只用两个摄像机矩阵和表中5-8列的图像坐标。把你得到的结果与表中2-4列的数据做比较。（c）考虑标定物角点I与P之间的钝角角点，用Q表示。假设点Q的像点分别是[196, 135]和[281, 237]。用立体视觉的方法计算点Q的3D坐标，并证明你的结果是合理的。

13.4.3 仿射方法讨论

主要问题在于是否真的需要估计摄像机模型中的11个参数。我们已经看到，在确定世界坐标系下的摄像机位姿时，只有3个独立的旋转参数和3个独立的平移参数，从实际图像坐标到以行列像素为单位的变换需要2个比例因子，以及焦距f，所以这11个参数并不都是独立参数。把它们作为独立参数对待，意味着旋转参数所确定的旋转矩阵不是正交的。对于精确调整好的摄像机，我们没必要用这么多的约束条件。但对于图像平面与光轴不垂直的情况，用较多的参数可以使模型更准确。为了估计比较多的自由参数，需要比较多的标定点。这些参数也不能明确反映摄像机的本质特征。但是仿射模型方法具有自身的优势，在图像行列不够垂直或者图像平面与光轴之间不够垂直的情况下，仿射模型仍然能够使用。不管是像素坐标还是实际图像坐标，都能够采用仿射模型，而且求解过程不需要迭代，可以很快算出结果。在13.7节，我们将介绍另一种标定方法，采用了更多的约束条件，能够克服上面所提到的问题。

习题13.21 标定相机

如果你没有相机，借一台或者买一台便宜的，找个硬盒子做标定物，在每个面上画上几个“X”。RH坐标系的原点位于盒子的一个角点处，三个坐标轴就是盒子的三条边。测量出盒子所有角点的坐标，以及相对RH坐标系“X”处的坐标。给盒子拍一张图片，在图片中找出15个角点和“X”点。用尺子量出这些点在图像中的坐标，单位为英寸。模仿本节所举的例子，求出摄像机矩阵和余差，对结果进行总结。对于标定中没有用到的那些点，用摄像机矩阵算出它们的3D坐标，看看结果如何。然后图像坐标改用mm为单位，再计算一次。

习题13.22 纹理映射

还是前面实验中用到的盒子，进行纹理映射。（a）首先，和上面一样得到一幅盒子的.pgm格式图片。（b）用第11章介绍的方法，建立从盒子的一面（用2D坐标）到包含你的照

片的图像阵列的映射。(c) 通过写入照片的像素值来更新盒子的.pgm图像文件。提示：对三角形进行映射比对平行四边形进行映射的效果要好，为什么？

13.5 使用结构光

在第一节中，我们提出了用结构光（structured light）进行测量，图13-4是关于结构光的示意图，现在我们具备了实现结构光的所有数学工具。图13-19详细显示了结构光的工作原理。物体表面的光照模式是，一幻灯投影仪把规则的栅格光线投射到表面上，然后用摄像机拍摄光栅覆盖的表面，效果与表面结构和表面位姿有关。由于光栅高度结构化，关于哪一些投影线产生了交叉点，成像系统具有明确的信息。假设某个时刻成像系统知道栅格交叉点${}^{G}P_{lm}$的像点是${}^{I}P_{uv}$。然后为了求出由特殊光照模式照明的3D表面点${}^{W}P_{lm}$，就产生4个可用的线性方程。我们必须知道摄像机标定矩阵**C**和投影仪标定矩阵**D**。系统的解$\mathbf{D}{}^{W}P_{lm} = {}^{G}P_{lm}$和$\mathbf{C}{}^{W}P_{lm} = {}^{I}P_{uv}$与13.3.3节给出的双摄像机体视的情况一样。

标定投影仪与标定摄像机方法很相似。打开投影仪，照射工作台平面。标定物放在桌面上，让它的一个角正好与投射的一个光栅交叉点重合，得到一组标定数据对<[${}^{G}P_{l}$, ${}^{G}P_{m}$], [${}^{W}P_{x}$, ${}^{W}P_{y}$, ${}^{W}P_{z}$]>，其中${}^{G}P_{l}$、${}^{G}P_{m}$等于产生交叉点的栅格线的序数（整数），${}^{W}P_{x}$, ${}^{W}P_{y}$, ${}^{W}P_{z}$是标定物角点的世界坐标。如果用仿射标定方法，可以把幻灯栅格线的顺序简化为$m = 1, 2, \cdots$，或者$l = 1, 2, \cdots$，因为仿射标定方法适合任何比例系数。

437

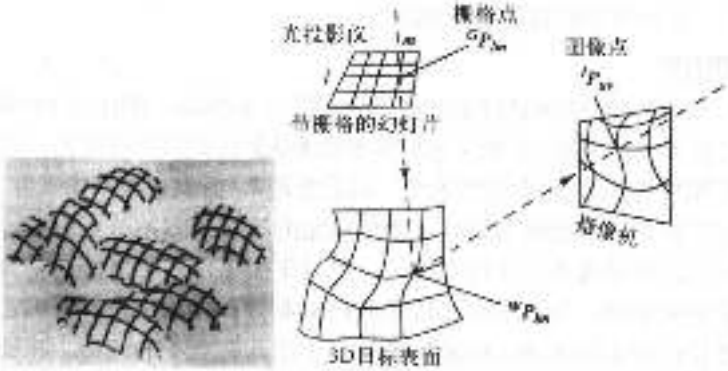


图13-19 左边的马铃薯被带栅格的幻灯照射。右边是结构光原理图。如果成像系统能够知道由哪条光线${}^{G}P_{lm}$产生了图像特征点${}^{I}P_{uv}$，那么就得到求表面点${}^{W}P_{lm}$的四个方程。摄像机矩阵和投影仪矩阵都要已知

算法13.2 利用条纹光以及标定过的摄像机和投影仪计算3D表面坐标

离线程序：

1. 标定摄像机，得到摄像机矩阵**C**。
2. 标定投影仪，得到投影仪矩阵**D**。

在线程序：

1. 输入摄像机矩阵**C**和投影仪矩阵**D**。
2. 输入场景的条纹光表面图像。
3. 抽取出亮线栅格及交叉点。
4. 确定所有栅格交叉点的标号l、m。
5. 对每个像点P_{uv}的投射点P_{lm}，用**C**、**D**和体视公式计算3D表面点P。
6. 输出网格图，格点表示3D点，亮条纹表示它们之间的连线。

习题13.23

按下列方式生成结构光并标定结构光投影仪。采用你最熟悉的图像编辑工具，制作一幅数字栅格图像，背景为黑色，或者在纸上画出栅格，用扫描仪扫成一幅数字图像。把便携式电脑与投影仪相连，显示出你做的数字图像，这样就把栅格投射到了3D空间中。调整投影仪，使它照向桌面工作区，把标定物放到桌子上，得出标定点，进行仿射标定计算，写出总结报告。

这种方法同样存在双摄像机立体视觉所存在的问题，尽管不是很严重。再看看关于马铃薯的那幅图（即图13-19），显而易见，在成像系统确定正确的栅格交叉点方面存在一定的问题。如果只需要表面形状而不需要表面位置，常常只要求栅格交叉点间的相对位置保持一致。请参考图13-20，以及Hu和Stockman、Shrikhande和Stockman于1989年的文献。工程上已经实现各种各样的解决方案，例如栅格线可以采用不同的形状和颜色。另一种解决方案是，快速改变栅格模式，使成像系统得到多幅图像，从这些图像可以唯一地确定栅格模式。白光投影仪的景深有限，栅格模式只有在景深范围内的光线才是清晰的。激光投影仪则不受这一限制，但某些物体的反射效果不好，原因是电压低以及一般激光的光谱范围有限。在许多受控环境中，结构光传感器使用起来非常方便。从一些公司可以买到现成的传感器设备，有的只有一束光，一条光带，或者是两条正交光带。

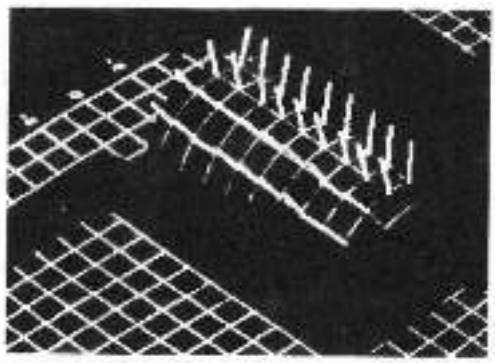


图13-20 通过投射栅格算出的表面法线。只需知道栅格线的顺序，不需知道栅格线的准确位置就可以算出法线（摄像机和投影仪都用透视模型）（由Shrikhande和Stockman,1989提供）

438

13.6 简单的位姿估计过程

我们要借助摄像机计算目标的几何形状和位姿。这一节我们学习一种简单的目标位姿计算方法，只根据二个图像点来计算。假设目标的几何模型已知，摄像机焦距f已知。之所以要讨论这个简单方法，是因为它不仅给出一种实际的位姿估计方法，同时也由此引出一些重要的概念：一个是逆透视（inverse perspective），即根据2D图像特征计算3D特征的透视变换；另一个是用最优化的方法，计算使3D目标点与2D图像点相匹配的最佳参数集。为了简化问题，假设世界坐标系与摄像机坐标系重合，这样就可以省去点表示符的上角标，因为这时不用需要指明坐标系（${}^{w}\mathbf{P}_i = {}^{c}\mathbf{P}_i = P_i$）。另一个简化是，只用实际空间坐标，不用图像量化或者是像素坐标。

三点透视问题（P3P）的环境参见图13-21。3D场景中的三点P_i在$u-v$图像平面上的对应点是Q_i。点P_i的坐标是我们要求的未知数。假设我们知道目标模型上哪些点受到关注（这是大假设），也确实知道这些点两两之间的距离。对于刚体来说，这些距离不随物体在空间中的运动而变化。在人机交互（HCI）应用中，点P_i也许是某个人的面部特征，如双眼和鼻尖。如果面部特征的距离合适，就能够检测出人脸。算出人脸的位姿后，就能够确定这个人在向何处看。在导航系统中，一点P_i也许是地图上位置已知的地理标记，导航机器人或者无人驾驶飞机通过下述方法能够算出自己相对标记的地理位置。

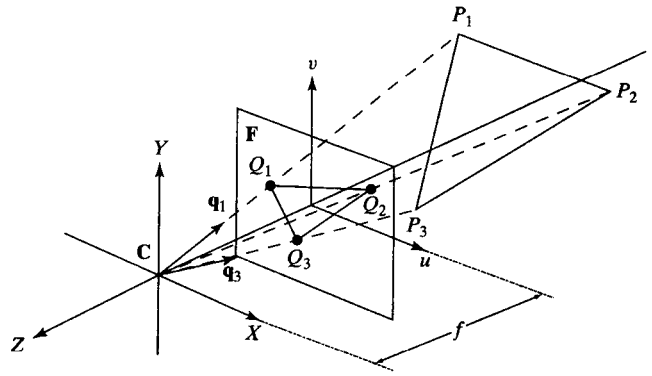


图13-21 位姿估计的简单情况：3D空间边长已知的三角形$P_1P_2P_3$，在$u-v$图像平面对应三角形$Q_1Q_2Q_3$。根据图像点Q_j的坐标可以求出三维点P_j的坐标。可以确定包含点P_j的物体相对摄像机坐标系的位姿。焦距长度f是指从摄影中心到图像平面的距离。图像上的点Q_j相对摄像机坐标系C的坐标为$Q_j = [u_j, v_j, -f]$

439 被观测点Q_i的图像位置是已知的。设$\mathbf{q}_i$是从原点沿Q_i方向的单位向量。3D点P_i也在同样的方向上。因此只要算出三个系数a_i，根据下列公式就可以从Q_i求出P_i的位置

$$P_i = a_i\mathbf{q}_i \tag{13-36}$$

利用公式（13-36）中的三个方程，可以推出三个点之间的距离公式，这三个距离根据目标模型也是可知的。

$$d_{mn} = \|P_m - P_n\| \quad (m \neq n) \tag{13-37}$$

根据观测值Q_i求P_i，利用点积及性质$\mathbf{q}_i \circ \mathbf{q}_i = \mathbf{1}$计算3D距离。

$$\begin{aligned} d_{mn}{}^2 &= \|a_m\mathbf{q_m} - a_n\mathbf{q_n}\|^2 \\ &= (a_m\mathbf{q_m} - a_n\mathbf{q_n}) \circ (a_m\mathbf{q_m} - a_n\mathbf{q_n}) \\ &= a_m{}^2 - 2a_ma_n(\mathbf{q_m} \circ \mathbf{q_n}) + a_n{}^2 \end{aligned} \tag{13-38}$$

现在得到关于未知量a_i的3个二次方程。左边的3个$d_{mn}{}^2$根据模型可以知道，而且根据图像点Q_i 3个$\mathbf{q_m} \circ \mathbf{q_n}$也可以算出。计算3个点$P_i$位置的P3P问题现在变成是求这3个二次方程，其中包含3个未知量。理论上有8个不同的三元组$[a_1, a_2, a_3]$能够满足方程（13-38）。参考图13-21，很容易可以看出，对于在坐标系一边的3个点位置对应参数是$[a_1, a_2, a_3]$，则在另一方的镜像必
440 然有另一组参数$[-a_1, -a_2, -a_3]$。如果$[a_1, a_2, a_3]$是方程组的解，则$[-a_1, -a_2, -a_3]$必然也是。我们最多只有4组表示实际位置的实数解，因为目标只可能在摄像机的一边。在Fishler和Bolles（1981）的文献中，特殊情况下4个位置都是有可能的，但一般情况下只有两个解。

现在看如何利用非线性优化求解未知数a_i（进而求P_i）。在后面的章节中进一步讨论其他优化方法。数学上，就是求下列3个函数中的a_i。

$$\begin{aligned} f(a_1, a_2, a_3) &= a_1{}^2 - 2a_1a_2(\mathbf{q_1} \circ \mathbf{q_2}) + a_2{}^2 - d_{12}{}^2 \\ g(a_1, a_2, a_3) &= a_2{}^2 - 2a_2a_3(\mathbf{q_2} \circ \mathbf{q_3}) + a_3{}^2 - d_{23}{}^2 \\ h(a_1, a_2, a_3) &= a_1{}^2 - 2a_1a_3(\mathbf{q_1} \circ \mathbf{q_3}) + a_3{}^2 - d_{13}{}^2 \end{aligned} \tag{13-39}$$

假设初始值在$[a_1, a_2, a_3]$附近，但是$f(a_1, a_2, a_3) \neq 0$。我们想算出一个增量$[\Delta_1, \Delta_2, \Delta_3]$，理想情况下使$f(a_1+\Delta_1, a_2+\Delta_2, a_3+\Delta_3)=0$，实际上是趋近于0。在$[a_1, a_2, a_3]$的邻域对$f$进行线性化，然后计算使结果为零的增量$[\Delta_1, \Delta_2, \Delta_3]$。

$$f(a_1+\Delta_1, a_2+\Delta_2, a_3+\Delta_3)=f(a_1, a_2, a_3)+\begin{bmatrix}\frac{\partial f}{\partial a_1} & \frac{\partial f}{\partial a_2} & \frac{\partial f}{\partial a_3}\end{bmatrix}\begin{bmatrix}\Delta_1\\ \Delta_2\\ \Delta_3\end{bmatrix}+\text{h.o.t.} \tag{13-40}$$

忽略公式（13-40）中的高阶项，并让左边等于0，就得到一个包含未知数$[\Delta_1, \Delta_2, \Delta_3]$的线性方程。同样道理，得出函数形式为$g$和$h$的两个方程。于是有下列矩阵方程：

$$\begin{bmatrix}0\\0\\0\end{bmatrix}=\begin{bmatrix}f(a_1, a_2, a_3)\\ g(a_1, a_2, a_3)\\ h(a_1, a_2, a_3)\end{bmatrix}+\begin{bmatrix}\frac{\partial f}{\partial a_1} & \frac{\partial f}{\partial a_2} & \frac{\partial f}{\partial a_3}\\ \frac{\partial g}{\partial a_1} & \frac{\partial g}{\partial a_2} & \frac{\partial g}{\partial a_3}\\ \frac{\partial h}{\partial a_1} & \frac{\partial h}{\partial a_2} & \frac{\partial h}{\partial a_3}\end{bmatrix}\begin{bmatrix}\Delta_1\\ \Delta_2\\ \Delta_3\end{bmatrix} \tag{13-41}$$

上面的偏微分矩阵就是雅可比矩阵$\mathbf{J}$。如果在点$[a_1, a_2, a_3]$处它是可逆的，那么就可以求得如下的参数增量：

$$\begin{bmatrix}\Delta_1\\ \Delta_2\\ \Delta_3\end{bmatrix}=-\mathbf{J}^{-1}(a_1, a_2, a_3)\begin{bmatrix}f(a_1, a_2, a_3)\\ g(a_1, a_2, a_3)\\ h(a_1, a_2, a_3)\end{bmatrix} \tag{13-42}$$

把该增量与上一步的参数值相加。用A^k表示参数的第k步迭代值，就得出我们熟悉的牛顿法表达形式。$\mathbf{f}$表示用函数f、g、h算出的值向量。

$$A^{k+1}=A^k-\mathbf{J}^{-1}(A^k)\mathbf{f}(A^k) \tag{13-43}$$

441

习题13.24 雅可比矩阵的定义

证明函数$\mathbf{f}(a_1, a_2, a_3)$的雅可比矩阵具有如下形式，其中$t_{ij}$表示点积$\mathbf{q}_i \cdot \mathbf{q}_j$。

$$\begin{aligned}\mathbf{J}(a_1, a_2, a_3)&=\begin{bmatrix}J_{11} & J_{12} & J_{13}\\ J_{21} & J_{22} & J_{23}\\ J_{31} & J_{32} & J_{33}\end{bmatrix}\\ &=\begin{bmatrix}(2a_1-2t_{12}a_2) & (2a_2-2t_{12}a_1) & 0\\ 0 & (2a_2-2t_{23}a_3) & (2a_3-2t_{23}a_2)\\ (2a_1-2t_{31}a_3) & 0 & (2a_3-2t_{31}a_1)\end{bmatrix}\end{aligned}$$

习题13.25 计算雅可比的逆

在上个习题中，求雅可比的逆矩阵$\mathbf{J}^{-1}$，用J_{ij}表示。

算法13.3对摄像机坐标系下三个3D点坐标位置的计算方法进行了总结。实验证明，一般情况下5到10次迭代后算法就会收敛，但是还不清楚如何对算法进行控制，以获得多个解。非线性优化有时需要一定的技巧，读者应通过阅读参考文献体会其中的细微差别。表13-3列出了P3P求解迭代过程中的性能指标变化情况。仿真实验中用焦距$f=30$的透镜对P_i进行投影，得到图像坐标Q_i。初始值的设置离实际值很远。经过几步迭代，算法很快收敛到实际值邻域内，并且最后算出的P_i与给定值非常接近，精确到小数点后两位十进制有效数字。如表13-3中

442

的3-5列所示，经过9步迭代后，模型边长和计算边长之差小于0.2个单位长度。如果初始值选为$a_i \approx 100$，迭代次数将减半。如果目标到摄像机的近似距离已知，那么这时的参数就是比较好的初始值。

算法13.3 根据三个图像点用P3P迭代求解法计算3D点的位置

输入三组3D和2D的对应点对($^{M}\mathbf{P}_i$, $^{F}\mathbf{Q}_i$)。$^{M}\mathbf{P}_i$是模型坐标，$^{F}\mathbf{Q}_i$是实际图像坐标。

输入摄像机焦距f和距离允许误差Δ。

输出三个模型点在摄像机坐标系中的位置$^{C}\mathbf{P}_i$。

1. 初始化
   (a) 根据模型点$^{M}\mathbf{P}_i$坐标计算距离的平方d_{mn}^2
   (b) 根据图像点$^{F}\mathbf{Q}_i$坐标计算单位向量$\mathbf{q}_i$和点积$2\mathbf{q}_m \circ \mathbf{q}_n$
   (c) 选择初始参数向量$\mathbf{A}^1 = [a_1, a_2, a_3]$（怎么选择？）
2. 迭代，直到$f(\mathbf{A}^k) \approx 0$
   (a) $\mathbf{A}^{k+1} = \mathbf{A}^k - \mathbf{J}^{-1}(\mathbf{A}^k)\,\mathbf{f}(\mathbf{A}^k)$
      i. $\mathbf{A}^k = \mathbf{A}^{k+1}$
      ii. 如果$\mathbf{J}^{-1}$存在，计算$\mathbf{J}^{-1}(\mathbf{A}^k)$
      iii.求$\mathbf{f}(\mathbf{A}^k) = [f(a_1^k, a_2^k, a_3^k), g(a_1^k, a_2^k, a_3^k), h(a_1^k, a_2^k, a_3^k)]^t$
   (b) 如果$\mathbf{f}(\mathbf{A}^{k+1})$在$0\pm\Delta$内，则停止；
      或者达到迭代次数，则停止。
3. 计算位姿。根据$\mathbf{A}^{k+1}$计算每个$^{C}\mathbf{P}_i = a_i^{k+1}\mathbf{q}_i$

表13-3 P3P求解法的迭代情况

It. k	$\lvert f(A^k)\rvert$	$\lvert g(A^k)\rvert$	$\lvert h(A^k)\rvert$	a_1	a_2	a_3
1	6.43e + 03	3.60e + 03	1.09e + 04	1.63e + 02	1.65e + 02	1.63e + 02
2	1.46e + 03	8.22e + 02	2.48e + 03	1.06e + 02	1.08e + 02	1.04e + 02
3	2.53e + 02	1.51e + 02	4.44e + 02	8.19e + 0 1	9.64e + 01	1.03e + 02
…		…			…	
8	2.68e + 00	6.45e − 01	5.78e + 00	8.414e + 01	9.127e + 01	8.926e + 01
9	5.00e − 02	3.87e − 02	1.71e − 01	8.414e + 01	9.12 6e + 01	8.925e + 01

It. k	p_{1x}	p_{1y}	P_{1z}	P_{2x}	P_{2y}	P_{2z}	P_{3x}	P_{3y}	P_{3z}
1	−36.9	−58.4	147.6	−34.4	−14.4	160.7	0.0	−14.5	162.4
2	−24.0	−38.0	96.0	−22.5	−9.3	105.2	0.0	−9.3	103.6
…	…	…	…	…					
8	−19.1	−30.2	76.2	−19.1	−7.9	88.9	0.0	−7.9	88.9
9	−19.1	−30.2	76.2	−19.1	−7.9	88.9	0.0	−7.9	88.9

注：焦距f = 30，仿真实验中图像点Q_j根据P_1 = [−19.05, −30.16, 76.20]、P_2 = [−19.05, −7.94, 88.90]和P_3 = [0.00, −7.94, 88.90]算出。初始值设为A^0 = [300, 300, 300]，Δ = 0.2。到第9次迭代P3P程序收敛到给定的P_j值，精确到小数点后两位十进制有效数字。

Ohmura等人（1988）开发了一个系统，能够以每秒10次的速度计算人头的位置。蓝色模型特征点$^{M}\mathbf{P}_j$取左眼左角、右眼右角和鼻子下面。（面部表情的变化对这些点影响不大。）由于做了蓝色标记，因此能够迅速找到对应的图像点$^{F}\mathbf{Q}_j$，结果也很稳定。利用算出的$^{C}\mathbf{P}_j$以及$^{M}\mathbf{P}_j$到$^{C}\mathbf{P}_j$的映射，可以求出人脸的位姿（用算法13.1）。Ballard和Stockman（1995）开发的系统，

在人脸上不做任何标记就能确定人眼和鼻子的位置，但速度要慢得多，因为要进行人眼和鼻子的识别。两个研究组都声明，求得的三点所构成的平面，其法向量的方向误差数量级为几度。如果三点${}^{c}\mathbf{P}_i$所在的平面与图像平面近似垂直，则图像点坐标${}^{I}\mathbf{Q}_i$有一个小误差就会在计算3D平面方向中带来很大的误差。为了避免这种情况，Ohmura等人（1988）安排摄像机轴线与人脸方向大约成20°夹角。 443

方程（13-38）总存在一个解，因此我们能够根据看到图像上的三点算出飞机的位姿实际选择一个好的模型很重要，知道飞机不是绿色的或者没有飞机存在就有帮助。模型验证同样重要，可以通过在目标模型上选择更多的点并在图像上进行验证。举个例子来说，为了区别一张脸的两个不同位姿，可以在利用人眼和鼻子计算出位姿之后，再找出耳朵、下巴和眉毛。下一节将对验证问题讨论的更多一些，也要考虑以像素坐标为单位的数字图像点，以及透视投影中的摄像机径向畸变。

习题13.26

讨论如何用P3P方法解决P5P问题。

习题13.27 WP3P问题*

想办法找到Huttenlocher和Ullman于1988年发表的论文，其中介绍了一种计算3点刚体位姿的方法，用的是一个弱透视投影模型。这个解法是封闭形式的，能够明显产生两个解，这一点与本章的P3P求法不同。对该解法进行编程，然后通过仿真实验进行测试，利用数学方法投影刚体上的3点得到实验用的数据，生成3对对应点$<P_i, Q_i>$。

13.7 改进的摄像机标定法*

我们现在讨论Roger Tsai（1987）提出的标定方法，该方法已经被广泛用于工业视觉系统，据称如果算法用的好的话，进行3D测量的精度达到1/4000，这是个非常好的效果。在第13.3节已经对标定思想进行了详细讨论，我们再讨论时就采用简化的表示符号。

- $\mathbf{P}=[x, y, z]$表示3D坐标系统的一个点。
- $\mathbf{p}=[u, v]$是实际图像平面上的一点。（可以把u轴看成是水平轴，方向向右；v轴是垂直轴，方向向上。）
- $\mathbf{a}=[r, c]$是用整数表示的图像阵列中的一个像素，r是像素的行坐标，c是像素的列坐标。（约定俗成，和上面的u、v相对应，r轴是垂直轴，方向向下，c轴是水平轴，方向向右。）

摄像机标定被看作是参数估计，我们求的是表征摄像机几何结构和位姿的摄像机参数（camera parameter），有两类不同的参数需要进行估计：

1. 内部参数
2. 外部参数 444

13.7.1 摄像机内部参数

内部参数（intrinsic parameter）是指真正的摄像机参数，与所用的光学部件有关，包括如下参数：

- 主点$[u_0, v_0]$，光轴与图像平面的交叉点。
- 比例因子$[d_u, d_v]$，与像素x和y的尺寸有关。

- 变形因子τ_i，与摄像机图像纵横比有关。
- 焦距f，光心到图像平面的距离。
- 摄像机畸变因子（κ_1），与摄像机径向畸变有关的比例因子。

这些定义以摄像机透镜的光心为参考点，摄像机坐标系的原点就在这一点。光轴通过光心与图像平面垂直，主点经常是图像的中心，但有时不是。比例因子d_x和d_y表示单个像素的水平尺寸和垂直尺寸，单位是实际长度单位如mm。对于特定的摄像机，假设u_0、v_0、d_x、d_y和变形因子τ_i都是已知的，那么只有焦距f和摄像机畸变因子κ_1要通过标定算出。

13.7.2 摄像机外部参数

外部参数（extrinsic parameter）描述摄像机在3D世界坐标系下的位置和方向（位姿），其中包括：

- 平移：

$$\mathbf{t} = [t_x \quad t_y \quad t_z]^T \tag{13-44}$$

- 旋转：

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13-45}$$

平移参数描述了摄像机在世界坐标系下的位置，旋转参数描述了摄像机的姿态。一开始我们就强调只有3个独立的旋转参数而不是9个。

下面要介绍的是主动的、有合适精度的、有效的和灵活的标定方法[参见Tsai(1987)]。虽然用这种方法并不能为某个特定的透镜建立出完美的模型，但这种方法可用于现用的任何摄像机和镜头。图13-22显示的是一种标定装置，它也可用于3D目标重建系统，后面将讨论这种系统。该装置有一个金属盘，上面涂着7×7的黑色小圆阵列，圆圈的中心表示点的位置。标定物安装在水平导轨上，标定物与水平导轨垂直，并能够沿导轨运动，每步间隔10mm。导轨上的位置决定了3D世界坐标系的坐标。

415



图13-22 通过2D运动模式生成多个3D空间特征点的标定装置

在图13-22所示的系统中，沿导轨的不同位置拍摄几幅图像，这些不同位置对应着到摄像机的不同距离。在标定摄像机过程中，摄像机本身不动，并且焦距不变，对于每幅图像，检测出圆圈，并算出中心点。图像处理的结果是3D已知点和2D图像点间的对应点集。要求对应点的组数$n>5$，我们把它们表示为

$$\{([x_i, y_i, z_i], [u_i, v_i]) | i = 1, \ldots, n\}$$

利用下面的公式能够由图像像素坐标$[r, c]$算出实值图像坐标$[u, v]$：

$$u = \tau_i d_x (c - u_0) \tag{13-46}$$

$$v = -d_y (r - v_0) \tag{13-47}$$

其中d_x和d_y是水平及垂直方向相邻两像素中心之间的距离，τ_i是摄像机图像变形因子。

图13-23显示的是该方法所用的成像几何模型。$\mathbf{P}_i=[x_i, y_i, z_i]$是3D空间的任意点，在图像平面上与之对应的点是$\mathbf{p}_i$。向量$\mathbf{r}_i$从光轴上的点$[0, 0, z_i]$到3D点$\mathbf{P}_i$。向量$\mathbf{s}_i$从主点$\mathbf{p}_0$到图像点$\mathbf{p}_i$。向量$\mathbf{s}_i$和向量$\mathbf{r}_i$平行。假设摄像机的所有径向畸变都沿着向量$\mathbf{s}_i$的方向产生。

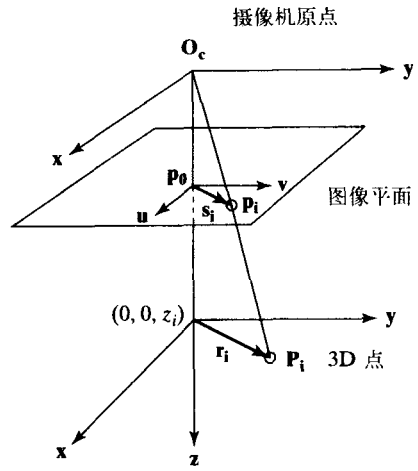


图13-23 Tsai标定方法的几何模型。图像上的点$\mathbf{p}_i=[u_i, v_i]$对应着标定物上的点$\mathbf{P}_i=[x_i, y_i, z_i]$。主点$\mathbf{p}_0=[u_0, v_0]$。径向畸变使图像点$\mathbf{p}_i$沿图像内的$\mathbf{p}_0-\mathbf{p}_i$方向变化

Tsai指出，第一阶段先计算多数外部参数，因为径向畸变沿向量$\mathbf{s}_i$的方向，不用考虑$\mathbf{s}_i$就能确定旋转矩阵。另外，不知道κ_1就能确定t_x和t_y。t_z的计算则必须等到第二阶段进行，因为t_z的变化会产生类似κ_1引起的图像效应。

446

不是直接求出所有的未知数，我们首先求解一组参数μ，从这组参数可以得到要求的外部参数。已知n组对应点$[x_i, y_i, z_i]$和$[u_i, v_i]$，$i=1, \cdots, n$，$n>5$。构造矩阵$\mathbf{A}$，每行为a_i:

$$a_i=[v_i x_i,\ v_i y_i,\ -u_i x_i,\ -u_i y_i,\ v_i]. \qquad (13\text{-}48)$$

设$\mu=[\mu_1, \mu_2, \mu_3, \mu_4, \mu_5]$是需要求的未知参数向量，其中旋转参数$r_{11}$、$r_{12}$、$r_{21}$和$r_{22}$与平移参数$t_x$、$t_y$之比构成$\mu$的各个元素。

$$\mu_1=\frac{r_{11}}{t_y} \qquad (13\text{-}49)$$

$$\mu_2=\frac{r_{12}}{t_y} \qquad (13\text{-}50)$$

$$\mu_3=\frac{r_{21}}{t_y} \qquad (13\text{-}51)$$

$$\mu_4=\frac{r_{22}}{t_y} \qquad (13\text{-}52)$$

$$\mu_5=\frac{t_x}{t_y} \qquad (13\text{-}53)$$

设向量$\mathbf{b}=[u_1, u_2, \cdots, u_n]$包含了$n$组对应点的图像横坐标$u_i$。因为$\mathbf{A}$和$\mathbf{b}$是已知的，从线性方程

$$\mathbf{A}\mu=\mathbf{b} \qquad (13\text{-}54)$$

447

能解出未知的参数向量μ。（参考Johnson、Riess和Arnold（1989）关于求解线性系统方程的内容。）下面就可以根据μ算出旋转参数和平移参数。

1. 设$U=\mu_1^2+\mu_2^2+\mu_3^2+\mu_4^2$。计算平移参数$t_y$的平方

$$t_y^2=\begin{cases}\dfrac{U-[U^2-4(\mu_1\mu_4-\mu_2\mu_3)^2]^{1/2}}{2(\mu_1\mu_4-\mu_2\mu_3)^2} & \text{if } (\mu_1\mu_4-\mu_2\mu_3)\neq 0\\ \dfrac{1}{\mu_1^2+\mu_2^2} & \text{if } (\mu_1^2+\mu_2^2)\neq 0\\ \dfrac{1}{\mu_3^2+\mu_4^2} & \text{if } (\mu_3^2+\mu_4^2)\neq 0\end{cases} \qquad (13\text{-}55)$$

2. 设$t_y=(t_y^2)^{1/2}$（正方根），然后根据算出的μ值计算4个旋转参数和平移参数t_x:

$$r_{11}=\mu_1 t_y \tag{13-56}$$

$$r_{12}=\mu_2 t_y \tag{13-57}$$

$$r_{21}=\mu_3 t_y \tag{13-58}$$

$$r_{22}=\mu_4 t_y \tag{13-59}$$

$$t_x=\mu_5 t_y \tag{13-60}$$

3. 为了确定t_y的正确正负号，选择一个目标点$\mathbf{P}$，它对应的图像点$[u, v]$远离图像中心（为了避免数值问题）。设$\mathbf{P}=[x, y, z]$，然后计算

$$\xi_x=r_{11}x+r_{12}y+t_x \tag{13-61}$$

$$\xi_y=r_{21}x+r_{22}y+t_y \tag{13-62}$$

这就好像把算出的旋转参数做为点$\mathbf{P}$的坐标x和y的系数。如果ξ_x与u的正负号一样，ξ_y与v的正负号一样，那么t_y的正负号就正确，否则就要变号。

4. 其余的旋转参数可按下面的公式计算:

$$r_{13}=\left(1-r_{11}^2-r_{12}^2\right)^{1/2} \tag{13-63}$$

$$r_{23}=\left(1-r_{21}^2-r_{22}^2\right)^{1/2} \tag{13-64}$$

$$r_{31}=\frac{1-r_{11}^2-r_{12}r_{21}}{r_{13}} \tag{13-65}$$

$$r_{32}=\frac{1-r_{21}r_{12}-r_{22}^2}{r_{23}} \tag{13-66}$$

448

$$r_{33}=(1-r_{31}r_{13}-r_{32}r_{23})^{1/2} \tag{13-67}$$

在推导这些公式时，用到了旋转矩阵$\mathbf{R}$的标准正交约束条件。由于方根运算的二值性，r_{23}、r_{31}和r_{32}的正负号也可能不对。在这一步中，如果下式

$$r_{11}r_{21}+r_{12}r_{22}$$

的符号为正，那r_{23}的符号就应该变号，这样才能保证旋转矩阵的正交性。另外两个在算出焦距之后再进行调整。

5. 现在从第2个线性系统方程计算焦距f和平移参数t_z。首先构造矩阵$\mathbf{A}'$，它的行为

$$a_i'=(r_{21}x_i+r_{22}y_i+t_y, v_i) \tag{13-68}$$

接下来构造向量$\mathbf{b}'$

$$b'_i=(r_{31}x_i+r_{32}y_i)v_i \tag{13-69}$$

解下面线性系统方程

$$\mathbf{A}'\mathbf{v}=\mathbf{b}' \tag{13-70}$$

其中$\mathbf{v}=(f, t_z)'$。到这里就得到了f和t_z的估计值。

6. 如果$f<0$，那么改变r_{13}、r_{23}、r_{31}、r_{32}、f和t_z的正负号。这样就保证符合右手坐标规划。

7. f和t_z的估计值用来计算摄像机畸变因子κ_1，以及改善f和t_z的值。这里用简化的摄像机畸变模型，实际图像坐标$[\hat{u}, \hat{v}]$根据测量值按下列公式计算:

$$\hat{u}=u(1+\kappa_1 r^2) \tag{13-71}$$

$$\hat{v} = v(1 + \kappa_1 r^2) \tag{13-72}$$

其中径向畸变项中的r由下式给出:

$$r = (u^2 + v^2)^{1/2} \tag{13-73}$$

用畸变因子修正透视投影方程，得到如下形式的非线性方程:

$$\left\{ v_i(1 + \kappa_1 r^2) = f \frac{r_{21}x_i + r_{22}y_i + r_{23}z_i + t_y}{r_{31}x_i + r_{32}y_i + r_{33}z_i + t_z} \right\} \quad i = 1, \ldots, n \tag{13-74}$$

利用非线性回归法求解这个系统，就能得出f、t_z和κ_1的值。

13.7.3 标定举例

通过例子看看如何进行摄像机标定。右表中的5组对应点是标定系统的输入。世界坐标系和**u**−**v**图像坐标系的单位用的都是cm。

i	x_i	y_i	z_i	u_i	v_i
1	0.00	5.00	0.00	−0.58	0.00
2	10.00	7.50	0.00	1.73	1.00
3	10.00	5.00	0.00	1.73	0.00
4	5.00	10.00	0.00	0.00	1.00
5	5.00	0.00	0.00	0.00	−1.00

449

图13-24-2.jpg

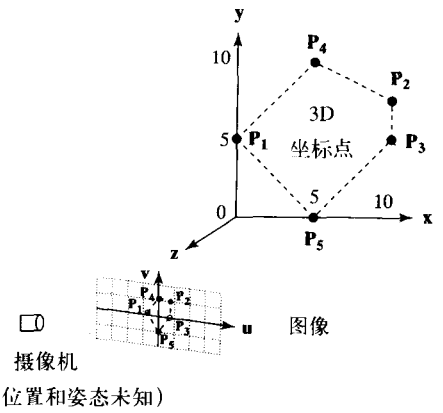


图13-24 3D坐标点和对应的2D图像点是标定程序的输入，标定程序用于计算摄像机参数，包括位置、姿态和焦距

450

图13-24-3.jpg

图13-24-4.jpg

图13-24-5.jpg

图13-24-6.jpg

图13-24-7.jpg

图13-24-8.jpg

图13-24-9.jpg

图13-24-10.jpg

图13-24-11.jpg

图13-24-12.jpg

图13-24-13.jpg

图13-24-14.jpg

图13-24-15.jpg

图13-24-16.jpg

图13-24-17.jpg

图13-24-18.jpg

图13-24-19.jpg

图13-24-20.jpg

图13-24-21.jpg

图13-24-22.jpg

图13-24-23.jpg

图13-24-24.jpg

图13-24-25.jpg

图13-24-26.jpg

图13-24-27.jpg

图13-24-28.jpg

图13-24-29.jpg

图13-24-30.jpg

图13-24-31.jpg

图13-24-32.jpg

图13-24-33.jpg

图13-24-34.jpg

图13-24-35.jpg

图13-24-36.jpg

图13-24-37.jpg

图13-24-38.jpg

图13-24-39.jpg

图13-24-40.jpg

图13-24-41.jpg

图13-24-42.jpg

$$U = \mu_1^2 + \mu_2^2 + \mu_3^2 + \mu_4^2 = 0.07$$

利用公式（13-55）中的第一个式子，得到

$$t_y^2 = \frac{U - [U^2 - 4(\mu_1\mu_4 - \mu_2\mu_3)^2]^{1/2}}{2(\mu_1\mu_4 - \mu_2\mu_3)^2} = 25$$

如果t_y取正方根5，则有

$$r_{11} = \mu_1 t_y = -0.87$$
$$r_{12} = \mu_2 t_y = 0$$
$$r_{21} = \mu_3 t_y = 0$$
$$r_{22} = \mu_4 t_y = -1.0$$
$$t_x = \mu_5 t_y = 4.33$$

为了确定t_y的正负号，接下来计算ξ_x和ξ_y，用的对应点是$\mathbf{P}_2$=(10.0，7.5，0.0) 和点$\mathbf{p}_2$=(1.73，1.0)，图像点$\mathbf{p}_2$远离图像中心。

$$\xi_x = r_{11}x + r_{12}y + t_x = (-0.87)(10) + 0 + 4.33 = -4.37$$

451

$$\xi_y = r_{21}x + r_{22}y + t_y = 0 + (-1.0)(7.5) + 5 = -2.5$$

因为ξ_x和ξ_y与$\mathbf{p}_2$的正负号不一致，所以t_y的正负号是错误的，对它取反号。于是有

$$t_y = -5$$
$$r_{11} = 0.87$$
$$r_{12} = 0$$
$$r_{21} = 0$$
$$r_{22} = 1.0$$
$$t_x = -4.33$$

继续计算其余的旋转参数:

$$r_{13} = \left(1 - r_{11}^2 - r_{12}^2\right)^{1/2} = 0.5$$
$$r_{23} = \left(1 - r_{21}^2 - r_{22}^2\right)^{1/2} = 0$$
$$r_{31} = \frac{1 - r_{11}^2 - r_{12}r_{21}}{r_{13}} = 0.5$$
$$r_{32} = \frac{1 - r_{21}r_{12} - r_{22}^2}{r_{23}} = 0$$
$$r_{33} = (1 - r_{31}r_{13} - r_{32}r_{23})^{1/2} = 0.87$$

经检查$r_{11}r_{21} + r_{12}r_{22} = 0$，其符号不为正，因此$r_{23}$不变号。

现在建立第二个线性系统方程如下:

$$r_{21}x_i + r_{22}y_i + t_y \quad v_i$$

$$\mathbf{A}' = \begin{bmatrix} 0.00 & 0.00 \\ 2.500 & -1.00 \\ 0.00 & 0.00 \\ 5.00 & -1.00 \\ -5.00 & 1.00 \end{bmatrix}$$

和

$$(r_{31}x_i + r_{32}y_i)v_i$$

$$\mathbf{b}' = \begin{bmatrix} 0.0 \\ 5.0 \\ 0.0 \\ 2.5 \\ -2.5 \end{bmatrix}$$

求解$\mathbf{A}'\mathbf{v} = \mathbf{b}'$得到向量$\mathbf{v} = [f, t_z]$。

$$f = -1.0$$

$$t_z = -7.5$$

因为f是一个负数，所以我们的坐标系不是右手坐标系。为了把z轴正过来，要改变r_{13}、r_{23}、r_{31}、r_{32}、f和t_z的正负号。最后结果是：

$$\mathbf{R} = \begin{bmatrix} 0.87 & 0.00 & -0.50 \\ 0.00 & -1.00 & 0.00 \\ -0.50 & 0.00 & 0.87 \end{bmatrix}$$

和

$$\mathbf{T} = \begin{bmatrix} -4.33 \\ -5.00 \\ 7.50 \end{bmatrix}$$

以及$f = 1$

由于所举的例子没有考虑畸变，上面便是标定的最后结果。图13-26从两个不同的视点对标定结果进行显示。

习题13.28

证明例子中得到的旋转矩阵**R**是标准正交的。

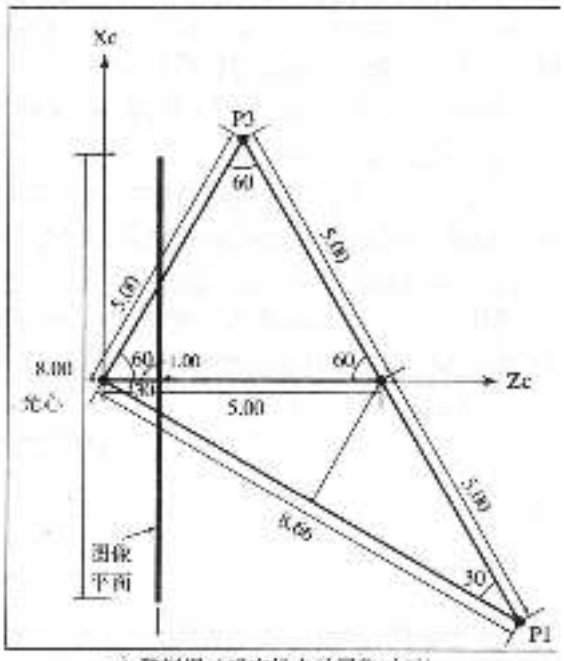
习题13.29 利用摄像机参数f

借助欧几里德几何原理和图13-26a，找出点P_1和P_3在图像平面上的投影。（图13-26a中的所有直线都是共面的）。验证得出的结果和例子中所给的p_1、p_3图像坐标相同。

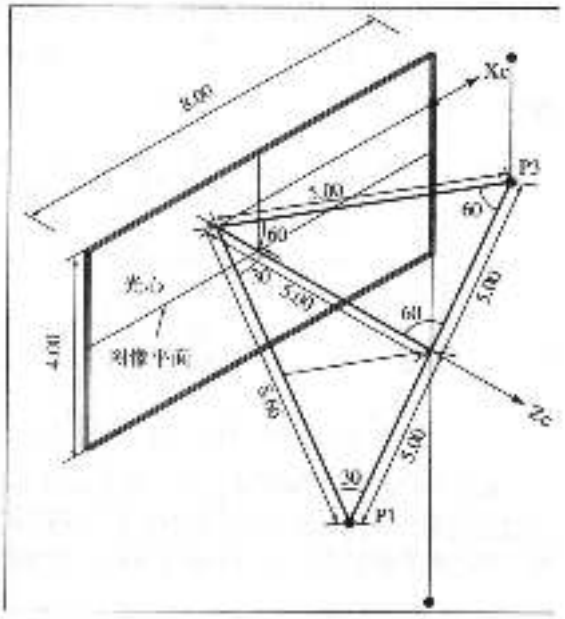
习题13.30 利用摄像机参数**R**和**T**

例子中给出的P_1和P_3相对的是世界坐标系，求它们在摄像机坐标系中的坐标。

- 利用欧几里德几何学和图13-26a。
- 利用标定得到的摄像机参数。



a）顶视图（用实线表示图像平面）



b）透视图

图13-26 摄像机和图像平面在世界坐标系下的两个视图，例子中利用Tsai的标定方法（Habib Abi-Rached提供）

13.8 位姿估计*

在工业视觉中，特别是机器人导航任务中，得到3D目标在工作区坐标系中位姿是非常重要的。由于摄像机在工作区的位姿可以通过标定算出，问题就变成要确定目标相对摄像机的位姿。本节给出的确定目标位姿的方法，其精度要比13.6节给出的简单方法高很多。位姿计
453 算中最基本的和最常用的方法是点对应方法。利用2D和3D线段的对应，2D椭圆和3D圆的对应，以及结合使用点对、线对、椭圆-圆对的内容，请参考Ji等人（1998）的工作。

13.8.1 2D-3D点对应求位姿

前面的一节中用到了摄像机模型，假设标定过的摄像机的内外参数都已经知道了。根据3D目标模型和2D图像之间的n对对应点确定目标位姿，这本质上是一个非线性问题。估计位姿参数要用非线性的的方法。但是在某些情况下，也能找到近似地线性解。

假设$[x, y, z]$是目标点${}^{M}\mathbf{P}$在模型坐标系中的坐标，物体坐标系到摄像机坐标系的变换关系为${}^{C}_{M}\mathbf{Tr} = \{\mathbf{R}, \mathbf{T}\}$，包括旋转矩阵$\mathbf{R}$和平移向量$\mathbf{T} = [t_x, t_y, t_z]$。然后将点${}^{M}\mathbf{P}$投影到图像平面上，产生投影坐标$[u, v]$，其中

$$u = f\frac{r_{11}x + r_{12}y + r_{13}z + t_x}{r_{31}x + r_{32}y + r_{33}z + t_z} \tag{13-75}$$

和

$$v = f\frac{r_{21}x + r_{22}y + r_{23}z + t_y}{r_{31}x + r_{32}y + r_{33}z + t_z} \tag{13-76}$$

其中f是摄像机的焦距，是已知的。

根据目标模型坐标系与摄像机坐标系之间地变换关系，能够得到目标在摄像机坐标系中的位姿。用前面的透视成像模型，在下列形式的12个方程中包含9个旋转参数和3个平移参数。

$$\mathbf{B}\,\mathbf{w} = \mathbf{0} \tag{13-77}$$

其中

$$\mathbf{B} = \begin{pmatrix} fx_1 & fy_1 & fz_1 & 0 & 0 & 0 & -u_1x_1 & -u_1y_1 & -u_1z_1 & f & 0 & -u_1 \\ 0 & 0 & 0 & fx_1 & fy_1 & fz_1 & -v_1x_1 & -v_1y_1 & -v_1z_1 & 0 & f & -v_1 \\ fx_2 & fy_2 & fz_2 & 0 & 0 & 0 & -u_2x_2 & -u_2y_2 & -u_2z_2 & f & 0 & -u_2 \\ 0 & 0 & 0 & fx_2 & fy_2 & fz_2 & -v_2x_2 & -v_2y_2 & -v_2z_2 & 0 & f & -v_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ fx_6 & fy_6 & fz_6 & 0 & 0 & 0 & -u_6x_6 & -u_6y_6 & -u_6z_6 & f & 0 & -u_6 \\ 0 & 0 & 0 & fx_6 & fy_6 & fz_6 & -v_6x_6 & -v_6y_6 & -v_6z_6 & 0 & f & -v_6 \end{pmatrix} \tag{13-78}$$

和

454 ~ 455

$$\mathbf{w} = (r_{11} \quad r_{12} \quad r_{13} \quad r_{21} \quad r_{22} \quad r_{23} \quad r_{31} \quad r_{32} \quad r_{33} \quad t_x \quad t_y \quad t_z)^T \tag{13-79}$$

如果想找到独立的位姿参数，而不仅仅求出模型点与图像点对应的仿射变换，就要对$\mathbf{R}$的元素附加条件，使$\mathbf{R}$满足实际旋转矩阵应满足的所有条件。特别地，旋转矩阵应是标准正交的，其行向量的幅值应等于1，行向量之间是正交的。用公式表示为:

$$\begin{aligned} \|R_1\| &= r_{11}^2 + r_{12}^2 + r_{13}^2 = 1 \\ \|R_2\| &= r_{21}^2 + r_{22}^2 + r_{23}^2 = 1 \\ \|R_3\| &= r_{31}^2 + r_{32}^2 + r_{33}^2 = 1 \end{aligned} \tag{13-80}$$

和

$$R_1 \circ R_2 = 0$$
$$R_1 \circ R_3 = 0 \tag{13-81}$$
$$R_2 \circ R_3 = 0$$

如果对**R**附加上这些条件后，问题就变成非线性的了。如果对**R**的行向量分别利用幅值约束条件，并独立进行计算的话，可以用线性约束优化技术计算**R**的行向量。（请参考Faugeras等1993年的文献，其中用到了类似的方法。）

13.8.2 约束线性最优化

对于公式（13-77）所示的系统，现在的问题是，求出使得$\|\mathbf{Bw}\|$最小，并且满足约束条件$\|\mathbf{w}'\|^2 = 1$的解向量**w**，其中这里**w**′是**w**中元素的子集。如果把这个约束施加到**R**的第一个行向量上，那么

$$\mathbf{w}' = \begin{pmatrix} r_{11} \\ r_{12} \\ r_{13} \end{pmatrix}$$

为了求解这个问题，有必要把原来的方程$\mathbf{Bw} = \mathbf{0}$重写为下列形式:

$$\mathbf{Cw}' + \mathbf{Dw}'' = \mathbf{0}$$

其中**w**″是由**w**中其余元素构成的向量。用上面的例子，把约束条件加到**R**的第1行，

$$\mathbf{w}'' = (r_{21} \quad r_{22} \quad r_{23} \quad r_{31} \quad r_{32} \quad r_{33} \quad t_x \quad t_y \quad t_z)^T$$

对于原来的问题，首先要使目标函数$\mathbf{O}=\mathbf{CW}'+\mathbf{DW}''$最小化，即

$$\min_{w',w''} \|\mathbf{Cw}' + \mathbf{Dw}''\|^2 \tag{13-82}$$

456

考虑约束条件$\|\mathbf{w}'\|^2 = 1$，用拉哥朗日乘子法，上面的式子就变为:

$$\min_{w',w''} \left[\|\mathbf{Cw}' + \mathbf{Dw}''\|^2 + \lambda(1 - \|\mathbf{w}'\|^2)\right] \tag{13-83}$$

求解上面的最小化问题。把目标函数分别对**w**′和**w**″求偏导数，并令其等于0。

$$\frac{\partial \mathbf{O}}{\partial \mathbf{w}'} = 2\mathbf{C}^{\mathrm{T}}(\mathbf{Cw}' + \mathbf{Dw}'') - 2\lambda\mathbf{w}' = 0 \tag{13-84}$$

$$\frac{\partial \mathbf{O}}{\partial \mathbf{w}''} = 2\mathbf{D}^{\mathrm{T}}(\mathbf{Cw}' + \mathbf{Dw}'') = 0 \tag{13-85}$$

从公式（13-85）可得

$$\mathbf{w}'' = -(\mathbf{D}^{\mathrm{T}}\mathbf{D})^{-1}\mathbf{D}^{\mathrm{T}}\mathbf{Cw}' \tag{13-86}$$

把公式（13-86）代入公式（13-84），得到

$$\lambda\mathbf{w}' = [\mathbf{C}^{\mathrm{T}}\mathbf{C} - \mathbf{C}^{\mathrm{T}}\mathbf{D}(\mathbf{D}^{\mathrm{T}}\mathbf{D})^{-1}\mathbf{D}^{\mathrm{T}}\mathbf{C}]\mathbf{w}' \tag{13-87}$$

可以看出，λ是下面矩阵的特征向量

$$\mathbf{M} = \mathbf{C}^{\mathrm{T}}\mathbf{C} - \mathbf{C}^{\mathrm{T}}\mathbf{D}(\mathbf{D}^{\mathrm{T}}\mathbf{D})^{-1}\mathbf{D}^{\mathrm{T}}\mathbf{C} \tag{13-88}$$

因此，要求的**w**′对应矩阵**M**的最小特征向量。对应的**w**″就可以直接通过公式（13-86）求出。应该注意地是，因为只对**R**的第一行施加幅值约束条件，所以得到的**w**″是不可靠的，这个结果还不能使用。但是求解向量**w**″提供了一个重要的信息，就是关于施加约束的行向量的

正负号。约束条件是$\|\mathbf{w}'\|^2 = 1$，而“$\mathbf{w}'$”的正负号并不受这个条件约束。所以要检查由$\mathbf{w}'$得到的解是否在物理上是可能的。特别地，平移t_z必须是正值，这样才能保证目标放在摄像机的前面。如果向量$\mathbf{w}''$中对应t_z的元素是负数的话，就意味着算出的$\mathbf{w}'$的幅值是正确的，但正负号不对，必须变号。那么$\mathbf{w}'$的最后表达式是:

457

$$\mathbf{w}' = \text{sign}(w''_9)\mathbf{w}' \tag{13-89}$$

13.8.3 计算变换Tr = {R,T}

首先通过上面计算的$\mathbf{w}'$得出行向量$\mathbf{R}_1$，这时$\mathbf{R}_1 = \mathbf{w}'$。矩阵$\mathbf{C}$和$\mathbf{D}$为:

$$\mathbf{C} = \begin{pmatrix} x_1 & y_1 & z_1 \\ 0 & 0 & 0 \\ x_2 & y_2 & z_2 \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ x_6 & y_6 & z_6 \\ 0 & 0 & 0 \end{pmatrix} \tag{13-90}$$

和

$$\mathbf{D} = \begin{pmatrix} 0 & 0 & 0 & -u_1x_1 & -u_1y_1 & -u_1z_1 & f & 0 & -u_1 \\ fx_1 & fy_1 & fz_1 & -v_1x_1 & -v_1y_1 & -v_1z_1 & 0 & f & -v_1 \\ 0 & 0 & 0 & -u_2x_2 & -u_2y_2 & -u_2z_2 & 0 & f & -u_2 \\ fx_2 & fy_2 & fz_2 & -v_2x_2 & -v_2y_2 & -v_2z_2 & 0 & f & -v_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & -u_6x_6 & -u_6y_6 & -u_6z_6 & f & 0 & -u_6 \\ fx_6 & fy_6 & fz_6 & -v_6x_6 & -v_6y_6 & -v_6z_6 & 0 & f & -v_6 \end{pmatrix} \tag{13-91}$$

然后用同样的方法，求行向量$\mathbf{R}_2$，并对$\mathbf{R}_2$进行幅值条件约束，这样$\mathbf{R}_2 = \mathbf{w}'$，矩阵$\mathbf{C}$和$\mathbf{D}$为:

$$\mathbf{C} = \begin{pmatrix} 0 & 0 & 0 \\ fx_1 & fy_1 & fz_1 \\ 0 & 0 & 0 \\ fx_2 & fy_2 & fz_2 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \\ fx_6 & fy_6 & fz_6 \end{pmatrix} \tag{13-92}$$

和

$$\mathbf{D} = \begin{pmatrix} fx_1 & fy_1 & fz_1 & -u_1x_1 & -u_1y_1 & -u_1z_1 & f & 0 & -u_1 \\ 0 & 0 & 0 & -v_1x_1 & -v_1y_1 & -v_1z_1 & 0 & f & -v_1 \\ fx_2 & fy_2 & fz_2 & -u_2x_2 & -u_2y_2 & -u_2z_2 & f & 0 & -u_2 \\ 0 & 0 & 0 & -v_2x_2 & -v_2y_2 & -v_2z_2 & 0 & f & -v_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ fx_6 & fy_6 & fz_6 & -u_6x_6 & -u_6y_6 & -u_6z_6 & f & 0 & -u_6 \\ 0 & 0 & 0 & -v_6x_6 & -v_6y_6 & -v_6z_6 & 0 & f & -v_6 \end{pmatrix} \tag{13-93}$$

如果用与$\mathbf{R}_1$、$\mathbf{R}_2$相同的求法求出$\mathbf{R}_3$，将不满足$\mathbf{R}_3$和$\mathbf{R}_1$、$\mathbf{R}_2$正交的条件。$\mathbf{R}_3$按如下方式计算:

458

$$\mathbf{R}_3 = \frac{\mathbf{R}_1 \times \mathbf{R}_2}{\|\mathbf{R}_1 \times \mathbf{R}_2\|} \tag{13-94}$$

这时$\mathbf{R}$的行向量几乎满足所有的约束条件，但只有一个：不能保证$\mathbf{R}_1$与$\mathbf{R}_2$是正交的。为了

解决这个不希望出现的情况，要对$\mathbf{R}_1$、$\mathbf{R}_2$和$\mathbf{R}_3$进行正交化处理，以保证旋转矩阵$\mathbf{R}$是标准正交的。这个可以像上面一样固定$\mathbf{R}_1$和$\mathbf{R}_3$并对$\mathbf{R}_2$重新计算如下：

$$\mathbf{R}_2 = \mathbf{R}_3 \times \mathbf{R}_1 \tag{13-95}$$

这样就算出了所有的旋转参数，而且它们满足必要的约束条件。用最小二乘法计算平移向量$\mathbf{T}$，采用新的、非齐次和超约束的12个方程：

$$\mathbf{A}\,\mathbf{t} = \mathbf{b} \tag{13-96}$$

其中

$$\mathbf{A} = \begin{pmatrix} f & 0 & -u_1 \\ 0 & f & -v_1 \\ f & 0 & -u_2 \\ 0 & f & -v_2 \\ \vdots & \vdots & \vdots \\ f & 0 & -u_6 \\ 0 & f & -v_6 \end{pmatrix} \tag{13-97}$$

和

$$\mathbf{b} = \begin{pmatrix} -f(r_{11}x_1 + r_{12}y_1 + r_{13}z_1) + u_1(r_{31}x_1 + r_{32}y_1 + r_{33}z_1) \\ -f(r_{21}x_1 + r_{22}y_1 + r_{23}z_1) + v_1(r_{31}x_1 + r_{32}y_1 + r_{33}z_1) \\ -f(r_{11}x_2 + r_{12}y_2 + r_{13}z_2) + u_1(r_{31}x_2 + r_{32}y_2 + r_{33}z_2) \\ -f(r_{21}x_2 + r_{22}y_2 + r_{23}z_2) + v_1(r_{31}x_2 + r_{32}y_2 + r_{33}z_2) \\ \vdots \\ -f(r_{11}x_6 + r_{12}y_6 + r_{13}z_6) + u_1(r_{31}x_6 + r_{32}y_6 + r_{33}z_6) \\ -f(r_{21}x_6 + r_{22}y_6 + r_{23}z_6) + v_1(r_{31}x_6 + r_{32}y_6 + r_{33}z_6) \end{pmatrix} \tag{13-98}$$

459

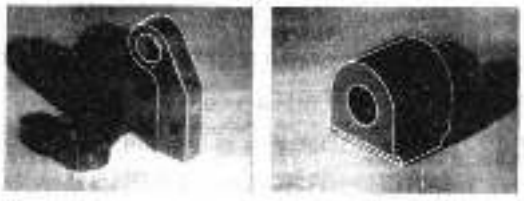
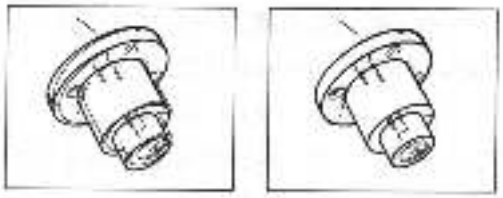


图13-27 利用约束线性优化方法，根据6个对应点计算姿态的实例（Mauro Costa提供）

13.8.4 位姿验证和位姿最优化

在评价位姿参数的质量时，应该有一种定量度量方法。在仿射标定方法中已经用到了一种度量方法，也就是模型位姿投影点与对应图像点之间距离的平方和。然而有的目标点被目标自己或者其他物体遮挡，这些点就要去掉。也可用其他类型的距离度量方法，例如豪斯多夫（Hausdorf）距离或者改进的豪斯多夫距离。（请参考Huttenlocher等人1993年的文献，以及Dubuisson与Jain1984年的文献。）也可用其他特征如边、角或孔等进行验证。

对位姿参数质量的度量可用于改进被估计的位姿参数。从概念上讲，我们能够对相差不大的参数做出评价，并保留最好的参数。假设每个旋转和平移参数分别有10个不同取值，采用笨方法进行最优搜索意味着将评价一百万套位姿参数，需要的计算量太大了，一般不这样做。非线性最优化方法，如牛顿方法或者鲍威尔方法（参考Press等人1992年的文献）可能会更快。图13-28显示了单目标图像的初始位姿估计，以及在初始解基础上进行非线性最优化处理后的结果。改进的位姿从观感效果上明显更好一些，这对于抓取目标来说比较有用，但对于识别来说就没有必要。



a) 初始位姿 b) 最终位姿

图13-28 非线性最优化前后的位姿情况（Mauro Costa提供）

13.9 3D目标重建

对3D建模来说3D测量是非常重要的。我们能够得到目标的深度图像，然后用来建立目标的计算机模型。这种3D目标重建的过程已经在医学和工业视觉方面得到应用，也用于建立虚拟现实环境所需的目标模型。这一节对目标建模进行一些必要的讨论，这也是下一章的主要内容。目标重建的过程共分为四个步骤：

1. 3D数据获取；
2. 图像配准；
3. 表面重建；
4. 优化。

460

在3D数据获取这一环节中，深度数据必须根据一系列物体表面的视图得到。一般8~10个视图就够了，但对于复杂物体或者是精度要求严格的情况下，还必须要有更多的视图。当然视图越多意味着计算量也就越大，因此并不是越多越好。

每幅视图是关于目标某部分的一幅深度图像，经常是配准后的灰度或者彩色图像。对所有视图的深度数据进行综合可得到目标的表面模型。亮度数据可用于图像配准过程，但真正重要的是在图形学纹理映射方面的应用，可以使目标视图更加贴近自然。把这些深度数据转换到一个3D坐标系的过程称为图像配准（registration）过程。

对这些数据做配准之后，就能看到3D点的云图（cloud of 3D points），但是要建立目标模型还需做很多工作。可以有两种方法表示3D目标：（1）用3D网格及格点间的连线表示出目标表面。（2）用一组3D体素表示出目标的整个体积。（参见第14章中关于这些表示方法的全面解释。）不同表示方法之间可以相互转换。

习题13.31 具有隐藏表面的物体

有的物体具有隐藏面，不管拍多少视图都看不到这些表面。简单画出一个这样的物体。为了简单，可以采用2D空间下的2D模型。

13.9.1 数据获取

利用最新扫描仪可以得到配准彩色图像的深度图像。我们介绍一种由现成商品组成的实验系统，并重点介绍基本操作过程。图13-29是一套专用的主动立体视觉系统，可用来得到深度数据和彩色数据。系统采用四个彩色视频摄像机，安装在铝杆上，摄像机与数字化电路板相连，数字电

461



图13-29 含4个摄像机的立体图像捕捉系统（Kari Pulli提供）

路板由计算机控制切换四路输入，产生640×480分辨率的图像。摄像机下面是一台投影仪，放在计算机控制的转盘上。投影仪发射出一条竖直的白色线形光带，可以手动调焦。光带照射到黑暗的室内，在得到深度图像后，打开两边的灯，拍摄彩色图像。

系统同时采用13.7节介绍的Tsai算法对摄像机进行标定。这个系统可以作为标准的两摄像机体视系统，或者是更稳健的四摄像机体视系统。无论那种情况，投影仪都用来发出竖直线形光带照射被扫描的目标。计算机控制转盘转动，光线以一定间隔从目标的左侧照射到右侧，间隔的多少可由用户选择，这样可以产生或低或高的分辨率。每到一个位置，摄像机拍摄黑暗中照射到目标上的光带图像。在每一幅图像上，光带与外极线的交点作为立体匹配的一点。图13-30显示基于两摄像机和一条光带的三角测量原理。两个相匹配的像素点确定3D空间中的一点。对于某条光带，沿光带对每个像素点计算相应的3D点，然后转动投影仪，投射一条新的光带，得到一幅新的图像，重复上述过程。结果就是一幅稠密深度图像。3D点和左图上的像素点对应，只要这个点对投影仪和右面的摄像机都是可见的。

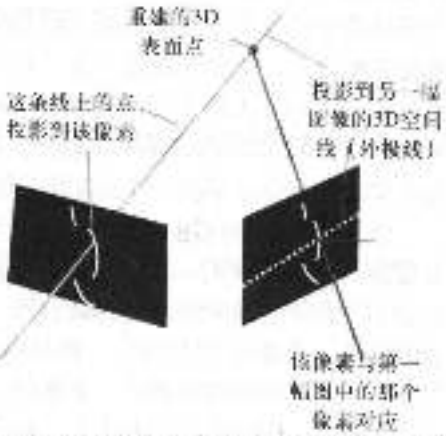


图13-30 两幅图像上光带与外极线的交点提供一对对应点（Kari Pulli提供）

用两台以上的摄像机可以提高图像捕捉系统的可靠性。一台摄像机作为基本摄像机，在该摄像机坐标系下计算深度图像。物体表面上的点，对于基本摄像机、投影仪和其他摄像机中至少一个必须是可见的。如果它只是对三台摄像机中的一个可见，那么系统便是两台摄像机立体视觉系统。如果它对于三台摄像机中的两台或者全部三台都是可见的话，冗余的图像就能使系统更加稳健。基本摄像机外加另两幅图像，我们就得到三个图像点，这样就有三个对应图像点参与三角计算，算出3D坐标。得出的三个结果有可能不同，但如果它们相差不是很大的话（也就是说，它们都在7mm³的体积范围内），就认为它们是有效的，可取算出的三个3D点的平均值作为最终结果。或者采用基线最宽的两摄像机测量结果，因为这个结果比其他两种情况更可靠。如果该点在全部四台摄像机中都是可见的，便有六种可能的组合。仍然可以检验他们是否都落在一个小的体积范围内，抛弃范围外的结果，采用结果平均值或者采用基线最宽的那对摄像机的测量结果。这样做的精度要高于只用一对固定摄像机的精度。（在测量车内的人体位置时，如图13-1所示，用这种方法计算x、y和z值的期望误差大约是2mm）。图13-31中显示利用该方法计算的玩具卡车的深度图像。该套3D卡车数据清楚地显示出卡车的外形。

462



图13-31 4摄像机主动体视系统得到的玩具卡车深度图像。为了看起来方便，深度点用亮度数据进行着色（Habib Abi-Rached提供）

13.9.2 视图配准

为了覆盖物体的整个表面，必须根据多幅视图得到深度数据。视图1到视图2的变换^{2}T，要么是通过精确的机械运动得到，要么是通过图像对应求出。如果用高精度设备，如标定好的机器人或者坐标测量机，来控制摄像机或者物体运动，那么系统就可以自动完成视图之间的变换。如果摄像机或物体的运动不是机器控制的，就必须有一种检测视图对应的方法，该
463 检测方法计算数据从一幅视图映射到另一幅视图的刚性变换。可以借助3D特征如角点和线段特征自动完成，基于这些特征可以得到一些3D-3D的对应点，从而算出变换关系。也可以通过交互的方式，比如允许用户在一对目标图像上选择对应点。无论那种情况，最初的变换都不会是完美的。机器人和测量机会产生伴随误差，当运动比较多时误差将会变大。自动寻找对应点的方法受到匹配算法的影响，也许会找到错误的对应点，或者特征不太准确。人工选点的方法也会出现误差，即使量化后能够找到正确的像素，但变换也有可能是错误的。

为了解决这些问题，多数图像配准方法采用迭代算法，初值采用估计的变换^{2}T。不管这个变换是怎么得到的，通过一个最小化策略对它不断修改。例如最近点迭代算法（ICP），使3D点^{2}T^{1}P和^{2}P之间的距离之和最小化，其中点^{1}P来自一幅视图，^{2}P是另一幅视图中与点^{2}T^{1}P相距最近的点。在此基础上的另一种方法是，在第二幅视图中寻找一点，沿从^{2}T^{1}P到表面的法线方向，在第二幅视图中插入一个邻域。（参见Chen和Medioni(1992)以及Dorai等人（1994）发表的文献）。当得到彩色数据时，就可以利用估计出的变换，把彩色数据从一个视图投影到另一个视图，并定义一个距离测度来表示它们的对齐程度，可以对这个距离进行迭代最小化，找出3D点之间的最佳变换。图13-32是用ICP算法对两幅沙发视图的配准过程。

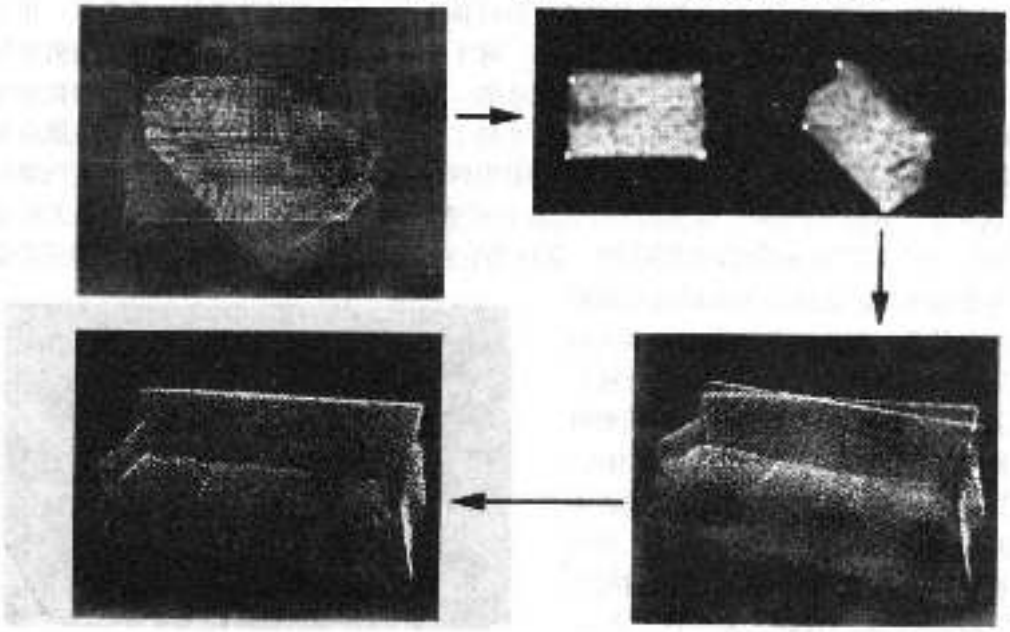


图13-32 （Kari Pulli提供）

（左上图）对两组深度数据进行配准

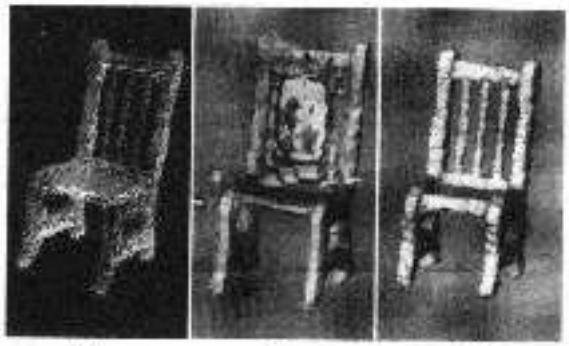
（右上图）左与深度视图对应的亮度图像上由用户选择四个点

（右下图）存在少量偏差的初始变换

（左下图）几次迭代后，两组深度数据得到很好地对齐

13.9.3 表面重建

一旦数据配准到同一个坐标系中，就可以开始重建工作。希望重建目标与实际物体在外形上尽可能相似并且保持其拓扑结构。图13-33显示在重建过程中可能出现的问题。配准的深度数据很密集，但是噪音干扰非常大。在实际椅子体外有多余的斑点，尤其是椅子的靠背部分。中间的重建结果十分粗糙。它把深度数据只看作是3D空间点的云图，而并没有考虑物体的几何形状以及深度数据点之间的邻接关系。在保持物体拓扑结构方面也很失败。右边的重建结果就比较好，它去掉了大部分噪声，并且保留了椅子靠背上的空隙。该重建是用空间切割（space-carving）算法生成的，下面介绍该算法。



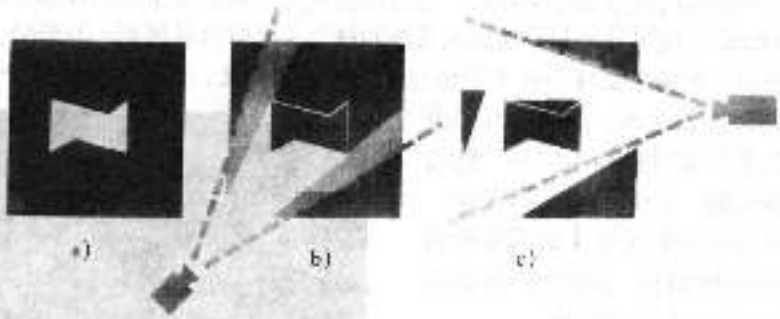
a) b) c)

图13-33 （Kari Pulli提供）

a) 配准的椅子深度数据
b) 重建过程出现的问题
c) 具有正确拓扑结构的粗略网络模型

13.9.4 空间切割算法

空间切割算法是由Curless和Levoy提出的，这里介绍的方法是由Pulli等人（1998）实现的算法。图13-34说明了它的基本思想。左图是要根据视图重建的目标。中图中一台摄像机在观察目标。根据点相对目标与摄像机的位置，把空间划分成不同的区域。目标的左侧和底部对 464
摄像机是可见的，扫描到的表面和摄像机之间的体积空间（浅灰色）位于目标前面，可以去掉。除了目标，如果还有背景数据，就可以去掉更多的体积空间（深灰色）。另外，目标后面的点不可以去掉，因为只有一台摄像机，不能告诉我们那些点是目标的一部分还是目标后面的空间。在右图中，另一台摄像机观察目标，可以切除更多的体积空间。通过足够多的视图就可以将大部分不想要的自由空间切除，只留下目标的体素模型。



a) b) c)

图13-34 空间切割思想（Kari Pulli提供）

a) 目标剖面图
b) 摄像机视图1能够去除浅灰色的空间
c) 摄像机视图2可以除掉其他一些空间（但还有一部分剩余）

空间切割算法将空间划分成小立方体或体素的集合，可以每次只处理一个体素。图13-35显示只有一幅视图如何确定立方体的状态。

- 在a情况下，立方体位于深度数据和传感器之间，因此这个立方体肯定不属于目标，不予考虑。

- 在b情况中，整个立方体位于深度数据后面。在传感器看来，立方体属于目标。
- 在c情况下，立方体既不完全在数据前面，也不是完全在数据后面，被认为是与目标表面交叉。

465 : 466

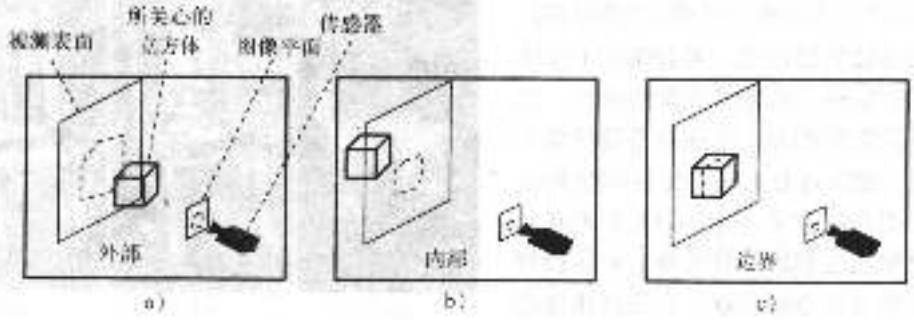


图13-35 与重建目标有关的立方体三种空间位置（Kari Pulli提供）

立方体标记过程实现如下：立方体的八个角投影到传感器的图像平面中，其中凸出的外形大致构成一个六边形。传感器到六边形的光线形成一个锥体，锥体的顶端被截去，这样它就包含该立方体。如果所有投影到六边形上的数据点位于被截锥体的后面（比传感器到立方体的最远角点还要远），立方体则位于目标之外；如果这些点比最近的立方体角点还要近，那么立方体便位于目标之内；否则的话，它便是一个边界立方体。

到现在为止，我们只看到一个立方体和一个传感器。而切割自由空间需要多个传感器（或视图）。对于所有传感器将上述立方体标记步骤都进行一遍。即使只有一个传感器告诉我们立方体位于目标的外部，那么就确定立方体位于目标之外。如果所有的传感器都表明立方体位于目标内部，这时才能说该立方体是目标的一部分。也许有视图能够说明立方体位于目标之外，但我们却没有这幅视图。第三种情况，如果立方体既不在目标内部也不在目标外部，那么它就是一个边界立方体。

用八叉树分层结构进行立方体标记，要比用固定大小的立方体更加有效。第14章将详细介绍八叉树（octree），这里只是简单使用八叉树的概念，比较容易理解。开始时，一个大立方体包围着深度数据。根据定义这个大立方体与深度数据相交叉，就将它分成八个较小的立方体。去掉位于目标外的立方体，而位于目标内的立方体可以看作是目标的一部分。然后对边界立方体作进一步的划分。继续这一过程直到产生期望的解为止。最后得到的八叉树就表示该3D目标。图13-36说明对椅子所进行的分层空间分割过程。



图13-36 分层空间切割，经过七次切割产生椅子的网格图（Kari Pulli提供）

为了看起来方便，八叉树可表示为如图13-36所示的3D网格形式。初始网格建立之后，通过使网络进一步简化并更好地与数据拟合，使网络得到优化。图13-37a显示配准的小狗深度数据，b是初始化网格，c~f是几步优化结果，这是Hoppe等人于1992年所做的工作，f是最后的网格图，

比开始的那幅更加简洁而且光滑。现在最后的网格图可用于图形学系统，进行目标实际视图的绘制，如图13-38所示；或者用于第14章所讲的基于模型的目标识别方面。 467

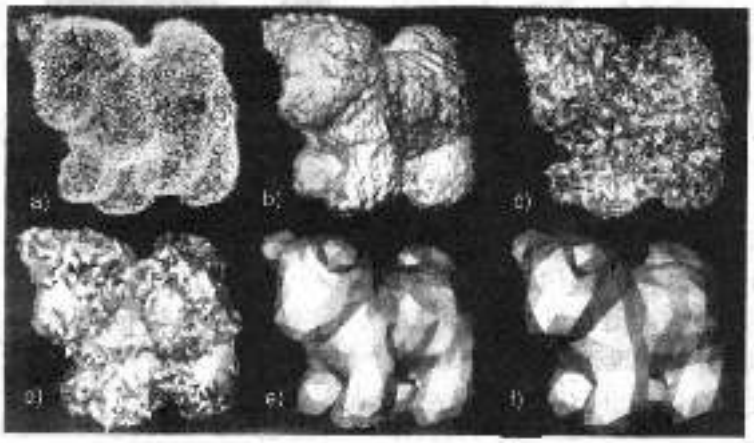
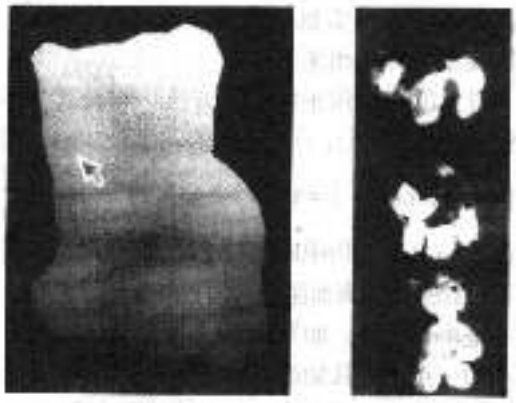


图13-37 配准的深度数据和建立小狗网格图的五个步骤（Kari Pulli提供）

13.10 从明暗恢复形状

第6章和第12章都讨论了具有光滑曲面的物体，如何对光线进行反射生成具有明暗效果的图像。下面我们简单介绍在一定条件下，如何根据图像的明暗效果计算目标的形状。

人们倾向于认为逐渐变深的表面逐渐远离我们的视线。在脸上使用化妆品，就可以改变别人对自己的感觉。把比脸部颜色更深的化妆品涂在脸颊外侧，会使得脸部看起来更窄一些，因为暗色调使我们感到表面离开视线的速度更快一些。同样道理，比脸部颜色浅的化妆品会产生相反的效果，给人一种脸部更丰满的感觉。使用朗伯反射公式，可将图像亮度变化映射成表面面元的法线方向。Horn和Bachman（1978）早期的工作，研究了月球拓扑结构的确定方法，遥远的太阳光线照射到月球上，并在遥远的地球上观察月球。这类方法已经发展成为从明暗恢复形状（shape from shading、SFS），即将图像中的明暗变化映射成场景中物体的表面方向。



a) b) 468

图13-38 （Kari Pulli提供）

a）小狗模型的伪彩色绘制图，用户可以对它进行操纵以选择一幅想要的视图

b）箭头所指的3D点表示小狗模型的鼻子，将它投影到三幅彩色小狗图像上，以选择像素点进行小狗实际颜色的绘制

定义109 从明暗恢复形状的方法，根据图像的明暗效果计算表面形状$\mathbf{n} = f(x, y)$，其中$\mathbf{n}$是表面在图像点(x, y)处的法线方向，$^{f}I[x, y]$是像素亮度。

图13-39说明了从明暗恢复形状的方法。左图是物体的一幅图像，它的表面基本上是朗伯反射表面，图像亮度与表面方向和光照方向的夹角成正比。右图显示了物体表面上几点处的表面方向。很明显，最亮的图像点说明了该点正对着光源，即图中X处指向我们的点的方向。

边缘点的表面方向与视线和表面边界垂直，这就完全限制了边缘点在3D空间中的方向。利用这些约束，表面方向可以传播到所有的图像点上，这就产生了部分本征图像。为了得到每个图像点处的深度z，我们可以给最亮的点赋值z_0，然后利用表面方向的变化将深度传播到图像的每一点。

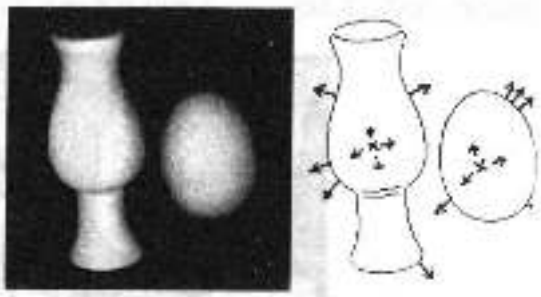


图 13-39

（左）朗伯表面物体的明暗效果图像，光源离摄像机比较近
（右）画出了边界处的表面方向，边界由Canny算子检测得到

习题13.32

假设朗伯表面的立方体，所有表面方向与光照方向至少成$\pi/6$夹角。很明显，最亮的图像点处的方向并不是指向光源的方向。为什么对鸡蛋和花瓶来说，最亮图像点处的方向是指向光源的方向，而对立方体来说却不是呢？

469

习题13.33

利用明暗分析，怎样才能知道目标边界是翼边还是刃边？

朗伯反射模型是$i=c\cos\theta$，其中常量c与光能、表面反照率、表面元素与光源和传感器之间的距离有关，这些因素都是常量因素。在距离上假设目标与光源及传感器的距离是目标直径的好几倍。经常也假设光照方向是已知的，但像上面看到的那样，光照方向有时根据较弱的假设条件计算出来。

目前的情况用正投影最方便。假设3D空间坐标$[x, y, z]$的参考坐标系是摄像机坐标系。被观测表面是$z=f(x, y)$。现在的问题是，对于每个图像点根据观测到的亮度值$^{x}I(x, y)$计算函数f的值。对$f(x, y)-z=0$进行微分可得$\frac{\partial f}{\partial x}\Delta x+\frac{\partial f}{\partial y}\Delta y-\Delta z=0$，其向量表达形式为$[p, q, -1]\circ[\Delta x, \Delta y, \Delta z]=0$，其中$p$和$q$分别表示$f$对$x$和$y$的偏微分。该等式定义了在表面点$[x, y, f(x, y)]$处的切平面方程，该点处的表面法线方向为$[p, q, -1]$，这不是单位向量。如果知道$[x_0, y_0, z_0]$是表面上的一点，并且知道$p$和$q$，那么上面的平面近似说明，可在该点附近的切平面内寻找近似的表面点$[x_0+\Delta x, y_0+\Delta y, z_0+\Delta z]$。只要能利用亮度图像和有关假设条件估计出$p$和$q$来，我们就可以做到这一点。

通过拍摄已知目标的图像，就能够将表面方向与亮度联系起来。对于图像亮度$I[x, y]$，只要知道点$[x, y, f(x, y)]$处的表面方向$[p, q, -1]$，就可以利用对应数据$\langle p, q, I[x, y]\rangle$计算映射。该映射使表面方向与它产生的明暗效果联系起来。图13-40显示了这种多对一的映射关系。所有与光照方向成θ角的表面将产生相同

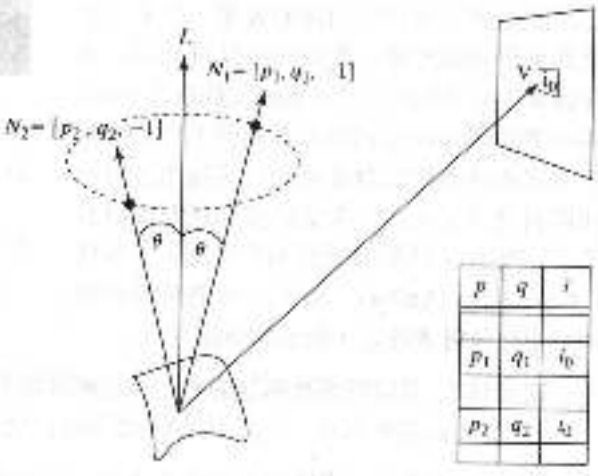
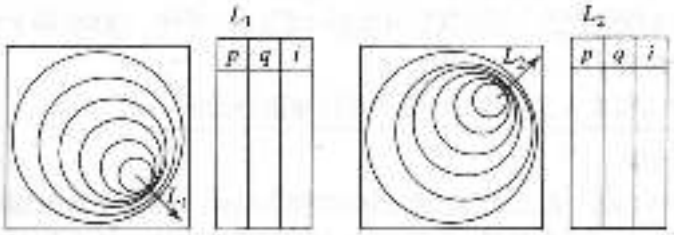


图 13-40

（左）沿锥筒的所有表面方向将产生相同的亮度
（右）反射映射表将表面方向和亮度联系起来，这是多对一的映射

的图像亮度。用来做标定的最好物体就是球体，因为（a）球体可以显示所有的表面方向，以及（b）根据图像点相对球心的位置和球的半径，可以很容易求出表面方向。图13-41显示球体的标定结果，采用两个不同的光源，对于每个光源，可以建立一个反射映射表，该反射映射表储存了产生某亮度的所有表面方向。 470

如图13-39和13-41所示，图像亮度对表面方位来说是一个很强的约束，但是并不能唯一地决定表面的法线方向，还需要其他约束条件。一般有两种方法：第一种就是利用空间邻域信息，例如，像素和它的4邻域产生5个明暗方程，对这些方程进行联立可求出通过这5点的光滑表面；第二种方法是采用多幅亮度图像，这样就可以利用一个像素的多个方程，而不考虑它的邻点。这种方法被称为光度立体（photometric stereo）。



a）照明来自右下方　　b）照明来自右上方

图13-41　用同一材质的朝伯标定球体作为研究目标，建立反射映射表。在被观测的球体图像中，对每个点$I(x, y)$利用解析几何求出p和q。我们可以将每个图像点$I(x, y)$处的对应数据$<p, q, I(x, y)>$插入映射表中。光源不同，得到的映射表就不同

习题13.34

已知标定球体，半径为r，位于摄像机坐标系$\mathbf{C}$:[x, y, z]的[0，0，100]处。根据图像位置[x, y]推导求p和q的公式。回想正投影是怎样去掉z坐标的。

13.10.1　光度立体

光度立体依次采用不同的光源照射目标，得到目标的多幅图像。对每个像素点都得到一组亮度值，然后通过查表得到相应的表面法线方向。如图13-41所示，该表通过离线的光度标定程序建立，算法13.4描述了这个过程。光度立体是一种快速方法，在受控环境中，这种方法十分有效。Ray等人（1983）指出，采用三个均衡的光源，即使对于反光物体也能得到很好的结果。但如果从明暗恢复形状需要严格控制环境的化，用结构光效果会更好，这是当前工业发展的趋势。 471

算法13.4　光度立体法：利用三个不同光源$^1\mathbf{L}$，$^2\mathbf{L}$，$^3\mathbf{L}$得到三幅图像$^1\mathbf{I}$，$^2\mathbf{I}$，$^3\mathbf{I}$，根据图像计算场景点的表面法线方向[p, q]

离线标定：

1. 把标定球放在场景中心。
2. 对于每一个光源$^i\mathbf{L}$，
   （a）打开光源$^i\mathbf{L}$。
   （b）拍摄标定球面的图像。

（c）建立反射映射$^j\mathbf{R}=\{<p, q, {}^j\mathbf{I}[\mathbf{x}, \mathbf{y}]>\}$，其中（$p$, q）与图像$^j\mathbf{I}$的亮度$^j\mathbf{I}[\mathbf{x}, \mathbf{y}]$对应。

在线表面测量：

1. 将被测目标放在场景中心。
2. 分别使用每个光源$^j\mathbf{L}$，快速拍摄三幅图像$^j\mathbf{I}$。
3. 对于每个图像点$[\mathbf{x}, \mathbf{y}]$
   （a）使用亮度$\mathbf{i}_j={}^j\mathbf{I}[\mathbf{x},\mathbf{y}]$检索反射映射表$^j\mathbf{R}$，并且访问与亮度$\mathbf{i}_j$对应的方向集$\mathbf{R}_j=\{(p, q)\}$。
   （b）取三个集合的交集$\mathbf{S}=\mathbf{R}_1\cap\mathbf{R}_2\cap\mathbf{R}_3$。
   （c）如果$\mathbf{S}$为空的话，那么赋值$\mathbf{N}[\mathbf{x},\mathbf{y}]$=NULL，否则，将$\mathbf{S}$中的平均方向向量赋值给$\mathbf{N}[\mathbf{x}, \mathbf{y}]$。
4. $\mathbf{N}[\mathbf{x}, \mathbf{y}]$作为存放表面法线方向的本征图像部分返回。

13.10.2 结合空间约束

人们提出了几种方法，通过对空间邻域应用明暗约束，确定光滑表面的函数$z=f(x, y)$。一种方法就是前面提到的，从最亮的图像点向表面传播的方法，最小化方法寻找符合约束条件的最佳函数。图13-42是最小化算法求得的一个结果，用网格表示算出的表面，其中两个合成目标和一个实际目标。结果的好坏，与这些数据对应的任务有关。这种方法在实际应用中不是很可靠。

从明暗恢复形状的研究工作证明，明暗信息对于表面形状来说是较强的约束。这是统计算机视觉问题一个很好的例子，输入，输出以及假设都进行了很清楚的定义。在有的情况下，很多数学算法都能产生很好的效果，但没有一项工作能在各种场景下都产生很好的效果。感兴趣的读者可以阅读参考文献，以深入了解这方面的内容，尤其是那些数学算法，在这里我们只做了简单介绍。

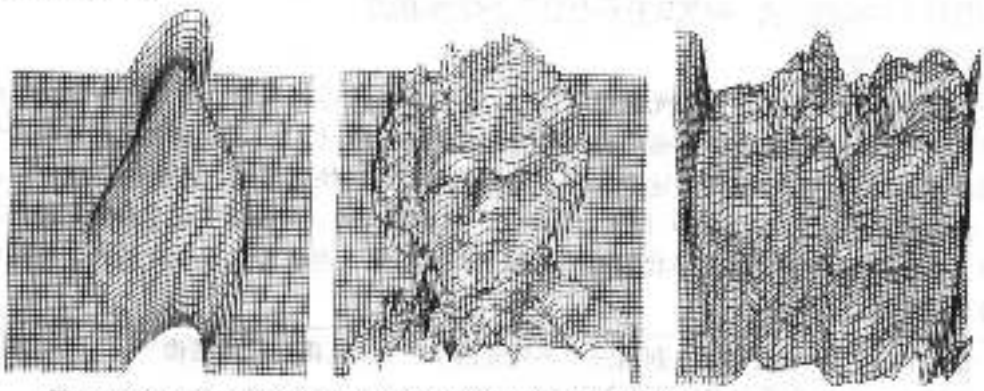


图13-42 Tsai-Shah算法分别用在合成图像和实际图像的结果（Mubarak Shah提供）

（左）对凸面的CAD模型应用漫反射光照模型产生图像，由图像算出表面

（中）从合成的莫扎特半身像得到的表面

（右）从青椒的实际图像得到的表面

13.11 从运动恢复结构

人类通过在环境中的运动感知到大量3D结构的信息。当我们或者目标产生运动时，我们

能够从图像序列中获取信息。利用流向量或者对应点，就可以重建3D场景的表面和角点，以及传感器在场景中的运动轨迹。这个直观过程可以改进为一类专门定义的数学问题。利用计算机视觉算法计算场景结构以及目标与观察者的运动，这种构建问题进行起来非常困难。研究虽然在平稳进步但是进展非常缓慢。

图13-43是一般情况，其中观察者和场景中的目标都可能在运动。目标和观察者之间的相对运动在图像中形成流向量。这些可以通过点匹配或光流的方法算出来。图13-44显示五个3D点有很大差异的两幅视图。文献中的不同情况，体现在问题定义和实现的算法上的不同。

472
~
473

问题定义中用到的3D目标也许是

- 点
- 线
- 平面片
- 曲面片

在一定假设条件下，算法不仅应该产生3D目标的结构，还应得到目标在摄像机坐标系中的运动情况。已有的许多算法都假设3D目标已被可靠地测量和匹配。测量和匹配是十分困难的，容易产生误差，目前为止很少有令人信服的演示例证。基于图像流的算法，拍取图像的时间间隔很小，试图计算稠密的3D结构。而基于特征对应的算法，可以容忍较长的时间间隔，但只能得到稀疏的3D结构。

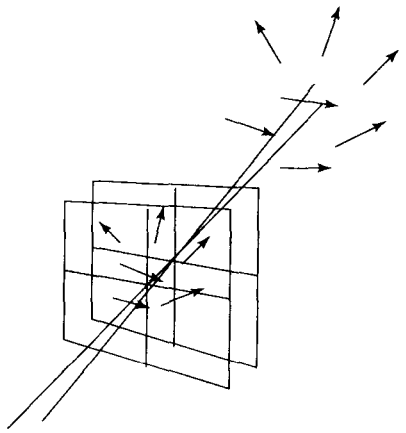


图13-43 场景中观察者和目标同时运动。3D点的运动投影成前后两幅图像间的2D流向量

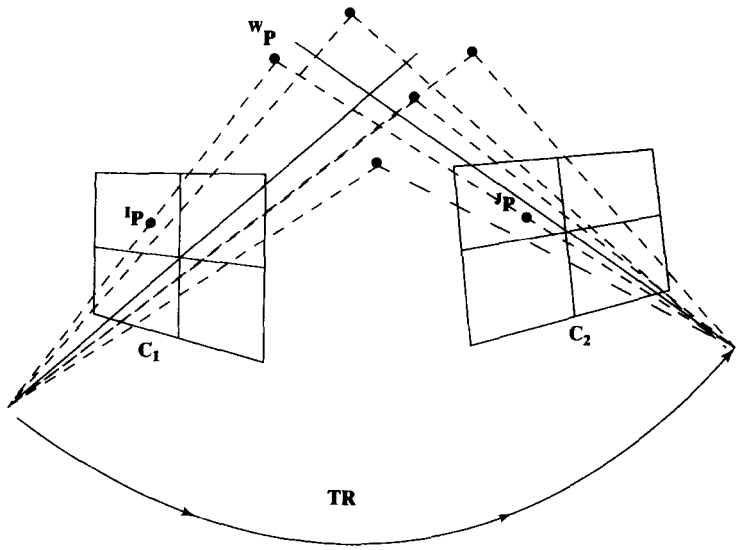


图13-44 场景中目标静止，观察者运动。3D目标点$^{W}\mathbf{P}$投影到两幅图像上形成2D图像点$^{I}\mathbf{P}$和$^{J}\mathbf{P}$，由于摄取图像的时间和空间差异很大，造成点的对应十分困难。如果找到了图像对应点，问题就变成计算相对运动**TR**和3D点$^{W}\mathbf{P}$的坐标

Ullman（1979）的早期工作表明，对于四点刚体结构，采用一台固定摄像机，根据这四点的三个正投影，从理论上可以算出四点刚体的结构和运动。十年之后，Huang和Lee（1989）证明只用两个正投影解决不了这个问题。对于从运动恢复形状，极简化的数学模型有趣但求解起来非常困难。由于噪声和错误匹配容易带来误差，所以它们并不是很实用。Haralick和Shapiro（1993）论述了几种数学途径，并且说明计算方法的稳健性。Brodsky等人（1999）利用移动的摄像机观察静态场景，计算出稠密的目标形状，给出了比较好的实验结果。在本书第1章，我们曾经提问是否能根据巴黎圣母院的视频建立它的3D模型。这就是一个从运动恢复结构的问题。而且现在商业上已经有了这样的计算机视觉系统。对这些方法的总结，可以参考Faugeras等人（1998）发表的论文。我们介绍了从运动恢复结构的一般性问题以及几个方面的内容，建议希望深入研究的读者阅读已出版的相关文献。

习题13.35 改进算法13.4

把交集计算放在离线阶段，可以提高算法13.4的效率。证明为什么可以这么做。选用什么样的数据结构才适合在线过程储存结果？

13.12 参考文献

仿射摄像机的标定方法参考的是Ballard和Brown（1982）以及Hall等人（1982）的早期工作。后者还提到采用一套标定好的摄像机和投影仪的结构光系统。现有几种行之有效的摄像机标定方法。对于目标识别，采用仿射透视模型甚至是弱透视模型精度常常就够用了。但对于检测或者是精确的位姿计算，就需要考虑径向畸变的模型，也就是应用广泛的Tsai方法，参见Tsai（1987）。很多机器视觉应用需要进行标定，但也有很多情况不需要标定，就像我们用视频摄像机扫描外景一样。我们不知道某时某点的焦距是多少，也不知道相对世界坐标系的位姿参数。但人类却能根据这些图像感知出世界的3D结构。假设只用了透视投影，3D结构就可以在不知道比例因子的情况下算出来。Faugeras等人的工作（1998）说明了如何根据图像序列建立建筑物的纹理映射3D模型。Brodsky等人（1999）给出了更一般表面结构的计算结果。

474
|
475

我们的P3P解法参考了Ohmura等人（1988）的工作。Linnainmaa等人（1988）的类似工作几乎在同一时间进行，但是要注意到Fischler和Bolles（1981）曾经研究过同样的问题，并且公布了封闭形式的解法。在图像序列中跟踪一个目标时，迭代求解方法只有一定的优势，因为可以利用一个起始点，该起始点能帮助去掉错误的解。另一种合适的模型是Huttenlocher和Ullman（1988）提出的弱透视投影模型，这是一种很好的近似方法，并且推导过程十分有建设性。Fischler和Bolles（1981）第一次正式定义并且研究了N点透视问题，并且给出了P3P封闭形式的解。他们同时说明了如何使用这种方法：首先假设N个对应点，计算出目标位姿，然后证明其他模型点在图像上可以找到对应点。他们把这个算法称为RANSAC，因为是随机选择对应点。如果能够得到特征点的属性，就应该避免这种随机性。

最近几年里在实验室受控环境下，人们在根据多幅视图建立物体3D模型方面做了大量的工作，开发出了很多系统和程序。我们的目标重建系统是由Pulli等人（1998）在华盛顿大学开发的。

1. Ballard, D., and C. Brown. 1982. *Computer Vision*. Prentice-Hall.
2. Ballard, P., and G. Stockman. 1995. Controlling a computer via facial aspect. *IEEE-Trans-SMC*, April 1995.

3. Brodsky, T., C. Fermuller, and Y. Aloimonos. 1999. Shape from video. *Proceedings of IEEE CVPR 1999,* Ft Collins, Co. (23–25 June 1999), 146–151.
4. Chen, Y., and G. Medioni. 1992. Object modeling by registration of multiple range images. *Int. J. Image and Vision Computing,* v. 10(3):145–155.
5. Craig, J. 1986. *Introduction to Robotics Mechanics and Control.* Addison-Wesley, Reading, MA.
6. Curless, B., and M. Levoy. 1996. A volumetric method for building complex models from range images. *ACM Siggraph '96,* 301–312.
7. Dorai, C., J. Weng, and A. Jain. 1994. Optimal registration of multiple range views. *Proc. 12th Int. Conf. Pattern Recognition,* Jerusalem, Israel (Oct. 1994), v. 1:569–571.
8. Dubuisson, M.-P., and A. K. Jain. 1984. A modified Hausdorff distance for object matching, *Proc. 12th Int. Conf. Pattern Recognition,* Jerusalem, Israel.
9. Faugeras, O. 1993. *Three-Dimensional Computer Vision, a Geometric Viewpoint.* The MIT Press, Cambridge, MA.
10. Faugeras, O., L. Robert, S. Laveau, G. Csurka, C. Zeller, C. Gauclin, and I. Zoghlami. 1998. 3-D reconstruction of urban scenes from image sequences. *Comput. Vision and Image Understanding,* v. 69(3):292–309.
11. Fischler, M., and R. Bolles. 1981. Random concensus: a paradigm for model fitting with applications in image analysis and automated cartography. *Communications of the ACM,* v. 24:381–395.
12. Forsyth, D., and others. 1991. Invariant descriptors for 3-d object recognition and pose. *IEEE Trans. Pattern Analysis and Machine Intelligence,* v. 13(10):971–991.
13. Hall, E., J. Tio, C. McPherson, and F. Sadjadi. 1982. Measuring curved surfaces for robot vision. *Computer,* v. 15(12):385–394.
14. Haralick, R. M., and L. G. Shapiro. 1993. *Computer and Robot Vision, Volume II.* Addison-Wesley, Reading, MA.
15. Heath, M. 1997. *Scientific Computing: An Introductory Survey.* McGraw-Hill, Inc., New York.
16. Hoppe, H., T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. 1992. Surface reconstruction from unorganized points. *Proc. SIGGRAPH '92,* 71–78.
17. Horn, B. K. P., and B. L. Bachman. 1978. Using synthetic images to register real images with surface models. *CACM 21* v. 11:914–924.
18. Hu, G., and G. Stockman. 1989. 3-D surface solution using structured light and constraint propagation. *IEEE-TPAMI,* v. 11(4):390–402.
19. Huang, T. S., and C. H. Lee. 1989. Motion and structure from orthographic views. *IEEE Trans. Pattern Analysis and Machine Intelligence,* v. 11:536–540.
20. Huttenlocher, D., and S. Ullman. 1988. Recognizing solid objects by alignment. *Proc. DARPA Spring Meeting,* 1114–1122.
21. Huttenlocher, D. P., G. A. Klanderman, and W. J. Rucklidge. 1993. Comparing images using the Hausdorf distance. *IEEE Trans. Pattern Analysis and Machine Intelligence,* v. 15(9):850–863.
22. Ikeuchi, K., and B. K. P. Horn. 1981. Numerical shape from shading and occluding boundaries, *Artificial Intelligence,* v. 17(1–3):141–184.
23. Ji, Q., M. S. Costa, R. M. Haralick, and L. G. Shapiro. 1998. An integrated technique for pose estimation from different geometric features. *Proc. Vision Interface '98,* Vancouver (June 18–20), 77–84.
24. Johnson, L. W., R. D. Riess, and J. T. Arnold. 1989. *Introduction to Linear Algebra.*

476

Addison-Wesley, Reading, MA.

25. Linnainmaa, S., D. Harwood, and L. Davis. 1988. Pose determination of a three-dimensional object using triangle pairs. IEEE Trans. Pattern Analysis and Machine Intelligence, v. 10(5):634–647.
26. Ohmura, K., A. Tomono, and A. Kobayashi. 1988. Method of detecting face direction using image processing for human interface. *SPIE Visual Communication and Image Processing,* v. 1001:625–632.
27. Pulli, K., H. Abi-Rached, T. Duchamp, L. G. Shapiro, and W. Stuetzle. 1998. Acquisition and visualization of colored 3D objects. *Proceedings of ICPR '98,* 11–15.
28. Ray, R., J. Birk and R. Kelley. 1983. Error analysis of surface normals determined by radiometry. *IEEE-TPAMI,* v. 5(6):631–644.
29. Shrikhande, N., and G. Stockman. 1989. Surface orientation from a projected grid. *IEEE-TPAMI,* v. 11(4):650–655.
30. Tsai, P.-S., and M. Shah. 1992. A fast linear shape from shading. *Proceedings IEEE Conf. Comput. Vision and Pattern Recognition* (June 1992), 734–736.
31. Tsai, R. 1987. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf cameras and lenses. *IEEE Trans. Robotics and Automation,* v. 3(4).
32. Ullman S. 1979. *The Interpretations of Visual Motion.* MIT Press, Cambridge, MA.
33. Vetterling, W. T. 1992. *Numerical Recipes in C.* Cambridge University Press, New York.
34. Zhang, R., P.-S. Tsai, J. Cryer, and M. Shah. 1999. Shape from shading: a survey. *IEEE-TPAMI,* v. 21(8):690–706.

第14章　3D模型和匹配

无论是计算机视觉还是计算机图形学都要用到3D目标的模型。计算机图形学中，目标必须用便于绘制和显示的结构表示出来。最常见的结构是3D网格结构，它是由3D点和连接这些点的边构成的多边形集合。与图形相关的硬件一般都支持网格表示。对于更平滑和更简单的表面，其他图形表示方法还有二次曲面、B样条表面和细分表面。除了3D形状信息外，图形表示还可以包含颜色和纹理信息，然后通过图形硬件把这些信息纹理映射到被绘制的目标。图14-1显示的是玩具狗的粗略3D网格模型，以及相同视点的纹理映射绘制图像。

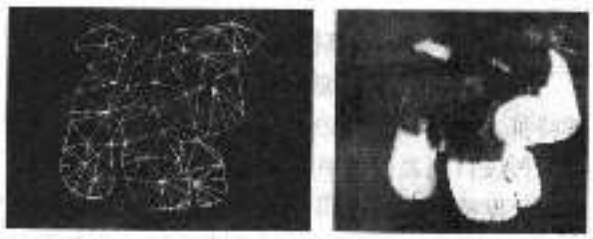


图14-1　玩具狗的3D网格模型和纹理映射绘制图像（Kari Pulli提供）

对于计算机视觉来说，目标表示必须符合目标识别的要求，这意味着目标表示和从图像中抽取的特征之间必须有一定的对应关系。3D目标识别中有几种常用的图像类型，如灰度图像、彩色图像和深度图像。经常需要把灰度图像或彩色图像配准到深度数据，这样可以给识别算法提供更丰富的特征集。大多数3D目标算法只是为特定的表示设计的，并不能推广到处理不同的特征，因此在讨论3D目标识别前，有必要了解一下通用的表示方法。总的来说，几何表示要用到点、线和面等；图符表示要用到单元成分以及它们之间的空间关系；功能表示要用到功能部件以及部件间的功能关系。我们首先讨论3D目标表示的最常用方法，然后介绍常见目标识别算法中用到的表示方法。

14.1　模型表示

计算机视觉开始于Robert在1965年进行的多面体识别工作，其中使用了简单的线框模型，以及与从图像中的直线段进行匹配。基于线段的模型今天仍然很流行，但也有其他一些模型能更精确地表示曲面甚至任意表面的目标数据。本节我们研究网格模型、表面-边-顶点模型、体素和八叉树模型、广义圆柱体模型、超二次曲面模型以及可变形模型。还要考虑真正3D模型和特征-视模型的不同之处，特征-视模型用一组2D视图来表示3D目标。

14.1.1　3D网格模型

3D网格是一种简单的几何表示，通过相连的顶点和边构成3D空间多边形来描述目标。任意多边形可构成任意结构的网格。由类型相同的多边形构成的网格是规则网格（regular mesh）。常用的三角形网格（triangular mesh）全部由三角形组成，图14-1就是一个三角形网格。网格可以用不同的分辨率表示目标物体，从粗略估算到很高的细节分辨均可。图14-2显示同一条狗不同分辨率的三个网格模型，它们可用于图形绘制或者利用深度数据进行目标识别。当用于识别时，要定义特征抽取算子，目的是从用于匹配的深度数据中抽取特征。在第14.4.1节对这些特征进行讨论。



图14-2 不同分辨率的小狐狸模型（Kari Pulli提供）

14.1.2 表面-边-顶点模型

因为早期的3D视觉系统大多处理多边形目标，在识别或位姿估计中，边就成了主要的局部特征。仅由目标的边和顶点组成的3D目标模型，称为线框（wire-frame）模型。线框表示中，假设目标表面是平面并且目标只含直线边。

线框模型广泛应用于计算机视觉中，它的一个推广形式是*表面-边-顶点*（surface-edge-vertex）模型。表面-边-顶点表示是一个数据结构，包括目标的顶点、表面和边，通常还包括一些拓扑关系，说明表面在边哪一侧，以及顶点在边的哪一端。当目标是多边形的时，表面是平面，边是直线段。这个模型也可以推广到包含曲边和曲面。

480

图14-3举例说明了表面-边-顶点数据结构，它在3D目标识别系统中表示目标模型的数据库。这个数据结构是分层的，最高层是世界，然后不断向下到最低层的表面和弧。图14-3的方框中带标记的字段{name，type，⟨entity⟩，transf}，表示⟨entity⟩类集合中的元素，集合中的每个元素都有名字、类型、指向⟨entity⟩的指针和3D变换。对⟨entity⟩进行3D变换将产生一个旋转和平移实例。例如世界有一个*object*集合，在这个集合中命名了不同的3D目标模型实例。任何给出的目标模型都在自己的坐标系中进行了定义，通过变换可以单独确定实例在世界坐标系中的位置。

每个目标模型都包括三个集合：边，顶点和面。顶点有一个相关的3D点和相交于此点的边的集合。边有起点、终点，左边的面、右边的面，如果不是直线边，还要有一条弧定义边的形态。面有一个定义其形状的表面和包含其外边界和孔边界的边界集合。边界有一个相关的面和边的集合。这里没有定义最低层的实体—弧，表面和点、表面和弧的表示与应用背景以及所需的精度与平滑性有关。它们可以用公式表示，或者进一步分解为表面片和弧段。点仅仅是坐标为(x, y, z)的向量。

481

图14-4显示一个简单的3D物体，可用表面-边-顶点方式进行表示。为了简单起见，只讨论几个可视表面和边。可视表面是F1、F2、F3、F4和F5，其中F1、F3、F4和F5是平面，F2是圆柱面。F1可用一条圆弧表示的边确定边界，F2需要两条这样的边界线确定。F3的边界由四条直边组成的外边界和一条圆弧构成的孔边界确定。F4和F5的边界都由四条直边组成的单一类型边界线确定。边E1把面F3和F5分开。如果把顶点V1作为边的起点，V2为终点，那么F3是这条边的左面，F5是右面。顶点V2有三条相关联的边即E1、E2和E3。

482

习题14.1 表面-边-顶点结构

使用图14-3的表示法，构造图14-4所示目标的整体模型，对3D目标的每个面、边及顶点进行命名，并在结构中使用这些名字。

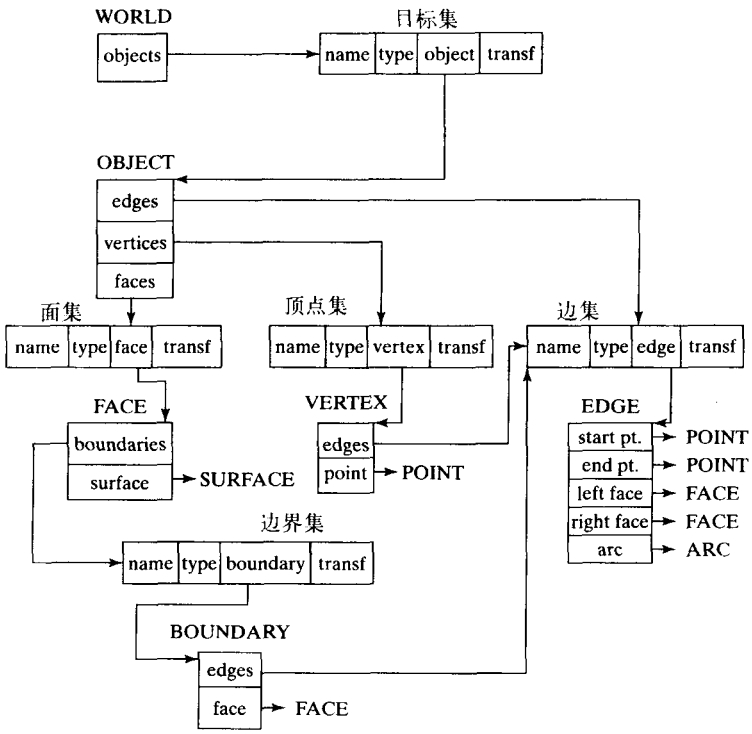


图14-3 表面-边-顶点数据结构

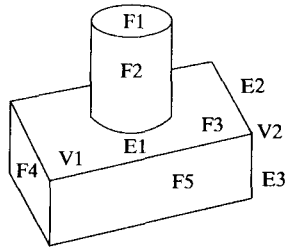


图14-4 平面和圆柱面组成的3D目标

14.1.3 广义圆柱体模型

广义圆柱体（generalized cylinder）是一种体积基元，用一条空间曲线轴和轴上各点处的截面函数进行定义。截面沿轴向密排形成旋转体。例如一般圆柱体是广义圆柱体，它的轴是直线段，截面是半径恒定的圆；圆锥体是广义圆柱体，它的轴是直线段，截面是圆，其半径从轴的一个端点以零开始增长，在另一个端点达到最大值；长方体是广义圆柱体，它的轴是直线段，截面是相同的矩形；圆环体是广义圆柱体，它的轴是圆，截面是相同的圆。

目标的广义圆柱体模型，包括广义圆柱体描述、广义圆柱体间的空间关系以及目标的全局特性。圆柱体可以用轴长度、平均截面宽度、两底面之比以及锥角进行描述。连接关系是最常见的空间关系。除了端点连接关系外，圆柱体之间也可能连在一起，使得一个圆柱体的端点成为另一个圆柱体的内部点。在这种情况中，可以用连接参数来描述这种连接关系，如圆柱体相接触的位置、倾角以及描述一个绕另一个旋转的环绕角。目标的全局特性可能包括圆柱体块数、细长圆柱体块数和连接的对称性。也可以用分层的广义圆柱体模型，其中在不

同层上表示不同细节的模型。例如可把人体粗略地模为棒状图（如图14-5所示），由表示头部、
483 躯干、手臂和腿部的圆柱体组成。在下一层，躯干可能分为脖子和躯干部分；手臂可分为三个圆柱体，分别表示上臂、前臂和手；腿也类似。再下一层，手可以分为手掌和五根手指，继续分的化，手指可以分为三节，当然拇指是两节。

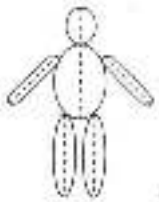


图14-5 人体的广义圆柱体粗略模型。虚线表示圆柱体的轴

三维广义圆柱体投影到图像中会产生两种不同的二维效果，即条带和椭圆。条带（ribbon）是圆柱体长度方向的投影，而椭圆（ellipse）是截面的投影。当然截面不一定是圆，所以投影也不一定是椭圆。某些广义圆柱体是完全对称的，所以没有长短之分。对这种情况，现在有算法能够从感模目标的图像中寻找条带。这些算法一般是寻找含有轴信息的长形区域。图14-6显示从2D形状确定广义圆柱体曲线轴的过程。



图14-6 从2D形状构造广义圆柱体（Gerard Medioni提供）

图14-7显示的是，为了制作合身的衣服而建立特定人体精细模型的步骤。在特定的测量环境中进行，还要从12个摄像头得到输入图像。6个摄像头均匀分布在2m的圆柱体空间，拍摄人体图像。其中一套安装位置较低，另一套安装位置较高，这样就可以拍摄2m高的人体。如图14-7所示，从6个摄像头得到的侧面轮廓用于拟合椭圆截面，从而获得圆柱体模型。除了要计算剪面轮廓上的点外，还要使用栅格光线通过三角测量计算3D表面上的点。用结构光数据算出凹陷处的有关数据，最终算出精细的三角网格模型。

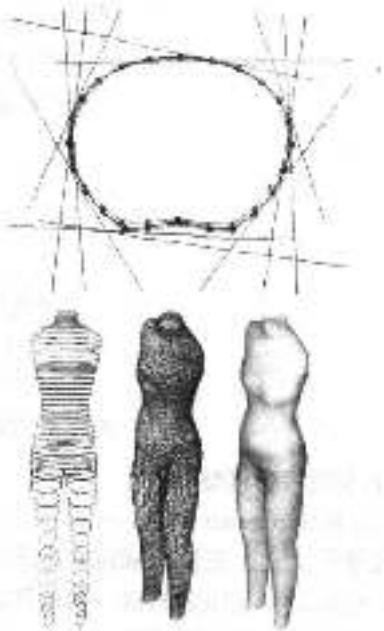


图14-7 为了制作合身的服装而建立人体模型的步骤。（上图）三条截面轮廓曲线。数据来自6个摄像头所拍的图像（直线表示侧面到摄像头的投影范围）。结构光特征使我们能够算出凹陷处的3D点位置。（底部）把椭圆截面与6个侧面轮廓拟合形成的广义圆柱体模型，三角形网格图和演染后的图像（由Helen Shen和香港科技大学计算机科学系的同事提供，项目得到香港工业技术发展委员会AF/183/97资助，中国SAR 1997）

习题14.2 广义圆柱体模型

构造飞机的广义圆柱体模型。飞机要有机身、机翼和机尾。每个机翼上都应附着一个发动机。尝试描述广义圆柱体之间的连接关系。

14.1.4 八叉树

八叉树（octree）是分层次的八叉树结构，树中的每个节点对应一个立方体区域。如果立方体完全包含于三维目标中，

那么对应节点标记为*full*；如果立方体不包含目标的任何部分，则标记为*empty*；如果立方体部分地与目标相交，则标记为*partial*。标记为*full*或*empty*的节点没有子节点；标记为*partial*的节点有八个子节点，分别代表这个立方体的八个部分。 484

可用$2^n \times 2^n \times 2^n$三维数组表示三维目标，其中$n$是整数，数组的元素称为体素（voxel），其值为1（满）或0（空），表示目标存在或者不存在。目标的八叉树编码等价于三维数组表示，但通常需要更少的空间。图14-8给出了目标和它的八叉树编码的简单示例，其中使用的是Jackins和Tanimoto（1980）的八分编号方式。 485

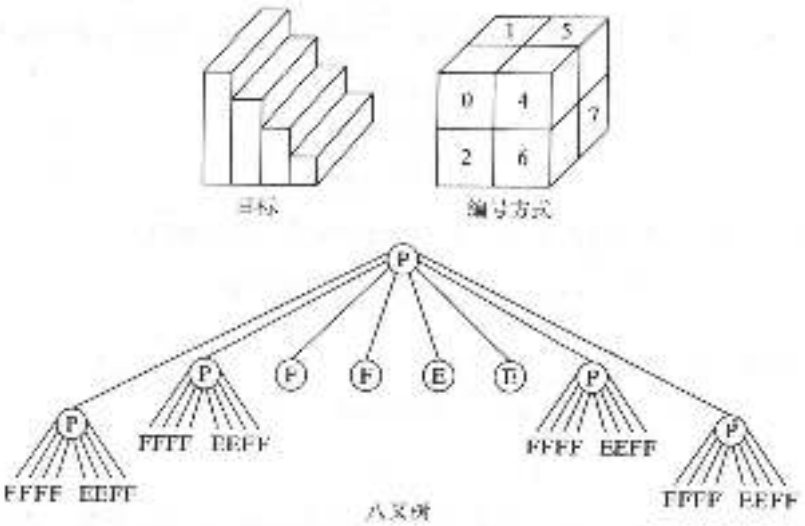


图14-8 三维目标及其八叉树编码的示例

习题14.3 八叉树

图14-11显示一把椅子的两幅视图，构造这把椅子的八叉树模型。假定座和靠背都是4voxels × 4voxels × 1voxel，每条腿是3voxels × 1voxel × 1voxel。

14.1.5 超二次曲面模型

超二次曲面模型最初是用在计算机图形学方面，后来经Pentland引入到计算机视觉领域。可以直观的把超二次曲面看作粘土块，能够通过变形和粘合形成目标模型。数学上，超二次曲面构成形状的参数化族，超二次曲面可以用向量S定义，其x、y和z元素分别是角度η和ω的函数，公式如下：

$$\mathbf{S}(\eta,\omega)=\begin{bmatrix}x\\y\\z\end{bmatrix}=\begin{bmatrix}a_1\cos^{\varepsilon_1}(\eta)\cos^{\varepsilon_2}(\omega)\\a_2\cos^{\varepsilon_1}(\eta)\sin^{\varepsilon_2}(\omega)\\a_3\sin^{\varepsilon_1}(\eta)\end{bmatrix} \tag{14-1}$$

其中$-\frac{\pi}{2}\leqslant\eta\leqslant\frac{\pi}{2}$，$-\pi\leqslant\omega<\pi$，参数$a_1$、$a_2$和$a_3$分别表示超二次曲面在$x$、$y$和$z$方向的尺寸，参数$\varepsilon_1$、$\varepsilon_2$表示在经度面和纬度面上的方度。

超二次曲面可以表示建筑物的某些部分，比如球体、椭球体、圆柱体、平形六面体和中间部分的形状。当ε_1和ε_2都是1时，生成的表面是椭球表面；如果$a_1=a_2=a_3$，则是球面。当$\varepsilon_1<1$和$\varepsilon_2=1$时，生成的表面是圆柱体表面。 486

超二次曲面不仅能够表示出完美的几何形状模型，而且能够表示变形后的几何形状，如经过锥化（tapering）和弯曲（bending）变形的几何形状。沿着z轴的线性锥化由以下变换给出：

$$x' = \left(\frac{k_x}{a_3}z + 1\right)x$$

$$y' = \left(\frac{k_y}{a_3}z + 1\right)y$$

$$z' = z$$

其中k_x和$k_y(-1 \leqslant k_x, k_y \leqslant 1)$分别是x和y平面关于z方向的锥化参数。弯曲变形由以下变换定义：

$$x' = x + \cos(\alpha)(\mathbf{R} - r),$$

$$y' = y + \sin(\alpha)(\mathbf{R} - r),$$

$$z' = \sin(\gamma)\left(\frac{1}{k} - r\right)$$

其中k是曲率，r是x和y元素在弯曲平面$z-r$上的投影，由以下公式给出：

$$r = \cos\left(\alpha - \tan^{-1}\left(\frac{y}{x}\right)\right)\sqrt{x^2 + y^2},$$

R是r的变换，由以下公式给出：

$$\mathbf{R} = k^{-1} - \cos(\gamma)(k^{-1} - r),$$

γ是弯曲角度

$$\gamma = zk^{-1}.$$

超二次曲面模型主要用来拟合深度数据，目前已有几种表面拟合的超二次曲面参数恢复方法。图14-9用超二次曲面拟合心脏左心室5个时刻的3D数据。这些是带参数函数的二次曲面扩展模型，其中参数不是常数而是函数。

487



图14-9 心脏收缩过程中，5个时刻的左心室拟合模型。使用了带参数函数的超二次曲面扩展模型（Jinah Park和Dimitris Metaxas提供）

14.2 实际3D模型与视类模型

上述的目标表示中，强调的都是目标的三维性质，而忽视了根据任意视点2D图像进行目标识别的问题。多数目标从不同视点观看，其结果是不同的。圆柱体从一个视点看可以投影为条带（参见14.1.3节），从另一个视点看又可以投影为矩阵。一般来讲，视点空间可以划分成视类（view classe）（又称为特征视）的有限集合。每个视类表示具有相同属性的视点的集合。这个属性可以是从这些视点能看到的目标相同表面，或者是能看到的相同线段，或者关系结构间的相关距离足够小，其中关系结构是从这些视点的线条图中抽取的（参见第11章）。将产生具有拓扑同构性线条图的视点分为一组，图14-10显示确定出的几个立方体视类。图14-11显示椅子的两幅视图，其中大部分可见面是相同的面。利用由区域基元确定的视图间的

相关距离，以及与封闭性有关的区域邻接关系，可以通过聚类算法将这些视图分为一类。许多不同但类似的视图构成一个视类，视图的数量应该是无限的。关键是一旦确定了目标的正确视类，为计算位姿所做的对应匹配运算就有了较强的约束，而且是二维匹配。

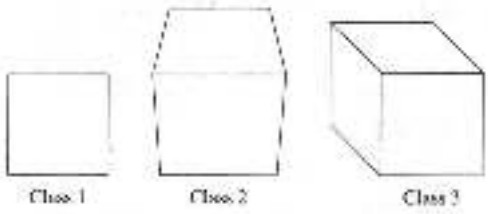


图14-10 立方体的三个视类，将产生拓扑同构性线条图的视点分为一类

488

视类由Koenderink和van Doorn（1979）提出。他们把提出的结构称为表象图（aspect graph）。从一组相连的视点看到的有本质差异的目标视图，就称为表象（aspect）。表象图的节点表示表象，相邻表象间用弧线连接。两表象边界上的外观变化称为视觉事件（visual event）。自动构造表象图的算法出现于80年代后期，但由于实际目标的结构非常庞大，所以这些算法并未广泛用于目标识别。相反，视类或者特征视的概念得到广泛使用。

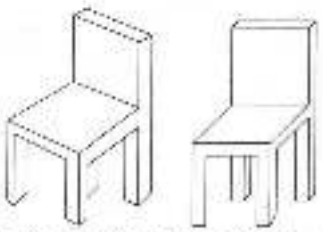


图14-11 属于同一视类的两幅椅子视图，根据低层的相关距离进行分类

习题14.4 视类模型

图14-11的两幅椅子视图属于三维目标的同一视类。画出椅子更常见的三个视类。

14.3 物理学模型和可变形模型

物理学模型（physics-based model）可用来建立被拍摄物理目标的外观模型和行为模型。本节所给的例子中，一个是人类心脏模型（参见图14-16），一个是电话听筒模型（参见图14-15）。可利用物理学原理建立实际物理系统的模型，或者用来模拟图像分析任务。建立心脏模型，目的是模拟目标随时间的形状变化和行为变化，从而了解心脏的活动情况；建立电话听筒模型，目的是获得静态测量的网格模型。

与物理学模型密切相关的一个术语是可变形模型（deformable model）。后者强调建立目标形状的变化模型。

物理学模型和可变形模型近期进展很快，这两个方向无论是在理论方面还是在应用方面都有很丰富的研究内容，它们比本书所涉及的内容要复杂得多。这里只做简单介绍，主要目的是让大家对该领域有所了解，并在课外主动阅读最新发表的文献。

14.3.1 蛇形活动轮廓模型

多数人都曾把橡皮筋套在伸出的手指上，手指可以看作2D空间中的五个点，橡皮筋则是通过五个点的封闭轮廓。橡皮筋的行为可以看作是活动轮廓（active contour），活动轮廓在图像中向最小能量状态的方向运动。橡皮筋趋于收缩以释放存储的能量，直到遇到支撑（手指）为止。图14-12（右）说明了这个原理，深色小区域好比我们的手指，这些小区域阻挡了橡皮筋的收缩。另一方面，橡皮筋不会无限收缩，即使只有一个点（或线）阻挡，橡皮筋也会均匀缠绕在该点周围。在模拟过程中能够限制高度弯曲。图14-12（左）显示气球膨胀时将会出现的情况，其中手指像抓球一样把气球抓在手中。类似的，可以设想一个虚拟气球在图像区域或图像点内膨胀。

489 图14-12说明了活动轮廓的一个重要优点：尽管要拟合的数据被分成片段，但轮廓结构仍然是完整的。更进一步，也可以得出其他特性，如平滑性、周界范围和简单曲线的特性。现在简单介绍如何用计算机算法模拟活动轮廓的行为。

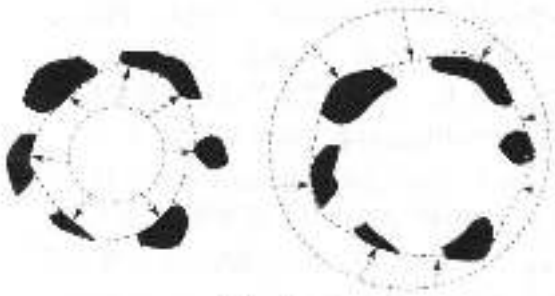


图 14-12
（左）2D气球走活动轮廓膨胀以适合2D数据点
（右）2D橡皮筋在2D数据点外伸展

为了模拟活动轮廓的行为，首先需要一个存储器状态以确定轮廓的结构和位置。考虑简单情况，在时刻t，有确定的N点集合，每个点位于$P_{i,t}$，与邻点$P_{i-1,t}$和$P_{i+1,t}$相连成环形。对于虚拟的橡皮筋，每个点受到来自两邻点的拉力，使点$P_{i,t}$加速移动到新位置$P_{i,t+\Delta t}$。图14-13（左）说明了这一点。一般认为每个点都有单位质量，这样很容易算出由力产生的加速度。由加速度可算出速度，由速度可算出位置。因此，t时刻存储器状态还应该包括每个点的加速度和速度，而且在仿真的初始状态这些加速度和速度可以不为零。还需要另一个数据成员，即布尔变量，用它指明是否因碰到数据点（称为硬约束（hard constraint））而使该点的运动停止。当然除了活动轮廓，还需要存储要处理的数据，可能是灰度图像、2D边缘点集、3D表面点集等。其表示方式可以用本章、第2章或第10章中介绍的方法。

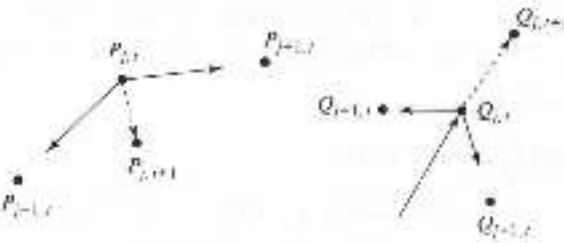


图 14-13
（左）拉伸橡皮筋上某点的力，使该点趋于向内运动
（右）气球上某点的膨胀力，如果超过来自邻点的弹力，就使该点趋于向外运动

490 活动轮廓上点运动的简单算法见算法14.1。除非轮廓点遇到硬约束或受到的合力为零，否则轮廓点会一直运动。算法可能永不终止，例如用活动轮廓跟踪正在讲话的两片嘴唇时！注意需要有轮廓的初始位置。

算法14.1 轮廓上的点$P_{i,t}$移动到下一个位置$P_{i,t+1}$的更新步骤

输入：t时刻N个数据点；每个$P_{i,t}$都有速度$V_{i,t}$和加速度$A_{i,t}$。

输出：$t+1$时刻N个数据点；每个$P_{i,t+1}$都有速度$V_{i,t+1}$和加速度$A_{i,t+1}$。

时间间隔Δt，对每个未因硬约束而停止的点$P_{i,t}$进行计算：

1. 用$P_{i,t}$的邻点计算$P_{i,t}$所受的合力。
2. 用合力计算加速度向量$A_{i,t+1}$。
3. 计算速度$V_{i,t+1} = V_{i,t} + A_{i,t}\Delta t$。
4. 计算新位置$P_{i,t+1} = P_{i,t} + V_{i,t}\Delta t$。
5. 如果$P_{i,t+1}$在数据点的允许范围内，就锁定这个位置。

算法14.1是欧拉算法的一个简单步骤。对于很小的时间间隔，欧拉算法根据力计算加速度，根据加速度计算速度，根据速度计算位置。当运动点遇到数据点、边或面片时，它的位置就

固定下来。该算法的计算代价一般很大，因为为了寻找这样的点需要对数据结构或图像进行搜索。

胡克定律建立了弹簧模型，它是物理学模型的常用组成部分。假设自然长度是L的弹簧连接点P_j和点P_k。作用在P_j上的力F与弹簧的伸长量（压缩量）成正比。

$$F = -k_L(\|P_j - P_k\| - L)\frac{P_j - P_k}{\|P_j - P_k\|} \tag{14-2}$$

这作为前面橡皮筋的模型足够了。如果弹簧系统无限振荡，就应该添加一个阻力。剩下的问题就是确定合适的长度L。如果建立已知目标的模型，如正在讲话的嘴唇，那么就能够确定N、L和k_L的实际值。K_L是刚度系数，它表示力与形变的关系。

能量最小化公式*

尽管以前有人用过活动轮廓的概念，但Kass、Witkin和Terzopoulos1987年的论文激起了计算机视觉领域对活动轮廓的兴趣。上面的多数讨论参考了他们称为“snake”的活动轮廓思想。用活动轮廓对数据进行拟合是一个最优化问题，即寻找服从某些硬约束的最小能量边界。一种实验方法是把如下三部分之和做为总能量：（1）内部轮廓能量（internal contour energy），由轮廓本身的拉伸和弯曲决定；（2）图像能量（image energy），它说明轮廓与图像亮度和梯 491
度的拟合程度；（3）外部能量（external energy），由约束力产生。约束信息由用户以交互的方式提供，或者由更高级的计算机视觉处理过程提供。

用$s\in[0, 1]$作为参数的轮廓表示为$v(s)=[x(s), y(s)]$，它是实际变量s的函数。问题是要找到这样的函数使如下定义的能量最小。

$$E_{contour} = \int_0^1 (E_{internal} + E_{image} + E_{constraints})ds \tag{14-3}$$

$$E_{internal} = \alpha(s)|\mathbf{v}'(s)|^2 + \beta(s)|\mathbf{v}''(s)|^2 \tag{14-4}$$

在每个点和活动轮廓间的距离平方上加上$E_{constraints}$，可以控制活动轮廓在某些指定点附近通过。E_{image}只是活动轮廓上的点与最近边缘点间的距离平方的和。内部能量的定义更加有趣。$E_{internal}$的第一部分限制小轮廓线段长度的较大变化，因为较低的能量意味着长度变化较小。第二部分限制曲率的大小。权重函数$\alpha(s)$和$\beta(s)$起调和作用，也允许形成尖角，或在柔和纹理上产生长纹理跳变。

活动轮廓对图像的拟合可用机翼或独木舟的制造情况进行说明。图14-14中，要把一根木条按一定间隔钉到横杆上，而横杆固定在硬壁上。木条平滑弯曲以适应横杆的空间分布，这样就形成光滑但可能复杂的一条曲线。与横杆接触相当于硬约束。因为木条在很多点上分配弯曲能量，这样就避免出现高曲率情况。通过计算机算法可以很容易生成这样的样条曲线，实际上图14-14就是用xfig工具中的算法生成的。

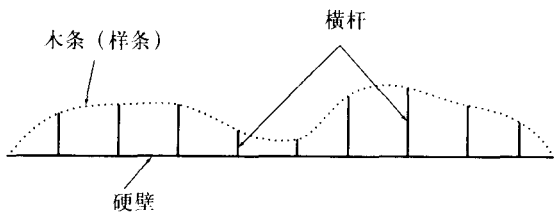


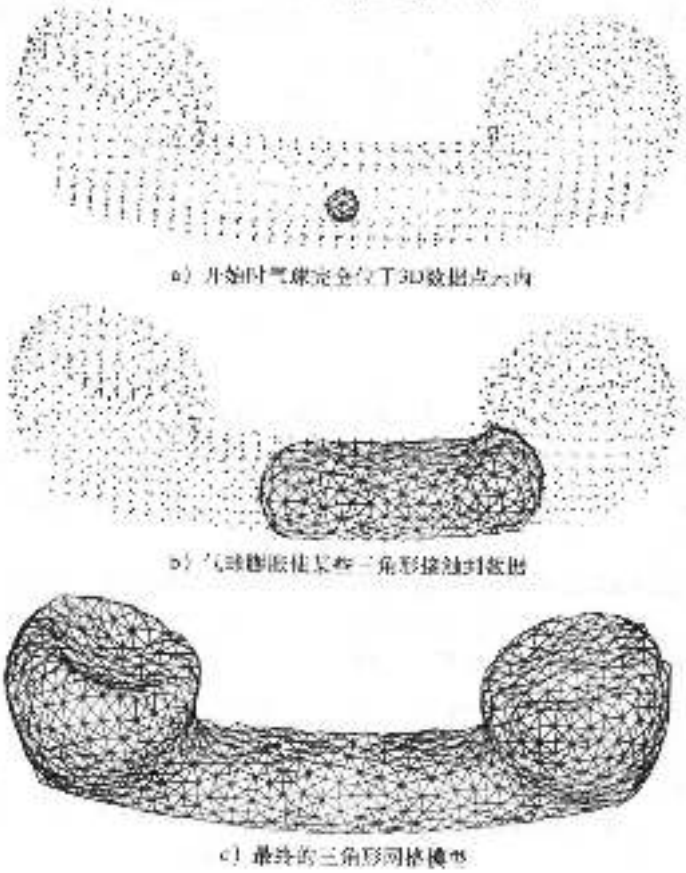
图14-14 附着在横杆上的木条形成低能量轮廓（光滑样条用xfig得到）

轮廓能量最小化的方法已经超出了本书的讨论范围。要控制好活动轮廓需要认真进

行数值编程。可以使用专业有限元数值分析软件包。1987年之后，有些研究工作采用了动态规划方法，代替Kass等人（1987）提出的尺度空间方法。感兴趣的读者可以在参考文献中找
492 到很多有意思的研究工作。

14.3.2 3D气球模型

气球模型可以是近似球体的网格模型。大多数美式足球由12块五边形或者20块六边形构成，这些形状可以划分成三角形。三角形的边可以采用弹簧模型，这样通过扩展或者收缩能够改变整个系统的形状。图14-15显示这样的球体模型。模型通过从内部扩展3D数据点云而构
493 成，这些点是电话听筒的测量数据。当某个顶点接触到测量数据时，算法就锁定该顶点的位置。膨胀力作用于每个顶点，其方向由内向外沿着表面法线指向顶点。为了检测与测量数据的接触情况，只需沿着法线方向搜索数据。当膨胀三角形变得足够大时，算法就把它细分成四个三角形。通过这种方式，球可以膨胀成伸长的数据形状，如图14-15b-c所示。通过距离扫描仪得到目标表面的不同视图，就能够算出3D数据点的位置，所有点都刚性变换到全局坐标系中。想像一下把不同表面网络缝合到一起的难度有多大，其中这些网格分别是从不同的视图得到的。当气球模型发生形变以拟合数据时，它能保持正确的拓扑结构和近似不变的三角形分布。只利用3D点集就想建立很好的表面模型是很困难的。



a）开始时气球完全位于3D数据点云内

b）气球膨胀使某些三角形接触到数据

c）最终的三角形网格模型

图14-15 用物理学方法进行三角形网格膨胀的三幅示意图，三角形网格拟合3D数据云的过程（Yang Chen 和 Gerard Medioni 提供）

14.3.3 建立心脏跳动模型

三角形网格常用来建立表面模型，而四面体可用于建立3D体积模型。每个四面体元素有四个顶点、四个面和六条边。基于材料性质对边分配硬度（stiffness）值。当模型上不同的点有力作用时，结构形状会发生变化。图14.16显示跳动心脏的两个状态，它们是根据标记磁共振图像算出来的。传感器可以标记活动组织的某些部分，这样可以测量它们的3D运动。心脏模型与数据吻合，表明模型反映了真正的物理过程和心脏跳动情况。拟合的模型点的运动可以解释心脏是如何工作的。模型四面体元素的形变，与所受的力和所模拟的组织硬度有关。

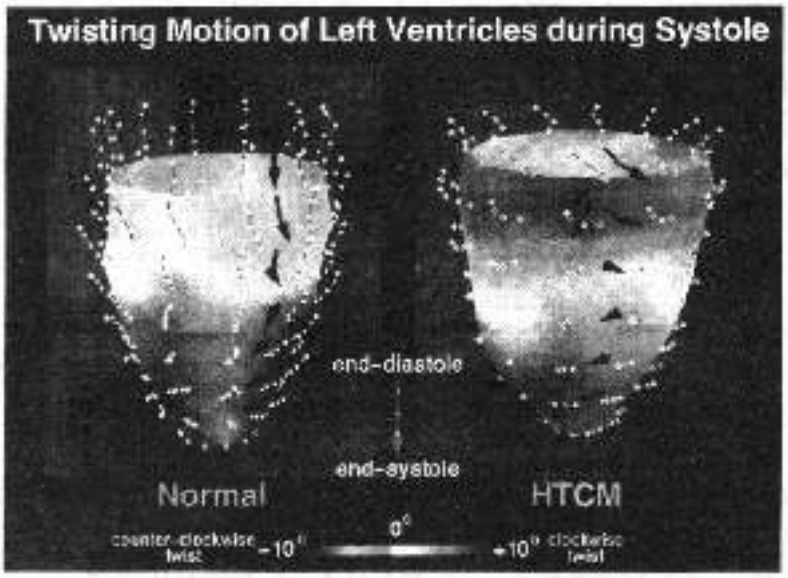


图14-16 跳动心脏的两个运动情况，由标记磁共振图像数据计算得出。传感器可以标记活动组织并测量它的3D运动。心脏模型与数据吻合，表明模型反映了真正的物理过程和心脏跳动情况。两个心脏的运动向量是不同的（Jinah Park和Dimitris Metaxas 提供）

14.4 3D目标识别范例

考察过不同的3D目标模型后，现在讨论3D目标识别最常用的几个范例。这实际上难度很大，因为所用的方法与应用、数据类型以及识别任务的要求密切相关。有几方面的因素可用来对目标识别问题进行分类或进行约束。这些因素包括：

- **我们的兴趣在于工程学还是认知科学？** 如果只是想得到直接面向应用的工程解，那么这个问题可能非常明确以至于很简单，就像要从混乱的钢柱堆中抓出一根。如果我们的兴趣是要理解人类的目标识别功能，这就意味着要在一般理论上有所发展，要与多种心理学数据相一致，这是一个困难得多的问题。
- **任务中涉及的是自然目标还是人造目标？** 人造目标通常比自然目标更规则，很多人造目标有刚性图标原型，具有已知的匹配范例可用。自然目标经历了长期复杂的自然变化过程（如地质的、生物的等等），这些变化的模型很难建立。而且，与人造目标所处的环境相比，自然目标的环境具有更少的约束，更难进行预测。例如室外环境自动导航的目标识别问题，要比工厂自动化中的目标识别与位姿确定问题困难的多。

- **目标表面是多面体、二次曲面还是自由形态的表面？**很多识别方案仅仅处理多面体表面，这使建模过程变得非常简单。最近研究人员们开始转向二次曲面，据称它可以模拟大约85%的人造目标。使用二次曲面最主要的好处是，可以用相同基元（可能带有参数拟合）对模型数据和测量数据进行描述。现在还不清楚用二次曲面建立刚体雕刻目标、自由形态目标的效果如何。雕刻目标，如跑车、涡轮叶片或者冰山，可能有很多不同的平滑弯曲的表面特征，这些特征很难分离为简单基元。
- **场景中只有一个目标还是有很多目标？**某些目标识别方案假定要识别的目标是单独存在的。在工程任务中，这有时是可能的，有时是不可能的。多目标环境要困难得多，因为
495 目标特征存在遮挡和混杂现象。全局性特征仅对单个目标是有效的。在多目标环境中分割问题是很重要的问题。
- **识别的目的是什么？**识别目标的目的可能是为了检查、抓取或避让。对于检查，至少要察看目标的部分细节，模型和测量精度必须足够高。抓取物体则有不同的要求。抓取任务不仅需要粗略的模型几何知识，还要考虑平衡、力度以及目标在工作空间中的可接近性。对于路径上的障碍物，机器人识别的目的是为了避让，这时只需要识别出目标大致的尺寸、形状和位置即可。
- **测量到的是2D数据还是3D数据？**人类对一只眼睛看到的图像就能运用自如。很多研究者设计的系统只使用2D亮度图像作为输入。通过视图变换，建立起2D图像特征和3D模型之间的关系。所以匹配过程需要找出这个变换而且进行目标识别。如果能够得到3D数据，则匹配就变得容易得多，这就是目前研究人员热衷于研究深度数据的原因。他们相信能够直接检测目标的表面形状和位置。这反过来又可直接用来检索可能的目标模型，还减少了计算配准变换时的歧义性。
- **目标模型是几何模型还是图符？**几何模型描述目标精确的3D形状，而图符则描述一类目标。几何模型广泛应用于工业机器视觉领域，其中要识别的目标来自预先规定的由少量目标组成的集合。CAD数据非常有用，其中要包括所有必要的几何细节。要识别不同类别的目标时，就需要用图符。比如在医疗成像中，每个器官都是一个新的目标类别，而每个人又有自己的特殊情况。人类生存环境中的很多目标如椅子，存在很多种类，这时只用几何模型就不能满足要求了。
- **目标模型是通过学习得到的还是预先确定的？**目标模型可能包含大量精确数据，这些数据很难由人类来提供。仅有CAD数据也是不够的，数据的附加机制如重要特征等，常常是必需的。借助传感器通过学习得到目标的几何知识，这种系统是迷人的理想系统。

14.4.1 几何模型比对匹配

3D目标比对识别与2D匹配的原理相同。（参见第11章的基本定义。）算法14.2是这种匹配方法的基本思想。

算法14.2 确定图像数据点集是否与3D目标模型匹配

1. 在模型点集和图像数据点集之间假设一个对应关系；
2. 利用这个对应关系求模型到数据的变换；
3. 把这个变换应用到模型点上，产生变换后的模型点集；
4. 对变换后的模型点集和数据点集进行比较，以证实假设的正确性。

下面我们讨论3D-3D和2D-3D两种情况。

1. 3D-3D比对

假设3D模型是3D模型点特征的集合，或者3D模型可以转化为3D模型点特征的集合。如果是深度数据，那么匹配就需要相应的3D数据点特征。比对过程是寻找从三个选定模型点特征到三个对应数据点特征之间的对应关系。这个对应关系决定了包括3D旋转和3D平移的3D变换，把这个变换应用到前述的三个模型点就会得到对应的三个数据点。第13章中的算法13.3就是用来完成这项工作的。如果点的对应关系正确并且没有噪声，那么可以通过这三对匹配点找到正确的3D变换。实际上很少有这种理想情况，所以一般用于组对应点以得到更稳健的结果。任何情况下，一旦算出可能的变换，就把这个变换应用到所有的模型点，产生变换后的模型点集合，可以直接用这个集合与数据点集合做比较，和2D中的情况一样，用验证程序确定变换过的模型点多大程度上与数据点对齐，并以此断言匹配成功或尝试另一个可能的对应关系。和2D中的情况一样，存在一些智能算法，通过局部特征焦点法或其他感知聚类技术来选择对应点。图14-17显示的是3D-3D对应情况，将3D椅子模型相交的三条边与3D网格数据集进行匹配。

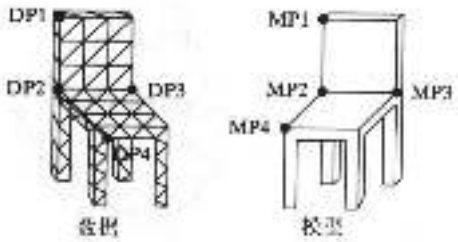


图14-17 3D模型点和3D网格数据点之间的对应情况，可以用3D-3D比对算法计算从模型到数据的变换

习题14.5 3D-3D特征比对

多面体目标中线段相交于一点是很常见的。考虑3D杯子目标，它有一个圆柱体部件，内有盛液体的圆柱体腔，一个半圆形把手，杯子的哪些特征能在3D数据中检测到，并且能用作匹配对应特征？

496
↓
497

这里特征抽取是个重要的问题。如果目标上面的特征点如角点、顶点、凹点等很容易找到，那么上面的程序就很适用。如果表面平滑，特征点很少甚至不存在，那么就需要更好的方法来寻找对应关系。Johnson和Hebert（1998）在CMU研究出的一种方法就能解决这个问题。他们的3D目标表示由以下部分组成：（1）目标的3D网格模型；（2）一组自旋图像（spin image），根据反映目标局部形状特征的网格模型构造得出。

已知3D目标的网格模型，就可以估算在每个网格顶点的表面法线。然后3D空间任意有向点与特定顶点处表面法线的关系，就可以用两个距离参数α和β来表示，其中α是点到表面法线的垂直距离，β是点到特定顶点切面的有向垂直距离。这段说明中，没有提到旋转角，因为旋转角具有歧义性。

自旋图像是2D直方图，可以针对网格中选定的顶点进行计算。构造每幅自旋图像都要有一组贡献点（contributing point）。贡献点规模的大小取决于两个自旋图像参数，即从贡献点到选定顶点的最大距离D，以及贡献点法线和选定顶点法线间允许夹角A。围绕指定的有向点o，基于贡献点集合C构造出自旋图像，其中C以指定的自旋图像参数A和D为基础进行选择。用累加数组$S[\alpha, \beta]$表示自旋图像，并初始化为零。然后对每个点$c \in C$，计算它关于选定的网格顶点o的距离参数α和β，并且将对应α和β的累加数组箱格增加1。注意累加数组中的箱格大小等于3D网格中顶点之间的平均距离。图14-18给出了自旋图像的几个例子。

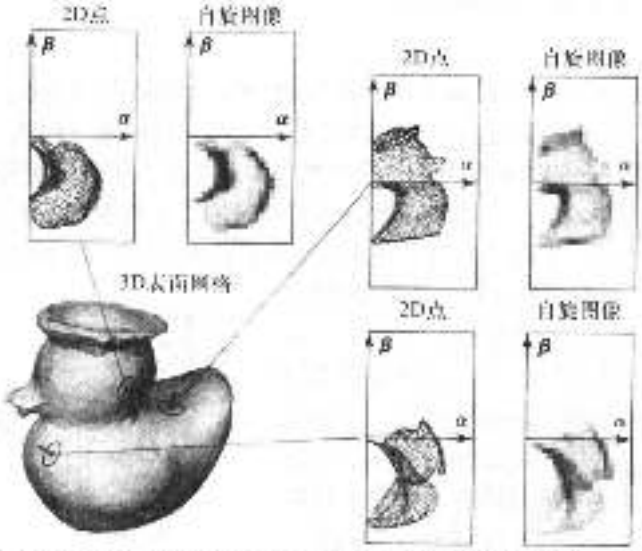


图14-18 自旋图像实例（图形由Andrew Johnson提供，经IEEE允许，再版自"Efficient Multiple Model Recognition in Cluttered 3-D Scenes"，作者A. E. Johnson和M. Hebert，IEEE计算机视觉和模式识别会议论文集，1998年6月，©1998 IEEE）

在网格模型的每个顶点构造自旋图像，这给出了网格各点的局部形状信息。为了匹配两个目标，要用到两组自旋图像。通过计算相关系数，比较两个目标各点对应的自旋图像。高度相关的点对，构成了目标匹配所需要的3D对应点对，利用几何一致性，对对应点对进行分组，并去掉不一致的对应点。然后就像一般比对方法一样，计算刚性变换，并用于验证匹配或去除匹配。图14-19显示对一幅难度大、内容混乱的图像进行自旋图像识别的情况。原图像中包含6个不同的目标，与数据库中的模型对应，而数据库中含20个目标模型。



图14-19 自旋图像识别系统（图形由Andrew Johnson提供，经IEEE允许，再版自"Efficient Multiple Model Recognition in Cluttered 3-D Scenes"，作者A. E. Johnson和M. Hebert，IEEE计算机视觉和模式识别会议论文集，1998年6月，©1998 IEEE）

2. 2D-3D比对

比对也可以用于2D-3D匹配，其中目标模型是三维的，而数据来自2D图像。这时，从模型点到数据点的变换更为复杂，除了3D旋转和3D平移，变换中还有透视投影成分。根据对应点、对应线段以及2D椭圆与3D圆加上单点的对应，或者是以上三类特征相结合，都可以估计出完整的变换。这给匹配提供了一个有力的工具。对应关系可以根据经验进行假设或者通过相

关匹配（参见第14.4.2节）得到，然后用对应关系确定出可能的变换。3D模型特征经变换产生2D数据特征。这里要提到一个在2D-2D比对中不曾出现的问题。在3D目标的任何2D透视图像中，一些变换得到的特征出现在背对摄像机的表面上，以及被离观察者更近的其他表面遮挡住的表面上。因此为了精确生成变换后的特征，并用来与图像特征作比较，必须应用隐藏特征算法。隐藏特征算法涉及到图形学绘制算法，如果用软件实现的话，运算速度会非常慢。如果有适当的网格模型和图形硬件，那么完全绘制就是可以的。其他常见的做法是，要么忽略隐藏特征问题，要么采用不能保证精度的近似算法，但这对于验证来讲足够了。

TRIBORS目标识别系统（Pulli和Shapiro,1996）采用多面体目标的视类模型，寻找模型线段三元组和2D图像线段三元组间的对应关系。在训练阶段对模型三元组进行排列，这样在匹配阶段，首先选择被检测到的概率较大的三元组，而那些概率较低的干脆不予考虑。用含9个参数的向量来描述三元组，这些参数用来描述被匹配视类中的三元组的外观特征。图14-20显示线段三元组的参数化情况。模型三元组与图像中有相同参数的三元组匹配。一旦假定了一个匹配，用数据三元组中的线段交点与模型中假设的3D对应顶点配对，并采用迭代式点对应外向算法（参见第13章）确定变换关系。然后对3D目标的线框模型进行变换，其中可见边缘通过隐藏线检测算法进行确定。对每条预测的边缘，确定最接近的图像线段，并且根据预测边缘与最接近图像线段之间的相似程度进行验证。图14-21显示TRIBORS系统的工作情况。

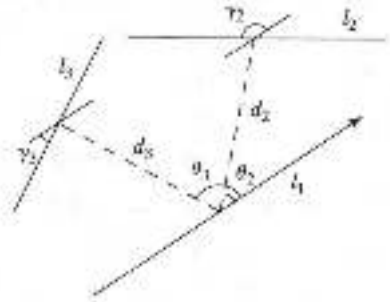


图14-20 TRIBORS系统中线段三元组的参数化。d_2和d_3分别是线段l_1中点到线段l_2和l_3中点的距离。角度γ_3和γ_2分别是线段l_3和l_2与线段l_1之间的夹角。角度θ_1和θ_2分别是线段l_1和图中所示l_1到l_2和l_3连接线之间的夹角



a) b) c)

图14-21 （Kari Pulli提供）

a）从真实图像中抽取的边缘

b）被匹配的线段三元组（粗线）和最新姿态估计结果

c）最终匹配结果和姿态估计结果

499

500

习题14.6 TRIBORS匹配

TRIBORS采用与线段三元组有关的9个参数，确定模型三元组与图像三元组间可能的匹配。生成3D多面体目标相同视类的几幅不同视图，比如用图14-11中的椅子目标。确定在所有视图中出现的三条主要线段，并计算图14-20所示的9个参数。计算不同的参数向量，它们之间相似程度如何？把这三个线段的9个参数与完全不同的其他三个线段参数进行比较，在线段三元组之间，这9个参数有明显的不同吗？

3. 光滑目标比对

我们已经讨论了3D网格模型到3D深度图像的比对，以及3D多面体模型到2D亮度图像的比对。现在要考虑的问题是，根据一个2D亮度图像识别自由形态的3D目标，并计算它的位姿。求解结果借用了模型的视类类型，但是视类的表示与TRIBORS所用的线段三元组集合有很大不同，并且匹配在最低层的边缘图像上进行。

这里讨论的算法以Chen和Stockman（1996）的工作为基础。他们建立的系统能够确定表面光滑的3D目标的位姿。在这个系统中，用一组$2\frac{1}{2}$D视图（称为模型表象）建立3D目
501 标的模型，把中心视点进行上、下、左、右旋转，$2\frac{1}{2}$D视图就是根据这5幅图像合成的。构造汽车模型表象的5幅输入图像参见图14-22，抽取出中间那幅边缘图的轮廓，将其分割为曲线段，求出曲线段的不变特征，作为识别模型表象的索引。

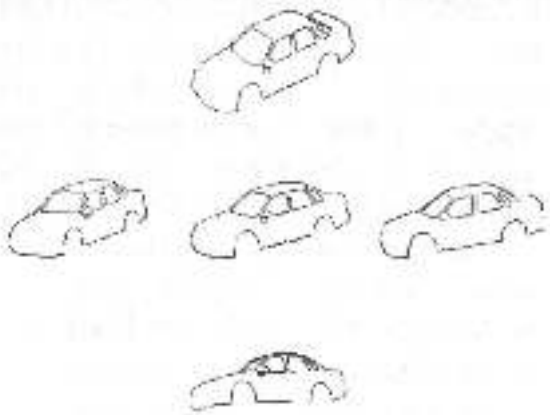


图14-22 构造汽车模型表象的5幅输入图像（JinLong Chen提供）

采用stereo-like计算，对于中间图像上的每个2D轮廓点[u，v]，计算3D边缘点[x，y，z]。用上下两幅图像计算目标边缘在y方向的曲率，用左右两幅图像计算目标边缘在x方向的曲率。同样用stereo-like计算，算出中间边缘图上折痕和标记点的3D位置。这样，$2\frac{1}{2}$D模型表象中包含3D边缘、折痕和标记点、对应于中间边缘图像以及在这些点处的x和y的曲率。基于这些信息，如果知道视图的参数，通过数学公式就能生成同一视类中任意视点的边缘图。视类可用下列特征进行描述：（1）3D点集和上面提到的曲率；（2）用作索引的不变特征集。根据中间和邻近边缘图之间的stereo-like对应关系，可以推导出这些3D点，参见第13章有关内容。不变特征则是从中间图像的2D边缘图推导出来的，参见第10章有关内容。

对要分析的图像进行处理，产生边缘图和一组曲线段，用曲线段给模型视图数据库建立索引，产生目标-视图假设。匹配过程对检索到的假设进行测试，其中每个假设都包括目
502 标的标识和近似位姿，把每个候选$2\frac{1}{2}$D模型表象与测量到的边缘图进行拟合，通过这种方法来进行验证。开始时，设定目标位姿为能生成中间表象的位姿，假设要对这个中间表象进行匹配。把模型表象的投影边缘图像与观测到的边缘图进行比较。多数情况中比较结果不会很好。因此匹配时要改进位姿参数可，以减小在投影模型边缘图和观测边缘图之间的2D距离。图14-23显示匹配的步骤。a图是从输入图像推导出的边缘图，b图是检索出的假设模型位姿。c图显示生成模型边界的几次迭代，d图显示第一个可接受的匹配结果。

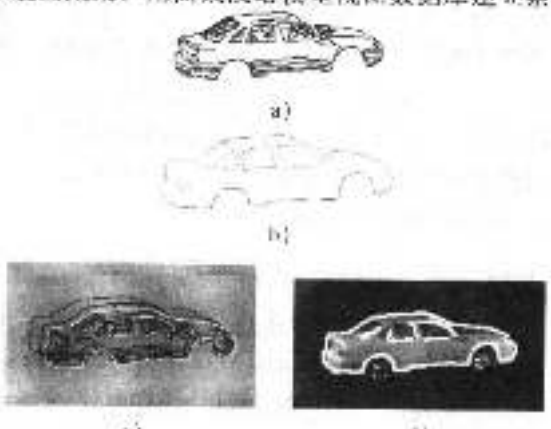


图14-23 匹配边缘图。比例系数s为1:2（JinLong Chen提供）

a）观测到的边缘图
b）模型边缘图
c）比对算法中轮廓的迭代
d）拟合的边缘图显示在原始图像上

习题14.7

就下面几种情况，解释需要多少目标模型和多少模型表象、解释需要多高的位姿精度。(a) 在洗车场，自动机械需要根据进入的汽车模型进行复位调整。(b) 在停车场，监控和登记系统需要识别进入和驶出的汽车模型，并记录相应的时间。(c) 计算机视觉系统需要扫描汽车报废场，记录存在多少残骸以及它们的种类。

习题14.8

得到弯曲目标的一个模型表象，说明通过轻微旋转，就能够生成目标的轮廓，如外径为10、内径为1的圆环面。中心的模型表象，视线与外圆垂直，沿模型轮廓确定一组3D点，并确定这些点的x、y方向的曲率。然后独立合成图像，说明轻微旋转时这个轮廓是如何变化的。 503

14.4.2 关系模型匹配

和二维匹配一样，3D目标识别可利用关系模型，这样就从几何模型转向图符模型。算法14.3总结了基本相关距离匹配技术，将其简化成单一关系，该匹配技术在第11章介绍过。具体采用什么模型和方法取决于图像数据是3D的还是2D的。

算法14.3 相关距离匹配技术：确定两个相关描述是否达到匹配的相似程度

P是模型部件集。

L是部件可能的标记集。

$\mathbf{R}_p$是部件关系。

$\mathbf{R}_l$是标记关系。

找到一个从**P**到**L**的映射f，它使误差$E_r(f)=|\mathbf{R}_p \circ f - \mathbf{R}_l| + |\mathbf{R}_l \circ f^{-1} - \mathbf{R}_p|$最小。采用解释树、离散松弛、概率松弛或第11章中介绍的其他方法。

1. 3D关系模型

三维关系模型由3D基元和3D空间关系组成。基元可以是体积、表面片或3D空间中的直线特征或曲线特征。广义圆柱体常常用作体积基元以及某类3D连接关系。几何离子（Geons）或几何图标（geometric ion）被认为是人类视觉所用的体积基元。在3D目标识别中也用到了几何离子。工业目标可以用平面、圆柱面和表面间的邻接关系进行表示。三维直线和曲线段具有多种空间关系，如连接、平行和共线等。

棒-盘-团（stick, plate, blob）模型用于建立3D目标的粗糙模型，也可对多部件的复杂人造目标进行描述和识别。其中部件可以是各种各样的平面或曲面。对于粗匹配的每个部分可归类为棒条、盘片和团，这与表面-边-顶点模型不同，后者试图对各部件进行精确描述。棒条是又细又长的部件，只有一个有效维。盘片是又平又宽的部件，它含两个接近的平面，平面通过一条薄边相连接。盘片有两个有效维。团是有三个有效维的部件。这三类部分都近似是凸的，所以棒条不能弯曲的很厉害，盘片表面不能折叠的很厉害，团虽然可以是崎岖不平的，但凹度不能太大。图14-24显示了棒条、盘片和团的几个例子。 504

棒-盘-团模型描述了棒条、盘片和团如何一起构造目标。这些描述也是粗略的，它们不能准确说明两个部分的实际相交点。棒条包括两个逆提端点、逻辑内部点集和逻辑质心，可以把这些点看作连接点。盘片包括边缘点集、表面点集和质心。团包括表面点集和质心。目标模型中只能用到这些信息。

棒-盘-团这种关系模型，是细节化图符目标模型很好的例子，图符目标模型在图符目标识别中得到了很成功的应用。模型由5个关系组成。一元简单部件（SIMPLE PARTS）关系列出了目标的各部件。每个部件都有几项属性描述，包括部件的类型（棒，盘或者团），也可能包括部件尺寸或形状的数量信息。连接/支持（CONNECTIONS/SUPPORTS）关系包含目标结构上最重要的信息。这个关系是六元形式（s_1，s_2，*SUPP-ORTS*，*HOW*）。元素s_1和s_2是简单部件，如果s_1支持s_2，SUPPORTS为真；反之为假。HOW描述了s_1和s_2的连接类型。

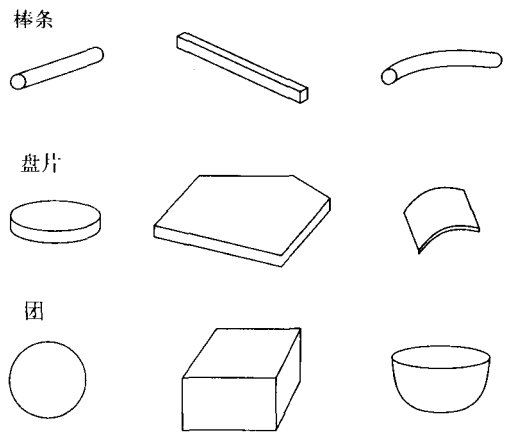
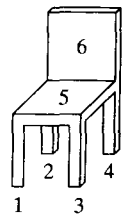


图14-24 棒条、盘片和团儿举例

其他4个关系表示约束情况。三元约束（TRIPLE CONSTRAINT）关系是四元形式（s_1，s_2，s_3，*SAME*），其中简单部件s_2接触s_1和s_3，如果s_1和s_3在相同端点（或表面）接触s_2，则SAME为真；反之为假。平行（PARALLEL）关系和垂直（PERPENDICULAR）关系是二元形式（s_1，s_2），其中简单部件s_1和s_2在模型中是平行的（或垂直的）。图14-25显示椅子的
505 棒-盘-团模型。无论部件的精确形状如何，所有有类似关系的椅子都应该与这个模型匹配。



SIMPLE-PARTS

PART#	TYPE
1	Stick
2	Stick
3	Stick
4	Stick
5	Plate
6	Plate

CONNECTS-SUPPORTS

SP1	SP2	SUPPORTS	HOW
1	5	True	end-edge
2	5	True	end-edge
3	5	True	end-edge
4	5	True	end-edge
5	6	True	edge-edge

TRIPLES

SP1	SP2	SP3	SAME
1	5	2	True
1	5	3	True
1	5	4	True
1	5	6	False
2	5	3	True
2	5	4	True
2	5	6	False
3	5	4	True
3	5	6	False
4	5	6	False

PARALLEL

SP1	SP2
1	2
1	3
1	4
2	3
2	4
3	4

PERPENDICULAR

SP1	SP2
1	5
2	5
3	5
4	5
5	6

图14-25 椅子目标棒-盘-团模型的整体关系结构

习题14.9 棒-盒-团模型

画出简单的多面体课桌图，构造该目标整体关系的棒-盒-团模型。

2. 视类关系模型

当数据由2D图像构成时，就可以用视类模型代替完全3D目标模型。训练数据可以是合成的图像，或者是目标的实际图像。训练数据可用来构造这些模型。根据目标的种类，从目标图像中抽取可用的2D特征集。从训练图像抽取的特征生成目标相应视图的相关描述。然后对这些相关描述进行聚类，形成目标的视类。每个视类用包含所有特征的组合相关描述来表示。这些特征是在该视类的所有视图中检测到的。综合相关描述是视类的关系模型。典型地，目标有5个视类，每个视类都有自己的相关描述。视类模型可用于完全相关匹配。如果数据库中有很多不同的模型，或者对于第11章介绍的相关索引，完全相关匹配的代价就很高。下面的例子采用相关索引方式。



图14-26 工件的左右图像，以及经图像处理去除阴影和高亮部分后得到的边缘图像（Mauro Costa提供）

506

由华盛顿大学Mauro Costa开发的RIO目标识别系统，根据2D图像识别多目标场景中的3D目标。用固定摄像头拍摄一对图像，一幅图像拍摄时用左侧的光源，另一幅图像用右侧的光源。用这两幅图像来确定哪个区域是阴影部分，哪个区域是高亮部分，这样就可以得到只含目标的高质量边缘图像。从边缘图像获得直线段和弧线段，并根据直线段和弧线段构造识别用的特征。图14-26显示的是左右图像对和抽取的边缘图像。图14-27显示从边缘图像中抽取的直线段和弧线段。

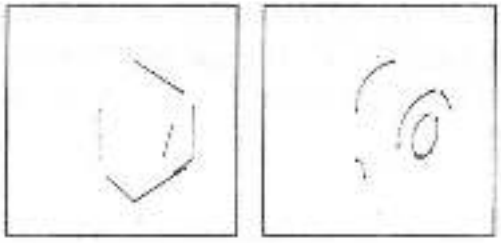


图14-27 从图14-26的边缘图像抽取的直线段和弧线段（Mauro Costa提供）

RIO目标可能有平面、圆柱面、线状图案的表面。这就产生很多实用的高级特征。RIO采用10种特征。它们是椭圆、同轴弧（2个、3个或多个）、平行线段对（远近均可）、线段三元组（U形和Z形）、L连接、Y连接和V连接。图14-28显示的是从图14-27的线段和弧线中构造出的一些特征。直线特征包括2个L连接和一对平行线。弧线类特征显示出3个同轴弧。注意不是所有的直线段或弧线段最终都会成为匹配用的特征。图14-29显示完整的RIO特征集合。



图14-28 根据图14-27中的直线和弧线构造的直线特征、弧线特征和椭圆特征（Mauro Costa提供）

507

除了标记特征，RIO还在特征之上采用标记二元关系来识别目标。RIO中使用的关系有：共用一条弧、共用一条线、共用两条线、同轴性、端点接近、以及包围/被包围，如图14-30所示。

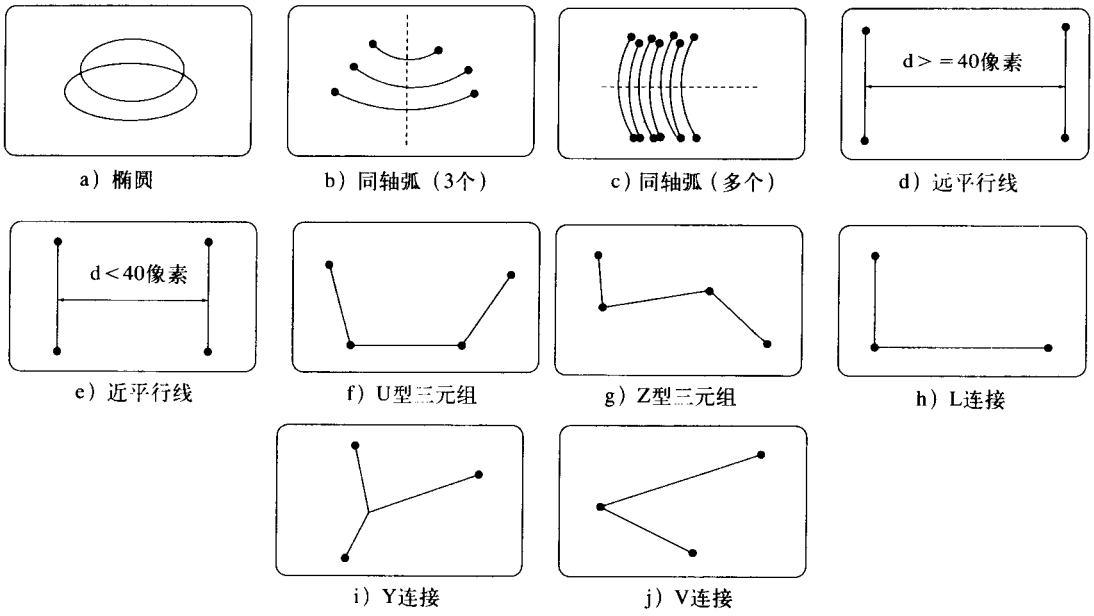


图14-29 RIO系统使用的特征（Mauro Costa提供）

模型视图的结构描述是图结构，图的节点是特征类型，图的边是关系类型。为了使图能用在相关索引程序中，把图分解成2-图的集合，每个2-图包括两个节点和节点间的一个关系。图14-31显示螺母的模型视图，表示三个特征及特征间关系的局部完全图，以及2-图分解。

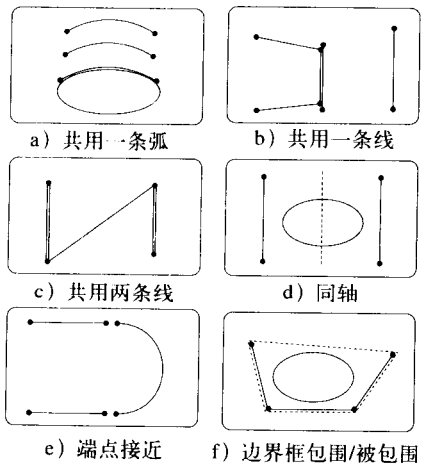


图14-30 样本特征对之间的关系（Mauro Costa提供）

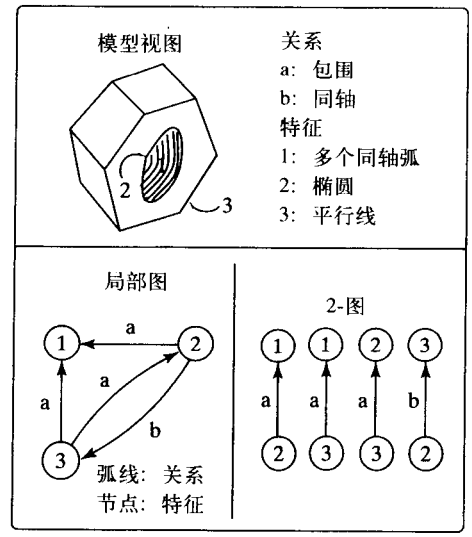


图14-31 螺母的图和对应的2-图（Mauro Costa 提供）

相关索引将未知图像与可能很大的目标视图数据库进行匹配，生成关于哪些目标存在于图像中的一组假设。离线预处理阶段建立数据结构，在线阶段进行匹配。离线阶段构造在线阶段要用的散列表。散列表的索引是四元组，表示目标模型视图的2-图。四元组的元素包括两个节点的类型和两个关系的类型。例如四元组（椭圆，远平行线，被包围，包围）的意思

是，2-图表示了椭圆特征和远平行线特征，其中椭圆被两条平行线段包围，也就是这两条平行线包围这个椭圆。因为大多数RIO关系都是对称的，所以这两个关系经常是相同的。比如，四元组（椭圆，一组同轴弧，共用一条弧，共用一条弧）描述了一个关系，其中椭圆和一组同轴弧共用一条弧。为了散列，把四元组的图符元素转换为数字，对数据库中的每个模型视图都进行预处理，对模型视图的每个2-图进行编码，产生四元组索引，将模型视图的名字和相关信息存储在散列表选定的箱格列表中。

构造好的散列表用于在线识别，对数据库中每个可能的模型视图，都使用一个用于投票的累加器。当分析场景时，抽取场景特征，构造形式为2-图集合的相关描述。然后，对相关描述中的每个2-图进行编码，产生一个索引，用这个索引来访问散列表。与所选箱格有关的列表被检索出来，列表包括具有该特殊2-图的所有模型视图。给这个列表中的每个模型视图都投一票。对图像中的所有2-图都执行上述过程。在程序结尾，得票最高的模型视图作为候选的假设。图14-32显示在线识别过程。图中所示的2-图转化为数字四元组（1，2，9，9），这个四元组在散列表中选定一个箱格，访问这个箱格，检索出含四个模型M_1、M_3、M_{23}、M_{81}的列表。这些模型视图的累加器都加1。

生成假设后，必须进行验证。相关索引在模型视图中提供了从2D图像特征到2D模型特征的对应关系。这些2D模型特征与假设目标的3D模型特征联系起来。RIO系统执行验证，采用相对应的2D-3D点对、2D-3D线段对和2D椭圆-3D圆对，计算从假设目标的3D模型到图像的变换关系。直线和弧线段投影到图像平面，通过计算某种距离确定验证是否成功，或者假设是否正确。图14-33和14-34显示RIO系统的运行过程。图14-33显示多目标场景的边缘图像，以及检测到的直线特征、圆弧特征和椭圆特征。图14-34显示系统产生的一次不正确假设和三次正确假设。不正确假设被验证程序取消，而正确假设通过了验证。第13章中给出了基于点对应的RIO位姿估计程序，图14-35显示完整的RIO系统的方框图。

508 ~ 510

511

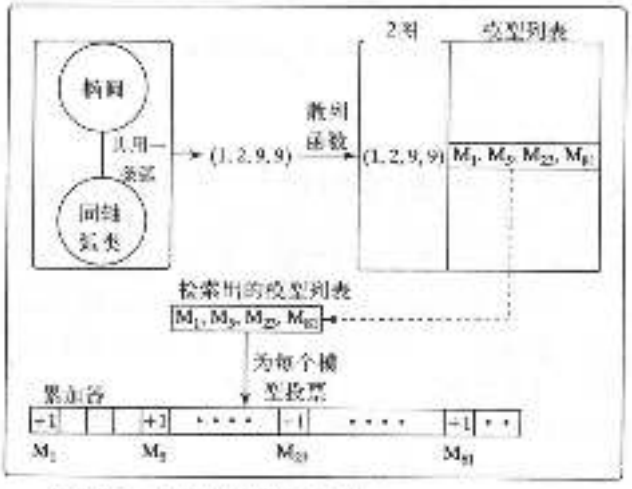


图14-32 相关索引的投票方案（Mauro Costa提供）

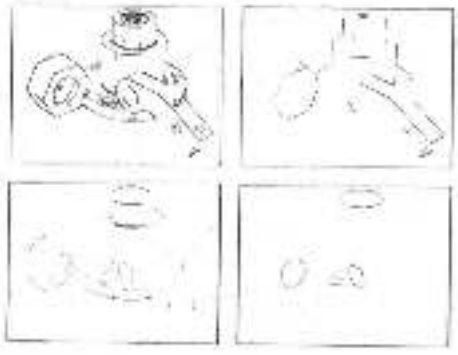


图14-33 测试图像及其直线特征、圆弧特征和椭圆特征（Mauro Costa提供）

习题14.10 相关索引

编写用于目标匹配的相关索引程序。该程序要采用存储的目标模型库，库中每个模型都用2-图集合表示，识别阶段的输入，是一个2-图集合表示的多目标图像，程序应该返回数据库

512

中每个模型的列表。数据库中模型的2-图至少有50%在图像中。

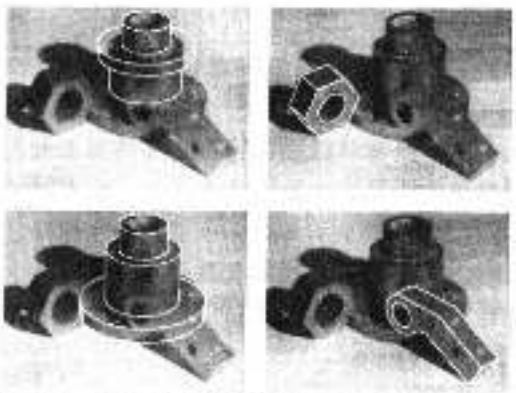


图14-34 一次不正确的假设（左上）和一次正确的假设（Mauro Costa提供）



图14-35 RIO目标识别系统的流程图

14.4.3 功能模型匹配

几何模型给出了特定物体的精确定义。CAD模型描述了单个目标所有重要的点和具体尺寸。关系模型描述了一类目标，更具一般性，但类中的每个成员都必须有相同的关系结构。例如一把椅子可能被描述为有一个靠背，一个座位和四角下的四条腿。另一把用底座和支架代替四条腿的椅子则与这个描述不匹配。基于功能的目标识别方法更进了一步，它试图通过目标的功能来定义目标的类别。因此椅子是某种东西，人可以坐在上面，椅子可能有很多不同的关系结构，只要满足一组功能约束，那么它就是椅子。

基于功能的目标识别，最先由Stark 和 Bowyer（1996）在他们的GRUFF（Generic Object Recognition Using Form and Function）系统中使用。GRUFF包含三级知识：

513

（1）所有目标的类别层次都在知识库中。

（2）根据功能属性对类别进行定义。

（3）知识基元是功能定义的基础。

1. 知识基元

知识基元是一个参数化过程，它实现了几何、物理学或因果关系的基本概念。知识基元

用3D形状描述的一部分作为输入，返回一个值，该值表示基元在多大程度上满足某种需求。6个GRUFF知识基元定义的概念包括:

- 相关姿态
- 尺度
- 邻近性
- 稳定性
- 空旷性
- 包围性

相关姿态（relative orientation）基元确定两个表面的相关姿态满足期望关系的程度。例如椅座上表面可能基本垂直于相邻的靠背表面。尺度（dimension）基元对6种可能的尺度类型进行尺度测试，这6种尺度类型是：宽度、深度、高度、面积、连续表面和体积。大部分物体中，一个部件的尺度约束着其他部件的尺度。邻近性（proximity）基元检验目标形状元素间定性的空间关系。例如水壶把手应该位于水壶的质心之上，这样才容易提起它。

所具有某种形状的物体以一定的方向和力度放在支撑面上，稳定性（stability）基元用来检验这时该物体的稳定性。空旷性（clearance）基元检验物体部件间特定的空间体积是否是空旷的。例如为了让人能坐在椅子上面，应该清空椅座上的立方体空间。最后，包围性（enclosure）基元测试目标必要的凹陷。例如高脚酒杯必须有用来盛酒的凹陷。

2. 功能属性

功能目标类别的定义规定了它必须有的与知识基元有关的功能属性。家具、器皿和工具类物体的GRUFF功能分类，由下列四种可能的模板确定:

- 提供稳定的X
- 提供X表面
- 提供X容积
- 提供X把手

其中X是模板的参数。例如椅子必须为坐在其上的人提供稳定的支持和可坐的表面。汤碗必须为汤提供容积空间。杯子必须包含可抓起的合适把手，把手要与杯子的尺度匹配。

3. 类别层次

GRUFF通过分类树表示出目标的各类别层次，树中列出了系统当前可识别的所有类别。在树的顶层是非常一般的类别，如家具和器皿。以下逐层具体化。比如家具中的类别有：椅子、桌子、长椅、书架和床。而且这些类别可以进一步分解。椅子可以分为：传统椅子、沙发、平衡椅和高脚椅等。图14-36显示GRUFF分类树的一部分。

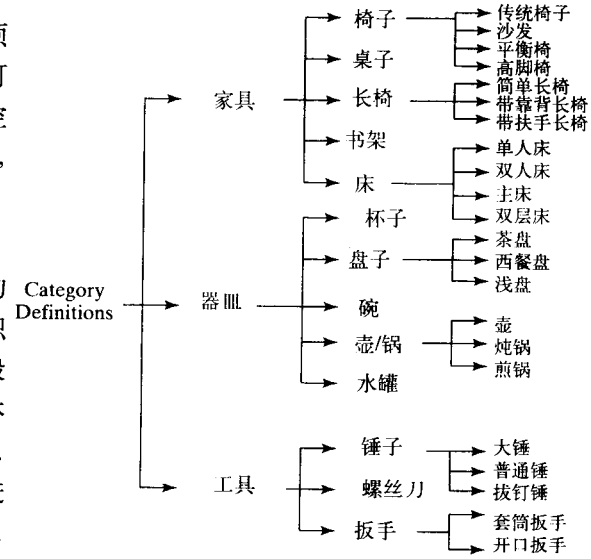
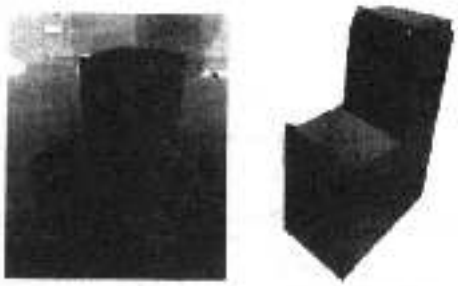


图14-36 GRUFF分类树的一部分（由 Louise Stark 和 Kevin Bowyer 提供）

514

GRUFF采用基于功能的目标分类方法，目的不是为了识别目标，而是推断用深度数据表示的观测目标是否具有该类成员应具有的功能。这个基于功能的分析过程，主要包括两个阶段：预处理阶段和识别阶段。预处理阶段与类别无关，以相同的方式处理所有的目标。在这个阶段，分析3D数据，列举所有可能的功能要素。识别阶段使用这些要素来构造索引，用这些索引给目标类别排序。索引由功能要素和它的面积及体积组成。以索引信息为基础，不再
515 搜索那些不可能匹配的类别。对余下的类别排序，用于进一步的评价。对每个类假设，首先调用它的每个知识基元，度量由数据得到的功能要素与其需求之间的符合程度。每个知识基元返回一个评价测度，然后综合这些测度形成最终的联合测度，联合测度描述来自数据的全部功能要素与假设类别的匹配程度。图14-37显示GRUFF系统的一种典型输入，图14-38显示数据分析中的功能推理部分。



深度图　　分割后的深度图

图14-37 GRUFF系统的输入数据（由Louise Stark和Kevin Bowyer提供）

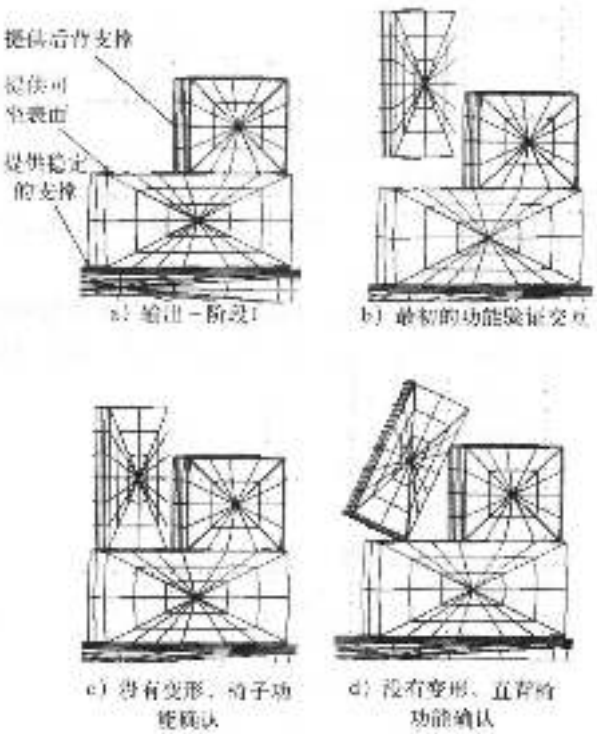


图14-38 GRUFF系统的处理过程（由Louise Stark和Kevin Bowyer提供）

习题14.11 功能目标识别

考虑两张桌子，一张四角下有四条腿，另一张有一个底座。功能目标识别系统使用两张桌子间的什么相似性，把二者分为同一类目标？

14.4.4 基于外观的识别

在大多数3D目标识别方案中，模型是根据目标2D图像得到的独立实体。这部分我们讨论的内容是，通过记忆很多目标的2D图像来学习目标，把未知目标的感测图像与记忆中的图像

进行匹配，从而识别出目标。在信号级（signal level）表示目标，直接对亮度图像进行匹配，而不使用高级特征，因此不需要消耗太多的时间，不需要进行难以测试的复杂编程。下面列出信号级识别存在的几个问题。基于外观的识别方法比较简单，可用大量的图像进行训练和测试。这种方法已经取得了较大的成果，最显著的成果在于人脸识别方面。在此我们讨论该方法在人脸识别方面的应用。

通过外观进行识别的方法简述如下：

- 在训练或学习阶段，建立标记图像数据库。DB = $\{<I_j[], L_j>_{j=1,\cdots k}\}$。其中$I_j$是第$j$个训练图像，而$L_j$是它的标记。
- 把未知图像I_u与数据库中的图像进行比较，把最接近的训练图像I_j的标记L_j赋给未知目标，从而识别出这个未知目标。最接近的训练图像I_j可通过使欧几里得距离$\|I_u[]-I_j[]\|$最小化来确定，或使点积$I_u \circ I_j$最大化来确定，这二者都在第5章中进行了定义。 516

当然每一步都有要强调的复杂性因素。

- 训练图像必须是被识别目标的典型实例。在人脸识别中（其他多数目标也如此），训练图像必须包括表情变化、照明变化、以及头部在2D和3D中的轻微转动。
- 目标区域要仔细选择，所有的人脸位置和尺寸必须大致相同。否则，需要对位置和尺寸参数进行确定。 517
- 因为这个方法并未把目标从背景中分离出来，所以结果中将包含背景，在训练中应该认真考虑这一点。
- 对人脸识别来讲，100 × 100的图像大小已经足够了。即使图像只有100 × 100这么小，所有图像的空间维数也是10 000。训练样本的数量有可能比这小很多，因此应该使用一些降低维数的方法。

大家应该考虑到戴眼镜和不戴眼镜的人脸差别，或者带天线和不带天线汽车的差别。如果出现其他不相关的变化，还能检测到这些差别吗？

现在考虑降低目标特征数量这一重要问题。对于人脸识别，维数可以从100 × 100降低到15，但识别率仍然维持在97%。对R × C的图像空间，第5章对不同基底进行了讨论，并说明一幅图像可以表示为有意义的图像之和，如跳变边缘图和波纹图等。另外，当图像被表示为标准正交基的线性组合时，图像能量恰好是系数平方之和。

1. 训练图像集的基

假设能够找到具有下列性质的一组标准正交基图像B：

1. B=$\{F_1, F_2, \ldots, F_m\}$，其中$m$远小于$N = R \times C$。

2. 在下列意义下，用这组基进行图像表示的质量平均来说是令人满意的。对于训练集中所有M个图像I_j，都有：

$$I_j^m = a_{j1}F_1 + a_{j2}F_2 + \cdots + a_{jm}F_m$$

和

$$\sum_{j=1}^{m} \left(\|I_j^m - I_j\|^2 / \|I_j\|\right)^2 < P\%$$

其中I_j^m是原始图像I_j的近似图像，I_j是m个基图像的线性组合。

图14-39中上面一行是6幅训练图像，图像来自Weizmann 学院的数据库。中间4幅是推导出来的基图像，用来进行人脸表示。其中最左侧是所有训练样本的平均结果。下面一行的6幅

人脸图像，是4个基向量的线性组合表示的结果，与原始的6幅图像相对应。几个不同的研究项目表明，$m = 15$或$m = 20$幅基图像足够用来表示数据库中的人脸图像（例如Pentland（1986）研究组3000张人脸图像的数据库），用F_i近似表示I_i的平均精度在5%以内。结果，用近似图像进行匹配与用原图像进行匹配将产生基本相同的结果。要强调的是，对于图14-39的数据库，
518 **每幅训练图像在内存中可以只用4个数值表示**，这样对未知图像就能够进行有效比较。只要4个基向量保存在内存中，需要时就可以重新生成与原始人脸图像非常近似的图像。（注意，第一个基向量是原始人脸图集的平均图，而不是标准正交集中的一个基。）



图14-39 （图像数据库由Weizmann学院Yael Moses提供，处理过的图像由John Weng提供）
（上行）6幅训练图像，人脸图像库中某人的原始图集
（中间行）平均训练图像和从协方差矩阵推导出的三个最重要的特征向量
（下行）库中的4幅图像的线性组合对上面一行图像进行表示的结果

2. 计算基图像

有了基集B，使需要的内存空间大大压缩，同时也加快了计算速度，因为m要比原始图像的像素数量N小很多。称基图像F_i为训练样本集的主元（principal component）。下面的算法14.4列出了使用主元进行基于外观的识别步骤。识别包括两部分：离线训练阶段和在线识别阶段。训练阶段中的第一步是，计算训练图像的平均图像，产生差图像集合Φ。集合中的每个元素都是某幅训练图像和平均图像的差。如果把每幅差图Φ视为N维向量，那么Φ就是N行
519 M列矩阵。下一步是计算训练图像的协方差矩阵Σ_Φ。由定义可知，在所有训练图像上，$\Sigma_\Phi[i, i]$是第i个像素的方差，而$\Sigma_\Phi[i, j]$是第i个像素和第j个像素的协方差。因为已经算出了平均图像和差图像，所以协方差矩阵可以定义为：

$$\Sigma_\Phi = \Phi^T \Phi \tag{14-5}$$

这个协方差矩阵非常大，为N × N，其中N是图像像素数，典型值是256 × 256甚至512 × 512。如果直接使用的话，在下一步计算特征向量和特征值时将非常耗费时间。（关于主元算法，请参见《Numerical Recipes in C》，Vetterling,1992。）我们利用下面的矩阵Σ'_Φ代替上面的Σ_Φ。

$$\Sigma'_\Phi = \Phi\Phi^T \tag{14-6}$$

这个矩阵要小很多，为m × m。Σ_Φ的特征向量和特征值与Σ_Φ的特征向量和特征值的关系是：

$$\Sigma_\Phi F = \lambda F \tag{14-7}$$

$$\Sigma'_\Phi F' = \lambda F' \tag{14-8}$$

$$F = \Phi^T F' \tag{14-9}$$

其中λ是Σ_Φ的特征值向量，F是Σ_Φ的特征向量，F'是Σ_Φ的特征向量。

这里介绍的主元分析方法，在人脸识别中的应用效果显著（参见Kirby和Sirovich（1990），Turk和Pentland（1991），以及Swets和Weng（1996））。有人怀疑对于高频变化的情况，这个方法就不大适用，因为这时即使图像产生微小变化，自相关作用也会下降很快，这就加重了目标分割的要求。脸部图像不会遇到这个问题。Swets 和Weng（1996）针对很多无纹理的目标得出了很好的结果，而Murase 和Nayar（1995）针对人脸也得到了很好的结果。在以10°间隔摄取的训练图像基础上，他们能够以2度的精度内插估计出3D目标的位姿。

Turk 和 Pentland（1991）对上面关注的两个问题提出了解决方法。首先，他们采用第9章中提到的运动技术，把头部从视频序列中分割出来，这样就能分割出脸部，进而对图像尺寸进行规范化处理。其次，他们采用高斯滤波方法对图像像素进行再次加权处理，使外围背景的像素值近似为零，同时保留中间的人脸像素值。

算法14.4　基于主成分基的外观识别

离线训练阶段

输入含M个标记训练图像的集合I，

产生基集B和每幅图像的系数向量。

$\mathbf{I} = \{I_1, I_2, \cdots, I_M\}$是训练图像集合。（输入）

$\mathbf{B} = \{F_1, F_2, \cdots, F_m\}$是基向量集合。（输出）

$\mathbf{A}_i = [a_{i1}, a_{i2}, \cdots, a_{im}]$是图像$I_i$的系数向量。（输出）

1. $\mathbf{I}_{mean} = \text{mean}(\mathbf{I})$。
2. $\Phi = \{\Phi_i | \Phi_i = I_i - \mathbf{I}_{mean}\}$，差图像集合。
3. Σ_Φ等于Φ的协方差矩阵。
4. 用主元方法计算Σ_Φ的特征值和特征向量。（参见正文）
5. 选择m个最重要的特征向量，构造向量B作为基集；从最大的特征值开始，按特征值减序依次选择对应的特征向量。
6. 用基向量的线性组合表示训练图像I_i，即

$I'_i = a_{i1}F_1 + a_{i2}F_2 + \cdots + a_{im}F_m$

在线识别阶段

输入基向量集合B，系数集合$\{\mathbf{A}_i\}$的数据库，测试图像$\mathbf{I}_u$。

输出$\mathbf{I}_u$的类标记。

1. 计算$\mathbf{I}_u$的系数向量$\mathbf{A}_u = [a_{u1}, a_{u2}, \cdots, a_{um}]$。
2. 在集合$\{\mathbf{A}_i\}$中找到向量$\mathbf{A}_u$的k个最近邻。
3. 通过k个最近邻的标记，确定$\mathbf{I}_u$的类别（如果近邻很远或与标记不一致的话，就有可能被拒绝）。

习题14.12

获得10幅人脸图像和10幅风景图像，所有图像大小都是$R \times C$。计算所有图像对之间的欧

几里得距离，把距离显示在20×20的上三角矩阵中。这一组人脸很相近吗？风景图像呢？最近距离和最远距离的比是多少？能用欧几里得距离进行图像数据库检索吗？请加以解释。

习题14.13

设I_t是未知目标的图像，B=$\{(I_1, L_1)\}$是标记过的训练图像的集合，假设所有的图像都是规范化的，即$\|I_j\|=1$。(a) 证明当$I_t \cdot I_j$取最大值时，$\|I_t - I_j\|$取最小值。(b) 如果没有$\|I_j\| = 1$的假设，上述结论是不正确的。为什么？

3. 最佳分类与快速搜索

主元分析使我们能够用压缩了的方式表示训练模式的子空间。算法14.4中，表示训练数据
520 的最佳基称为最佳描述特征（most expressive feature, MEF）。John Weng的工作已经证明，尽
521 管最佳描述特征能够理想地表示训练图像的子空间，却不能很好地表示不同类图像间的差异。Weng提出最佳分类特征（most discriminating feature, MDF）的概念，通过判别分析可以推导出最佳分类特征。MDF重点在于区别不同类目标的图像差异。图14-40对MEF和MDF做了对比。原始数据坐标是(x_1, x_2)。y_1是发生最大变化的方向，y_2与y_1正交。因此y_1和y_2坐标是最佳描述特征。向量的原始类别是通过主次轴分别与y_1和y_2重合的椭圆表示的。(第3章首次给出了2D情况下寻找主次轴的算法。)在两类之间，y_1和y_2的阈值很难判别。经判别分析算出的MDF轴z_1和z_2，允许基于z_1的阈值实现对训练样本的理想分离。

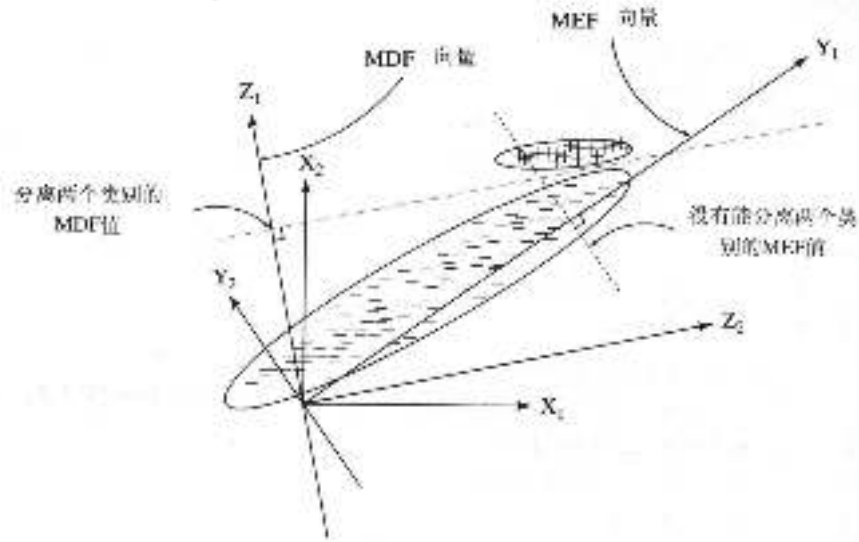


图14-40 由方差矩阵的特征向量确定的最佳描述特征，可以很好的表示数据，但不能很好的表示类间差异。可以通过判别分析来找到子空间，这些子空间强调类间差异（J.Swets和J.Weng提供）

Weng及其同事对特征空间外观识别方法做出的另一个改进是，提出了搜索树构造程序，用该方法在含S个训练样本的数据库中寻找最近邻，所用的搜索时间是O（$\log_2 S$）。第4章介绍过用于目标分类的决策树。对于树上的每个决策点，把一幅未知图像投影到最佳分类子空间，需要对下一步的分支做出决策。在决策树的不同节点所用的MDF，随着训练样本的不同而不同，而且能够转向所需要的特殊分支决策。这是最近发展起来的理论，感兴趣的读者最好参
522 考有关文献以了解更详细的内容。

习题14.14

（a）得到300幅人脸图像，把每幅图像用$R \times C$维的向量表示。（b）计算这300幅样本图像的散布矩阵和平均图像。（c）计算散布矩阵的m个最大特征值和相对应的m个特征向量，使散布矩阵95%的能量得以保留（d）随机选择5幅原始人脸图像，把它们表示成m个最佳特征向量的线性组合。（e）显示算出的5幅近似图像并与原图像做比较。

14.5 参考文献

网格模型来自计算机图形学，在计算机图形学中常称为多边形网格。Foley等人（1996）的图形学教材是这方面很好的参考资料。表面—边—顶点模型参考的是麻萨诸塞大学70年代的VISIONS系统。本书中的结构模型参考了Camps（1992）近期的工作。广义圆柱体模型首先由Binford提出，Nevatia和Binford（1977）用来处理深度数据。Rom和Medioni（1993）的文章讨论了从2D数据计算圆柱体。八叉树首先由Hunter（1978）提出，并由Jackins和Tanimoto（1980）进行了更进一步的推广。这些在Samet（1990）的书中进行了详细介绍。超二次曲面模型的讨论绝大部分参考Gupta、Bogoni和Bajcsy（1989）的工作，左心室的例子参考的是Park、Metaxas和Axel（1996）的工作，在可变形模型讨论中也用到了这个例子。

视类模型概念一般认为应归功于Koenderink和van Doorn（1979）。Camps等人（1992）、Pulli（1996）和Costa（1995）使用视类模型识别三维目标。比对匹配由Lowe（1987）提出，并由Huttenlocher和Ullmann（1990）进行了详细分析。3D-3D比对内容参考了Johnson和Hebert（1998）的工作，而2D-3D讨论参考了Pulli和Shapiro（1996）的工作，对平滑目标比对识别问题，参考的是Jin-Long Chen和Stockman（1996）的工作，也涉及Basri和Ullman（1988）的原始工作。匹配棒—盘—团模型在Shapiro等人（1984）的工作中有所描述，相关匹配在Shapiro和Haralick（1981，1985）中进行了汇总讨论。相关索引在Costa和Shapiro（1995）的工作中可以找到。功能目标识别参考的是Stark和Bowyer（1996）的工作。

Kirby和Sirovich（1990）研究了脸部图像压缩的问题，Turk和Pentland（1991）进行了更有效的脸部识别研究。Swets和Weng（1996）提出一般的学习系统，称为SHOSLIF，他们对主元方法进行了改进，通过采用MDF并且构造出树结构的数据库，可以在$\log_2 N$时间内搜索出最近邻。Murase和Nayar（1994）也提出一种有效的搜索方法，在以10°间隔摄取的训练视图基础上，以2°的精度通过内插估计出3D目标的位姿。另外针对几种目标但不是人脸，发现 523
20维或者更少维的特征空间就能保证有很好的性能。本章通过外观进行识别的内容大部分参考Swets和Weng（1996）的工作，并多次引用Turk和Pentland（1991）的工作。

能量最小化在70年代用于轮廓平滑。但是Kass、Witkin和Terzopoulos（1987）提出活动轮廓的论文激起了其他研究人员的兴趣，很快就有了拟合以及跟踪表面和体积的应用情况。Amini等人（1988）利用动态规划使活动轮廓与图像拟合。在医疗图像方面的例子参见Yue等（1995）。物理学模型和可变形模型的研究和应用进展很快，Chen和Medioni（1995）以及Park、Metaxas和Axel（1996）的工作就是两个很好的实例。

1. Amini, A., S. Tehrani, and T. Weymouth. 1988. Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints. *Proc. IEEE Int. Conf. Comput. Vision*, 95–99.

2. Basri, R., and S. Ullman. 1988. The alignment of objects with smooth surfaces. *Proc. 2nd Intern. Conf. Comput. Vision*, 482–488.

3. Biederman, I. 1985. Human image understanding: recent research and theory. *Comput. Vision, Graphics, and Image Proc.*, v. 32(1):29–73.
4. Camps, O. I., L. G. Shapiro, and R. M. Haralick. 1992. Image prediction for computer vision. In *Three-dimensional Object Recognition Systems,* A. Jain and P. Flynn, eds. Elsevier Science Publishers BV, Amsterdam.
5. Chen, J. L., and G. Stockman. 1996. Determining pose of 3D objects with curved surfaces. *IEEE Trans. Pattern Analysis and Machine Intelligence,* v. 18(1):57–62.
6. Chen, Y., and G. Medioni. 1995. Description of complex objects from multiple range images using an inflating balloon model. *Comput. Vision and Image Understanding,* v. 61(3):325–334.
7. Costa, M. S., and L. G. Shapiro. 1995. Scene analysis using appearance-based models and relational indexing. *IEEE Symp. Comput. Vision* (Nov. 1995), 103–108.
8. Foley, J., A. van Dam, S. Feiner, and J. Hughes. 1996. *Computer Graphics: Principles and Practice.* Addison-Wesley, Reading, MA.
9. Gupta, A., L. Bogoni, and R. Bajcsy. 1989. Quantitative and qualitative measures for the evaluation of the superquadric model. *Proc. IEEE Workshop on Interpretation of 3D Scenes,* 162–169.
10. Hunter, G. M. 1978. *Efficient Computation and Data Structures for Graphics.* Ph.D. Dissertation, Princeton University, Princeton, NJ.
11. Huttenlocher, D. P., and S. Ullman. 1990. Recognizing solid objects by alignment with an image. *Int. J. Comput. Vision,* v. 5(2):195–212.
12. Jackins, C. L., and S. L. Tanimoto. 1980. Oct-trees and their use in representing three-dimensional objects. *Comput. Graphics and Image Proc.*, v. 14:249–270.
13. Johnson, A. E., and M. Hebert. 1998. Efficient multiple model recognition in cluttered 3-D scenes. *Proc. IEEE Conf. Comput. Vision and Pattern Recognition,* 671–677.
14. Kass, M., A. Witkin, and D. Terzopoulos. 1987. Snakes: active contour models. *Proc. First Int. Conf. Comput. Vision,* London, UK, 259–269.
15. Kirby, M., and L. Sirovich. 1990. Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. and Machine Intelligence,* v. 12(1):103–108.
16. Koenderink, J. J., and A. J. van Doorn. 1979. The internal representation of solid shape with respect to vision. *Biological Cybernetics,* v. 32:211–216.
17. Lowe, D. G. 1987. The viewpoint consistency constraint. *Int. J. Comput. Vision,* v. 1:57–72.
18. Murase, H., and S. Nayar. 1995. Parametric appearance representation. In *3D Object Representations in Computer Vision,* J. Ponce and M. Herbert, eds. Springer-Verlag.
19. Nevatia, R., and T. O. Binford. 1977. Description and recognition of curved objects. *Artificial Intelligence,* v. 8:77–98.
20. Park, J., D. Metaxas, and L. Axel. 1996. Analysis of left ventricular wall motion based on volumetric deformable models and MRI-SPAMM. *Medical Image Anal. J.,* v. 1(1):53–71.
21. Pentland, N. P. 1986. Perceptual organization and the representation of natural form. *Artificial Intelligence,* v. 28:29–73.
22. Pulli, K., and L. G. Shapiro. 1996. Triplet-based object recognition using synthetic and real probability models. *Proc. ICPR96,* v. IV:75–79.
23. Roberts, L. G. 1977. Machine perception of three-dimensional solids. In *Computer Methods in Image Analysis,* J. K. Aggarwal, R. O. Duda, and A. Rosenfeld, eds. IEEE

Computer Society Press, Los Alamitos, CA.

24. Rom, H., and G. Medioni. 1993. Hierarchical decomposition and axial shape description. *IEEE Trans. Pattern Anal. and Machine Intelligence,* v. 15(10):973–981.
25. Samet, H. 1990. *Design and Analysis of Spatial Data Structures.* Addison-Wesley, Reading, MA.
26. Shapiro, L. G., J. D. Moriarty, R. M. Haralick, and P. G. Mulgaonkar. 1984. Matching three-dimensional objects using a relational paradigm. *Pattern Recog.,* v. 17(4):385–405.
27. Shapiro, L. G., and R. M. Haralick. 1981. Structural descriptions and inexact matching. *IEEE Trans. Pattern Anal. and Machine Intelligence,* v. PAMI-3(5):504–519.
28. Shapiro, L. G., and R. M. Haralick. 1985. A metric for comparing relational descriptions. *IEEE Trans. Pattern Anal. and Machine Intelligence,* v. PAMI-7(1):90–94.
29. Stark, L., and K. Bowyer. 1996. *Generic Object Recognition Using Form and Function.* World Scientific Publishing Co. Pte. Ltd., Singapore.
30. Swets, D., and J. Weng. 1996. Using discriminant eigenfeatures for image retrieval. *IEEE Trans. Pattern Anal. and Machine Intelligence,* v. 18:831–836. 525
31. Turk, M., and A. Pentland. 1991. Eigenfaces for recognition. *J. Cognitive Neuroscience,* v. 3(1):71–86.
32. Yue, Z., A. Goshtasby, and L. Ackerman. 1995. Automatic detection of rib borders in chest radiographs. *IEEE Trans. Medical Imaging,* v. 14(3):525, 536. 526

第15章 虚拟现实

假设外科医生要做一个去除病人脑部肿瘤的手术计划。在诊断过程中，得到了病人头骨和大脑的三维（3D）图像。借助虚拟现实（Virtual Reality，VR）技术，外科医生可以在三维数据模型上进行演练，而不需要实际的对象，通过尝试不同的入口路径、不同的操作手术，可以为病人选择一种最好的治疗方案。如果把总的大脑图谱（atlas）与三维图像数据对应起来，使外科医生可以分层观察大脑结构，并对不同手术方案的结果进行评价，以免损坏重要的大脑组织。虚拟现实技术推动了虚拟外科手术的发展，图15-1显示的是一幅用于虚拟现实系统的大脑模型绘制图。

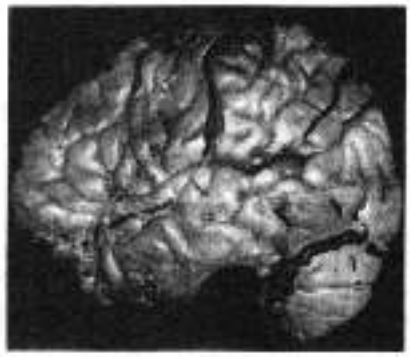


图15-1 基于3D网格模型绘制出的大脑图像，模型来自病人的核磁共振图像（华盛顿大学人脑项目组提供）

虚拟现实是一个新兴领域，一般认为它是计算机图形学的一个子领域，因为计算机生成的图像是虚拟现实系统的一个重要组成部分。虚拟现实应用系统本身非常重要，应该进行研究。在很多方面虚拟现实系统与本书的内容也是密切相关的，例如：（a）需要获取图像和处理图像；（b）需要高质量的立体显像，使用户在虚拟环境中有身临其境的感觉；（c）采用共同的数学模型，使实际空间与模型空间的3D点相对应；（d）有时需要用机器视觉技术测量用户或其他实际物体的位置。在仿真器工程（特别是飞行模拟器）、遥操作及计算机游戏方面，虚拟现实技术已经逐渐走向成熟。

15.1 虚拟现实系统的特征

本节首先列出虚拟现实系统或虚拟环境（Virtual Environment，VE）的重要特征，然后介绍几个的应用实例。

- 操作者对模型进行操作，模拟对实际物体的各种可能的操作。
- 具有高分辨率、高速度的显示技术，使用户深陷虚拟环境之中，并有身临真实环境的感觉。
- 用户能与模型环境顺利交互，并能使模型环境发生改变。
- 3D视觉反馈起相当大的作用。为了更好地观察目标，虚拟现实系统一般允许用户改变观察视点，或者控制目标物进行旋转和平移。尽管视觉反馈很重要，但其他反馈（如触觉反馈、运动觉反馈、力觉反馈或听觉反馈）也应该有，这样才能使用户感到物体的存在或者听到它们的碰撞声，等等。

图15-2说明操作者在实际环境中进行操作的情况，而图15-3说明人在虚拟环境中进行操作的情况。

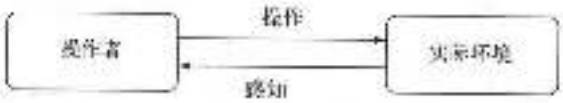


图15-2 实际环境下的操作

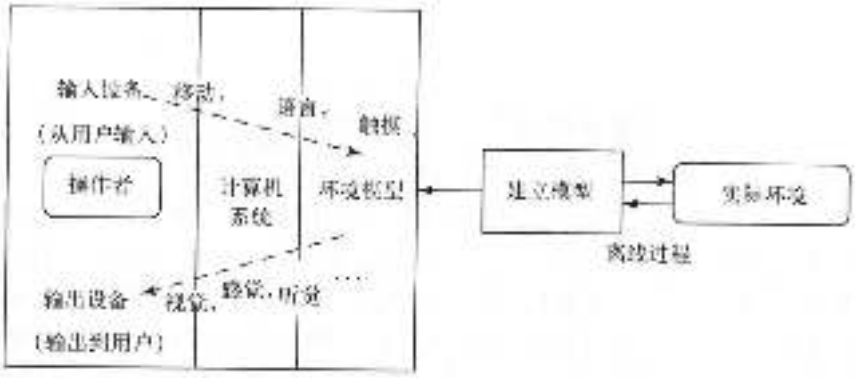


图15-3 虚拟环境下的操作

习题15.1 书与电影

- 人们在读书时会被书中的情节深深地吸引。参考上述虚拟现实系统的4个特征，考虑这种读书情况具有上述哪些特征？不具有哪些特征？（注意有的书允许读者在不同的续篇中进行选择）
- 人们看电影时会沉浸于其中，特别是当看宽荧幕电影或立体电影时。参考上述虚拟现实系统的4个特征，考虑这种看电影情况具有上述哪些特征？不具有哪些特征？

528

- 你知道或者玩过具有上述4个特征的视频游戏吗？请加以说明。

15.2 虚拟现实的应用

虚拟现实常常与新的应用需求相联系，而新硬件（一般价钱较贵）的出现使这些应用成为可能。下面的几个常见系统，都在某种程度上采用了虚拟现实技术。

15.2.1 建筑漫游

用户可以和房屋建筑模型进行交互，在房屋内虚拟行走，透过虚拟窗户欣赏窗外的虚拟风景。如果是杰斐逊（Jefferson）的Monticello建筑模型，用户能够看到杰斐逊的古董收藏品，独一无二的床及炮弹钟。许多这样的历史古迹和虚拟博物馆目前正在进行数字化。简单情况下，用户可以通过万维网及一般的平面显示器，走进这些虚拟环境。也许不允许用户修改Monticello建筑模型，但用户可以捡起并查看其中的古玩。如果用户计划建一座真正的房子，在建成之前他可以修改建筑模型，还可以改变墙面设计和家具布局。

与建筑漫游类似的是虚拟飞越（Flyby），人们能够欣赏美国亚利桑那州的大峡谷，就好比在峡谷上面飞行一样。简单情况下，用户只是乘坐飞机的一名旅客，自己决定不了视点；复杂情况下，用户是飞行员，可以改变路线并以多种方式欣赏美景。

15.2.2 飞行仿真

利用仿真飞行器，用户能控制飞行器飞过各种地形，并在各种机场起飞和着陆。30年来，越来越多的人们满头大汗，心情激动地走出仿真训练器，这说明他们曾深深地沉浸在虚拟环境之中。

15.2.3 解剖组织的交互式分割

虚拟现实系统可以辅助进行医学个体识别，以及根据3D数据建立解剖模型。换句话说，虚拟现实系统支持交互式组织分割。例如，通过立体图像向用户显示三维 MRI数据。通过与

这些数据进行交互。用户可以在血管中心或心脏周围做一系列的点标记。系统中需要的立体显示设备和3D输入设备将在15.5节进行介绍。 529

现在出现了更多的虚拟现实系统，如病痛管理系统、恐惧症治疗系统、弱视辅助系统、驾驶仿真器、科学可视化及虚拟教室等。图15-4所示的是目前在华盛顿大学的虚拟现实项目：动态虚拟运动场（Dynamic Virtual Playground）。虚拟运动场是一个原型系统，用来研究在虚拟情况下同时进行的多项比赛活动。它可用来模拟学校实验室，其中每组学生参与不同的研究项目。

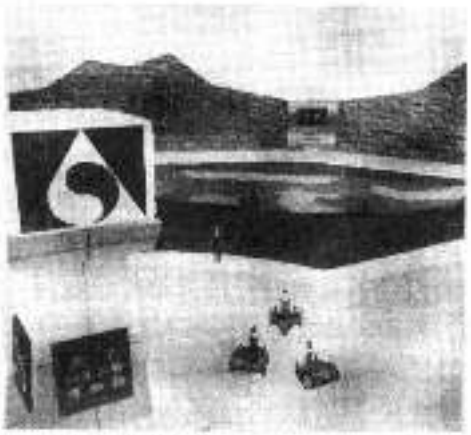


图15-4 动态虚拟运动场是一个实验环境，用户可在其中进行比赛（华盛顿大学HIT实验室提供）

15.3 增强现实

承包商如果要重新建立现有场所的模型，需要知道当前水管、煤气管道和电路的分布情况。从现有的地图、蓝图或CAD文件可以得到这些数据。下面的情形是一个增强现实（Augmented Reality，AR）（又称为混合现实（Mixed Reality））的实例。承包商戴着头戴式显示器（head-mounted display, HMD），计算机图形叠加在他所看到的实际场景上面。当他看向地面时，他看到的是埋有水管的蓝线；当他看向墙面时，他看到的是表示电路的红线和表示水管的蓝线。在某种意义上说，增强现实使承包商具有超人的能力，他的视力能穿透墙壁！

建立这样的增强现实系统需要解决如下问题：

- 建立物体的3D模型，以增强实际视图。
- 通过标定使实际工作空间与3D模型空间对应起来。
- 跟踪用户姿态，以确定用户在实际工作空间中的视点。
- 实时显示的内容，是实际图像和基于模型生成的计算机图形相结合的产物。
- 对头部运动的响应时间以及图像与图形之间的配准精度，会严重影响系统的有效性。

增强现实环境如图15-5所示，请把该图与本章中的其他示图进行对比。增强现实有很多用途，下面是它的几种应用情况： 530

- 增强现实辅助外科手术。对实际病人做手术的外科医生要观察CAT扫描数据，包括根据病人活体图片设计的手术路径计划（这个计划可能已经通过前面的虚拟现实系统制定好了）。
- 在个人电脑（PC）的主板检查中，检查人员把一块新的PC主板与CAD模型做比较，证实所有要求的元器件和引线齐全。主板要精确放在一个夹具中，使从摄像机得到的图像与CAD模型精确配准。检查人员通过一台大显示器观察图像。
- 汽车驾驶员观看显示器，显示器显示前面的地形特征、仪表盘内的放映机把建筑物和街道的名字投影到挡风玻璃上。
- 几个人在开会，想对他们建立的计算机模型进行讨论。他们要能看到模型，指出并讨论它的特点，互相还能看到对方及所在的环境。远程会议就属于这种情况。参加会议的人有的离会场很遥远，他们不仅想看到计算机模型，还想看到其他参加会议的人员。

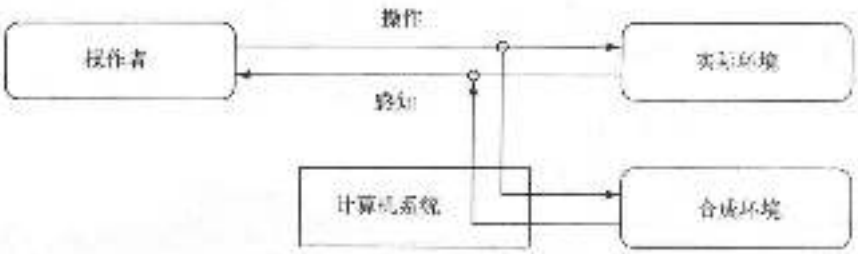


图15-5 增强现实环境下的操作

图15-6是一个远程会议系统。用户的桌面上放了两张卡。每张卡背景为白色，上面有一个方形黑色区域。黑色区域内显示一幅人脸图像。用计算机视觉技术找到卡片，用统计模式识别方法识别出卡内的模式，从而确定每张卡的含义。利用增强现实技术，远端人员的图像显示在一张卡上，要讨论的模型图片显示在第二张卡上。

531

图15-7显示同一房间的两个人，正通过透明护目镜观看网页，他们正在讨论这些网页。

532

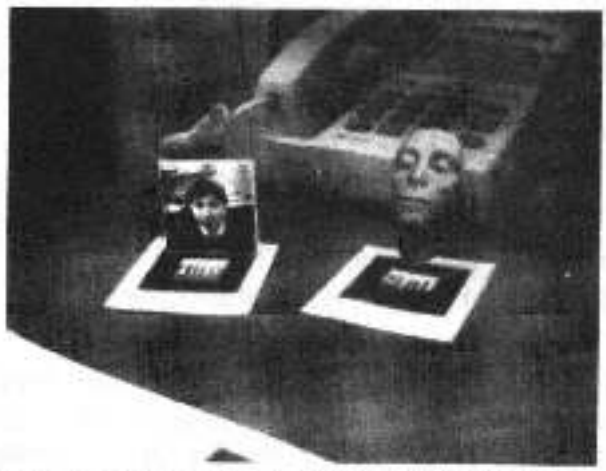


图15-6 增强现实技术在远程会议中的应用（华盛顿大学HIT实验室提供）

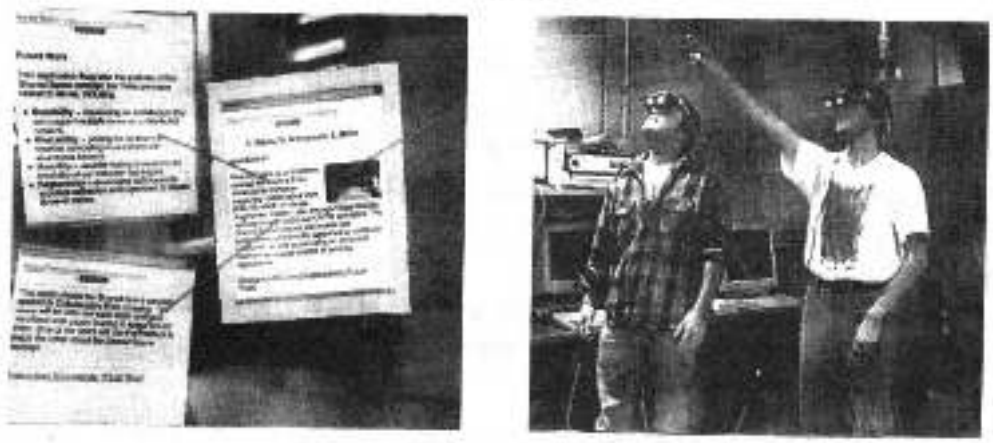


图15-7 两人戴着透明的，增强现实护目镜（右图），他们能够看到真实世界，也能够看到计算机生成的图像。他们两个正在观看上面的网页，网页就像是在空间漂浮着（华盛顿大学HIT实验室提供）

15.4 遥操作

遥操作（teleoperation）是一个工程学科，它极大地丰富了虚拟现实的研究内容，特别是通过传感器和执行器把操作者与环境合二为一。请将图15-8与本章其他图做个比较，看看它们之间有什么异同。利用遥操作技术，操作者能够对远程实际环境中的工作情况进行控制，而机

器人或类似机器人的机械根据操作者的命令在实际环境中完成操作。成功应用的例子如下：

1. 在美国火星探路者控制任务中，操作者通过计算机向火星上的导航机器人发送命令，让它向前走10 cm并抽取土壤样品。附近的登陆车（登陆车把机器人运到火星上）上面安装有摄像机，摄像机摄取的图像被传送到地面。由于距离遥远，传送图像和命令大约需要11分钟。



图15-8 遥操作系统

2. 一次小事故之后，在核电厂的危险区域，通过无线电通信，操作者控制远程机器人用真空吸尘器打扫放射性废物。操作者戴着头戴式显示器，其中显示的是受污染区域的情况，与机器人身上安装的摄像机拍到的实况一样。用仿真杆模拟实际真空杆，仿真杆内的传感器测量杆的位置和运动，并产生控制信号控制远处的实际真空吸尘器动作。

3. 外科医生进行远程外科手术，缝合一个类似缝棒球的虚拟物体。传感器精确记载这个缝合过程的运动参数，然后传送给远程机器人，远程机器人正在缝合一只真正的狗身上的伤口。这个实验曾经实现过。要把外科医生的精湛手术传送到其他到不了的地方，在狗身上做的这个试验也许是完成了第一步。

4.（未来情景）一个动脉硬化病人去医院做清理动脉斑的手术。通过MRI机得到病人身体的实时3D视图。把微型机器人放入血管进行全身血管的清理。在以前诊断中得到的绘制图的帮助下，主治大夫通过3D输入设备指出脉管系统的哪个区域需要清理。MRI设备然后就工作在交替模式，一种模式是像以前那样用来成像，另一种模式用来控制微型机器人在指定区域内进行清理手术。

图15-9显示的是一个实际系统，Kyushu电力公司的遥控机器人正在修理高压电线。

在讨论实现虚拟环境所需的设备及数学模型之前，我们先给大家讲一段有趣的故事。有人设计了一辆遥控铲土机，使远程操作者能控制对土壤、煤炭的搬运工作等。操作者戴着数据手套（data glove），其中的传感器能够测量手掌和手指的位置，系统再把这个信号转化成控制铲土机的信号。铲土机上有两台摄像机，为操作者的头戴式显示器提供左、右两幅图像。操作者用于抓一把桌子上的锯屑，远处的铲土机将仿效这个动作抓一次实际工作环境中的煤炭。假设操作者鼻子发痒，并用带数据手套的手去挠痒！当这只手向鼻子移动时，实际环境中的铲土机末端执行器就会向摄像机移动。图像结果又通过头戴式显示器反馈给操作者，他就会感到脸上被沉重的铁铲击打了一下！这只是虚拟的一击，操作者本身没有受到伤害，但是，他心理上感到非常不舒服，需要把工作停下来。（如果系

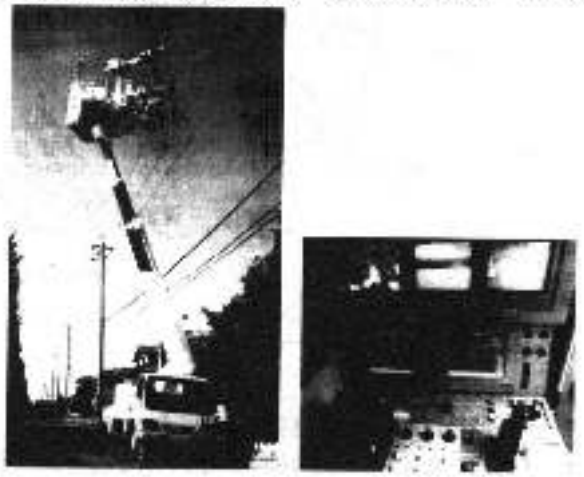


图15-9 Kyushu电力公司的Hot-Line遥控机器人系统。遥控机器人正修理高压电线（左图）系统的操作界面（右图）（Blake Hannaford提供，经MIT出版社允许，K.Goldberg再版，*The Robot in the Garden*, Cambridge, MA: The MIT Press, 2000）

534 统设计的不好，就有可能真的损坏安装在铲土机上的摄像机了。）

定义110 遥操作指操作者对远程实际环境中的实际设备进行遥控操作。来自远程环境的反馈信号和来自设备的控制信号，使操作者产生一种幻觉，感到自己处于实际环境当中。

定义111 虚拟现实是一种为用户合成的现实，通过丰富的现实模型和输入输出设备，由计算机系统产生。操作者感到是在对实际物体进行操作，而实际上不是这样。操作者能感觉到并能改变的这个虚假环境称为虚拟环境。

定义112 增强现实或混合现实是实际环境和虚拟环境相结合的产物。计算机系统合成的输出与对实际环境感知的数据进行融合，来加强人对现实的理解。

定义113 合成环境在遥操作、增强现实或虚拟现实系统中，由计算机系统和沉浸式I/O设备为操作者产生的一种环境。（有时上述几种情况都被称为虚拟环境。）

15.5 虚拟现实设备

为了使操作者沉浸于合成环境之中，常常要用到一些仪器设备。参考图15-10，可以看出它就是前面讨论的增强现实系统，其中用到的设备在遥操作、增强现实或虚拟现实中都要用到。承包商要重新建立一个建筑物的模型，他在建筑物内走动，并在墙上做标记，墙内有水管和电路。承包商通过透明HMD观看实际墙壁，HMD中的光学器件把计算机生成的图像叠加到所看到的景物图像上面，计算机生成的图像中反映水管和电路的布局。

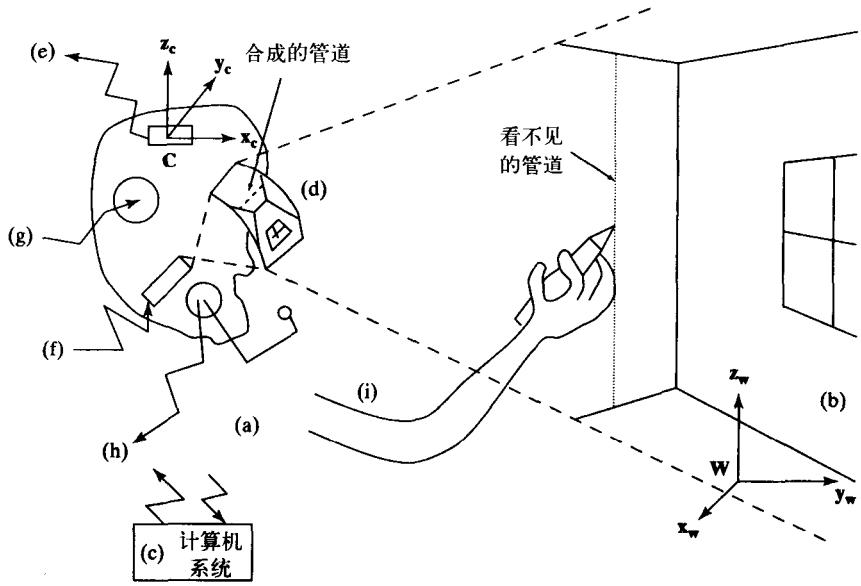


图15-10 增强现实系统中操作者使用的设备。操作者对墙内的管子位置进行标记，根据房屋CAD 模型确定看不见的管道位置

承包商（a）通过计算机系统与增强的实际环境（b）交互，计算机系统（c）在图像（d, e, f）基础上附加其他非视觉特征，并使用语音（g, h）与承包商交流，承包商自由走动并在墙上做标记（i）。当他走动时，通过分光镜（d）能看到真实世界的图像。位姿传感器（e）把人头的位置与姿态送给计算机，然后计算机利用这些参数，参考水管和电路的三维CAD模型生成新的图像。把生成的图像投影到操作者能看到的镜面上（f），操作者看到的是对实际环境增强了的视图。

15.5.1 头戴式显示器

透明（see-through）头戴式显示器（HMD）如图15-11的左边所示。分光镜允许镜外实际场景的光线进入，但内部作为反射镜反射来自计算机生成的图像，结果使操作者看到了增强的图像。注意，由于头戴式显示器或头戴内的光学器件很小，显示的图像也很小。另一种设计是用不透明（opaque）HMD，如图15-11的右边所示。注意，所有的元器件都在HMD内部，并随HMD移动，包括反射镜、摄像头及图形显示器。合成图像是由摄像机捕捉的数字图像和计算机生成的图像综合得出的。透明设计能够产生分辨率较高的图像，而不透明设计可以更好地控制用户的视线。 535

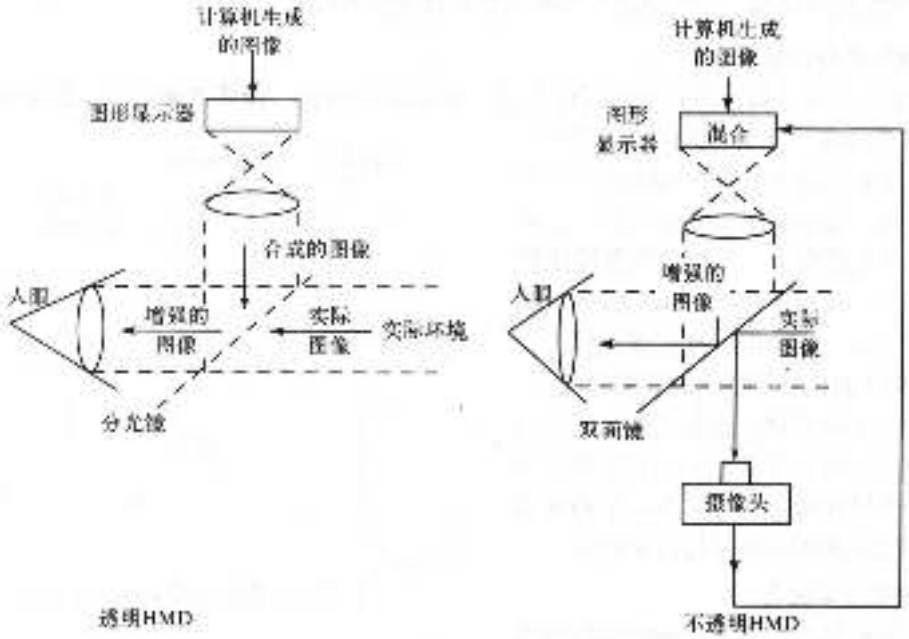


图15-11 头戴式显示器示意图

习题15.2 增强现实系统的配准精度

参见图15-10，(a) 当输出操作者的头部位姿信息时，如果位姿传感器的方位角产生2°的误差，那么操作者标记的垂直管线与管线的实际位置将产生多大误差（单位cm）？

(b) 假设视觉显示器的视场角是120°，覆盖500个像素。(如果需要，你可以假设摄像机镜头焦距为2.0 cm。也可以假设镜头距离被观测的墙壁3m)。增强图像中管子的投影和管子在图像中的实际位置之间的水平距离将怎样？单位采用像素。

习题15.3 增强现实系统的配准精度

这个问题与习题15.2有关系，而且使用同样的设备，讨论对汽车仪表盘的校验问题。由操作者对真实仪表盘和增强现实系统中的CAD模型做比较。与前面问题不同的是，所有的CAD特征在实际图像中对操作者都是可见的。操作者要检查仪表是否存在，是否正常工作。这些仪表包括里程表、油压指示、无线电等。增强现实系统同时启动测试设备，并向操作者提供信息，告诉他下一步要找的是什么。假设视场角是60°，镜头距离仪表盘大约60cm。和上个问题一样假设位姿传感器方位角有2°误差，(a) 如果增强图像中有一个表示无线电按钮位置的

红圆圈，由方位角引起的水平配准误差是多少？（b）如果配准误差很大，请找出一种可使误差自动减小的计算机算法。你的方法对任何HMD都有效吗？（c）如果配准误差控制的比较小，则操作者的校验工作做的就会比较好。请找出一种能自动产生较小配准误差的计算机算法。你的方法对任何HMD都有效吗？（d）针对这个校验问题需要用增强现实系统吗？能采用全自动方法吗？试加以说明。

习题15.4 多操作者的增强现实系统

假设几名外科医生组成一个治疗小组，每个人都戴着HMD，观看叠加在实际病人视图上的外科计划和解剖结构。这可能吗？解释为什么可能或不可能。

15.5.2 虚拟灵巧手术

虚拟现实系统支持基于模型的灵巧手术，如图15-12所示。操作者通过从上面投影到反射镜的立体显示器，观看合成的工作空间。这样允许双手在镜子下面的实际3D工作空间自由移动，操作镜子下面的工具。需要仔细跟踪工具的位姿，并把它的图像反投影到镜子上，使操作者看到相应的反馈图像。图中显示一位操作者正在虚拟器官上做切口。3D工具的各种技术将在后面介绍。显然，3D位姿的精度、感知速度和显示器刷新频率对系统至关重要。除了常见的手术实习和外科计划方面的应用，艺术家也可用这种系统对3D模型进行数字雕刻。

536
|
537

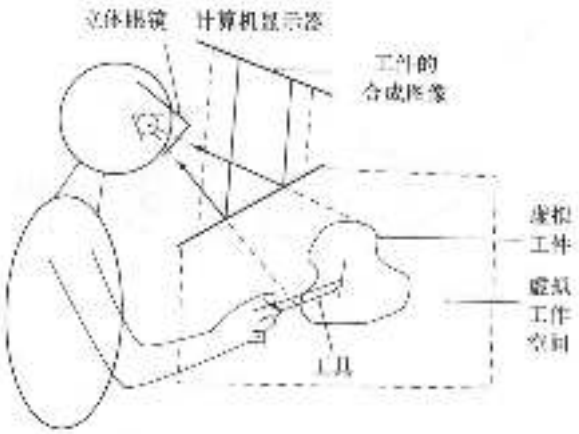


图15-12 虚拟灵巧手术台示意图

15.5.3 立体显示设备

立体视觉也许是感知10m内3D物体的最重要视觉信息，同时也是从虚拟现实系统反馈信息的主要手段。立体显示一般有两种设计方式。对不透明的HMD，可以把分离的图片送到双眼，利用第12章讲的数学模型，以合适的视差把左、右图合成起来。不透明HMD可以生无限虚拟世界产生较宽的视场，给操作者带来很逼真的身临其境的感觉。但制造这些设备比较困难，因为需要适应不同的操作者，注意多个用户可沉浸于同样的虚拟世界，每个用户都戴着HMD，HMD各有一个位姿传感器，视点不同，在用户的眼前出现的虚拟环境也不同。另一种方法是用常规的图形显示器，如图15-12所示，随时间交替显示左、右图，操作者戴着立体眼镜，立体眼镜在时间上与显示同步，使左眼只看到奇数帧，右眼只看到偶数帧，或者左眼只看到偶数帧，右眼只看到奇数帧。这种随时间交替的输入信号，人眼感觉的效果是3D的。用这种方案设计的系统，价钱不贵且容易使用，但身临其境的程度有限，因为所有的图像都限制在显示屏上，因此称为鱼缸虚拟现实。由于显示屏的切削，使虚拟世界的视图受到限制。另外，只有一台显示器与一个用户同步，而其他观看显示的用户不能控制观看的视点，即使移动身体也控制不了观看的视点。采用环形显示可以减弱鱼缸效应，用户就好比处在一个洞穴之中，但对于多个用户仍不能完全产生身临其境的立体效果。

538

15.6 虚拟现实感知设备

15.6.1 视觉

如前所述，虚拟现实系统的视觉输出一般包括立体显示。立体显示可以通过HMD实现，也可以通过立体眼镜观看标准的计算机显示器实现。通过VE中测量用户位姿的传感器将该输出提供给用户，以便显示合适的视图图像。

输入到VR系统的视觉设备，如果有的话，通常具有跟踪用户眼部、头部或身体位姿的功能，并把位姿信息输入到建模系统。眼部跟踪器能够提供凝视的方向。这些设备可装到HMD上，或者与操作者完全分开。基于HMD的头部位姿跟踪可用一台摄像机实现，即利用摄像机跟踪HMD上的特征点。最近的VE研究结果表明，利用来自多台摄像机的最佳视图能够跟踪人手、人头、人脚和四肢。视觉输入设备的一个优点是，作为穿戴设备可以使用户活动不受约束。但是，尽管用户活动不受设备的约束，但设备对工作空间仍然有所限制。有的人眼跟踪器依赖专门的红外照明；有的身体跟踪器依赖受控的背景颜色。

15.6.2 听觉

计算机语音输入已有15年以上的历史，很多系统如电话系统和家庭PC应用中都要用到。语音输入具有天生不需要专门学习的优点。当操作者双眼和双手忙于其他工作而没有空闲时，就有必要使用语音。类似地，语音输出也是一种方便的通信方式，它不需要显示器。

声音输出可以改善接口。如当文件夹图标被拖到垃圾箱时，金属碰撞声证实文件已被删除。它也能提高VE内身临其境的程度，如驾驶虚拟交通工具的操作者能听到发动机的声音和煞车时轮子发出的声音。或者，远程操作者能够通过所放音乐的疯狂程度而感到辐射能量的大小。

定义114 **语音合成**通过编码产生声音数据或声音控制信息，这种声音不是自然产生的声音。

15.6.3 位姿

3D位姿传感器用来测量人体某部分或所持工具的位置和姿态。6自由度传感器包括HMD 539
上常用的Polyhemus传感器、游戏棒及更新型设备如Green与Halliday（1996）描述的bat设备。*x-y-z*位置传感器包括sparking styllus 和各种机械设备。也有装在人身体上的机械关节，通过确定这些关节的位置，由计算机把位姿信息输出给穿戴者，不过这种设备不太常见。这种设备具有力反馈功能，下面进行介绍。

15.6.4 触觉

人通过触觉、力觉和运动觉与外部世界相互作用。触觉的产生，是由于皮肤的神经能感知温度、硬度和表面光滑度。肢体和肌肉的神经能感知肢体的位姿和肌肉的松紧以及它们的变化。前庭系统的神经能感知身体的运动。

定义115 人的**触觉**包括接触的感觉（接触觉）和身体位置、力或运动的感觉（肌肉运动觉）。各种机电设备能够提供力的输入和输出。

15.6.5 运动觉

人对运动的理解是多种感知系统共同起作用的结果。正如在电影或仿真器中出现的那样，视觉显示足以引起令人难受的晕动病。在各种VE系统中，由踏车、机械机构、振动台或离心机给人体带来的运动刺激，使人身临其境的感觉比视觉显示所引起的感觉更强烈。另外，前庭系统会因受到冷空气或水流刺激而产生反应。在运动感知方面计算机视觉具有重要的作用。

理想情况下，操作者在实际环境中自由运动，由跟踪摄像机拍摄一系列图像，再对这些图像进行分析，得到解释运动的所有信息，可将这种解释映射到人体的计算机模型上。已有跟踪人体运动的商业系统。典型的系统依赖人体上专门放置的标记，这样做是为了简化图像分割和特征抽取。这种运动测量系统，在研究各种运动和在整形外科设备中得到了应用。有的研究系统不需要在人体上放置标记，直接对人头、双手和双脚的运动进行跟踪。

15.7 简单3D模型绘制

为了创建虚拟场景，需要建立目标模型以及由模型生成图像的软件工具。目标模型可以是复杂的网格模型，就像第13章和第14章中介绍的那样，或者是较简单的线框模型。建立这些模型可以借助计算机辅助设计软件包，如AUTOCAD。图15-13显示的是汽车线框模型，用的是交互式CAD软件工具。



图15-13 福特概念车的线框模型（福特汽车公司提供）

一旦建立了3D模型，就可以显示它在任意视点、不同光照下的视图。

定义116 **绘制**即从模型生成图像的过程。

540 绘制可以认为分成两步：

1. 对于选定的视点，确定模型的哪个表面是可见的。

2. 确定创建图像上对应点的像素值。

从概念上来说，步骤1就是从视点沿期望的方向到目标构造一条光线，光线与目标的首次
541 交叉点就是沿这条光线可见表面上的一点。这个概念称为**光线跟踪**（ray tracking），许多算法能够实现这一点，目前的计算机采用z-buffer硬件机制，能够快速执行步骤1。

步骤2有简有繁。简单时，目标具有特殊的颜色，它由反射属性已知的特殊材料制成。光线来自一定方向的点光源，也存在一些环境光。数学模型如第6章介绍的Phong明暗模型，可用来确定目标物上一片小区域所对应的像素颜色。图15-14是汽车线框模型的简单绘制图。如果加上多个光源、面光源、阴影、透明表面和交叉反射等要素，可以创建更逼真的图像（以牺牲速度为代价）。

图15-15是两幅绘制图像，用到几个不同的交通工具模型。图像中的一些目标采用前面介绍的方法进行绘制，另一些带纹理的表面由于模式复杂，绘制起来将很费时间。（参见左图中



图15-14 汽车的绘制图像（福特汽车公司提供）



图15-15 基于汽车模型和其他模型绘制的场景（Habib Abi-Rached提供）

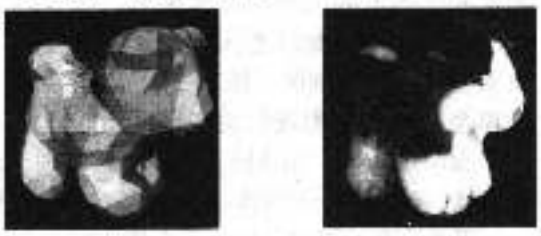
靠左边的建筑物和附近的人行道。）对这些表面采用的是纹理映射，而不是绘制。纹理映射将在下一节讨论。

15.8 实际图像和合成图像融合

基于模型的合成绘制也就做到这个程度，可以看出结果不是很逼真，而且为了改善逼真程度所需要做的运算又很费时间。具有复杂纹理的现存图像，不仅能够提高绘制图像的逼真程度，而且能够加快绘制的速度。纹理，可以是人工生成的模式或者是一幅实际图像（可以是其中一部分）。在绘制表面的过程中，不是用单一的颜色值去染色，而是把给定的纹理"粘贴"或"涂刷"在表面上。这就产生了纹理映射（texture mapping），其中可见像素的最终像素值从所给纹理图像的像素中选取。

定义117 纹理映射是把纹理贴到一个光滑表面的过程，这样就建立了表面的纹理图像。 542

图15-15中经纹理映射的表面是多边形平面，这是纹理映射的最简单的表面。纹理映射也可针对更复杂的曲面，如在橘子上涂刷粗糙的果皮纹理。如果目标是自由形态而且用网格模型表示，则可以分块贴加纹理。但对于复杂目标，需要用更高级的方法。计算机图形学的最新技术是用目标的实际图像提供所需的纹理。图15-16a是重建小狗的粗略网格模型（见第13章），图15-16b是该模型的纹理映射图像。在这个例子中，纹理来自小狗的实际图像，所拍图像的视点与模型显示的视点一致。实际上，我们希望显示任意视点的纹理映射图像，就像图像绘制的情况一样。



a) b)

图15-16 小狗的粗略网格模型和纹理图像（Kari Pulli提供）

定义118 基于图像的绘制，是指在一组实际目标图像的基础上，产生任意视点的合成图像的技术。

基于图像的绘制不需要目标的几何模型，只需存储大量不同视点的图像，在这些图像间进行插值后生成任意视图。然而如果我们有目标的网格模型和少量从各视点得到的实际图像，那么几何模型加上已有的图像就可以产生很逼真的绘制图像。第13章描述的重建系统，根据一组深度图像和相关的颜色图像建立目标的粗略网格模型，该重建系统也能根据用户选择的各个视点绘制出目标图像，所用的技术称为基于视图的纹理化（view-based texturing）。图15-17显示这种方法的基本原理。左边，用户用鼠标控制目标的伪彩色重现图，并旋转它到希望的视点位置；中间，是最靠近该视点的三幅图像；右边，在期望方向上生成的小狗纹理映射图像。为生成图像中的每个非背景像素，从图像上的某

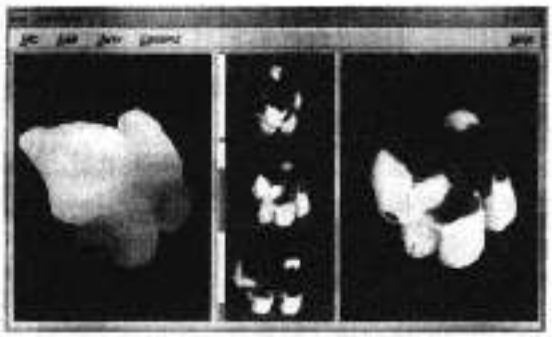


图15-17 （Kari Pulli提供）参见彩图15-17

（左）小狗模型的深度图像

（中间）附近视点的三幅真彩色视图像

（右）对视图像素进行加权得到的绘制图像。

个像素到三维模型发出一条光线，然后从模型分别到中间三幅图像的对应像素发出一条光线。对这三个像素的颜色值进行融合，就得到了生成图像上所选像素的值。三个像素的作用不是平均处理的，融合算法要考虑存储视图与要绘制视图之间的相似性，从目标模型到存储视图像素的光线方向以及存储视图像素与视图边界的靠近程度。也可用z-buffer算法软件，去掉因太远而实际上不能落在表面上的像素。

543

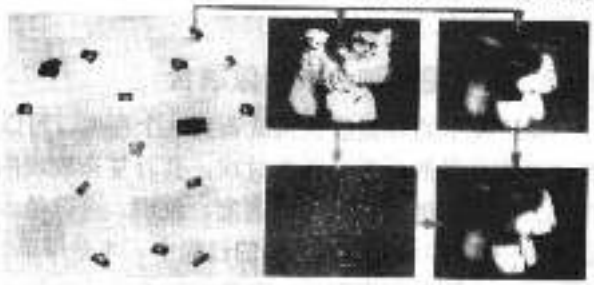


图15-18 由少量目标视图生成的配准深度图像和彩色图像，可用来生成高质量的绘制图像，而不需构造目标的全三维模型（Kari Pulli提供）参见彩图15-18

（左图）可能的视点

（中上图）某视点对应的深度图像

（右上）同一视点对应的彩色图像

（中下）根据深度数据建立的网格模型

（右下）把彩色数据纹理映射到网格模型得到的绘制图

构造真实目标的全三维网格模型要花费很长时间，基于视图的纹理化技术实际上不需要完整的几何模型，只用一组配准的原始深度图像和彩色图像就能够生成绘制图。图15-18利用小狗模型对这个过程进行说明。所用的不是全网格模型，而是针对每幅样本视图生成部分网格模型。这些部分网格模型与目标的彩色图像一起产生绘制图像，其效果和使用全网格模型的效果一样逼真。

544

再者，有的真实目标不适合建立实体模型。当一个目标具有较薄的部件，如船帆或植物的叶子，需要网格模型有很高的分辨率表现出拓扑结构。因为有可能从几幅视图得到深度数据与颜色数据，基于视图的纹理化技术仍然可以采用，从而产生在任意方向上都很逼真的目标图像。图15-19显示对花篮的绘制过程。



图15-19 由于目标的部件太薄，用同样的技术建立全三维模型几乎是不可能的（Kari Pulli提供）参见彩图15-19

（左上）三幅不同的目标彩色图像

（左下）把三幅原始图像的像素点映射到新的视点，产生不同视点的三幅新图像

（右）最后的绘制图像，三幅新图像的加权结果

习题15.5 增强立方体图像

用第13章的方法标定摄像机，标定物是一个立方体或小盒子（用7个可见角点作为控制点）。设计一个二维的模面，模面上的3个点表示数字3。对于3D立方体的上部，参考第11章构造映射函数g，把点$[x_w, y_w, z_w]$从3D立方体顶部映射到2D点$[x_i, y_i]$，在正方形内建立模面的模型。这个映射应该是线性的，并把立方体上部的四个角点映射到模面的四个角点。生成并打印图像，图像包含实际标定场景的每个点（除了立方体上部）的像素。图像中应该包括模面模型的像素。用同样的映射函数g重复上面的过程，其中图像采用扫描得到的任意方形图像，可以用你自己的脸部图像。

习题15.6 合成立方体图像

用习题15.5中得到的实际摄像机投影矩阵，或用第13章中的某个投影矩阵。(a) 合成一幅

立方体图像，立方体位于摄像机的视场内。(b) 生成与 (a) 一样的图像，要求立方体的一个面就是上个问题中的模面。(*c) 用两张不同的脸部照片对立方体的两个面进行纹理映射，用第11章介绍的映射方法。 545

15.9 人机交互与心理问题

显然，采用本章介绍的设备，在很多方面能够提高人机接口的品质和带宽。在虚拟环境中，主要目标是提高身临其境的程度。如虚拟外科手术需要通过视觉、触觉和力觉进行高品质人机交互。人体之间的差别为虚拟现实工程系统的建立带来了困难。例如，由于头部尺寸和形状不同，使得HMD的设计工作变得复杂；人类视觉系统的差异，使基于立体融合的显示控制发生困难。另外，不同的人对客观上一样的颜色、粗糙度和声音等的敏感情况也有稍许差异。

虚拟现实系统会产生令人不舒服的感觉，如晕动症通常就是不希望出现的。其他可能还有眼疲劳、劳累和受挫。如果立体融合子系统与操作者或现实匹配的不好，就会产生这些效果。更坏的情况是，仿真飞行器的操作者突然在虚拟环境中消失！这些问题给虚拟现实系统的设计人员带来很多挑战。

15.10 参考文献

Durlach和Mavor (1995) 编著的书中，综述了虚拟现实在艺术领域的现状。同年Bartield和Furness所编著的书中，搜集了大量关于虚拟现实研究的最近的文章。这两本书提供了丰富的背景知识，包括设备描述、定义、优秀的实例、图表和分类别的参考文献。

虚拟现实在医学上有多种用途。在Posten的Serra (1996) 发表的论文中，讨论了虚拟现实在计划外科手术和交互式3D图像分割方面的应用。我们的虚拟灵巧手术台就是受到了这篇文章的启发。虚拟现实在健康恢复和恐怖症治疗方面的应用，在Strickland (1997) 发表的一系列文章中有所描述。建立目标模型并对它们的行为和运动进行定义是当前研究的难点。关于问题的描述以及解决问题的方法，请参考Green和Halliday (1996) 以及Deering所发表的论文。虚拟现实给用户带来的身临其境的感觉，一方面对用户有所帮助，另一方面也会使用户难受或者恼怒，特别是当立体视觉与运动图像显示不恰当时，这部分内容请参考Viire (1997) 发表的文章。Stuart (1996) 编著的书中，包含许多实用的表格，总结了人类的感知特征和虚拟现实系统中的输入输出设备的特点。基于视图的纹理化部分参考的是Pulli等人 (1997) 的工作。Ohta和Tamura (1999) 编著的《Mixed Reality》一书中，对最近的一些工作进行了收集整理，其中包括本书所讨论的一些内容。

1. Barfield, W., and T. Furness III, eds. 1995. *Virtual Environments and Advanced Interface Design.* Oxford University Press.
2. Biocca, F., and F. Levy. 1995. *Communication in the Age of Virtual Reality,* F. Biocca and M. R. Levy, eds. L. Erlbaum Associates, Hillsdale, NJ. 546
3. Deering, M. 1996. The Holosketch VR Sketching System, *Communications of the ACM,* v. 39(5):54–61.
4. Durlach, N., and A. Mavor, eds. 1995. *Virtual Reality: Scientific and Technological Challenges.* National Research Council, National Academy Press, Washington, D.C.
5. Green, M., and S. Halliday. 1996. A geometric modeling and animation system for virtual reality. *Communications of the ACM,* v. 39(5):46–53.
6. Ohta, Y., and H. Tamura, eds. 1999. *Mixed Reality: Merging Real and Virtual Worlds.*

Ohmsha, Ltd., Tokyo, Japan; also distributed by Springer-Verlag, New York.

7. Poston, T., and L. Serra. 1996. Dextrous virtual work. *Communications of the ACM,* v. 39(5):37–45.
8. Pulli, K., M. Cohen, T. Duchamp, H. Hoppe, L. Shapiro, and W. Stuetzle. 1997. View-based rendering: visualizing real objects from scanned range and color data. *Proc. 8th Eurographics Workshop on Rendering* (June 1997).
9. Strickland, D., ed. 1997. Special issue on VR and health care. *Communications of the ACM,* v. 40(8).
10. Stuart, R. 1996. The Design of Virtual Environments. McGraw-Hill, New York.
11. Viire, E. 1997. Health and safety issues for VR. *Communications of the ACM,* v. 40(8):40–41.

547

第16章 案例研究

本章描述两个不同的商业系统，它们都利用计算机视觉和模式识别技术，解决实际应用中遇到的问题。这些案例集成了不同的硬件和算法，通过了解这些案例，使我们对完整的系统设计有所了解。其中用到的大多数方法（不是全部），已经在本书前面的章节中讨论过。第一个案例是IBM公司开发的Veggie Vision系统，用于超市收款台进行商品识别。另一个是虹膜识别系统，在自动柜员机（ATM）或安全设备中进行身份验证或者身份识别。

16.1 Veggie Vision系统

条形码的使用极大地减少了超市售货员的劳动强度，但处理不同商品的劳动强度仍然很大。有的商品，如马铃薯或苹果可预先进行包装并打上条形码，以便能够像灌装和箱装产品一样进行处理。然而许多商品是散装的，主要是为了方便客户单个挑选，例如西红柿或者青豆。顾客可以把散装商品放在塑料袋中，也可以不放。在一般商店中，用台秤称取散装商品的重量。收款员可能要确认商品类型并把代码输入机器。这个过程很有必要进行自动处理。为什么不在台秤上面安装一个摄像头，借助它来自动识别商品的类型呢？如果这样，就能大大简化收款员的工作，而且能够改善对存货的管理。事实上，在IBM的T. J. Watson研究中心，已经开发出一个称为Veggie Vision的系统。实验室的实验证实了系统的有效性，现在正在进行实地试验。自动识别系统还存在其他方面的优点，例如可以根据商品的大小和成熟度进行更详细的定价。后面我们会更详细地讨论超市商品销售问题，以及IBM的解决方法。特别感谢Bolle，Connell等人（1996），他们提供了Veggie Vision系统的相关文档。读者可以参考本章列出的参考资料，从他们发表的文章中得到更多的信息。 548

16.1.1 应用场合和要求

美国市场大概有m=350种不同的商品，但是一个商店可能只卖150种左右。这些数字都说明了商品识别自动化是一个困难的问题。为了节约资金，自动识别系统应在一秒内做出识别判断，所采用的计算设备成本应不超过目前超市使用的扫描仪和计算机的成本。希望新设备所占用的空间与当前使用设备的空间一样大，而且不要改变商店现有的内部环境。

几方面因素决定了这种系统必须适应商店环境的变化。首先，不同商店的商品种类也不同。其次，同一家商店的商品会随季节而变化，甚至是每天都在变化。例如香蕉刚到时是绿色的，以后会逐渐变黄。有效的系统必须能够适应这种变化，而且能够进行扩展以处理新的商品。

最后，整个系统的操作，对操作员来说必须是可接受的。这包括最初学习如何使用系统，系统的自动化操作方式，以及当自动化程序因故出现问题时要由操作员做出决策。整个系统，包括机器和操作员，必须比目前大部分商店中的手工操作更加有效。图16-1显示所预期的整体系统。



图16-1 超市蔬菜识别系统的设计草图（需要人机交互时就显示出合适的商品类型）（R.Bolle和J.Connell提供） 549

通过触摸屏向收款员显示结果，当自动化系统确定不了时允许由收款员进行确认。

16.1.2 系统设计

1. 硬件组成

扫描硬件所占的空间必须与现在使用的台秤和条形码扫描仪所占的空间差不多，而且不需特别改造就能在各种商店环境下运行。图16-2是设计出来的扫描仪原理图。在光源和摄像头上都用到了偏振滤光片，为了滤去商品的镜面反射，摄像头滤光片的方向与照明滤光片的方向垂直。选用的数字信号处理芯片(DSP)，可以在一秒内完成图像处理运算。彩色摄像头和DSP是收款机的低速输入设备，系统只需要一套识别器和一台放置商品的台秤。

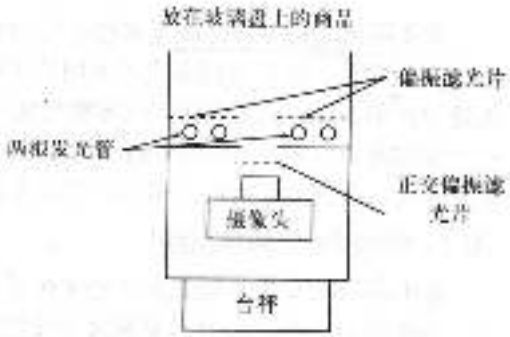


图16-2 滤光片相垂直的扫描仪设计方案。图中显示彩色摄像头和现有台秤及条形读出器中用的偏振光源

2.目标表示与识别

前面的应用实例表明彩色直方图是行之有效的特征，这一点得到了研究和开发结果的证实。经典纹理特征在该问题上的使用效果并不好，因此出现了一些面向问题的特征，如下所述。也用到了简单的形状特征。基于商品的图像，结合颜色、纹理、形状和大小直方图等特征，构成d>100维的特征向量Q，用来表示放在台秤上的未知商品。图16-3显示某些苹果（左边）和橘子（右边）的彩色直方图。

550

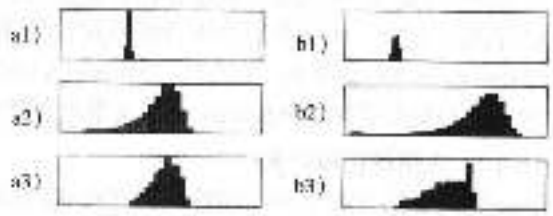


图16-3 苹果（左边）和橘子（右边）的彩色直方图。从上到下，分别是色调、饱和度和强度的直方图（R.Bolle和J.Connell提供）

为了使系统适应变化的情况，采用最近邻分类方法。商品的特征向量标记样本存储在一个数组中，最多m=350个类别，每个类别有10个样本，这样就存储了3500个样本。采用DSP，可以很容易地在一秒之内对查询特征向量Q和全部3500个标记样本做比较，这样就能找到k个最近邻。并没有用特别的数据结构去组织这些训练样本，这样做便于进行更新。存储样本向量时同时存储关联信息，这样可以记录向量的历史，而且当样本过时时可以从内存中删除该向量。

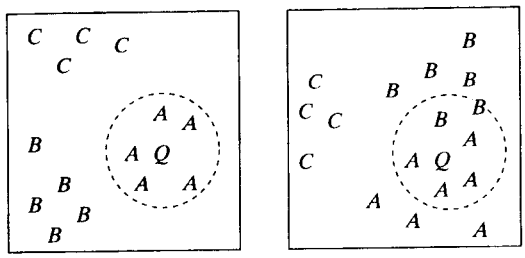
查询特征向量Q与L类中的第j个训练样本之间的距离$^{L}P_{j}$，采用第8章的计算方法得到。由于每个特征都是根据直方图算出的，距离$d(Q,\ ^{L}P_{j})$就是Q与$^{L}P_{j}$之差的绝对值。

$$d^{L}=d(Q,\ ^{L}P_{j})=\sum_{f=1}^{f}w_{f}\,d(Q,\ ^{L}P_{j,f}) \tag{16-1}$$

通过阈值来控制识别过程，并决定结果的确定性。用距离阈值t确定Q是否与样本P_{j}有足够的接近程度。设k_{t}表示从内存选出的与Q近邻的样本个数。识别过程在下一节做介绍。

16.1.3 识别过程

识别台秤上商品的总算法，请参考算法16.1。在第16.1.4节中，对一些步骤进行了更加详细的描述。利用训练样本中的最近邻进行目标识别，原理图参见图16-4。



a）确定　　b）不确定

图16-4　在特征空间内进行决策的原理图。显示的是二维特征空间，而实际特征空间是d维的

a）当Q中t内的所有训练样本来自同一类时的“确定”识别结果

b）不确定时的策略：要么重新扫描商品，要么系统要求操作员从接近的A类和B类中进行选择

算法16.1　Veggie Vision识别商品流程

1. 操作者进行控制，拍摄光源关闭和光源打开时的图像，并从背景中抽取出前景商品。
2. 绘制颜色特征、纹理特征、形状特征和大小特征的直方图；把他们结合到一起构成特征向量Q。
3. 将Q与内存中的每个训练样本${}^{L}P_{j}$做比较；舍弃偏差大于t的样本；按升序排列剩余的样本。
4. 如果求得的K个最近邻都具有相同的标记L，那么标记L将作为Q标示的等价量被返回，而且识别结果是确定的。在这情况下，系统能自动做出决定。
5. 如果第一次的判断不确定，请求操作者再放一次商品，然后重复第1~4步。
6. 如果第二次的判断仍不确定，就按排列顺序显示N个可选标记，由操作者来确定。
7. 如果适当的话，把Q加入到训练样本集中，可能要删掉其他训练样本。

16.1.4　详细分析

1.获取商品图像

在商店环境下，要求在几乎不需要控制的情况下获取图像。特别地，台秤内的摄像头将感测或反射来自上面的光线。获得台秤上商品的两幅图像，拍第一幅图像时台秤内的光源打开，拍第二幅图像时该光源关闭。要分割的三个区域是：（1）商品区域，光源关闭时对应暗区域，光源打开时对应亮区域。（2）在两幅图像中背景区域具有类似的亮度。（3）如果商品装在塑料袋中，光源关闭时对应区域不太暗，光源打开时对应区域不太亮。对于该项工程，设定阈值使商品区域能够从塑料袋和背景区域中分割出来。为得到高质量的颜色信息，利用台秤内的偏振光来抑制镜面反射，因为它并不表示商品的表面。另外，对光照条件进行控制，使要成像的商品表面不受台秤外面光照的影响。这样即使房间的照明发生变化，感测到的颜色将是一致的。

2. 计算特征

只针对商品所占的区域计算特征。特征必须具有旋转不变性，但不具有尺度不变性。绘制四种类型特征的直方图并进行整合，得到一个向量Q，用来表示未知的商品。这四种特征是彩色、纹理、形状和尺寸。

把每个像素的颜色值从RGB空间转换到HSI空间，坐标为(h, s, i)。绘制h、s和i的直方图。

不考虑亮度或饱和度比较低的像素，因为它们在转换时会引起数值不稳定。用商品区域的总面积对这三个直方图进行规范化处理。图16-3显示出苹果和橘子的颜色直方图。

只对原始彩色图像每个像素的绿色通道计算**纹理**特征。利用大小不等的center-surround模板进行纹理特征计算。center-surround模板，其中心的盒形区域具有正权值，周围的背景区域具有负权值。利用子采样图像可以加速计算。把对模板的正负响应绘成直方图。中心峰值的大小给出了目标的总纹理信息。如果中心峰值较大，说明对模板的许多响应幅度较低或者说纹理非常细小。直方图的伸展范围说明纹理的对比程度，例如叶子的阴影与其细微表面条纹的对比。直方图的不对称性说明纹理成分大小相对模板尺度而言的一些信息，例如直方图向正向偏移表示商品具有较大的叶子，比起那些更卷缩的叶子如欧芹叶子，该种商品的叶子之间的缝隙较小。

551
~
552

测量**形状**的方法比较简单。对商品区域的边界进行平滑和跟踪，计算每个边界像素处的曲率。只利用区域外部的边界线段，图像边框和图像中商品之间相接触的线段不用。为了更好地聚类，画出曲率平方的直方图。球形商品将产生与半径对应的较窄的峰值。实际位置能区分柚子和柠檬之间的区别。狭长商品，如香蕉和胡萝卜，产生的数值范围较宽，在零值附近有一个尖峰。叶子多的蔬菜，曲率的分布较宽。

第四个直方图特征是**尺寸**。对每个前景像素计算尺寸值，而不仅仅是对像素进行计数。在二值前景模板的四个方向（水平、垂直和两对角线）计算游程长度，建立四幅有向图像。在每幅有向图像中，像素的有向尺寸就是它所在游程的总长度。前景像素的尺寸取自该像素处的最小有向值。目标尺寸就确定了，不需要参数模型和其他任何分割，只要把前景和背景分开就行了。一串葡萄分割成“膨胀的云”前景模板。外面碰伤的像素具有较小的游程长度，而内部像素具有较大的游程长度。于是尺寸直方图将由两个峰值，一个表示单个的葡萄，另一个表示葡萄串的总尺寸。在尺寸直方图上胡萝卜在特征宽度处将有一个窄的峰值，这正好与樱桃西红柿的宽度类似，形状直方图上零曲率附近的峰值表示这是长条状的，而西红柿没有。

3.监督学习

最近邻分类方法计算时间不长，训练简单，适应性较强。开始时，可用部分库存商品对系统进行训练，并设计类标识（存货代码）。系统投入使用后，操作者可以提出要求，把一个新的特征向量Q加入到训练样本的数据集中。一个新样本与该类样本的几何结构或者所用要素做比较，如果证明它是冗余的，则可以删除该样本。训练时，基于已有样本进行正确分类的新样本，如果它在最佳匹配距离t_2之内，就不保存；否则，就保存。这样允许在特征空间内构成多个模式。例如，一种模式只用于识别椰菜头，而另一种模式识别带长茎的椰菜。类别的样本个数最多为M，如果超过M就去掉使用率最小的样本。如果一个样本是最接近的，则计数加1即$I+$，否则减1即$I-$。

553

对每个商店都从头训练Veggie Vision系统是没有必要的。实验表明，如果使用另一个商店的样品进行训练，识别性能将会降低。然而，如上所述，系统具有自适应性。一开始系统的训练是基于上一个商店的商品，这时人员干预的频率较高，但人员的总工作量比从头开始训练的方案要小得多。

16.1.5 性能分析

一段时间以来，研究人员已经公布了一些实验结果，在后面的参考文献中可以看到具体内容。最近的一项研究中，采用了5300幅图像，涉及4个不同的商店。系统确认并正确的概率

是89%；识别正确或者将正确类别作为首选项提供给操作员的概率是93%；识别正确或者提供前4个正确选项的概率是96%。可能要求操作员重放一次商品。如果Veggie Vision系统在二次尝试中都是不确定的，就由收款员触摸显示图标进行确认。可以看出，即使每一次都由操作员通过触摸式CRT做出决断，该系统也会极大地减少工作人员的劳动量。

习题16.1

假设香蕉是矩形的，他们的形状直方图看起来将是怎样的？

习题16.2

画出红苹果和黄香蕉的颜色、纹理和形状直方图，并进行比较。

习题16.3

如果顾客把3个苹果和2个橘子放到一个塑料袋中，识别系统还能够应付吗？如果能，怎样实现？

16.2 基于虹膜的身份识别

现在介绍通过扫描人眼的虹膜纹理进行身份识别的系统。ATM环境下的传感器硬件是由Sensar制造的，能够运行IriScan的特征抽取与匹配软件。我们特别感谢Sensar的Gary Zhang（1998）和剑桥大学的John Daugman（1994,1998），他们提供了该系统的有关信息和图表。

身份识别一直以来都是一个重要的社会问题。对于商业和法律事务，需要进行正确的身份识别。例如，一个人从银行账户中取出现金或变更居住地址。进行身份确认时，这个人要向有控制权的另一个人出示证件，例如身份证或者出生证。当今世界的许多事务是通过机器 554
或计算机网络进行的，常常要用帐号和口令，或者帐号和个人识别号（PIN）来保证安全性和私密性。不管是否允许，其他人能够得到这些代码，然后就可以在不负责任或者不受控制的情况下进行交易。

身份识别除了在电子商务方面有着非常重要的应用，在警务工作方面也有很重要的应用价值。指纹已经得到普遍使用。对犯罪现场进行指纹检查，也许能识别出到过现场的有关人员。指纹也用于协作场合下的身份识别，例如用来识别安全环境中的工人。研究和应用指纹的历史超过一百年。人们已经研发出一些电子装置，使合作者的指纹能够很容易地输入到计算机网路或其他系统当中（参见Jain等人，（1999）的文章）。对于身份识别和验证系统，人脸识别技术也在逐渐发展之中。这些系统具有不依靠知识进行身份识别的能力，比如在飞机场、银行或者旅馆场所进行识别。他们在警务和安全场合尤其有用，但也在可接受性和保护隐私方面存在一些问题。

16.2.1 对识别系统的要求

考虑系统执行下面的两种操作之一：（a）从一大堆人中识别出一个人，不管他们是否合作，（b）确认一名合作者，验明他的身份，后一种情况常常称作验证，系统应满足的要求有的不是显而易见的，因此我们把它们列出来。系统设计受特殊生物特征的限制，也受测量方式和机器代码方式的限制。三个重要的生物特征是个人的外观特征：（1）指纹，（2）人脸，（3）人眼虹膜。下面将会讨论，虹膜能够比指纹或人脸提供出更好的信息。

1. 系统必须是在对个人影响最小的情况下获取信息的。
2. 一段时间前后，同一人的生物特征码前后差异必须很小。

3. 个人的生物特征必须与他人的生物特征有明显区别（人群对象随情况而变化）。

4. 系统对“虚假数据”（例如打印在纸上的图像）有较强的免疫力。

5. 对于特殊应用，系统的性价比要高。

在继续讨论虹膜扫描系统之前，需要对不同的生物特征在上述几个方面进行比较。除了以上提到的生物特征，我们也对DNA进行分析。

555 **1. 获得信息方便与否。**指纹（一般），人脸（好），虹膜（好），DNA（差）。对于指纹，有廉价的数字扫描仪，但需要用户提供指纹；人脸可以用廉价的视频摄像头方便地摄取图像；要获得高质量的虹膜图像需要更贵的光学设备和更多的控制操作；当然DNA的获得是一个昂贵的离线实验过程，通常在重要的法律案件中才使用。

2. 小类内差异。指纹（好），人脸（一般），虹膜（很好），DNA（很好）。值得注意的是，提取指纹时会产生很大的形变，人脸外观会随姿势、心情、头发和年龄而变化。虹膜纹理在孩子出生之前就已经形成，而且一生当中变化很小，已经开发出的扫描系统能够对虹膜进行一致性编码。

3. 大类间差异。指纹（好），人脸（好），虹膜（很好），DNA（很好）。虽然指纹在专家控制下能够很好的进行区分，但自动执行的效果就没有那么好。如果只是根据人脸进行识别，多数人都能找到与自己长相类似的第二个人，尤其是双胞胎。双胞胎自己还有1%的出错率！另外，双胞胎有相同的DNA，有趣的是，双胞胎没有相同的虹膜纹理。事实上，来自同一个体的双眼纹理就像来自不同个体的眼睛纹理一样，是不相关的。

556 **4. 防止假冒方面。**指纹（好），人脸（好），虹膜（很好），DNA（很好）。使用指纹或者人脸的一些系统，可能会被照片或简单外表模型所欺骗。眼睛虹膜内的瞳孔，其大小变化可通过监测系统进行跟踪，从而识破假冒者的诡计。

5. 性价比。指纹（一般），人脸（一般），虹膜（一般），DNA（差）。检测指纹和人脸的系统比较便宜，但对特征进行匹配的算法比较复杂。虹膜扫描系统比较昂贵，而匹配算法简单。DNA识别，在时间、人力和原材料方面都是非常昂贵的。

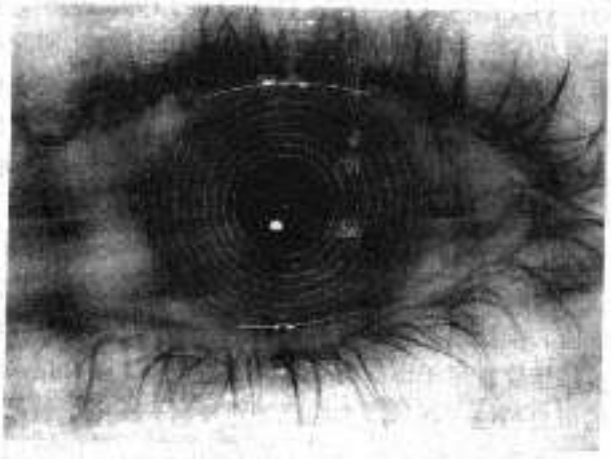


图16-5 人眼的窄视场图像。图像处理识别出虹膜的8条圆环，据此抽取出虹膜纹理特征（剑桥大学John Daugman提供，请参考www.cl.cam.ac.uk/~jgd1000/）

16.2.2 系统设计

下面具体讨论虹膜扫描技术在ATM机客户识别系统中的应用情况。当客户靠近ATM机，系统就扫描客户某只眼睛的虹膜，Sensar…Secure™系统根据客户记录识别该用户的身份。然后用户获得帐号，作为附加的安全防护措施，也许还要输入密码。虹膜扫描技术也可用于其他方面，例如打开安全门，这时只需要对原系统的设计参数进行一些调整，关于参数设置如下所述。

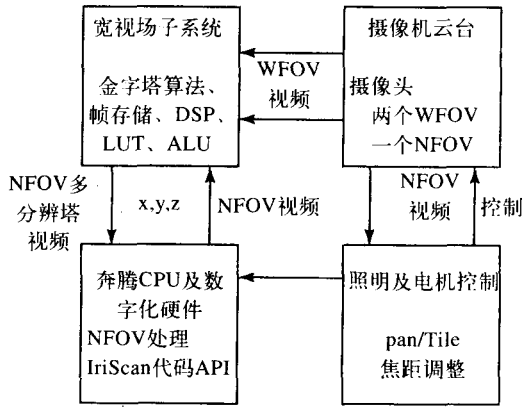
对于较大的三维视场，虹膜是一个相对很小的目标，要得到高分辨率的虹膜图像，需要精密扫描仪器和特殊光学器件。对于人们排着长队等待机器扫描的情况，为了确定排在最前

面的那个人，需要进行3D立体分析。一旦对该人的眼睛定位扫描完毕，就利用专用软件得到d = 2048维的二值向量Q，该向量表示虹膜的灰度纹理。把该向量与表示某群体客户的一组向量进行匹配，通过计算最小海明距离决定匹配结果。海明距离就是两个二值向量中有差别的位数。

1.硬件组成

Sersar...Secure™分布式处理结构参见图16-6。系统主要由4个单元组成：（1）通用计算机，提供用户界面以及测量控制和视频处理单元的接口，（2）摄像机云台，上面安装三个摄像头，捕捉宽视场图像和近视场图像，（3）云台控制单元，（4）视频处理单元，有专门硬件进行立体视频的实时处理。 557

根据两个宽视场摄像头捕捉的视频流，确定视场中最前面个体的位置。两视频流传送到信号处理单元，用多分辨金字塔进行实际立体视觉处理。某只眼的x-y-z位置被传送到主单元，然后利用这个信息去控制摄像机云台，从而得到人眼的近视场图像。这个过程的周期为半秒钟，这样可以跟踪以较低速度移动的人眼。主单元对近视场视频进行处理，从而确定人眼区域并抽取虹膜代码。总过程参见算法16.2。



WFOV表示宽视场
NFOV表示近视场

图16-6 Sersar…Secure™分布式处理结构

该系统的感测硬件比其他系统中用到的硬件要复杂得多，这一点限制了该系统的实际应用范围。造价高的原因主要是被动感知引起的，因为客户更愿意接受被动感知。客户在工作区内可以自由移动，因此系统必须能够确定客户的位置。这种情况就需要进行宽视场感测，以便找到要跟踪的目标，通过近视场感测得到所需的眼部图像。实时立体视觉需要使用特殊的硬件，采用两级分辨来加速在两视频流中寻找对应点的运算。

算法16.2 根据虹膜图像识别最前面的人员身份ID

1. 用宽视场视频和基于相关的立体视觉算法，确定最前面的人头位置。
2. 用模板法确定人脸特征的位置，然后确定左（右眼）位置$[x, y, z]$。
3. 根据$[x, y, z]$，利用近视场单色摄像头对准人眼中心，捕捉人眼的清晰图像I。
4. 采用专门的图像处理软件，从人眼图像I得到2048位虹膜代码Q。
5. 采用异或（XOR）运算将虹膜代码Q与数据库中的代码相匹配。如果两代码的差异小于K位，则返回该人的ID；否则返回“reject”。

2. 表示

人眼与身份的最终表示只是一个2048维的二值向量。图16-7用图示的方式表示一个向量，其中黑色表示0，白色表示1。将Gabor滤波器与虹膜图像进行邻域相关计算，结果的正负号确定了代码的每一位。在相关计算之前，必须对人眼图像进行旋转规范化处理。

如图16-8所示，通过二维Gabor小波与虹膜上(ρ_0, ϕ_0)处虹膜图像的相关运算，确定虹膜代码的每一位，所用小波的散差参数为α和β（解调）。小波沿ρ方向的截面是散差参数为α的 558

高斯函数，而沿ϕ方向的截面是经正弦波调制的散差参数为β的高斯函数。小波与图像函数的每次相关结果都产生如下的复数c：

$$c = \int_{\rho}\int_{\phi} f(\rho,\phi)[e^{-i2\pi(\phi-\phi_0)}e^{-(\phi-\phi_0)^2/\alpha^2}e^{-(\rho-\rho_0)^2/\beta^2}]\rho d\rho d\phi \tag{16-2}$$

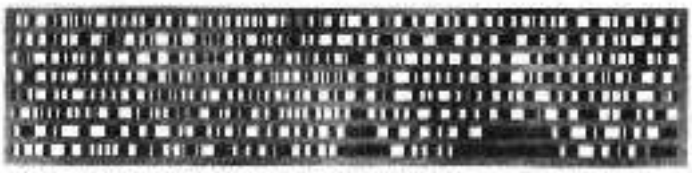


图16-7 2048位代码的图形表示，将不同大小的Gabor滤波器用于图16-5中的八条圆环内，所得结果的正负号用2048位代码表示（剑桥大学John Daugman提供）

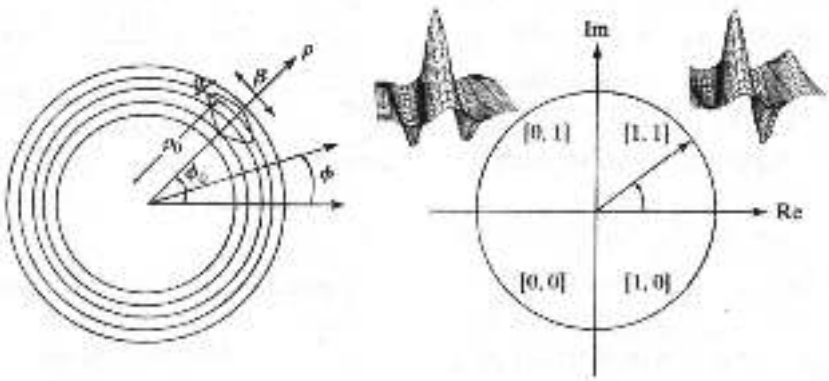


图16-8 （剑桥大学John Daugman提供）

（左图）半径$\rho_1 \leq \rho \leq \rho_2$的虹膜环形区域，Gabor小波位于$(\phi_0, \rho_0)$，小波的散差参数是$\alpha$和$\beta$

（右图）复数值二维Gabor小波的形状

根据正负号可以把复数值相关结果转化成虹膜代码的两位。如果（$Re(c) \geqslant 0.0$），那么$b_{real} = 1$；否则$b_{real} = 0$。如果（$Im(c) \geqslant 0.0$）那么$b_{imag} = 1$；否则$b_{imag} = 0$。显然，虹膜图像绕视线的任何旋转都会影响位置参数ϕ_0的位置。由于要根据宽视场图像中的双眼位置信息得到近视场图像，旋转幅度将不会很大。在匹配期间，进行轻微旋转之后，再匹配虹膜代码，最后得到旋转后代码与数据库候选代码的最佳匹配结果。ρ轴根据瞳孔边界和虹膜外边界进行确定。假设这两个
559 边界是圆形，但不一定是同心的。根据边界的综合信息找到这两个圆，与圆形霍夫变换的运算方式一样。通过两组参数ρ、x_c和y_c确定边界，这两组参数使沿圆周的梯度幅度最大。

$$\max_{(\rho,x_c,y_c)}\left|\frac{\partial}{\partial\rho}\oint\frac{f(x,y)}{2\pi\rho}ds\right| \tag{16-3}$$

3. 识别过程

识别过程参见算法16.2。前面讨论的内容，涉及识别过程的每一个重要环节。下一小节讨论系统的性能问题。

16.2.3 系统性能

关于虹膜图像捕捉与识别所用的时间，与使用情况有关，通常在1到5秒之间。对于ATM系统，这个时间是合适的，但在机场安检系统中要识别走动的人员，这个时间就显得太慢了。

生ATM应用中，主要是控制摄像头机械运动比较费时间。大约有90%的时间花费在图像捕捉上。关于图像运算，大概需要200*msec*来确定虹膜边界的位置和生成虹膜代码。匹配速度大约是每秒10万人。

最重要的指标是系统在识别中出错的概率。根据多次实验结果建立的理论模型，Daugman（1998）做出如下关于误差率的估计结果。如果要验明某个人的身份，2048位代码的70%就必须得到匹配，这时的误识率大约是$1/(6\times10^5)$，而拒真率是1/46 000，如果阈值降低到66%，那么误识率和拒真率相同，大约是一百万分之一。

我们对上面估计概率所用的模型做个简单的总结。读者如果想了解详细内容，请参考Daugman（1998）的论文。对300个体的虹膜图像进行两两比较，产生如图16-9所示的结果。结果发现（a）不同个体的虹膜，其海明距离（超过20万对）的分布范围在0.4~0.6位之间；（b）观测到的分布结果，与N = 266自由度，$p = 0.5 = q$的二项分布情况吻合得非常好；（c）令人惊奇的是，同一个体的两眼虹膜分布，与不同个体间的分布结果类似，这表明同一个体的双眼虹膜是不相关的，就像两个不同个体的虹膜一样。图16-9绘出不同个体代码间的海明距离分布（右边），以及同一个体代码间的海明距离分布（左边），交叉点处的概率是10^{-6}。决策阈值不一定要设在交叉点处，如果最大能容忍30%代码位的距离，那么误识率为60亿分之一，如果错误接受的代价比错误拒绝的代价大得多，那么这个误识率是满足要求的。 560

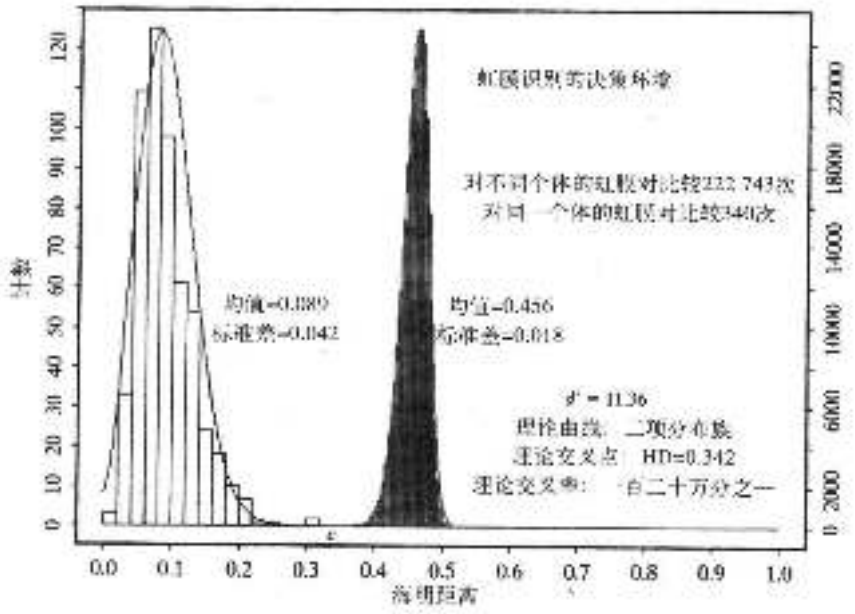


图16-9 同一个体的虹膜海明距离分布（左边），不同个体的虹膜海明距离分布（右边）。交叉点位于代码位的0.34处，其中拒真率等于误识率，两个大约都是10^{-6}（剑桥大学John Daugman提供）

16.3 参考文献

1. Bolle, R., J. Connell, N. Haas, R. Mohan, and G. Taubin. 1996. VeggieVision: a produce recognition system. *Proc. IEEE Workshop on Applications of Comput. Vision.*

2. Camus, T., U. M. Cahn von Seelen, G. G. Zhang, P. L. Venetianer, and M. Salganicoff. 1988. Sensar...Secure™ Iris Identification System, in *Proc. IEEE Workshop on Applications of Comput. Vision* (9–21 Oct. 1998), Princeton, NJ, 254–255.
3. Daugman, J. 1994. Biometric Personal Identification System Based on Iris Analysis. U.S. Patent No. 5,291,560 issued to John Daugman (March 1, 1994).
4. Daugman, J. 1998. Recognizing persons by their iris patterns. In *Biometrics: Personal Identification for a Networked Society,* A. Jain, R. Bolle, and S. Pankanti, eds. Kluwer Academic, Dordrecht, Netherlands and Norwell, MA.
5. DellaVecchia, M., T. Chmielewski, T. Camus, M. Salganicoff, and M. Negin. 1998. Methodology and apparatus for using the human iris as a robust biometric. *Ophthalmic Technologies VIII: SPIE Proc.* (Jan. 1998), 24–30.

561

索　引

索引中的页码为英文原书页码，与书中边栏页码一致。

A

accidental alignments（偶然对齐），394
accumulator array, Hough transform（累加数组，霍夫变换），304~309
active contours models（活动轮廓模型），489~492
addition,image（相加，图像），12
additive color systems（加色系统），192
affine calibration matrix（仿射标定矩阵），431~437
affine mapping, 2D object recognition via（仿射映射，二维目标识别），341~350
affine mapping functions（仿射映射函数），329~339
  rotation（旋转），330~331
  scaling（缩放），329~330
  translation（平移），332
affine method（仿射方法），436~437
affine transformations（仿射变换），338~341
  3D（三维），413~421
affine warp（仿射变形），334~335
aggregating: consistent neighboring edges into curves（相邻连贯的边缘生成曲线），301~303
  motion trajectories（运动轨迹），321~324
albedo, definition（反照率，定义），204
algorithms, alignment（算法，比对），497
  back-propagation（反向传播），126
  border（边界查找），295~297
  boundary matching（边界匹配），237~238
  classical clustering（经典聚类），281~282
  classical connected components,using union-find（经典连通成分标记，利用并查），61~62, 65
  classification（分类），101,103~104
  computing 3D surface coordinates using calibrated camera and projector（用标定好的摄像机和投影仪计算3D表面坐标），438
  computing output image from input image（从输入图像计算输出图像），139~141
  conventions for defining（常规要求），555~556
  converting RGB（red-blue-green） encoding to HSI（hue-saturation-intensity）encoding（RGB（红-蓝-绿）编码到HIS（色调-饱和度-强度）编码的变换），196
  cylindrical warp of image region（图像区域的圆柱变形），364~366
  decision procedure（决策过程），99
  deriving motion vectors for interesting points（计算兴趣点的运动向量），260
  detecting interesting image points（检测感兴趣的图像点），258
  discrete relaxation labeling（离散松弛标记），357
  finding straight line segments（直线段检测），305~306,308
  flow of produce identification in Veggie Vision（Veggie Vision识别商品流程），552
  geometric hashing offline preprocessing（几何散列离线预处理），349
  geometric hashing online recognition（几何散列在线识别），350
  graph-matching（图匹配），106
  Greedy Exchange（贪婪交换），267,268,270
  histogram equalization（直方图均衡化），133
  holecounting（孔计数），6
  Hough transform for finding circles（霍夫变换检测圆），310
  Hough transform for finding straight lines（霍夫变换检测直线），306
  identification by iris-scanning system（利用虹膜扫描系统识别身份），558
  image histogram（图像直方图），84
  interpretation tree search（解释树搜索），354
  isodata clustering（isodata聚类），284
  iterative K-means clustering（迭代K-均值聚类），282
  iterative P3P solution（P3P迭代求解法），443
  labeling block edges via backtracking（回溯法标记模块边缘），381
  labeling block edges via discrete relaxation（离散松弛法标记模块边缘），382
  labeling edges of blocks via backtracking（回溯法标记模块边缘），381
  local-feature-focus（局部特征聚焦法），343
  O'Gorman and Clowes method for extracting straight lines（抽取直线的O'Gorman/Clowes方法），

305,306,308
perceptron learning（感知器学习），122~123
photometric stereo with three light sources（利用三光源的光度立体），473
RAG（region adjacency graphs）（区域邻接图），82
recognition-by-appearance using basis of principal components（基于主成分基的表象识别），520
recursive labeling（递归标记），57~59
relational distance matching（相关距离匹配）504
rigid transformation for aligning model triangle with congruent world triangle with（将模型三角形与相合实际三角形对齐的刚体变换），421
row-by-row labeling（逐行标记），59
Shi's clustering procedure（Shi的聚类过程），289
single update stage for active contour（活动轮廓的一个更新步骤），491
space-carving（空间切割），464~467
tracking edges of binary edge image（二值边缘图像的边缘跟踪），303
transformation from model features to image features using pose clustering（通过位姿聚类寻找从模型特征到图像特征的变换），344
union-find（并查），59
using color and motion to track ASL gestures（利用颜色和运动跟踪ASL手势），322~324
watch-gear inspection（手表齿轮检查），68~71
alignment(s): accidental,（对齐：偶然），394
matching geometric models via（几何模型匹配），496~504
smooth object（光滑目标），501~504
3D-3D（三维到三维），496~498
via transformation calculus（利用变换演算），419~421
2D-3D（二维到三维），498~501参见matching
ambient light（环境光），207~208
definition（定义），208
analog images,definition（模拟图像，定义），29
angular field of view（角视场），31
ANNs(artificial neural networks)（人工神经网络），120~126
Mach band effect produced by（产生的马赫带效应），153~155
appearance,3D object recognition by,（外观，识别3D物体），516~523
application problems（应用问题），3~10
applications,of binary morphology(应用，二值形态学的），68~71
AR(augmented reality)（增强现实），530~532
architectural walkthrough（建筑漫游），529
arcs,detecting with Hough transform(弧，霍夫变换检测），303~312
area（面积），73
arrays:CCD（阵列：CCD）24~25, 26
PARENT（父节点向量），59~61
pixel（像素），43~45
arrow junctions（箭头连接），377
artificial neural networks(ANNs)（人工神经网络），120~126
Mach band effect produced by（产生的马赫带效应），153~155
artificial neurons（AN）（人工神经元），120~122
aspect graphs（表象图），488~489
aspect（表象），488
assignment,definition（分配，定义），351
auditory output, virtual reality（VR）systems（听觉输出，虚拟现实系统），539
augmented reality（AR）（增强现实），530~532
definition（定义），535，参见virtual reality（VR）systems
autocorrelation, measuring texture by power spectrum and（自相关，用功率谱测量纹理），221~223
automatic thresholding（自动阈值化），85~89
axis（axes）:best（轴：最佳轴），79~81
ellipse,lengths and orientations（椭圆，长度和方向）78~79
with least second moment（具有最小二阶矩），81

B

B^+-tree indexes（B^+-树索引），245~247
background pixels（背景像素），51
backprojection（反投影），200
back-propagation algorithm（后向传播算法），126
backtracking, labeling block edges via（回溯法，标记模块边缘），381
balloon models,3D（气球模型，3D），493~494
bandpass filtering（带通滤波），181
basis,orthogonal,using（基，正交，利用），160~162
basis images:computing（基图像：计算），519~521
for set of training images（训练图像集），518~519
bay_above_bay（湾在湾之上），112
bay_above_lake（湾在湖之上），112
bay_num（湾数），112
Bayesian classifier,definition（贝叶斯分类器，定义），115
Bayesian decision-making（贝叶斯决策），114~115
bays（湾），104,112
best affine calibration matrix（最佳仿射标定矩阵），

431~437
best axis（最佳轴），79~81
binary decision trees,definition（二叉决策树，定义），108
binary image（s）（二值图像），24
  analyzing（分析），51~91
  closing（闭合），65, 67
  definition（定义），30
  dilation（膨胀），65, 66~67
  erosion（腐蚀），65, 66~67
  labeling connected components（标记连通成分），56~63
  morphology（形态学），63~73
  opening（开启），65, 67
  run-coded（游程编码），37
  translation（平移），66
binary morphology（二值形态学），63~73
  applications（应用），68~71
  basic operations（基本运算），65~68
  conditional dilation（条件膨胀），71~73
  in medical imaging（医学成像），69
  structuring elements（结构元），63~65, 68~71
binary partition（二值分解），217
  local（局部），217
bins/binning（箱格/分箱），85,308,346
binsize（箱格大小），85
blade（刃边），372
  definition（定义），374
blobs（团），13;504~506
block（s）:labeling edges of via backtracking（模块：回溯法标记边缘），381
  labeling edges of via discrete relaxation（离散松弛法标记边缘），382
  labeling of line drawings of（线条图标记），377~383
blooming（光晕），28
blur, relating resolution to（模糊，分辨率与），406
Boolean features（布尔特征），112
border algorithm（边界查找算法），295~297
boresighted multispectral sensors（视轴多谱传感器），46
boundary（ies）:coding（边界：编码），292~293
  cues from（线索），393~394
  extraction（抽取），295
  illumination（光照），374
  interpreting shape from（恢复形状），391~392
  matching（匹配），237~238
  in space-time（时空），321
bounding box（边界框）76
box filter,definition（盒形滤波器，定义），136
box smoothing masks（盒形平滑模板），144
Burns line finder（Burns直线检测器），311~312

C

calibration: best affine calibration matrix（标定：最佳标定矩阵），431~437
  of cameras（摄像机）431~437
  of cameras improved method of（改进的摄像机方法），444~453
  of cameras example（摄像机举例），449~453
  of projectors（投影仪），437
camera coordinate frame C（摄像机坐标系C），44
camera effects:definitions（摄影特效：定义），272~273
  ignoring（忽略），274~276
camera model（摄像机模型），422~430
  parameters（参数），436~437
camera pan, definition（摄像机扫视，定义），272
camera zoom, definition（摄像机变焦，定义），272
camera（s）:calibration（摄像机：标定），431~437
  calibration example（标定举例），449~453
  calibration, improved（标定，改进），444~453
  CCD（charge-coupled device）（电荷耦合器件），22~24
  computing 3D points using multiple（利用多个摄像机计算3D点），428~430
  data acquisition using（获取数据），461~463
  extrinsic parameters（外部参数），445~449
  human eye as（人眼），26~27
  image formation in（图像形成），24~26
  intrinsic parameters（内部参数），445
  posing for stereo configuration（构成体视系统的~位姿），411~413
  video（视频），26
Canny edge detector（Canny边缘检测算子），157~158
  and linker（连接算子），297~301
case studies: identifying humans via iris of eye（案例研究：基于虹膜的身份识别），554~561
  Veggie Vision, 548~554
category hierarchy,GRUFF（类别层次，GRUFF），515~516
Cauchy-Schwartz Inequality（柯西-施瓦茨不等式），161,162
CCDs (charge-coupled devices) :arrays（电荷耦合器件：阵列），24~25, 26
  cameras（摄像机），22~24, 24~26
  variations（差异），28
centroid（中心），73
chain code, Freeman（链码，Freeman），293

changes, detecting in video (变化，检测视频中)，272~277
character recognition (字符识别)，98~100
charge-coupled devices (CCDs) : arrays (电荷耦合器件:阵列)，24~25, 26
cameras (摄像机)，22~24, 24~26
variations (差异)，28
centroid (中心)，73
child nodes (子节点)，107
chromatic distortion (彩色畸变)，29
chrominance (色度)，197
circle of confusion (模糊圈)，24
circles,finding with Hough transform (圆，用霍夫变换检测)，309~310
circularity (圆度)，74
class mean,nearest,used in classification (类均值，最近的，用在分类中)，101~103
classes,definition (类别，定义)，94
classical connected components algorithm, using union-find (经典连通成分算法，利用并查)，61~62,65
classification:algorithm (分类: 算法)，101, 103~104
color used for (基于颜色)，198~199
common model for (一般模型)，94~97
definition (定义)，94
fuzzy (模糊)，124
nearest class mean used in (最近类别均值)，101~103
nearest neighbors used in (最近邻)，103~104.
参见decision trees
classification system(s):building (分类系统:建立)，95~96
evaluating error rate of (错误率评估)，96
false alarms and false dismissals (误报和漏报)，96~97
classifier (s) (分类器)，95
definition (定义)，94
implementing (实现)，101~104
clearance primitive (空旷性基元)，514
clipping (削波)，28
closed form solutions for parameters (参数的封闭解)，314~315
closing of binary images (二值图像的闭运算)，65
definition (定义)，67
clustering (聚类) 119
classical,algorithms (经典的，算法)，282~282
isodata，282~284
iterative K-means (迭代K-均值) 282
methods (方法)，281
methods based on histograms (直方图方法)，284~286
Ohlander's recursive histogram-based technique (Ohlander 递归直方图技术)，285~286, 287
pose (位姿)，344~346
Shi's graph-partitioning technique (Shi的图分割技术)，286~289
CMY (cyan-magenta-yellow) subtractive color system (CMY (青-品红-黄) 减色系统)，193~194
code, Freeman chain (码，Freeman链码)，293
coding, boundary (编码，边界)，292~293
collision (冲突)，245
color (颜色)，187~211
applications (应用)，209
CMY (cyan-magenta-yellow) subtractive color system (CMY (青色-品红-黄色) 减色系统)，193~194
cube (立方体)，194
hexacone (六棱锥)，194, 195
histograms (直方图)，199~201, 231, 233
HSI (hue-saturation-intensity) HSI (色调-饱和度-亮度) 194~197
human perception (人类感知)，209~210
images (图像)，45~46
layout (分布)，232~233
physics of (物理学)，188~191
pseudo (伪彩色)，210
RGB (red-green-blue) basis for (RGB (红-绿-蓝) 基)，191~193
segmentation (分割)，201~202, 322~324
similarity measures (相似性度量)，231~244
triangle (三角形)，193,195
used in Veggie Vision (用于Veggi Vision系统)，552
using for classification (分类)，198~199
compression:data (压缩: 数据)，36
with JPEG (Joint Photographic Experts Group) format (JPEG (联合摄影专家组) 格式)，38~39
lossless (无损)，36
lossy (有损) 36
MPEG,for video (MPEG，视频)，261~262
With Motion JPEG (Joint Photographic Experts Group) format (运动JPEG格式)，40
computer vision,definition (计算机视觉，定义)，1
conditional dilation:in binary morphology (条件膨胀: 在二值形态学中)，71~73
definition, (定义)，72
conditioning images (处理图像)，128~186
confusion matrix (混淆矩阵)，106~107
definition (定义)，106
connected components:algorithm,classical,using union-find (连通成分: 算法, 经典的, 利用并查)，61~62,65

labeling（标记），56~63
labeling,using run-lenth encoding for（标记，利用游程编码），62~63
consistent labeling（一致性标记），351~353
definition（定义），351
constrained linear optimization（约束线性优化），456~457
constraints: epipolar（约束：外极），402~403
hard（硬），490
integrating spatial（综合空间），472
ordering（顺序），403
relational,symbolic matching and（关系，图符匹配），401
3D object recognition（3D目标识别），495~496
content-based image retrieval（基于内容的图像检索），226~250
indexing for with multiple distance measures（基于内容的多距离测度图像索引），248
continuous relaxation labeling（连续松弛标记），356~359
contours:active contour models（轮廓：活动轮廓模型），489~492
detecting with Hough transform for lines and arcs（用霍夫变换检测直线和圆弧），303~312
identifying regions by（区域识别），295~312
identifying with Canny edge detector and linker（用Canny边缘检测算子和连接算子识别），297~301
internal contour energy（内部轮廓能量），491
intrinsic images（本征图像），371~377
of moving objects（运动目标），321
contrast,detecting（对比度，检测），141~143
contrast stretching,definition（对比度扩展，定义），132
contributing points（贡献点），498
control points（控制点），332~334
definition（定义），333
converting: RGB（red-green-blue） encoding to HSI（hue-saturation-intensity） encoding,（变换：RGB（红-绿-蓝）编码到HSI（色调-饱和度-亮度）编码），196
RGB (red-green-blue) to YUV（RGB（红-绿-蓝）到YUV），197
convolution（卷积），128
cross correlation and（交叉相关），167~172
definition（定义），169
operation（运算），169~172
theorem（定理），182~183
co-occurrence matrices（共生矩阵），217~220
coordinate frames（坐标系），328~329, 413~415, 43~45
coordinate systems（坐标系），30~31,413~415
raster-oriented（光栅），30
coordinates, homogeneous（坐标，齐次），329
corner（s）（角点），377
detecting（检测），320~321
patterns（模式），4~6
correlation（相关），128
correspondence（s）:cross-correlation（对应：交叉相关），400~401
epipolar constraint（外极线约束），402~403
error versus coverage（误差与场景覆盖），403
establishing（建立），400~403
ordering constraint（顺序约束），403
pose from 2D-3D point（2D-3D点对应求位姿），455~456
symbolic matching and relational constraints（图符匹配和相关约束），401
in 3D-3D alignment（3D-3D比对），496~498
counting: holes（计数：孔），4~6
objects in an image（图像中的目标），54~56
coverage, error versus（场景覆盖，误差与），403
crease (s)（折痕），373,377
definition（定义），374
cross correlation（交叉相关），400~401
convolution and（卷积），167~172
definition（定义），169
normalized（规范化），170
crossbar inspection（检查交叉支撑杆），4~6
cubes: in octrees（立方体：在八叉树中），484~485
used in space-carving algorithm（用在空间切割算法中），466~467
cues: boundaries and virtual lines（线索：边界和虚拟直线），393~394
depth from focus（根据焦距变化求深度），393
motion phenomena（运动现象），393
from non-accidental alignments（非偶然对齐），394
shape from boundary（从边界恢复形状），391~392
shape from shading（从明暗恢复形状），388
shape from texture（从纹理恢复形状），388~391
3D in 2D images（2D图像中的3D），383~388
vanishing points（消隐点），392
curves: aggregating consistent neighboring edges into（曲线，相邻连贯的边缘生成），301~303
detecting with Hough transform（用霍夫变换检测），303~312
segmenting via fitting（基于拟合的曲线分段），317
cyan-magentag-yellow（CMY）subtractive color system（青-品红-黄（CMY）减色系统），193~194
cylinders: generalized-cylinder models（圆柱体：广义圆

柱模型），483~484
cylindrical warp,of image region（圆柱变形，图像区域），364~366

D

darkening with distance（随距离增大而变暗），206~207
data: acquisition in 3D object reconstruction（数据：在3D目标重建中获得），461~463
compression（压缩），36
gloves（手套），534
multidimensional, decisions using（多维，决策），117~119
range（深度），463~464,465
databases:image（数据库：图像），226~230
image,queries（图像，查询），228~230
organizing（组织），244~248
QBIC（Query by Image Content）（图像内容查询），226~227
Decathlete game（Decathlete游戏），255~256
decision tree（s）（决策树），98, 107~114, 522
automatic construction of（自动构造），109
binary, definition（二叉，定义），108
nodes（节点），107
decision-making:Bayesian（决策：贝叶斯），114~115
multidimensional data used for（多维数据），117~118
defect_cue（圆盘形结构元用于扩大瑕疵）,69,71
definition tree（s）,GRUFF（定义树，GRUFF），515
deformable models, physics-based and（可变形模型，物理学模型），489~495
density,and direction of edges in analyzing texture（在纹理分析中的边缘密度和方向），215~217
depth: cues（深度：线索）42
human perception of（人类感知），394
interpreting via focus（根据焦距变化求），393
3D cues in 2D images（2D图像中的3D线索），383~388
depth of field（景深），393
definition（定义），405
focus and（焦距），404~406
depth perception,stereo（深度感知，立体），397~403
derivative masks（微分模板），141~144
properties of（特性），143~144
detection: human edge（检测:人类视觉的边缘），153~155
LOG edge,Gaussian filtering and（LOG边缘，高斯滤波器），149~157
dextrous virtual work（虚拟灵巧手术），537~538
DFT（discrete Fourier transform）（离散傅里叶变换），179~181
difference operators for 2D images（2D图像的差分算子），144~149
differencing 1D signals（1D信号差分），141~144
differencing masks, detecting edges using（差分模板，用于检测边缘），141~149
diffuse: definition（漫反射：定义），204
reflection（反射），204~205
digital image（s）（数字图像），3
definition（定义），29
formats（格式），35~40
picture functions and（图像函数），29~35
problems with（问题），27~29
dilation: of binary images（膨胀，二值图像的），65
of binary images, definition（二值图像的，定义），66~67
conditional（条件），71-73
conditional, definition（条件，定义），72
dimensionality, high（维数，高），316
dimensions primitive（尺度基元），514
direction and density of edges in analyzing texture（纹理分析中边缘的方向和密度），215~217
discrete Fourier transform（DFT）（离散傅里叶变换），179~181
discrete relaxation:labeling（离散松弛：标记），354~356,357
labeling block edges via（标记模块边缘），382
discrimination, improving（辨别，改进），521~523
disparity, definition（视差，定义），398
dissolve, definition（溶变，定义），272~273
distance: darkening with（距离，变暗），206~207
image distance measures（图像距离测度），230~244
measures,multiple,indexing for content-based image retrieval with（测度，多，基于内容的多距离测度图像索引），248
pick-and-click（挑选-点击），234~235
relational,matching（相关，匹配），359~363
distortion:chromatic（畸变：彩色），29
geometric（几何），27
radial（径向），366~367
distribution: normal, definition（分布：正态，定义），116
parametric models for（参数模型），116~117
probability（概率），114~115
document retrieval（DR）（文档检索），97~98
DR（document retrieval）（文档检索），97~98
dynamic thresholding（动态阈值化），89

E

edge（s）（边缘），377

aggregating consistent neighboring edges into curves (相邻连贯的边缘生成曲线), 301~303
block,labeling via backtracking (模块, 回溯法标记), 381
block,labeling via discrete relaxation (模块, 离散松弛法标记), 382
density and direction in analyzing texture (纹理分析中边缘的密度和方向), 215~217
detecting using differencing masks (用差分模板检测), 141~149
detecting with LOG filter (用LOG滤波器检测), 151~153
human, detection of (人类, 检测), 153~155
jump (跳跃), 373
LOG,Gaussian filtering and detection of (LOG,高斯滤波器和检测), 149~157
surface-edge-vertex models (表面-边-顶点模型), 480~483
edge detector, Canny (边缘检测算子, Canny), 157~158
and linker (连接), 297~301
edgeness per unit area (每单位面积的边缘数), 216
eigenspace recognition by appearance (特征空间表象识别), 522
8-neighbors (8-邻域), 52
elastic matching (弹性匹配), 240
electromagnetic spectrum (电磁谱), 188
ellipse (椭圆), 484
axes,lengths and orientations (轴, 长度和方向), 78~79
empirical error rate,definition (经验错误率, 定义), 96
empirical interpretation of error (误差的经验解释), 315
empirical reject rate,definition (经验拒绝率, 定义), 96
encapsulated postscript (EPS) format (EPS格式), 39
enclosure primitive (包围性基元), 514
encoding: octrees (编码: 八叉树), 485~486
RGB (red-green-blue) ,conversion to HSI (hue-saturation-intensity) (RGB (红-绿-蓝), 到HSI (色调-饱和度-亮度) 的转换), 196
run-length,using for connected components labeling (游程, 用于连通成分标记), 62~63
YUV (YUV),197
energy,minimizing (能量, 最小化), 491~492
enhancing images (图像增强), 11~12,128~186
definition (定义), 130
entropy: computations (熵: 计算), 110
of a set of events, definition (一组事件, 定义), 109
epipolar:constraint (外极线: 约束), 402~403
geometry (几何), 402~403
lines,definition (直线, 定义), 402
plane, definition (平面, 定义), 402
epipole, definition (外极点, 定义), 402
EPS (encapsulated postscript) format (EPS格式), 39
equalization,histogram (均衡化, 直方图), 132~134
erosion of binary images (二值图像腐蚀), 65
definition (定义), 66~67
error (s):coverage versus (误差, 与场景覆盖), 403
definition (定义), 316
empirical interpretation of (经验解释), 315
false alarms and false dismissals (误报和漏报), 96~97
rate, classification system (率、分类系统), 96
statistical interpretation of (统计解释), 315~316
estimation: pose (估计: 位姿), 453~460
pose estimation procedure (位姿估计过程), 439~444
Euclidean distance: definition (欧几里得距离, 定义), 100
scaled definition (尺度比定义), 103
even functions (偶函数), 176
external corners (外角), 4~6
external energy (外部能量), 492
extracting non-iconic representations (抽取非图像表示), 14
extractor,feature (外部, 特征), 94
extremal axis length (极轴长度), 77
extremal points (极点), 76~78
extrinsic camera parameters (外部摄像机参数), 445~449
eye, as camera (眼睛, 像摄像机), 26-27, 参见iris-scanning system

F

face(s) (人脸), 377
finding (检测), 240~241
identifying (识别), 201~202
fade, definition (淡变, 定义), 272~273
false alarms, (误报) 96~97
false dismissals (漏报), 96~97
fast Fourier transform (快速傅里叶变换), 181~182
feature extraction (特征抽取), 498
feature extractor (特征抽取算子), 94
feature vector representation (特征向量表示), 100
feedforward networks, multilayer (前向网络, 多层), 123~126
field of view (FOV) :angular (视场: 角), 31
definition (定义), 31
file formats: GIF (Graphics Interchange Format) (文件格式: GIF), 38
JPEG (Joint Photographic Experts Group) (联合摄

影专家组），38~39
MPEG（Motion Picture Experts Group）for video（视频运动图像专家组），39~40
TIFF (Tag Image File Format)（标记图像文件格式），38
参见formats
file headers（文件头），36
filtering: bandpass（滤波器：带通），181
Gaussian LOG edge detection and（高斯、LOG边缘检测），149~151
images（图像），128~186
LOG, Marr-Hildreth theory（LOG，马尔-海尔德斯理论），155~157
median（中值），137~141
filter (s):box,definition（滤波器：盒形，定义），136
Gaussian（高斯），136~137
LOG, detecting edges with（LOG，检测边缘），151~153
masks as matched（匹配滤波模板），158~167
find procedure（find过程），59~60
fishtank virtual reality（鱼缸虚拟现实），539
fitting: constraints（拟合：约束），317
models to segments（线段拟合模型），312~317
problems（问题），316~317
segmenting curves via（曲线分段），317
flesh finding（人体检测），241~242
flight simulation（飞行仿真），529
FOC (focus of contraction), definition（收缩中心，定义），255
focus: depth of field and（焦距：景深），404~406
features（特征），341~342
interpreting depth from（根据焦距变化求深度），393
focus of contraction(FOC)（收缩中心），254 definition（定义），255
focus of expansion（FOE）（膨胀中心），254,393
definition（定义），255
FOE（focus of expansion）（膨胀中心），254,255
foreground pixels（前景像素），51
foreshortening（透视缩短），42,385~386
definition（定义），384
fork junctions（叉连接），377
formats: commonly used（格式：常用的），36~37
comparison of（比较），40
digital image（数字图像），35~40
EPS (encapsulated postscript)（封装的PostScript），39
GIF（Graphics Interchange Format）（图形交换格式），38
JPEG（Joint Photographic Experts Group）（联合摄影专家组），38~39
MPEG（Motion Picture Experts Group）for video（运动图像专家组），39~40
PostScript, 39
TIFF（Tag Image File Format），（标记图像文件格式），38
4-neighbors（4-邻域），52
4-tuples（4元组），508
Fourier analysis（傅里叶分析），172
Fourier basis（傅里叶基），174~175
image processing operations using（图像处理运算），175
Fourier power spectrum,definition（傅里叶功率谱，定义），177
Fourier transform（傅里叶变换），223
definition（定义），177
discrete（离散），179~181
fast（快速），181~182
FOV (field,of view):angular（视场：角），31
definition（定义），31
frame buffer（帧缓存区），23~24
frame grabber（帧捕捉器），23
frames of reference（参考坐标系），42~45
Freeman chain code（Freeman链码），293
Frei-Chen basis（Frei-Chen 基），163~167
frequency, spatial, analysis of using sinusoids（频率，空间，利用正弦波分析），172~184
front image plane（前图像平面），395
functional models,matching（功能模型，匹配），513~514
functional properties,GRUFF（功能属性，GRUFF），514~515
functions, odd and even（函数，奇偶），176
fuzzy classification（模糊分类），124

G

games, Decathlete（游戏，Decathlete），255~256
Gamma correction（Gamma校正），131
gates,AND,OR,and NOT（门，与、或、非），121，125
Gaussian filter（高斯滤波器），136~137
definition（定义），137
Gaussian filtering,LOG edge detection and（高斯滤波器，LOG边缘检测），149~157
Gaussian function,definition（高斯函数，定义），149
Gaussian noise（高斯噪声），136~137,315
Gaussian smoothing（高斯平滑），156
masks（模板），144
Gaussians, useful properties of（高斯，有效特性），151
gear_body（圆盘形结构元，去掉齿轮轮齿的部分），68,71

generalized-cylinder models（广义圆柱模型），483~484
Generic Object Recognition Using Form and Function（GRUFF）system（GRUFF系统），513~516,517
geometric distortion（几何畸变），27
geometric hashing（几何散列），346~350
geometric icons（几何图标），504~506
geometric models, matching via alignment（几何模型，比对匹配），496~504
geometry,used in Tsai calibration method（几何学，Tsai标定方法），446~447
geons（几何离子），504~506
GIF（Graphics Interchange Format）（图形交换格式），38
gradient, texture（梯度，纹理），385~387
Graphics Interchange Format（GIF）（图形交换格式），38
graph-matching algorithms（图匹配算法），106
graph-partitioning clustering technique，Shi' s（图分割聚类技术，Shi的）286~289
graphs:aspect（图：表象），488~489
region adjacency（区域邻接），81~82
region adjacency,definition（区域邻接，定义），81~82
gray-level mapping（灰度级映射），130~134
gray-scale image（s）:definition（灰度级图像，定义），30
thresholding（阈值化），83~89
grayval/binsize（灰度值/箱格大小），85
Greedy Exchange algorithm（贪婪交换算法），267, 268, 270
grids（栅格），437~439
group homogeneity（组内均衡性），86~88
GRUFF （Generic Object Recognition Using Form and Function）system（GRUFF系统），513~516,517
category hierarchy（类别层次），515~516
definition tree（定义树），515
functional properties（功能属性），514~515
knowledge primitives（知识基元），514
processing by（处理），517

H

Hamming distances,（海明距离），560~561
haptic sense, definition（触觉，定义），540
hard constraints（硬约束），490
hash function（散列函数），244
hash indexes（散列索引），244~245
hash table（s）（散列表），244
in relational indexing（相关索引），508
hashing, geometric,（散列，几何），346~350
HCI issues, in virtual reality（VR）systems（人机交互问题，在虚拟现实系统中），546
head-mounted displays（HMDs）（头戴式显示器），530,535~537
heuristics,for detection of zoom（启发式，检测变焦），276
hexacone（六棱锥），194,195
hidden units（隐层神经元），124
high contrast, detecting（高对比度，检测），141~143
high dimensionality（高维数），316
highlight, definition（高亮区，定义），206
histogram（s）: clustering methods based on（直方图:聚类方法），284~286
color（颜色），199~201,231,233
comparing（比较），274,276
definition（定义），84
equalization（均衡化），132-34
mode seeking（模示搜索），284-85
Ohlander's recursive histogram based clustering technique（Ohlander递归直方图聚类技术），285~286,287
shape（形状），236~237
texture（纹理），235
using for threshold selection（阈值选择），83~85
HMDs（head-mounted displays）（头戴式显示器），530,535~537
hole_mask（八边形结构元，直径比圆孔稍大），68,70
hole_ring（像素环，标记圆孔中心位置的像素），68, 69, 70
holes, counting（孔，计数），4~6
homogeneous coordinates, definition（齐次坐标，定义），329
Hough transform: algorithm（霍夫变换：算法），306
Burns line finder using principles of（利用原理的Burns直线检测），311
for detecting lines and circular arcs(检测直线和圆弧），303-12
encoding gradient direction with（编码梯度方向），318
extensions（扩展），310
finding circles with,（检测圆），309~310
generalized（广义），310
HSI（hue-saturation- intensity）（色调-饱和度-亮度），194~197
encoding, conversion from RGB （red-green- blue）encoding（编码，从RGB（红-绿-蓝）编码的转换），196
HSV（hue-saturation-value）system（HSV（色调-饱和

度-值）系统），194
hue-saturation-intensity（HSI）（色调-饱和度-亮度），194~197
encoding,conversion from RGB （red-green-blue）encoding（编码，从RGB（红-绿-蓝）编码的转换），196
human body,3D models（人体，3D模型），485
human edge detection（人类视觉的边缘检测）153~155
human heart,modeling motion of（人体心脏，跳动模型），494~495
human perception: color（人类感知：颜色），209
depth（深度），394
shading used in（基于明暗信息）208~209
hyperplanes（超平面），122

I

IBM（IBM）,226~227
identification:of humans via iris of eye（识别：基于虹膜的身份），554~561
requirements for identification systems（对识别系统的要求），555~557
identifying regions:classical clustering algorithms（区域识别：经典的聚类算法），281~282
clustering methods（聚类方法），281
in image segmentation（图像分割），280~291
region growing（区域增长），289~291
IDFT（inverse discrete Fourier transform）（离散傅里叶反变换），180~181
illuminated objects,sensing（被照射物体，感测），189
illumination boundary,definition（光照边界，定义），374
image addition（图像相加），12
image-based rendering, definition（基于图像的绘制，定义），543
image collections（图片收藏库），227~228
image data（图像数据），36
image databases（图像数据库），3~4,226~230
queries（查询），228~230
image distance measures（图像距离测度），230~244
image energy（图像能量），491~492
image enhancement: convolution and cross correlation（图像增强：卷积和交叉相关），167~172
definition（定义），130
detecting edges using differencing masks（差分模板检测边缘），141~149
Gaussian filtering and log edge detection（高斯滤波器和LOG边缘检测），149~157
gray-level mapping（灰度级映射），130~132
histogram equalization（直方图均衡化），132~134
image smoothing（图像平滑），136~137
median filtering（中值滤波），137~141
removal of small image regions（去除小图像区域），134~135
image file formats:comparison of（图像文件格式：比较），40
GIF（Graphics Interchange Format）（图形交换格式），38
TIFF（Tag Image File Format）（标记图像文件格式），38
image file header（图像文件头），36
image flow:computing（图像流：计算），262~263
definition（定义），255
equation（公式），263~264
solving for by propagating constraints（传播约束求解），264~265
image formation（图像形成），24~26
image histograms（图像直方图），83~85
image operations（图像运算），10~14
image plane,front（图像平面，前），395
image processing（图像处理），128~186
definition（定义），15~16
Fourier basis used for（采用傅里叶基），175
image quantization, spatial measurement and（图像量化，空间度量）,31~35
image registration, definition（图像配准，定义），327
image representation, imaging and（图像表示，成像），21~50
image restoration,definition（图像恢复，定义），130
image segmentation（图像分割），279~325
identifying regions（区域分割），280~291
image subtraction（图像相减），12,253~254
image understanding,definition（图像理解，定义），15~16
image warping（图像变形），12
imagery,real and synthetic（图像，真实和合成），542~545
image（s）:acquisition of（图像：获取），461~463
analog, definition（模拟，定义），29
basis,computing（基，计算），519~521
basis, for set of training images(基，对于训练图像集），518~519
binary,analyzing（二值化，分析），51~91
binary, definition（二值化，定义），30
color（颜色），45~46
computing features from（计算特征），13
computing output from input（从输入计算输出），

139~141
content-based, indexing for retrieval with multiple distance measures (基于内容的，多距离测度图像索引)，248
counting objects in (目标计数)，54~56
digital,definition (数字，定义)，29
digital, formats (数字，格式)，35~40
digital, picture functions and (数字，图像函数)，29~35
digital,problems with; (数字，问题)，27~29
enhancing (增强)，11~12
filtering and enhancing (滤波和增强)，128~186
gray-scale, definition (灰度级，定义)，30
gray-scale, thresholding (灰度级，阈值化)，83~89
improving (改善)，129~130
intrinsic (本征)，371~376
labeled (标记的)，292
labeled, definition (标记的，定义)，30
masks applied to (模板运算)，53~54
matching in 2D (2D匹配)，326~370
multiple (多幅)，12
multispectral (多谱)，45~46,210
multispectral, definition (多谱，定义)，30
perceiving 3D from 2D (2D图像中的3D信息)，371~409
pseudo-colored (伪彩色)，30
range (深度)，47~49
raw (原始)，35
removing small regions from (去除小区域)，134~135
retrieving content-based (基于内容检索)，226~250
run-coded binary (游程编码二值) 37
smoothing (平滑)，136~137
thematic (主题)，30,210
tracking edges of binary edge image (二值边缘图像的边缘跟踪)，303
training, basis images for set of (训练，基图像)，518~519
2D,3D structure from (2D, 3D结构)，42
2D,difference operators for (2D，差分算子)，144~149
2D,motion from sequences of (2D，序列求运动)，251~278
3D cues in 2D images (2D图像中的3D线索)，383~388
types of (类型)，29-31, 参见perspective imaging models
three-dimensional (3D) images 3维图像
two-dimensional (2D) images 2维图像
imaging: devices (成像：设备)，22~27
image representation and (图像表示)，21~50, 参见perspective imaging models
independent test data,definition (独立测试数据，定义)，96
indexes:B+-tree (索引：B^+-树)，245~247
hash (散列)，244~245
K-d tree (K-d树)，247
R-tree (R-树)，247~248
spatial (空间)，247~248
standard (标准)，244~247
indexing: for content-based image retrieval with multiple distance measures (索引：基于内容的多距离测度图像索引)，248
relational (相关)，363~364,508,510,511, 参见RIO object recognition system
input images,computing from output images (输入图像：计算输出图像)，139~141
inspection,crossbars (检查，交叉支撑杆)，4~6
integrated tracking (集成跟踪)，271~272
integrating,spatial constraints (综合，空间约束)，472
intensity (强度)，393
mapping (映射)，131
values (值)，46, 参见HSI (hue-saturation-intensity)
interactive segmentation of anatomical structure (解剖组织的交互式分割)，529
interest operators (兴趣算子)，257~258
interesting points (兴趣点)，256~261
internal corners (内角)，4~6
interposition (穿插)，42
definition (定义)，384
interpretation trees (IT) (解释树)，352~354
definition (定义)，352
line drawing (线条图)，380
intrinsic camera parameters (内部摄像机参数)，445
intrinsic images (本征图像)，371~376
scene values (场景值)，375
invariant features (不变特征)，14
inverse discrete Fourier transform (IDFT) (离散傅里叶反变换)，180~181
inverse perspective (逆透视)，439
iris-scanning system (虹膜扫描系统)，554~561
hardware components (硬件组成)，557~558
performance (性能)，560~561
representation in (表示)，558~560
system design (系统设计)，557~560
isodata clustering (isodata 聚类)，282~284

IT (interpretation trees) (解释树), 352~354
definition (定义), 352
line drawing (线条图), 380

J

Jacobian matrix (雅可比矩阵), 441~442
Joint Photographic Experts Group (JPEG): (联合摄影专家组),
format (格式), 38~39
Motion (运动), 39~40
JPEG (Joint Photographic Experts Group): (联合摄影专家组),
format (格式), 38~39
Motion (运动), 39~40
jump edge (跳跃边缘), 373
definition (定义), 374
junction pixels (连接像素), 301
junctions (连接), 377
types of (类型), 377~378

K

K-d tree indexes (K-d树索引), 247
K-means clustering, iterative (K-均值聚类, 迭代), 282
keywords (关键词), 228~229
knowledge-based thresholding (基于知识的阈值化), 89
knowledge-directed thresholding (面向知识的阈值化), 285
knowledge primitives,GRUFF (知识基元, GRUFF), 514

L

L-junctions (L连接), 377
label,definition (标记, 定义), 351
LABEL field (LABEL字段), 62~63
labeled image (s) (标记图像), 292
definition (定义), 30
labeling: block edges via discrete relaxation (标记: 离散松弛法标记模块边缘), 382
connected components (连通成分), 56~63
connected components,using run-length encoding for (连通成分: 游程编码), 62~63
consistent (一致性), 351~353
continuous relaxation (连续松弛), 356~359
cubes (立方体), 466~467
discrete relaxation (离散松弛), 354~356,357
edges of blocks via backtracking (回溯法标记模块边缘), 381
line drawings of blocks (模块线条图), 377~383
lines via relaxation (松弛法线段), 381~383
terms (术语),377
labeling algorithms:recursive (标记算法: 递归), 57~59
row-by-row (逐行), 59
labels function (标记函数), 61
lake_num (湖数), 112
lakes (湖), 112
Lambertian reflectance model (朗伯反射模型), 469~471
Lambertian reflection (朗伯反射), 204~205
laser light projectors (激光投影仪), 438
Laws texture energy measures (Laws纹理能量测度), 220~222,224
layout,color (分布, 颜色), 232~233
leaf nodes (叶子节点), 107,245~247
quadtree (四叉树), 294, 参见nodes
learning:machine (学习: 机器), 119
supervised (监督), 119
supervised,on Veggie Vision (监督, Veggie Vision), 553~554
unsupervised (无监督), 119
least-squares:error criteria, definition (最小二乘误差指标, 定义), 313
method (方法), 312~314
problem defining (问题,定义), 431~436
lenses (镜头) 24, 参见thin lens equation
LIDAR (light detection and range) devices (光检测与测距设备), 47~48
lid_bottomof_image (盖在图像底部), 112
lid_num (盖数), 112
lid_right_of_bay (盖在湾右侧), 112
lids (盖), 104,112
light:ambient (光: 环境), 207~208
ambient,definition (环境, 定义), 208
darkening with distance (随距离增加而变暗), 206~207
diffuse reflection of (漫反射), 204~205
radiation from one source of (来自单一光源的辐射), 203~204
sensing (感测), 21~22,189
specular reflection of (镜面反射), 205~206
structured (结构), 437~439
use of (使用), 41~42
white, definition (白, 定义), 189
light detection and range (LIDAR) (光检测与测距), devices (设备), 47~48
limb (翼边), 373
definition (定义), 374
line drawings:interpretation tree for (线条图: 解释树), 380

labeling drawings of blocks（模块的线条图标记），377~383
linear optimization,constrained（线性优化，约束），456~457
linear transformations,scaling（线性变换，缩放），329~330
lines:Burns line finder（线段:Burns直线检测器），311~312
detecting with Hough transform（用霍夫变换检测），303~312
epipolar（外极线），402
fitting（拟合），313~314
labeling via relaxation（松弛法标记），381~383
straight,finding（直线，检测），304~309
virtual（虚拟），393~394
linker, Canny edge detector and（连接算子，Canny边缘检测算子），297~301
local binary paritition（局部二值分解），217
local-feature-focus method,of object recognition（局部特征焦点法，目标识别），341~344
location of model point and image point（模型点和图像点的位置），335~338
LOG edges,Gaussian filtering and detection of(LOG边缘，高斯滤波器LOG边缘检测），149~157
LOG filtering,Marr-Hildreth theory（LOG滤波器，Marr-Hildreth理论），155~157
LOG filters,detecting edges with（LOG滤波器，边缘检测），151~153
looming（渐显），393
lossless compression,definition（无损压缩，定义），36
lossy compression,definition（有损压缩，定义），36
low-level features,detection of（低层特征，检测），129~130
luminance（亮度），197

M

Mach band effect, artificial neural network （ANN） used to produce（马赫带效应，人工神经网络产生），153~155
machine learning（机器学习），119
machine vision,definition（机器视觉，定义），1
Magic Value（魔值），37
magnetic resonance angiography （MRA）（核磁共振血管造影术），47
magnetic resonance imaging （MRI）（核磁共振成像），6~7,47,210
mapping: affine, 2D object recognition via（映射：仿射，2D目标识别），341~350
functions,affine（函数，仿射），329~339
gray-level（灰度级），130~134
polynomial（多项式），367
texture（纹理），542~545
mark,definition（标记，定义），374
Marr-Hildreth theory（Marr-Hildreth理论），155~157
mask（s）（模板），134
applying to images（图像），53~54
box smoothing（盒形平滑），144
derivative（导数），141~144
differencing, detecting edges using（差分，边缘检测），141~149
Gaussian smoothing（高斯平滑），144,151,152
for implementation of LOG filter（LOG滤波器），151,152~153
as matched filters（匹配滤波器），158~167
operations defined via（运算定义），167~168
origins（原点），54
Prewitt（Prewitt）146,148~149,307
properties of derivative and smoothing（导数和平滑模板的特性），143~144
Roberts（Robert），146~147
Sobel（Sobel），146,147
matching:boundary（匹配：边界），237~238
elastic（弹性），240
functional models（功能模型），513~514
geometric models via alignment（几何模型比对），496~504
relational distance（相关距离），359~363
relational, 2D object recognition via,（相关，2D目标识别），350~364
relational models（关系模型），504~513
sketch（简图），238~240
symbolic,and relational constraints（图符、相关约束），401
in 2D（二维），326~370
3D models and（3D维模型），479~526,参见alignment
mathematical morphology（数学形态学），63
matrix: best affine calibration（矩阵：最佳仿射标定），431~437
co-occurrence（共生），217~220
perspective transformation（透视变换），423~426
MaxCol（最大列），56
max-error criteria,definition（最大误差指标，定义），313
maximum intensity projection （MIP）（最大强度投影），47
MaxRow（最大行），56
MDFs （most discriminating features）（最佳分类特征），

522
mean radial distance（平均径向距离），75
measurement, spatial, image quantization and（度量，空间，图像量化），31~35
measure(s):color similarity（度量：颜色相似度），231~233
distance,indexing for content-based image retrieval with multiple（距离，基于内容的多距离测度图像索引），248
image distance（图像距离），230~244
object presence and relational similarity（目标检测及空间关系度量），240~244
measuring:shape similarity（度量：形状相似性），235~240
texture（纹理），215~223
texture similarity（纹理相似性），233~235
median:definition（中值，定义），138
filtering（滤波），137~141
MEFs （most expressive features）（最佳描述特征），521~522
memory,faster search of（内存，快速搜索），521~523
mesh:balloon models for 3D（网格：3D气球模型），493~494
models（模型），472,480,481
regular（规则），480
triangular（三角形），480
method of least squares（最小二乘法），312~314
microdensitometer（测微密度计），45
MIP (maximum intensity projection)（最大强度投影），47
mixed reality（混合现实），530
definition（定义），535
models:active contour（模型：活动轮廓），489~492
balloon for 3D（3D气球），493~494
fitting to segments（线段拟合），312~317
generalized-cylinder（广义圆柱），483~484
matching functional（功能匹配），513~514
matching geometric via alignment（几何模型比对匹配），496~504
mesh（网格），472,480,481
parametric（参数），116~117
perceptron（感知器），120~123
perspective imaging（透视成像），395~397
physics-based and deformable（物理学和可变形），489~495
relational,matching（关系，匹配），504~513
surface-edge-vertex（表面-边-顶点），480~483
3D,and matching（3D，和匹配），479~526
3D relational（3D关系），504~506
true 3D versus view-class（实际3D与视类），488~489
2D-3D alignment（2D-3D比对），498~501
view-class relational（视类关系），506~513
wire-frame（线框），480，参见three dimensional (3D) models
moment.（矩），参见second moment, second-order
morphology,binary image（形态学，二值图像），63~73
most discriminating features（MDFs）（最佳分类特征），522
most expressive features（MEFs）（最佳描述特征），521~522
motion:aggregating motion trajectories（运动：运动轨迹聚类），321~324
coherence, segmentation using（一致性，运动一致性分割），321~324
computing paths of moving points（计算运动点路径），265~272
modeling of human heart（建立心脏跳动模型），494~495
phenomena（现象），393
phenomena and applications（现象和应用），251~253
structure perceived from（从运动恢复结构），472~475
from 2D image sequences（从2D图像序列求运动），251~278
in virtual reality (VR) systems（虚拟现实系统），540
motion field:definition（运动场：定义），254~255
point correspondences used to compute（点对应计算），256~271
Motion Joint Photographic Experts Group（JPEG）format（MPEG格式），39~40
Motion JPEG （Joint Photographic Experts Group）format（MPEG格式），39~40
motion parallax（运动视差），386
definition（定义），387
Motion Picture Experts Group （MPEG）:compression of video（运动图像专家组：视频压缩），261~262
format for video（视频格式），39~40
motion vectors:computing（运动向量：计算），254~265
deriving for interesting points（计算兴趣点的运动向量），260
MPEG （Motion Picture Experts Group）: compression of video（运动图像专家组：视频压缩）261~262
format for video（视频格式），39~40
MRA (magnetic resonance angiography)（核磁共振血管造影术），47
MRI (magnetic resonance imaging)（核磁共振成像），6~7,47,210
multidimensional data,decisions using（多维数据，多维数据决策），117~119

multilayer feedforward network（多层前向网络），123~126
multiple images,combining（多幅图像，组合），12
multispectral image（s）（多谱图像），45~46,210
definition（定义），30

N

nearest-neighbor rule（最近邻规则），103
nearest neighbors,used in classification（最近邻，分类），103~104
Necker Cube/Phenomena（内克立方体/现象），382
neighborhoods,pixels and（邻域，像素），51~52
neighbors:nearest,used in classification（邻域：最近邻，分类），103~104
utility function（功能函数）,57~58
neural nets,artificial（神经网络，人工），119~126
neurons（神经元），119~120
artificial（人工），120~122
nodes（节点），107,245~247
octree（八叉树），484~485
quadtree（四叉树），294
noise,Gaussian（噪声，高斯），136~137,315
nominal resolution,definition（标称分辨率，定义），31
non-accidental alignments（非偶然对齐），394
non-iconic representations,extracting(非图像表示,抽取），14
nonlinear optimization（非线性优化），316
nonlinear warping（非线性变形），364~368
nonmaximum suppression（非最大抑制），299
normalized dot product（规范化点积），161~162
normalized RGB coordinates（规范化RGB坐标），192~193
notation（s）（符号），29~31
pixel values（像素值），23

O

object coordinate frame O（物体坐标系O），44
object counting（目标计数），54~56
object pose computation,3D sensing and（目标位姿计算，3D感知），410~478
object presence,and relational similarity measures（目标检测及空间关系度量），240~244
object recognition（目标识别），335~338
2D,via affine mapping（仿射映射法2D目标识别），341~350
2D,via relational matching（相关匹配法2D目标识别），350~364
3D,classifying（3D，分类），495~496
3D, paradigms（3D目标识别范例），495~523
of 3D objects by appearance（基于外观的3D目标识别），516~523
3D object recognition paradigms（3D目标识别范例），495~523
by appearance（基于外观），516~523
eigenspace recognition by appearance（特征空间外观识别）,522
Generic Object Recognition Using Form and Function（GRUFF）system（GRUFF系统），513~516,517
geometric hashing method（几何散列方法），346~350
local-feature-focus method（局部特征聚焦法），341~344
RIO system（RIO系统），506~513
TRIBORS system（TRIBORS系统），499~501，参见recognition
object reconstruction,3D（目标重构，3D），460~468
occlusion（遮挡）,383~384
octree(s)（八叉树），467,484~486
odd functions（奇函数），176
offline preprocessing（离线预处理），348,349,508
O'Gorman and Clowes algorithm（O'Gorman 和Clowes的算法），305,306,308
Ohlander's recursive histogram-based clustering technique（Ohlander递归直方图聚类技术），285~286,287
one-dimensional（1D）signals,differencing（1D信号，差分），141~144
online recognition（在线识别），348,350
opening of binary images（二值图像开运算），65~66
definition（定义），67
operations, defining via masks（运算，模板运算定义），167~169
operator（s）:Canny（算子：Canny），157~158
difference（差分），144~149
interest（兴趣），257~258
Prewitt（Prewitt），146,148~149
Roberts cross（Roberts交叉算子）,146~147
Sobel（Sobel），146,147
optimization:constrained linear（优化：约束线性），456~457
nonlinear（非线性），316
and verification of pose（位姿验证）,460
ordering constraint（顺序约束），403
organizing databases（组织数据库），244~248
origins, mask（原点，模板），54
orthogonal basis,using（正交基，用），160~162

orthogonal transforms,definition（正交变换，定义），331~332
orthographic projection（s）（正投影）,426~428,470
orthonormal transforms,definition（标准正交变换，定义），331~332
Otsu method,automatic thresholding（Otsu方法，自动阈值化），85~89
outliers（局外点），315,316
output images,computing from input images（输出图像，从输入图像计算），139~141
overlays（覆盖图），292

P

page description language（页面描述语言PDL）,39
panning（扫视），274
paradigms,3D object recognition（范例，3D目标识别）,495~523
parallax,motion（视差，运动），386,387
parallel implementation（并行实现），172
parallel list（PTLIST）array（并行结构）,304,305
parameters:camera model（参数：摄像机模型），436~437
  closed form solutions for（封闭解），314~315
  extrinsic camera（外部摄像机），445~449
  intrinsic camera（内部摄像机），445
parametric models,for distribution（参数模型，分布）,116~117
PARENT arrays（PARENT数组）,59~61
part, definition（部件，定义），351
pattern recognition: concepts（模式识别：概念），92~127
  problems（问题），92~93
PBM (Portable Bit Map)（可转移式点阵图，PBM格式）,37,38
PDL (page description language)（页面描述语言，PDL格式）,39
perception: human color（感知：人类色感）,209~210
  shading used in（明暗信息）,208~209
  of structure from motion（从运动恢复结构）,472~475
perceptron model（感知器模型），120~123
perimeter length（周长），74
perspective: imaging model（透视：成像模型），395~397
  inverse（逆），439
  projections（投影），426~428
  transformation matrix（变换矩阵），423~426
perspective scaling（透视缩放），385,386
  definition（定义），384
Perspective 3 Point Problem（P3P）（三点透视问题）,439~444
PGM（Portable Gray Map）（PGM格式）,37~38
Phong model of shading（phong明暗模型），208
photography, model（摄影，模型），22
photometric stereo（光度立体），471~472
physics-based models,deformable and（物理学模型，可变形模型），489~495
pick-and-click distance（挑选-点击距离）,234~235
picture function（s）:2D（图像函数：2D），175~179
  definition（定义），30
  digital images and（数字图像），29~35
pixel arrays（像素阵列），43~45
pixel coordinate frame I（像素坐标系I），43~44
pixel values,notations（像素值，符号），23
pixels background（像素：背景），51
  changing values of（改变像素值），10~11
  definition（定义），3
  foreground（前景），51
  junction（连接），301
  neighborhoods and（邻域），51~52
    参见external corners:internal corners
planes, front image（平面，前图像平面），395
plates（盘片），504~506
point correspondences, computing motion field with（点对应，计算运动场），256~261
point operator, definition（点算子，定义），131
points: computing 3D points using multiple cameras（点：多摄像机3D点计算），428~430
  contributing（贡献），498
  control（控制），332~334
  location of（位置），335~338
  pose from 2D-3D point correspondences（2D-3D点对应求位姿）,455~456
  representation of 2D（2D点的表示），328~329
  参见 Perspective 3 Point Problem（P3P）
polygonal approximation（多边形逼近），293
polygons,Voronoi（多边形，Voronoi），214~215
polynomial mappings（多项式映射），367
Portable Bit Map（PBG格式）,37~38
Portable Gray Map（PGM格式）,37~38
pose: clustering（位姿：聚类），344~346
  definition（定义），344
  estimation（估计），453~460
  estimation procedure（估计过程），439~444
  object pose computation and 3D sensing（3D感知与目标位姿计算），410~478

from 2D-3D point correspondences (2D-3D点对应) 455~456
verification and optimization of (验证和优化), 460
in virtual reality (VR) systems (虚拟现实系统), 539~540
PostScript (PostScript格式) ,39
power spectrum (功率谱), 177~179
measuring texture by autocorrelation and (用自相关和功率谱度量纹理), 221~223
precision: definition (查准率, 定义), 97
recall versus (查全率), 97~98
preprocessing,offline (预处理, 离线), 348,349,508
Prewitt,Judith (Judith Prewitt博士) ,146,148
Prewitt masks (Prewitt模板), 307
Prewitt operator (Prewitt算子), 146
primary key (主键), 244
primitives, knowledge (基元, 知识), 514
prior_neighbors function (prior_neighbors函数) ,61
probability distribution (概率分布), 114~115
projection (s) :orthographic (投影: 正投影), 426~428,470
weak perspective (弱透视), 426~428
projectors:calibration of (投影仪, 标定), 437
laser light (激光), 438
replacing camera with (代替摄像机), 412~413
property tables, regions represented by (特征表, 区域表示), 294~295
proximity primitive (邻近性基元), 514
pseudo color (伪彩色), 210
pseudo-colored images (伪彩色图像), 30
P3P (Perspective 3 Point Problem) (三点透视问题) 439~444
solution (解), 442~443
PTLIST (parallel list) array (并行结构PTLIST) ,304,305
pyramids,interpretation tree for line drawings of (塔状物, 塔状物线条图的解释树), 380

Q

QBE (query-by-example) (示例查询), 229~230
QBIC (Query by Image Content) database (QBIC数据库), 226~227
quadtrees (四叉树), 247-248
regions represented by (区域表示), 294
quantization: effects (量化: 效果), 29
image, spatial measurement and (图像, 空间度量), 31~35
special quantization effects (空间量化效果), 33
queries,image database (查询, 图像数据库), 228~230
query-by-example (QBE) (示例查询), 229~230
Query by Image Content (QBIC) database (QBIC数据库), 226~227
quicksort,modifying (快速排序, 改进的), 139

R

R-tree indexes (R-树索引), 247~248
radial distance:mean (径向距离: 均值), 75
standard deviation of (标准差), 75
radial distortion,.rectifying (径向畸变, 矫正), 366~367
radiation,from one light source (照射,来自单一光源) ,203~204
RAG (region adjacency graphs) (区域邻接图), 81~82
definition (定义), 81~82
ramp (斜坡), 33
range:data (深度数据) ,463~464,465
images (图像), 47~49
scanners (扫描仪) ,47~49
raster order (光栅顺序), 35
raster-oriented coordinate systems (光栅坐标系), 30
raw images (原始图像), 35
ray tracing (光线跟踪), 541~545
real image eoordinate frame F (实际图像坐标系), 44
real imagey, composing (实际图像, 合成), 542~545
recall:definition (查全率, 定义), 98
precision versus (查准率), 97~98
receiver operating curve (ROC) (受试者操作曲线), 97
receptors,sensitivity of (敏感性,感受器的), 190~191
recognition:by alignment (识别, 比对), 337
character (字符), 98~100
eigenspace,by appearance (特征空间, 基于表象), 522
online (在线), 348,350
structural pattern (结构模式), 105
structural techniques (结构方法), 104~106
in Veggie Vision (Veggie Vision系统) 550~551, 参见object recognition
recognition-by-alignment,definition (比对识别, 定义), 337
reconstruction:3D object (重构: 3D目标), 460~468
surface (表面), 464
recursive labeling algorithm (递归标记算法), 57~59
red-green-blue (RGB) :basis for color (红-绿-蓝 (RGB): 颜色基), 191~193
conversion to YUV (到YUV的转换), 197
encoding, conversion to HSI (hue-saturation-intensity) encoding (编码, 到HSI (色度-饱和度-亮度) 编码的转换), 196
reference frames (参考坐标系), 42~45,328~329
reflectance,Lambertian (反射, 朗伯) 469~471

reflection(s)（反射），338~339
diffuse（漫反射），204~205
specular（镜面）205~206
specular,definition（镜面，定义），206
region adjacency graphs（RAG）（区域邻接图），81~82
definition（定义），81~82
region properties（区域特征），73~81
regions（区域），377
boundary coding representing（边界编码表示），292~293
corners（角点），320~321
future（未来），297
growing（增长），289~291
identifying by contours（轮廓分割），295~312
identifying image segmentation（图像分割），280~291
labeled images representing（标记图像表示），292
labeled,finding borders of（标记的，边界检测），295~297
overlays representing（覆盖图表示），292
past（过去），297
property tables representing（特征表表示），294~295
quadtrees representing（四叉树表示），294
representing（表示），291~295
ribbons（条带），317~320
tracking existing boundaries of（边界跟踪），295~297
registration:image（配准，图像），327
of views（视图），463~464
reject class, definition（拒绝类别，定义），94
relation, definition（关系，定义），351
relational constraints, symbolic matching and（相关约束，图符匹配），401
relational description, definition（关系描述，定义），359
relational distance,matching（相关距离，匹配），359~363
relational indexing（相关索引），363~364,508,510,511参见RIO object recognition system
relational matching, 2D object recognition via（相关匹配，2D目标识别），350~364
relational models:matching（关系模型：匹配），504~513
view-class（视类），506~513
relational similarity measures, object presence and（空间关系度量，目标检测），240~244
relative orientation primitive（关系方向基元），514
relaxation:continuous,（松弛：连续），356~359
discrete（离散），354~356,357
discrete labeling block edges via（离散松弛法标记模块边缘），382
labeling lines via（松弛法线段标记），381~383
removing:salt-and-pepper noise（去除:椒盐噪声），134~135
small components（小成分），135~136
small regions from images（小图像区域），134~135
rendering: 3D models（绘制：3D模型），540~542
definition（定义），540
image-based（基于图像的），534
representation:of 3D models（表示:3D模型），480~487
feature vector（特征向量），100
features used for（特征），98~100
in iris-scanning system（虹膜扫描系统中），558~560
mesh models（网格模型），472,480,481
surface-edge-vertex models（表面-边-顶点模型），480
in Veggie Vision（Veggie Vision中），550~551
resolution:definition（分辨率:定义），31
nominal,definition（标称，定义），31
relating to blur（与模糊），406
subpixel,definition（亚像素级，定义），31
resolving power, definition（分辨力，定义），406
restoration, image, definition（恢复，图像，定义），130
retrieval:of content-based images（检索：基于内容的图像），226~250
image, indexing for content-based with multiple distance measures（图像，基于内容的多距离测度图像索引），248
problems（问题），3~4
RGB (red-blue- green): basis for color（RGB（红-绿-蓝）:颜色基），191~193
encoding ,conversion to HSI (hue-saturation-intensity) encoding（编码，到HIS（色度-饱和度-亮度）编码的转换），196
conversion to YUV（到YUV的转换），197
ribbon(s)（条带），317~320,484
definition（定义），318
detecting straight（检测直带），319~320
rigid transformations（刚体变换），331~332
RIO object recognition system（RIO目标识别系统），506~513
features employed by（采用特征）,507~508,509
RMSE （root-mean-square error）,definition（均方根误差，定义），313
Roberts basis（Robert基），162~163
Roberts masks（Robert模板），146~147
robots, vision-guided（机器人，视觉引导），9~10
ROC (receiver operating curve)（受试者操作曲线），97
root-mean-square error （RMSE）,definition（均方根误差，定义），313
rotation: arbitrary（旋转：任意），418~419
parameters for camera position（摄像机位置参数），445~449

2D（二维），330~331,332~334
3D（三维），415~418
row-by-row labeling algorithm（逐行标记算法），59
run-coded binary images（游程编码二值图像），37
run-length encoding,using for connected components labeling（游程编码，连通成分标记），62~63
run-of-signs test（符号变化检验），315

S

salt-and pepper noise,removing（椒盐噪声，去除），134~135
sampling_ring_spacer（圆盘形结构元，把齿轮体稍微扩大一点），68,71
sampling_ring_width（圆盘形结构元，把齿轮体扩大到齿尖部分），68,71
satellite images（卫星图像），8~9
saturation（饱和度）参见HSU（hue-saturation-intensity）scaled Euclidean distance, definition, 103
scaling: perspective（缩放：透视），385,386
2D（二维），329~330, 332~334
3D（三维），415
scanners, range（扫描仪，距离），47~49
scattering（散射），27
scene change, definition（场景变换，定义），272
SE（synthetic environment）（合成环境），535, 参见visual environment（VE）
searching, faster（搜索，快速），521~522
second moment: about axis（二阶矩：轴），80
axis with least（最小轴），81
second-order: column moment（二阶：列矩），77
mixed moment（混合矩），77
row moment（行矩），77
segmentation: color（分割：颜色），201~202,322~324
image（图像），279~325
using motion coherence（运动一致性分割），321~324
texture（纹理），223~224
segmenting: curves via fitting（分段：曲线拟合），317
video sequences（视频序列），273~274
segments: finding straight line（线段：检测直线），304~309
models fitted to（拟合模型），312~317
self-occluding surface（自遮挡表面），373
sensiny: illuminated objects（感测被照射物体），189
light（光线），21~22
sensing devices, virtual reality（VR）systems（感知设备，虚拟现实系统），539~540
sensor/transducer（传感器/变换器），94
sensors（传感器），45~49
LIDAR (light detection and range)（光检测与测距），47~48
multispectral（多谱），45~46
SFS（shape from shading）（从明暗恢复形状），388
shading（明暗分析），187~211,203~209
computing shape from（从明暗计算形状），468~472
human perception using（基于明暗信息的人类感知），208~209
interpreting shape from（从明暗解释形状），388
Phong model（phong模型），208
shadows（阴影），393
shape(s):computing from shading（形状：从明暗计算），468~472
histograms（直方图），236~237
interpreting from boundaries（从边界恢复），391~392
interpreting from shading（从明暗恢复），388
interpreting from texture（从纹理恢复），388~391
similarity measures（相似性度量），235~240
used in Veggie Vision（Veggie Vision使用），553
shape-from-shading,definition（从明暗恢复形状，定义），469
shear（切变），338
Shi's graph-partitioning clustering technique（Shi的图分割聚类技术），286~289
shift theorem（移位定理），183~184
shot change,definition（镜头切换，定义），272
signal level（信号级），516
signals:differencing 1D（信号：1D信号差分），141~144
representing as combination of basis signals（表示为基信号的组合），160~161
television,YIO and YUV for（电视，YIO和YUV），197~198
similarity: color（相似性：颜色），231~233
relational（关系），240~244
shape（形状），235~240
texture（纹理），233~235
sinusoids, analysis of spatial frequency using（正弦波，空间频率分析），172~184
size,used in Veggie Vision（尺寸，Veggie Vision使用），553
sketch matching（简图匹配），238~240
slant,definition（俯仰角，定义），389
small components,removing（小成分，去除），135~136
small image regions,removing（小图像区域，去除），134~135
smooth object alignment（光滑目标比对），501~504
smoothing: Gaussian（平滑：高斯），156
image（图像），136~137

smoothing masks（平滑模板），144,167~169
properties of（特性），144
snakes（蛇形），489~492
Sobel masks（Sobel模板），146,147
sonification,definition（语音合成，定义），539
space-carving（空间切割），464~467
spatial constraints,integrating（空间约束，综合），472
spatial frequency, analysis of using sinusoids（空间频率，正弦波分析），172~184
spatial indexing（空间索引），247-48
spatial measurement, image quantization and（空间度量，图像量化与），31~35
spatial quantization effects（空间量化效果），33
spatial relationships（空间关系），242~244
spatio-temporal gradient magnitude（时空梯度幅值），321
specular reflection（镜面反射），205~206
definition（定义），206
spin images（自旋图像），498,499,500
stability primitive（稳定性基元），514
standard deviation（标准差），102
of radial distance（径向距离），75
standard indexes（标准索引），244~247
statistical interpretation of error（误差的统计解释），315~316
stereo:acquisition system（立体：数据获取），461~463
configuration（结构），411~413
depth perception from（立体视觉求深度），397~403
displays（显示），399~400
photometric（光度），471~472
vision, establishing correspondences in（视觉，建立对应关系），400~403
stereoscopic display devices（立体显示设备），538~539
sticks（棒条），504~506
stiffness（硬度），494~495
still photos, JPEG （Joint Photographic Experts Group）format（静止图像，JPEG格式），38~39
storing video sequences（存储视频序列），277
straight lines, finding segments of（直线，检测直线段），304~309
straight ribbons,detecting（直带，检测），319~320
stretching（扩展），130~131
contrast, definition（对比度，定义），132
strobe light,use of（闪光灯，使用），41~42
strongback（硬壁），492
structural pattern recognition（结构模式识别），105
structural techniques, recognition（结构方法，识别），104~106
structured light,using（结构光，用），42,437~439
structure (s):corners（结构：角点），320~321
identifying higher-level（识别更高层），317~321
perceiving from motion（从运动恢复结构），472~475
ribbons（条带），317~320
3D,from 2D images（从二维图像到三维结构），42
union-find（并查），59~60
structuring elements, binary morphology（结构元，二值形态学），63~65,68~71
subpixel resolution,definition（亚像素分辨率，定义），31
subtraction, image（相减，图像），12,253~254
superquadrics（超二次），486~487
surface-edge-vertex models（表面-边-顶点模型），480~483
surface reconstruction（表面重构），464
surfaces, self-occluding（表面，自遮挡），373
surveillance（监视），253
symbolic matching,relational constraints and（图符匹配，相关约束和），401
synthetic environment（SE）,definition（合成环境，定义），535.参见visual environment（VE）
synthetic imagery,composing（合成图像，融合），542~545
system error, evaluating（系统错误，估计），96

T

T-junctions（T连接），377,384
Tag Image File Format (TIFF)（标记图像文件格式），38
teleoperation（遥操作），533~535
definition（定义），535
television signals,YIQ and YUV for（电视信号，YIO和YUV），197~198
temporal redundancy（时间冗余），40
tests,run-of-signs（检验，符号变化），315
tetrahedral elements（四面体元素），494~495
texels（纹理素），213
texture described based on（基于纹理素的描述），214~215
text applications（文本应用），8
texture（纹理），212~225
description vector（描述向量），233~235
energy（能量），220~212,224
histograms（直方图），235
interpreting shape from（从纹理恢复形状），388~391
measuring by autocorrelation and power spectrum（自相关和功率谱度量），221~223
measuring by binary partition（二值分解度量），217

measuring by co-occurrence matrices and features (共生矩阵和特征度量), 217~220
measuring by edge density and direction (边缘密度和方向度量), 215~217
measuring by texture energy (纹理能量度量), 220~221,224
quantitative measures of (定量纹理测度), 215~223
segmentation (分割), 223~224
similarity measures (相似性度量), 233~235
statistical approach (统计方法), 214
structural approach (结构方法), 213
texel-based descriptions of (基于纹理素的描述), 214~215
used in Veggie Vision (Veggie Vision使用), 552~553
texture gradient (纹理梯度), 42,385~387
definition (定义), 387
texture mapping (纹理映射), 542~545
definition (定义), 542
texturing, view-based (纹理化, 基于视图), 543~545
thematic images (主题图像), 30,210
theorems: convolution (定理: 卷积), 182~183
shift (移位), 183
thin lens equation (薄透镜方程) ,403~406
three-dimensional (3D) cues, in 2D images (三维线索, 二维图像), 383~388
three-dimensional (3D) images: interpreting from boundaries (三维图像: 边界解释), 391~392
interpreting from shading (明暗解释) ,388
interpreting from texture (纹理解释), 388~391
interpreting from vanishing points (消隐点解释) ,392
labeling line drawings used to portray (线条图标记), 377~383
perceiving from 2D images (2D图像中的3D信息) ,371~409
three-dimensional (3D) models:alignment,3D-3D (三维模型: 比对, 3D-3D), 496~498
alignment,2D-3D (比对, 2D-3D), 498~501
balloon (气球) ,493~494
generalized-cylinder (广义圆柱体), 483~484
human body (人体), 485
human heart (人体心脏) ,494~495
matching and (匹配), 479~526
mesh (网格), 472,480,481
octrees (八叉树), 484~486
physics-based and deformable (物理学和可变形), 489~495
relational (关系的), 504~506
rendering (绘制), 540~542
representation methods (表示方法), 480~487
superquadrics (超二次), 486~487
surface-edge-vertex (表面-边-顶点), 480~483
true versus view-class models (实际3D模型, 视类3D模型) ,488~489.参见models
three-dimensional (3D) objects: recognition by appearance (三维目标: 基于表象的识别), 516~523
reconstruction (重构), 460~468
RIO object recognition system (RIO目标识别系统), 506~513
three-dimensional (3D) points, computing using multiple cameras (多摄像机3D点计算), 428~430
three-dimensional (3D) sensing, object pose computation and (3D感知与目标位姿计算) ,410~478
three-dimensional (3D) structure, from 2D images (从二维图像到三维结构) ,42
three-dimensional (3D) -3D alignment (3D-3D比对), 496~498
three-dimensional (3D) :affine transformations (三维: 仿射变换), 413~421
classifying 3D object recognition (3D目标识别分类) , 495~496
object recognition paradigms (目标识别范例) ,495~523
pose from 2D-3D point correspondences (2D-3D点对应求位姿) ,455~456
threshold: above (阈值化: 上), 83
below (下), 83
inside (内), 83
outside (外), 83
threshold values (阈值), 24
thresholding:automatic (阈值化: 自动), 85~89
dynamic (动态), 89
gray-scale images (灰度图像) ,83~89
knowledge-based (基于知识), 89
knowledge-directed (面向知识), 285
TIF format (TIF格式) , 参见 TIFF
TIFF (Tag Image File Format) (标记图像文件格式), 38
tilt,definition (倾斜角,定义), 389
tip_spacing (圆盘形结构元, 直径等于齿尖轮廓的直径) ,68,71
tracking, integrated (跟踪, 集成) ,271~272
training images, basis images for (训练图像, 基图像), 518~519
trajectories: aggregating motion trajectories (轨迹, 运动轨迹聚类) ,321~324
computing (计算), 265~272
trajectory of i, definition (i的轨迹, 定义), 267
transformation (s) :2D (变换: 2D), 327
3D affine (3D仿射), 338~341,413~421

alignment via transformation calculus（基于变换的比对），419~421
computing Tr={RT}（计算Tr={RT}），458~459
linear（线性），329~340
from model features to image features using local-feature-focus method（用局部特征焦点法寻找从模型特征到图像特征的变换），343
perspective transformation matrix（透视变换矩阵），423~426
using pose clustering（利用姿态聚类），344
rigid（刚性），331~332
transform（s）:Fourier（变换：傅里叶），177,179~181
orthogonal and orthormal（正交和标准正交），331~332
translation:of binary images, definition（平移，二值图像，定义），66
parameters for camera position（摄像机位置参数），445~449
2D（二维），332~334
3D（三维），415
tree indexes:B+（树索引：B+），245~247
K-d（K-d树），247
R-（R-树），247-48参见triangle-tree
trees参见binary decision trees: decision trees; definition trees; interpretation trees; octrees; quadtrees; tree indexes; triangle-tree
triangle-tree（三叉树），248
triangles,aligning（三角形，比对），419~421
triangulation（三角测量），48
TRIBORS object recognition system（TRIBORS目标识别系统），499~501
trichromatic encoding（三基色编码），191~193
triplets（三元组），499~501
true 3D models, versus view-class models（实际3D模型，视类模型），488~489
Tsai calibration method（Tsai标定方法），444~453
two-class problems（二类问题），96~97
two-dimensional（2D）images（2D图像），21
difference operators for（差分算子），144~149
motion from sequences of（从2D图像序列求运动），251~278
perceiving 3D images from（2D图像中的3D信息），371~409
3D cues in（3D线索），383~388
3D structure from（三维结构），42
types of（类型）29-31
two-dimensional（2D） models, 2D-3D alignment（2D模型，2D-3D比对），498~501
two-dimensional（2D） object recognition via relational matching（相关匹配法2D目标识别），350~364
two-dimensional（2D）picture functions（二维图像函数），175~179
two-dimensions（2D）transformation,definition（二维变换，定义），327
two-dimensions（2D）：matching in（二维：匹配），326~370
pose from 2-D and 3D point correspondences（2D-3D点对应求位姿），455~456
registration of data（数据配准），326~328

U

union-find algorithms（并查算法），59
union-find structure（并查结构），59~60
union procedure（合并过程），59~60

V

vanishing point（s）（消隐点），42,392
VE（visual environment）（虚拟环境），535,538,539
vector space（向量空间），160, 162
of all signals（信号），158~160
definitions（定义），159
vector（s）:feature（向量：特征），100
motion（运动），254~265,321~324
motion, deriving for interesting points（运动，计算兴趣点的运动向量），260
texture description（纹理描述），233~235
Veggie Vision（Veggie Vision系统），548~554
application domain and requirements（应用场合和要求），549~550
computing features（计算特征），552~553
hardware components（硬件组成），550
identification procedure（识别过程），551
obtaining images of produce（获取商品图像），551~552
performance（性能分析），554
representation and recognition（表示与识别），550~551
supervised learning on（监督学习），553~554
system design（系统设计），550~551
verification: definition（验证：定义），93
and optimization of pose（位姿最优化），460
vertex,surface-edge-vertex models（顶点，表面-边-顶点模型），480~483
video: cameras（视频：摄像机），26
detecting significant changes in（检测视频显著变化），272~277
MPEG（Motion Picture Experts Group）compression

of (MPEG压缩) 261~262
MPEG format for (MPEG格式), 39~40
segmenting sequences (序列分割), 273~274
storing sequences of (存储视频子序列), 277
view-based texturing (基于视图的纹理化), 543~545
view-class models:relational (视类模型: 关系), 506~513
versus true 3D models (实际3D模型), 488~489
view classes (视类), 488
virtual lines: cues from (虚拟直线: 线索), 393~394
definition (定义), 394
virtual reality:definition (虚拟现实: 定义), 535
devices (设备), 535~539
fishtank (鱼缸), 539
virtual reality (VR) systems (虚拟现实系统) 527~547
applications (应用), 529~530
architectural walkthrough (建筑漫游), 529
augmented reality (增强现实), 530~532
dextrous virtual work (虚拟灵巧手术), 537~538
features (特征), 528~529
flight simulation (飞行仿真), 529
haptic sense and (触觉), 540
HCI and psychological issues (人机交互和心理问题), 546
head-mounted displays (HMDs) (头戴式显示器), 530,535~537
interactive segmentation of anatomical structure (解剖组织的交互式分割), 529
motion in (运动觉), 540
sensing devices (感知设备), 539~540
stereoscopic display devices (立体显示设备), 538~539
teleoperation (遥操作), 533~535
visual output (视觉输出) ,539
vision,stereo (视觉, 立体视觉), 397~403
visual environment (VE) (虚拟环境), 535,538,539
visual event (视觉事件), 489
visual output, virtual reality (VR) systems (视觉输出, 虚拟现实系统), 539
Voronoi polygons (Voronoi多边形), 214~215
voxels (体素), 485
VR (virtual reality) systems (虚拟现实系统), 527~547

W

warp, affine (变形, 仿射), 334~335
warping: images (变形: 图像), 12
nonlinear (非线性), 364~368
wavelets (小波), 182
weak perspective projections (弱透视投影仪), 426~428
weights (权) ,54
white light,definition (白光, 定义), 189
wipe,definition (擦除, 定义), 272~273
wire-frame models (线框模型), 480
within-group variance (组内方差), 86~88
world coordinate frame W (世界坐标系W), 44
warp-around (逆变), 28

X

X-ray devices (X射线设备), 46~47

Y

YIQ, encoding for television signals (YIQ, 电视信号编码), 197~198
YUV:conversion from RGB (red-blue-green) to, 197
(YUV: RGB (红-绿-蓝) 到YUV的转换), 197
encoding for television signals (电视信号编码), 197~198

Z

z-buffer (Z缓存) ,542
zero crossings (零交叉) 143,144,154,155
zoom,camera,definition (变焦, 摄像机, 定义), 272
zooming (变焦), 254~255, 274~275