



毕业设计（论文）

中文题目：图像特征点及特征区域

的提取算法及应用

英文题目：Study and Application of Extracting Algorithms

for the Feature Points and Regions of Image

学院：计算机与信息技术学院

专业：计算机科学与技术

学生姓名：和睿

学号：07281125

指导教师：孙永奇

2011 年 06 月 2 日

中文摘要

图像中的特征点是指数字图像中局部特征较明显的像素点。图像特征区域可以轮廓线或边缘线的形式刻画图像中的主要物体。以特征点或特征区域的形式描述图像特征简单、直观且无需利用图像中所有像素点的信息。近年来，基于特征点及特征区域的算法已广泛应用于图像处理、图像配准及模式识别等领域。其中，图像拼接技术和图像检索技术就是基于这些特征提取算法的典型应用。图像特征点和特征区域提取算法及其应用的研究，具有理论意义和实用价值。

本文首先阐述了 Harris 特征点、SIFT 特征点及 Canny 特征区域提取算法的基本原理。分析了算法的详细实施步骤，并研究了具体实现时要解决的一些细节问题。最后，在 MATLAB7.0 下实现了这三个算法并进行了测试实验。实验结果表明与 Harris 算法相比 SIFT 算法具有尺度不变性，但是其计算量较大，实时性较差，这为本文进一步研究基于这些算法的图像拼接和检索方法奠定了理论与技术上的基础。

在完成 Harris 特征点提取算法的基础上，本文将其应用于图像的特征点匹配，实现了图像拼接算法。根据这个算法，可以将两幅有重叠区域的较小图像拼接成一幅大图像。另外，本文还将 Canny 特征区域提取算法应用于图像检索方法中。提出了一种基于 Canny 特征区域和相似度矩阵匹配的基于图像内容的检索算法。并以手枪图像的检索为例，对算法进行了实验测试。实验结果表明，该图像检索算法具有较高的查准率、查全率和较低的误检率。

关键词：Harris 特征；SIFT 特征；Canny 特征；图像拼接；图像检索

Abstract

The feature points in image are often characterized as those pixels with eminent local feature. The feature region can be interpreted as the region encircled by the outlines of the objects in an image. It is convenient and audio-visual to indicate the feature in image by the feature points and the feature region instead of all pixel information. Recently, the algorithm based on the feature points and feature region has been widely applied to the technology of image processing, image registration and pattern recognition. Among these applications, the technologies of image registration and image retrieval are the most typical application based on these algorithms. Algorithms for extracting image feature points and region as well as their applications have both theoretical and applicable value.

The thesis first introduces the basic principle of algorithms for extracting the Harris feature points, SIFT feature points and Canny feature regions in image. The procedures for implementing these algorithms in detail are analyzed in the thesis. Based on MATLAB7.0, the algorithms are implemented and their performances are tested by experiment. The experimental results demonstrate that, compared with Harris feature points, SIFT feature points in an image is invariant for scale change, but the execution of the SIFT algorithm takes great time cost, and it's not applicable for real-time operation. Thus, the study lays further foundation for the theoretical and technical basis for image registration and image retrieval based on the algorithms.

By applying the extracting feature points of Harris algorithm to the

correspondence of features in image, the algorithm for image registration that combines two images into a broadened one is implemented. And what's more, a content-based image retrieval method based on the algorithm for extracting Canny feature region and the algorithm for image correspondence by similarity of image matrix is designed, and the method to the retrieval of gun images is applied. The experimental results testified the great performance of this method in terms of a high recall ratio, precision ratio and low false alarm ratio in the retrieval.

Key words: Harris feature; SIFT feature; Canny feature; Image registration; Image retrieval

目 录

第 1 章 绪论.....	1
1.1 选题背景和意义.....	1
1.2 国内外研究现状.....	5
1.3 本文的主要工作.....	7
1.4 论文内容组织.....	7
第 2 章 图像中特征点及特征区域的提取.....	9
2.1 概述.....	9
2.2 Harris 特征点提取算法原理.....	9
2.3 SIFT 特征点提取算法原理.....	10
2.3.1 高斯金字塔及 Dog 差分金字塔.....	11
2.3.2 Dog 金字塔中的极值点.....	12
2.3.3 极值点的精确定位与筛选.....	13
2.3.4 边缘响应点的筛除.....	15
2.3.5 特征点方向参数的确定.....	15
2.3.6 特征点描述符的生成.....	16
2.4 Canny 特征区域提取算法原理.....	17
2.4.1 图像梯度及幅值的求取.....	18
2.4.2 梯度非极大值抑制.....	18
2.4.3 双阈值提取边缘点.....	19
2.5 小结.....	20
第 3 章 特征点及特征区域提取算法的实现.....	21
3.1 概述.....	21
3.2 实验环境.....	21
3.3 Harris 特征点提取算法的实现.....	23
3.3.1 Harris 算法实现步骤.....	23
3.3.2 实验测试及分析.....	26
3.4 SIFT 特征点提取算法的实现.....	27
3.4.1 SIFT 算法实现步骤.....	27
3.4.2 实验测试及分析.....	32
3.5 Canny 特征区域提取算法的实现.....	33
3.5.1 Canny 算法实现步骤.....	34
3.5.2 实验测试及分析.....	36
3.6 算法测试界面的设计.....	36
3.7 小结.....	38

第 4 章	Harris 特征点在图像拼接中的应用	38
4.1	概述.....	38
4.2	拼接算法实现.....	39
4.2.1	Harris 特征点匹配.....	40
4.2.2	图像的几何变换.....	41
4.2.3	接缝融合.....	42
4.3	实验测试及分析.....	42
第 5 章	Canny 特征在图像检索中的应用	45
5.1	问题引入.....	45
5.2	基于矩阵相似度的图像检索算法.....	46
5.2.1	算法的实现.....	46
5.3	实验测试与结果分析.....	50
5.3.1	基于原始图像样本的图像检索	50
5.3.2	基于图像 Canny 特征区域的图像检索.....	52
5.4	小结.....	54
第 6 章	总结与展望.....	56
6.1	工作总结.....	56
6.2	进一步研究展望.....	57
参考文献	59

第 1 章 绪论

1.1 选题背景和意义

数字图像(Digital Image)^[1]又称数码图像或数位图像,是二维图像用有限数字数值像素的表示。每个图像的像素通常对应于二维空间中一个特定的“位置”,并且有一个或者多个与那个点相关的采样值组成该点的像素值。根据采样数目及特性的不同,数字图像可以大致划分为二值图像(Binary Image)、灰度图像(Gray Scale Image)、彩色图像 (Color Image)及三维图像 (3D Image)等。其中,灰度图像和彩色图像是应用较为广泛的图像表示方法。灰度图像可形象的理解为一个以像素点为元素的一维矩阵,矩阵中的每个像素可以由 0(黑)到 255(白)的亮度值表示,0-255 之间表示不同的灰度级。彩色图像由由红、绿、蓝三幅不同颜色的灰度图像组合而成,因此可理解为一个三维矩阵。每个矩阵中的元素分别在 0 至 255 间取值,最后的像素颜色由对应的三个取值所决定。基于这些特性,彩色图像又称为 RGB 图像。

在数字图像中,图像特征是表征一个图像最基本的属性或特征,这种特征可以是人类视觉能够识别的自然特征;也可以是人为定义的某些特征。常用的图像特征可大致分为:颜色特征、纹理特征、形状特征、点特征、空间关系特征等,例如较直观的特征有:图像的色调、图像中物体的个数及物体形状类型、图像的尺寸或扭曲程度等。特征的精确定义往往由具体的应用决定。在这些应用中,特征是一个数字图像中“有趣”的部分,是图像分析算法的起点。

图像特征提取^[2]是计算机视觉和图像处理中的一个概念。它指的是使

用计算机提取图像信息，决定图像中的每个像素点是否属于一个图像特征。特征提取的结果是把图像上的点分为不同的子集，这些子集往往属于孤立的点、连续的曲线或者连续的区域。图像特征的提取素来是计算机视觉和图像处理的研究热点。高效的图像特征提取方法可在相对较低的时间和空间代价下提取图像中具有“可重复性”的有效特征，即对同一场景的不同图像所提取的特征应该是相同的。同时，这些特征的提取将不随图像的移动、旋转或形变发生改变，对噪声或光照变化也具有较强的稳定性。

图像中的特征点是指数字图像中局部特征较明显的像素点，如：沿该点向水平或垂直方向移动时像素点灰度变化明显。图像特征区域可以轮廓线或边缘线的形式刻画图像中的主要物体。以特征点或特征区域的形式描述图像特征简单直观且无需利用图像中所有像素点的信息。更重要的，点特征或区域特征的提取容易保证“可重复性”并且对外界干扰的鲁棒性也较强。近年来，基于特征点及特征区域的算法已广泛应用于图像处理、图像配准及模式识别等领域。其中，图像拼接技术和图像检索技术就是基于这类特征提取算法的典型应用。

(1) 图像拼接技术

图像拼接技术^[3]就是将数张有重叠区域的图像拼接成一幅大型的无缝高分辨率图像的技术。作为计算机视觉、图像处理和计算机图形学的研究热点，图像拼接技术近年来已广泛应用于虚拟现实领域、医学图像处理领域、遥感技术领域、航空航天领域(如图 1.1)和军事领域。图像特征匹配和图像融合^[2]是图像拼接的两个关键技术。图像特征匹配是图像融合的基础,而且图像特征匹配算法的计算量一般非常大,因此图像拼接技术的发展很大程度上取决于图像特征匹配技术的创新。基于特征点的图像特征匹配方法是目前图像特征匹配技术中应用较多的方法。这类算法无需要利用图像中所有像素点的信息,仅需提取若干特征明显的点以实现匹配从而提

高了算法的效率并减少了算法的运算量。稳定有效的点特征提取是特征匹配的前提，Moravec 特征点、Harris 特征点、SIFT 特征点都是目前应用广泛的点特征。完成点特征的提取后，即须对这些特征点进行匹配。特征匹配的目标是找到待拼接图像序列中最为相似的“特征点对”并基于它们完成图像的几何变换实现拼接。目前主要的匹配方法^[3]有基于像素差平方和(SSD)描述符的方法、基于互相关(CC)描述符的方法和基于归一化互相关(NCC)描述符的方法。根据匹配的结果完成图像拼接后，往往发现拼接处有较为明显的接缝。为了消除这类接缝，我们就要进行图像融合^[3]。帽子函数加权平均法、渐入渐除法、中值滤波融合法等都是高效的图像融合方法。



图 1.1 利用数据拼接制作完成的“中国第一幅全月球影像图”

(2) 图像检索技术

图像检索技术^[4]始于 20 世纪 70 年代,主要包括三个方面:对用户需求的分析和转化,形成合乎规范的检索数据库的提问;收集和加工图像资源,提取特征,分析并进行标引,建立图像的索引数据库;根据相似度算法,计算用户提问与索引数据库中记录的相似度大小,提取出满足阈值的记录

作为结果,按照相似度降序的方式输出检索结果;根据检索特征类型分类,可进一步分为基于文本的图像检索技术(Text-based Image Retrieval, 简称 TBIR)和基于内容的图像检索(Content-based Image Retrieval, 简称 CBIR)技术。基于文本的图像检索技术主要从图像名称、图像尺寸、压缩类型、作者、年代等方面标引图像,一般以关键词形式的提问查询图像,或者是根据等级目录的形式浏览查找特定类目下的图像。基于内容的图像检索以图像内颜色分布情况、色调和纹路等作为检索特征,以用户提供的目标图像为特征模板,从图像数据库中检出与特征模板相似的其他图像。查准率、查全率和误检率是衡量图像检索性能的主要指标^[5],以下为这三个指标的定义。

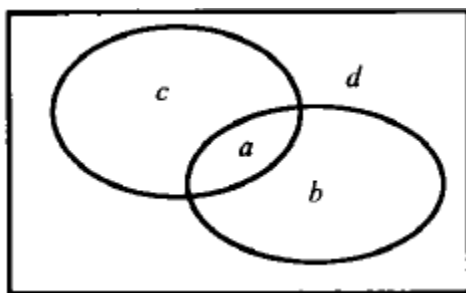


图 1.2 图像检索性能指标的定义

在图 1.2 中, 令 $a + b + c + d = Q$, $a + c = A$, $a + b = B$, 则 Q 为整个图像数据库; A 代表与目标图像相关或内容相符的图像集合, B 代表检索出的图像集合。则查全率 R 、查准率 P 和误检率 F 的定义分别如下:

$$R = p(B | A) = \frac{p(A \cup B)}{p(A)} = \frac{a}{a + c}$$

$$P = p(A | B) = \frac{p(A \cup B)}{p(B)} = \frac{a}{a + b}$$

$$F = \frac{b}{b + d}$$

因此，高性能的图像检索方法在检索图像时具有较高的查准率、查全率和较低的误检率。

1.2 国内外研究现状

图像拼接技术可以用来建立大视角的高分辨率图像，其质量主要依赖于图像特征的匹配精度，基于特征点进行匹配的方法是目目前图像特征匹配技术中应用最广泛的方法。Moravec^[3]于 1977 年提出了利用灰度方差提取点特征的算法^[1]。基于该算法提取特征点简单快捷，计算量小。但是，该算法对噪声影响十分敏感。于 1988 年提出的 Harris 算法^[6]可有效检测图像中的角点。该算法对 Moravec 算法进行了改进，以一阶偏导数来描述亮度变化，在图像旋转、灰度变化和噪声影响的状况下具有较强的稳定性。然而，当对图像的尺度进行缩放时，约有 50%的特征检测不到。1999 年 British Columbia 大学大卫·劳伊（David G. Lowe）教授总结了现有的基于不变量技术的特征检测方法，并正式提出了一种基于尺度空间的、对图像缩放、旋转甚至仿射变换保持不变性的图像局部特征描述算法—SIFT^[7](Scale-invariant feature transform, 尺度不变特征转换)，这种算法在 2004 年被加以完善。在国内，图像拼接技术的发展也很迅速。1997 年，王小睿等分别利用序贯相似度检测(SSDA)和归一化积相关两种相似度度量方法^[8]，建立模板图像和输入图像之间的相似性度量值，再利用模拟退火算法随机寻优，快速准确地找到最佳匹配点，从而实现图像的拼接。2004 年，赵向阳、杜立民提出了一种基于特征点匹配的图像自动拼接算法^[9]，其中使用了 Harris 算法提取特征点并进行匹配。该算法采用了鲁棒变换估计技术，在一定程度上提高了匹配算法的稳健性，但是计算速度比较慢。

图像检索技术是多媒体应用的关键技术，其目的就是为了解决图像数据库与用户需求的匹配和选择问题。20 世纪 70 年代有关图像检索的研究

主要借用了文本索引技术，即基于文本的图像检索技术(TBIR)。查询操作是基于图像的文本描述，检索方法是对图像文件建立关键词或文本标题等描述信息，然后将图像的存储路径和图像描述信息之间建立联系。但是，这种方法有其不便之处：一是必须由人工完整地标注所有图像，对于小图像集合问题不大，但对于无法计数的图像，用这种方法显然不可行；二是图像所包含的信息量极其丰富，不同用户对同一张图像的描述不尽相同，导致对图像的标注没有统一标准。90 年代以后，出现了基于内容的图像检索技术(CBIR)^[10]：事先抽取图像数据库中所有文件的特征，用户检索的过程是提供一个目标图像，系统抽取该目标图像的特征作为特征模板，然后同数据库中所有图像的特征进行比较，并将与特征模版相似的图像返回给用户。目前国外已有不少基于内容的图像检索系统。由 Virage 公司开发的基于内容的图像搜索引擎支持基于颜色、颜色布局、纹理和结构的可视化查询，甚至支持上述四种查询方式中任意组合形式的查询。由 Excalibu 技术公司开发的基于内容的图像检索的工具可利用纹理、颜色结构和纵横比等作为检索特征。由哥伦比亚大学开发的 VisualSEEK 和 WebSEEK 是一对用于图像检索的“姊妹系统”。VisualSEEK 用于搜索视觉特征，WebSEEK 用于搜索万维网中的文本和图像信息。该系统以颜色集和基于小波变换的纹理特征作为检索特征，同时为加快检索速度，采用了基于二叉树的检索算法。在国内，清华大学曾做过关于敦煌壁画的图像检索研究，用壁画的主色调来判断其绘制的年代，并且开发了一个基于内容的图像系统 IngRetr。该系统根据纹理、颜色直方图、和主题词等对图像的内容进行描述，采用相似索引技术加快检索速度。中国科学院计算技术研究所研究的多媒体信息检索系统 MIRES 也可在 Internet 上按内容对图像信息或文本信息进行检索。然而，上述许多基于内容的图像检索方法对类似枪支图像等一系列色彩单一、灰度变化不明显的图像的检索性能往

往不尽人意。图像中的轮廓线、边缘线等特征有必要成为图像检索算法设计过程中的考虑因素。

1.3 本文的主要工作

本文首先阐述了图像中的 Harris 特征点、SIFT 特征点和 Canny 特征区域提取算法的一般原理。结合 MATLAB 7.0^[11]下的相关程序代码，分析了这三种算法的具体实现方法。然后，根据测试实验的结果总结了这三种算法的性能特点。根据算法的这些特点，讨论了将特征点提取算法应用到图像拼接时应考虑的因素并指出了将 Canny 算法^[12]应用到图像检索技术中的优势与可行性。

然后，本文分析了基于 Harris 特征点的图像拼接算法的具体实现。根据在 MATLAB 下的测试实验，分析了该算法的性能。

最后，结合矩阵相似度算法^[13]，本文详细阐述了基于 Canny 特征区域完成图像检索的具体方法。并根据在 MATLAB7.0 下的对比实验，证明了该方法在手枪图像检索中的良好性能。

1.4 论文内容组织

本文共分为六章，各章内容如下：

第 1 章，绪论。介绍了特征点及特征区域提取算法提出的背景和应用，以及目前国内外这些应用技术的研究现状。

第 2 章，特征点及特征区域提取算法相关原理。主要阐述了图像中 Harris 特征点、SIFT 特征点及 Canny 特征区域的提取算法的基本原理。

第 3 章，特征点及特征区域提取算法的实现。结合 MATLAB 下的相关代码，详细分析了 Harris、SIFT 特征点及 Canny 特征区域提取算法的

实现步骤并通过实验测试比较总结了三种算法的性能。

第 4 章，Harris 特征点在图像拼接中的应用。阐述了基于 Harris 特征点完成图像特征匹配继而实现图像拼接的实现步骤，给出了 MATLAB7.0 下该方法的测试实例，并分析了测试结果。

第 5 章，Canny 特征在图像检索中的应用。提出并阐述了基于 Canny 特征区域的枪支图像检索方法的实现原理及步骤。最后，以手枪图像的检索为例，通过在 MATLAB7.0 下的实验测试，证明了该方法在手枪图像检索中的良好性能。

第 6 章，总结与展望。总结论文工作，并展望下一步研究工作。

第 2 章 图像中特征点及特征区域的提取

2.1 概述

图像中的特征点是指数字图像中局部特征较明显的像素点。图像特征区域是以轮廓线或边缘线的形式刻画图像中的主要物体的方法。以特征点或特征区域的形式描述图像特征简单、直观且无需利用图像中所有像素点的信息。

目前，应用较为广泛的特征点提取算法是 Harris 特征点提取算法和 SIFT 特征点提取算法。Harris 特征点提取算法能够有效提取图像中的“角点”。这些特征点明显直观，但是 Harris 算法在尺度变化下较敏感。SIFT 特征点提取算法在不同的尺度空间下提取特征点，该算法稳定性强，且具有尺度不变形，但提取算法运算量较大。

Canny 算法是典型的特征区域提取算法。该算法可提取图像中主要物体的轮廓线且能有效地抑制噪声并精确确定边缘的位置。

下面将分别阐述上述算法的相关原理。

2.2 Harris 特征点提取算法原理

Harris 算法^[6]是 C. Harris 和 J. Stephens 在 1988 年提出的一种基于信号的点特征提取算法，用以提取沿水平、垂直方向移动时灰度变化都很明显的点，即角点。

Harris 算法在 Moravec 算法^[3]的基础上进行了改进，定义了任意方向上的自相关值，即一组方形区域窗口中图像灰度误差的总和，它的二阶泰勒级数展开式近似为：

$$E(u, v) = [u \ v] \begin{bmatrix} u \\ v \end{bmatrix} \quad (2-1)$$

(u, v) 为图像中的任意像素点， M 是一个 2×2 的对称矩阵：

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} = e^{-\frac{x^2+y^2}{2\sigma^2}} \otimes \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = w(x, y) \otimes \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (2-2)$$

$w(x, y)$ 为高斯滤波算子。 I_x, I_y 分别为图像在 x, y 方向上的梯度值。

设 λ_1 和 λ_2 为矩阵 M 的两个特征值。通过对特征值分析，可得出三种情况：

(1) 如果两个特征值都很小，意味着窗口所处区域灰度变化很小，沿任意方向移动时，函数 E 都发生很小的改变。

(2) 如果一个特征值很大，而另一个特征值很小，意味着窗口所处灰度值呈现屋脊状，例如边缘。此时，沿着边缘方向移动 E 的变化很小，而垂直边缘移动时 E 的变化较大。

(3) 如果两个特征值都很大，表明窗口所处区域灰度值呈现为尖峰状，如角点。此时，沿任意方向移动 E 的变化都急剧增大。

从性能上看 Harris 算法计算简单，能在纹理特征丰富的图像中提取出大量的特征点且对图像的旋转变化、亮度变化、视角变化和噪声影响具有较强的鲁棒性。

2.3 SIFT 特征点提取算法原理

1999 年 British Columbia 大学的 David G. Lowe 总结了现有的基于不变量技术的特征检测方法，并提出了一种基于尺度空间的、对图像缩放、旋转甚至仿射变换保持不变性的图像局部特征描述算法 SIFT^[7]。该算法具有以下特点：

(1) 尺度不变性。SIFT 特征是图像的局部特征，其对旋转、尺度

缩放、亮度变化保持不变性，对视角变化、仿射变换、噪声也保持一定程度的稳定性；

(2) 独特性 (Distinctiveness)。信息量丰富，适用于在海量特征数据库中快速、准确的匹配；

(3) 多量性。即使少数的几个物体也可以产生大量的 SIFT 特征向量；

(4) 可扩展性。可以很方便的与其他形式的特征向量进行联合。

Lowe 在图像二维平面空间和 Dog(Difference-of-Gaussian)尺度空间中同时检测局部极值以作为特征点，以使特征具备良好的独特性和稳定性。Dog 算子定义为经两个模糊系数 σ 取值不同的高斯滤波算子滤波后的图像的差分，它具有计算简单的特点，是归一化 Log (Laplacian-of-Gaussian)算子的近似。下面详细阐述 SIFT 特征点提取算法的原理。

2.3.1 高斯金字塔及 Dog 差分金字塔

假设原始灰度图像为 $I(x, y)$ ，对 $I(x, y)$ 进行高斯滤波公式为：

$$L(x, y, \sigma) = e^{-\frac{x^2+y^2}{2\sigma^2}} \otimes I(x, y) \quad (2-3)$$

首先对原始灰度图像进行 5 次高斯滤波，每一次滤波时滤波因子 σ 的值是上一次的 k 倍，初始 σ 值可取为 1.52。至此，第一阶图像产生。接着对刚才那一阶的第三层图像进行降采样，以得到下一阶的第一张图像，并仿照第一阶图像的生成方法继续进行滤波，此过程往复进行直至生成 N 阶每阶含 5 张图像的“图像金字塔”，如图 2.1 所示。 k 值可取为 1.4， N 值可取为 $\lceil \log_2 \min(h, w) \rceil - 3$ 。 H 和 w 分别代表原始图像以像素点为单位下的高度和宽度。注意：Lowe 论文中的“尺度”概念即每一步滤波当中 σ 的取

值。之所以要对每阶第三层图像进行降采样是因为随着尺度的逐次增大，图像所包含的梯度信息逐渐减少，即相邻像素间的差异越来越小。这样就会导致存在大量冗余信息，为了减少不必要的计算，我们须进行降采样处理。

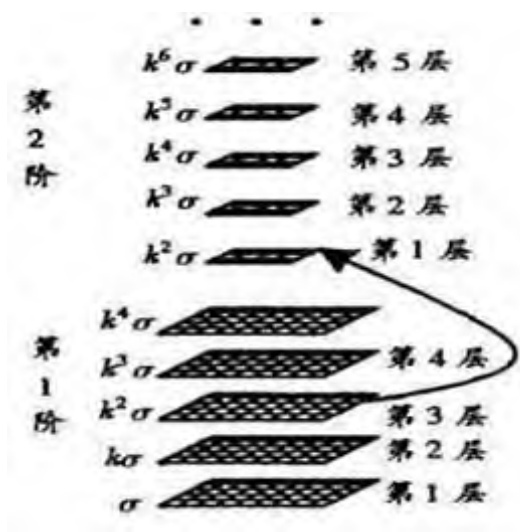


图 2.1 高斯金字塔的构建过

接着，将每一阶内相邻两张图像的对应像素值相减得到 Dog 差分金字塔,如图 2.2。Dog 函数可表述为：

$$D(x,y,\sigma) = L(x,y,k\sigma) - L(x,y,\sigma) \tag{2-4}$$

2.3.2 Dog 金字塔中的极值点

为了寻找 Dog 函数的极值点，Dog 金字塔中每阶除底层和顶层外的图像(图 2.2 中标有五角星的层)中的每一个像素点都要和它所有的相邻点比较，以确保在尺度空间和二维图像空间均检测到局部极值点。如图 2.3，

标记为叉号的像素若比其周围的 26 个像素点都大或都小，则认为该点是一个局部极值点，并记下它在所处二维图像空间中的坐标和对应尺度。

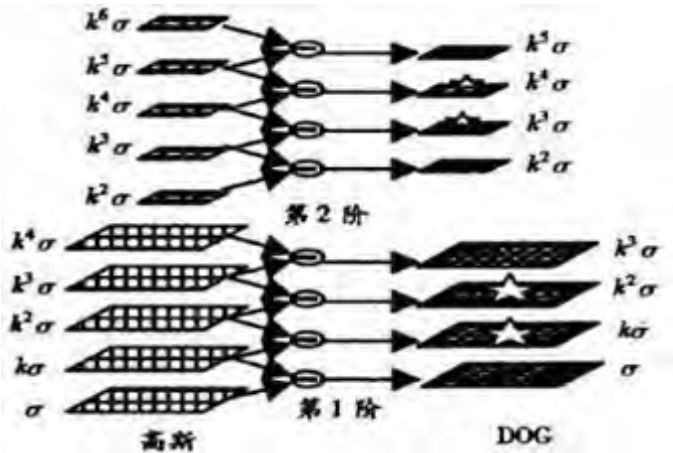


图 2.2 Dog 金字塔的构建过程

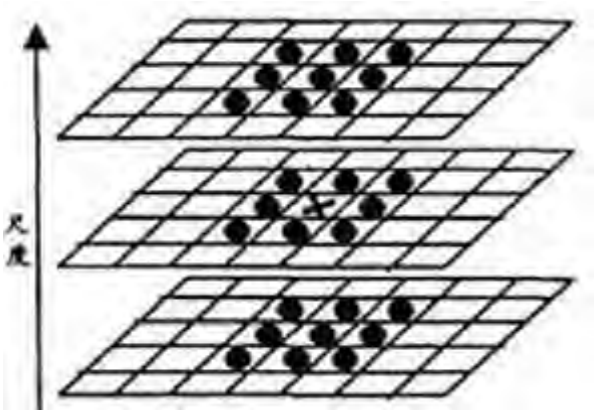


图 2.3 Dog 金字塔中极值点的提取

2.3.3 极值点的精确定位与筛选

在 Dog 尺度空间中检测到局部极值点还要经过进一步检验才能精确定位其位置。为了提高极值点的稳定性，需要对 Dog 函数进行曲线拟合。

定义：一个像素点的位置向量为 $\alpha = \begin{bmatrix} x \\ y \\ \sigma \end{bmatrix}$ ，其中 x 和 y 分别为像素点在所

处二维图像空间中的横坐标和纵坐标， σ 为对应的尺度。

利用 Dog 函数 Taylor 展开式：

$$D(\alpha) = D(\alpha_0) + \left(\frac{\partial D}{\partial \alpha} \Big|_{\alpha=\alpha_0}\right)^T (\alpha - \alpha_0) + \frac{1}{2} (\alpha - \alpha_0)^T \left(\frac{\partial^2 D}{\partial \alpha^2} \Big|_{\alpha=\alpha_0}\right)^T (\alpha - \alpha_0) \quad (2-5)$$

其中， α_0 为之前检测到的极值点的位置向量。

$$\frac{\partial D}{\partial \alpha} = \begin{bmatrix} \frac{\partial D}{\partial x} \\ \frac{\partial D}{\partial y} \\ \frac{\partial D}{\partial \sigma} \end{bmatrix} \quad \frac{\partial^2 D}{\partial \alpha^2} = \begin{bmatrix} \frac{\partial^2 D}{\partial x^2} & \frac{\partial^2 D}{\partial xy} & \frac{\partial^2 D}{\partial x\sigma} \\ \frac{\partial^2 D}{\partial yx} & \frac{\partial^2 D}{\partial y^2} & \frac{\partial^2 D}{\partial y\sigma} \\ \frac{\partial^2 D}{\partial \sigma x} & \frac{\partial^2 D}{\partial \sigma y} & \frac{\partial^2 D}{\partial \sigma^2} \end{bmatrix}$$

$$\text{令 } \Delta\alpha = \alpha - \alpha_0 = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta \sigma \end{bmatrix}, \text{ 则}$$

$$D(\alpha) = D(\alpha_0) + \left(\frac{\partial D}{\partial \alpha} \Big|_{\alpha=\alpha_0}\right)^T \Delta\alpha + \frac{1}{2} \Delta\alpha^T \left(\frac{\partial^2 D}{\partial \alpha^2} \Big|_{\alpha=\alpha_0}\right)^T \Delta\alpha$$

上式对 $\Delta\alpha$ 求导，并令其为 0 以求取极值对应的 $\Delta\alpha$ ，经计算有当 $D(\alpha)$ 取得极值时

$$\Delta\alpha = - \left(\left(\frac{\partial^2 D}{\partial \alpha^2} \Big|_{\alpha=\alpha_0} \right)^T \right)^{-1} \left(\frac{\partial D}{\partial \alpha} \Big|_{\alpha=\alpha_0} \right)^T$$

如果 $\Delta\alpha$ 的 Δx 或 Δy 值大于 0.5 或小于 -0.5，则要在原 α_0 的 x 或 y 上加 1 或减 1 以进行修正。若 $\Delta\alpha$ 的 $\Delta\sigma$ 值的绝对值大于 $1.52 * \sqrt{1.4} - 1.52$ ，则要在原 α_0 上将 σ 修正为 $1.4 * \sigma$ 或 $\sigma / 1.4$ 。将新得到的 α_0 往复执行上述过程直至无需进行修正为止。至此，极值点得到精确定位。接着将此时的 $\Delta\alpha$

带入公式 2-5，得到

$$D(\alpha) = D(\alpha_0) + \frac{1}{2} \left(\frac{\partial D}{\partial \alpha} \Big|_{\alpha=\alpha_0} \right)^T \Delta \alpha$$

若函数值的绝对值大于 0.03，则该点被接受。

2.3.4 边缘响应点的筛除

仅仅去除低对比度的极值点是远远不够的。Dog 函数在图像边缘有较强的边缘响应，因此我们还需要排除边缘响应点。Dog 函数的峰值点在横跨边缘的方向有较大的主曲率，而在垂直边缘的方向有较小的主曲率。主曲率可以通过计算在该点位置的 2×2 的 Hessian 矩阵得到，矩阵中的导数可由采样点相邻差来估计：

$$H = \begin{bmatrix} \frac{\partial^2 D}{\partial x^2} & \frac{\partial^2 D}{\partial xy} \\ \frac{\partial^2 D}{\partial xy} & \frac{\partial^2 D}{\partial y^2} \end{bmatrix} \quad (2-6)$$

若 $\frac{tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}$ (通常取 r 取 10)，则保留该点作为特征点。

2.3.5 特征点方向参数的确定

用特征点邻域像素的梯度方向分布特性，可以为每个特征点指定方向参数，从而使描述子对图像旋转具有不变性。像素点 (x, y) 的梯度可表示为：

$$grad(x, y) \Rightarrow \frac{\partial L}{\partial x} \quad \frac{\partial L}{\partial y}$$

梯度幅值为：

$$m(x, y) = \sqrt{\left(\frac{L(x+1, y) - L(x-1, y)}{2} \right)^2 + \left(\frac{L(x, y+1) - L(x, y-1)}{2} \right)^2} \quad (2-7)$$

梯度方向为：

$$\theta(x, y) = \arctan\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right) \quad (2-8)$$

确定特征点的方向采用梯度直方图统计法：统计以特征点为原点，一定区域内的图像像素点对特征点方向生成所作的贡献。特征点的主方向为极值点周围区域内梯度直方图的主峰值，如图 2.4 所示。

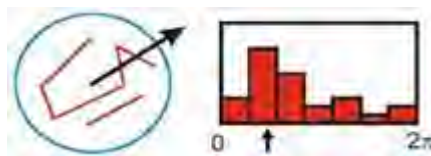


图 2.4 采用梯度直方图确定特征点方向

2.3.6 特征点描述符的生成

首先将坐标轴旋转为 SIFT 特征点的主方向，以确保旋转不变性。接下来以特征点为中心取 8×8 的窗口。如图 2.5(a) 中的中央黑点为当前特征点的位置，每个小格代表关键点邻域所在尺度空间的一个像素，箭头方向代表这些像素点的梯度方向，箭头长度代表梯度模值，图中蓝色的圈代表高斯加权的范围(越靠近关键点的像素梯度方向信息贡献越大)。然后在每 4×4 的小块上计算 8 个方向的梯度方向直方图，绘制每个梯度方向的累加值，即可形成一个种子点，如图 2.5(b)。此图是一个关键点由 2×2 共 4 个种子点组成，每个种子点有 8 个方向向量信息。这种邻域方向性信息联合的思想增强了算法抗噪声的能力，同时对于含有定位误差的特征匹配也提供了较好的容错性。

实际计算过程中，为了增强匹配的稳健性，Lowe 建议对每个关键点使用 4×4 共 16 个种子点来描述，这样对于一个关键点就可以产生 128 个数据，即最终形成 128 维的 SIFT 特征向量。此时 SIFT 特征向量已经去除

了尺度变化、旋转等几何变形因素的影响。

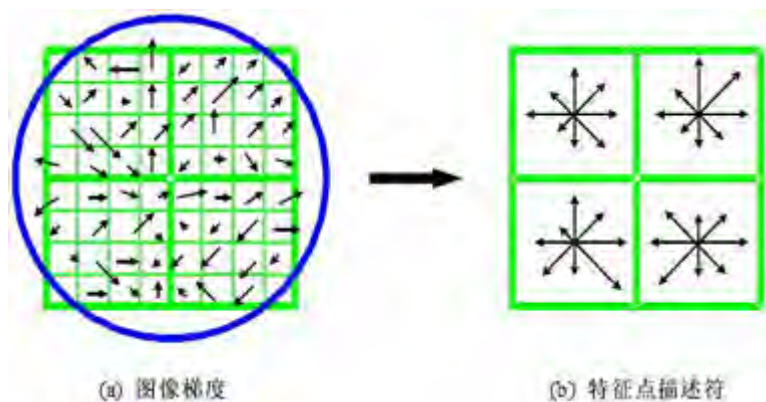


图 2.5 特征点描述符的生成

至此即可确定 SIFT 特征点的位置、尺度和方向。要计算它们在原始图像中的坐标需将其特征点的 y 和 x 值分别乘以一个尺寸变换系数 P , $P=2^K$ (K 为该点所属图像对应的阶序减 1)。

SIFT 算法克服了 Harris 算法的缺陷,但由于 SIFT 特征点需用 128 维的向量来描述,故而算法计算量较大。同时,基于 SIFT 特征点的向量进行特征匹配时,图像必须满足足够多的纹理,否则构造出的 128 维向量区别性就不是足够大,容易造成误匹配。

2.4 Canny 特征区域提取算法原理

Canny 边缘检测算法^[12]是 John F. Canny 于 1986 年开发出来的一个特征区域的提取算法。Canny 特征区域多表现为图像中物体的边缘或轮廓的形式。Canny 的目标是找到一个最优的边缘检测算法,最优边缘检测的含义是:

(1)最优检测: 算法能够尽可能多地标识出图像中的实际边缘,漏检真实边缘的概率和误检非边缘的概率都尽可能小;

(2)最优定位准则: 检测到的边缘点的位置距离实际边缘点的位置最

近, 或者是由于噪声影响引起检测出的边缘偏离物体的真实边缘的程度最小;

(3)检测点与边缘点一一对应: 算法检测的边缘点与实际边缘点应该是一一对应。

2.4.1 图像梯度及幅值的求取

对源图像进行高斯滤波, 则

$$S(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \otimes I(x, y) = w(x, y, \delta) \otimes I(x, y)$$

其中, $I(x, y)$ 为原始灰度图像, $w(x, y, \delta)$ 为高斯滤波算子。接着, 计算滤波后的图像 $S(x, y)$ 中各像素点的梯度值。可近似认为:

$$D_x(x, y) = \frac{\partial S}{\partial x} = 0.5 \times (S(x+1, y) - S(x-1, y))$$

$$D_y(x, y) = \frac{\partial S}{\partial y} = 0.5 \times (S(x, y+1) - S(x, y-1))$$

此过程相当于用模板 $H_1 = \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}$ 和 $H_2 = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}$ 对 $S(x, y)$ 分别滤波。

则 $S(x, y)$ 内每个像素点的梯度大小为:

$$M(x, y) = \sqrt{D_x^2(x, y) + D_y^2(x, y)}, \text{ 梯度方向为: } \theta(x, y) = \arctan(D_y(x, y)/D_x(x, y))。$$

2.4.2 梯度非极大值抑制

按每个像素点的梯度方向值, 可以把该值归入 $-\frac{\pi}{2} \sim \frac{\pi}{2}$ 四种方向类型中的一类, 如图 2.6 所示。

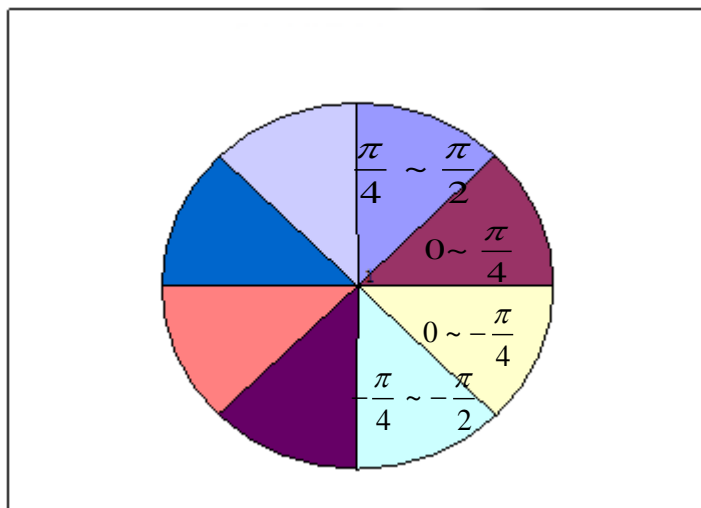


图 2.6 像素点的梯度方向范围

当像素点 $M(x, y)$ 的梯度值方向在 $-\frac{\pi}{4} \sim -\frac{\pi}{2}$ 范围内时，该点须与 $M(x, y+1)$ 和 $M(x, y-1)$ 两点比较，只有当 $M(x, y)$ 值大于这两点时，才保留该点。

同理，梯度值方向处于其他范围的像素点须与其梯度正方向、负方向上的像素点比较，以确定是否保留该点。

2.4.3 双阈值提取边缘点

双阈值算法对梯度非极大值抑制的图象作用两个阈值 τ_1 和 τ_2 ，且 $2\tau_1 \approx \tau_2$ ，从而可以得到两个阈值边缘图象 $N_1[i, j]$ 和 $N_2[i, j]$ 。由于 $N_2[i, j]$ 使用高阈值得到，因而含有很少的假边缘，但有间断(不闭合)。双阈值法要在 $N_2[i, j]$ 中把边缘连接成轮廓，当到达轮廓的端点时，该算法就在 $N_1[i, j]$ 的 8 邻点位置寻找可以连接到轮廓上的边缘，这样，算法不断地在 $N_1[i, j]$ 中收集边缘，直到将 $N_2[i, j]$ 连接起来为止。

2.5 小结

本章阐述了数字图像中的 Harris 特征点、SIFT 特征点及 Canny 特征区域提取算法的基本原理。

Harris 特征点是图像中沿水平及垂直方向移动时图像灰度变化明显的像素点，即角点。Harris 特征点提取算法快捷、简单且计算量小，对噪声影响、光照影响、旋转变换、视角变化均具有较强的鲁棒性。但是，该算法对图像尺度的变化较为敏感。SIFT 特征点提取算法以 128 维向量描述特征点，SIFT 特征点不仅对噪声、光照等具有较强的鲁棒性，同时它也克服了 Harris 算法对尺度敏感的缺点。SIFT 特征点具有尺度不变性。但是，SIFT 算法计算量大，且在匹配时的精度依赖与源图像中的纹理特点。Canny 特征区域提取算法可以方便的提取图像中以轮廓线形式描述的特征区域。该算法使得图像仅保留边缘线等有效信息，排除了色调、曝光度等因素的影响。

Harris 和 SIFT 算法各具优缺点。实际应用中，应根据实际情况，结合两种算法的性能特性来选择特征点提取算法。Canny 特征区域简化了图像信息，为方便、直观的识别图像特征创造了条件。

第 3 章 特征点及特征区域提取算法的实现

3.1 概述

基于 MATLAB 在数字图像处理方面的优势^[14]，本次工作选择 MATLAB7.0 作为实验平台，详细分析了在该环境下实现 Harris 特征点、SIFT 特征点及 Canny 特征区域提取算法的步骤并根据实验测试结果总结比较了三种算法的性能特性。最后，设计了一种基于 MATLAB GUI 的算法测试界面。

3.2 实验环境

MATLAB 是矩阵实验室(Matrix Laboratory)的简称,是美国 MathWorks 公司出品的商业数学软件。它将数值分析、矩阵计算、科学数据可视化以及非线性动态系统的建模和仿真等诸多强大功能集成在一个易于使用的视窗环境中,为科学研究、工程设计以及必须进行有效数值计算的众多科学领域提供了一种全面的解决方案,并在很大程度上摆脱了传统非交互式程序设计语言(如 C、Fortran)的编辑模式。该软件具有如下公认的优势与特性:

(1) 友好的工作平台和编程环境

MATLAB 由一些列工具组成,其中许多工具采用的是图像用户界面,包括 MATLAB 桌面和命令窗口、编辑器和调试器等。随着 MATLAB 的商业化以及软件本身的不断升级, MATLAB 的用户界面也越来越精致,更接近 Windows 的标准界面你,人机交互性更强,操作更简单。

(2) 简单易用的编程语言

MATLAB 语言是一种高级的矩阵语言，它包括控制语句、函数、数据结构、输入和输出和面向对象编程的特点。**MATLAB** 语言是基于流行的 C++ 语言的，因此语法特征与 C++ 语言极为相似，而且更加简单，更加符合科技人员对数学表达式的书写格式。

(3) 强大的科学计算机数据处理能力

MATLAB 是一个包含大量计算算法的集合，其拥有 600 多个工程中要用到的数学运算函数，可以方便地实现用户所需的各种计算功能。这些函数集涵盖了从最简单的函数到诸如矩阵运算、特征向量计算、傅里叶变换等复杂函数。

(4) 出色的图形处理功能

MATLAB 具有方便的数据可视化功能，能够将向量和矩阵用图形的形式表现出来。该软件不仅在一般数据可视化软件都具有的功能(例如二维曲线和三维曲面的绘制和处理等)方面更加完善，而且对于一些其他软件所没有的功能(例如图像的光照处理、色度处理以及四维数据的表现等)，**MATLAB** 同样表现出了出色的处理能力。同时，**MATLAB** 还着重在图形用户界面(GUI)^[14]的制作上做了很大的改善，用户可方便的实现类似 MFC 的交互式图形化界面。

此外，在数字图像处理方面，与 **OpenCv** 平台相比，在 **MATLAB** 下编写数字图像处理程序更加简单、方便。程序编写者可实时且直观的查看图像矩阵中元素值的动态变化情况并且可利用 **MATLAB** 自带的函数方便的完成图像处理并随时查看处理效果。通过该软件内自带的工具(如 **MATCOM**)，**MATLAB** 函数文件可以被方便的转化为 **cpp** 文件。近年来，已有越来越多的图像处理工作运用了 **VC++** 和 **MATLAB** 的混合编程方法。

正是基于 **MATLAB** 的这些优点，本次工作以 **MATLAB 7.0** 作为实验

平台。以下为本文实验所用主机的基本配置：

操作系统：Microsoft Windows XP Professional (5.1,版本 2600)

处理器：Intel(R) Pentium(R) Dual CPU T2390 @ 1.86GHz(2 CPUs)

内存：2046MB RAM

硬盘：120G

DirectX 版本：DirectX 9.0c (4.09.0000.0904)

3.3 Harris 特征点提取算法的实现

Harris 特征点可形象的表现为图像中的“角点”。本节将结合 MATLAB7.0 下编写的部分程序代码，详细分析 Harris 特征点提取算法的实现步骤并通过实例测试算法的性能。

3.3.1 Harris 算法实现步骤

不妨令源灰度图像为 IMG ，其宽度为 w ，高度为 h 。

步骤 1：求取 IMG 中的每个像素点 x 方向及 y 方向的方向导数。任取像素点坐标 (i, j) ，则他们在 IMG 中的灰度值为 $IMG(i, j)$ 。可近似认为：

$$\frac{\partial IMG}{\partial x}_{(a,b)} = 0.5 \times (IMG(i+1, j) - IMG(i, j) + IMG(i+1, j-1) - IMG(i, j-1))$$

$$\frac{\partial IMG}{\partial y}_{(a,b)} = 0.5 \times (IMG(i, j+1) - IMG(i, j) + IMG(i+1, j+1) - IMG(i+1, j))$$

这样分别以 $f_x = \begin{bmatrix} -0.5 & 0.5 \\ -0.5 & 0.5 \end{bmatrix}$ 和 $f_y = \begin{bmatrix} 0.5 & 0.5 \\ -0.5 & -0.5 \end{bmatrix}$ 作为滤波模版对 IMG 进

行滤波即可求得 IMG 中各像素点的方向导数，滤波后形成矩阵 I_x 和 I_y ，

以下为该步骤的相关代码。


```

if size(img, 3)>1
    img=rgb2gray(img);
end
fx = [-.5, .5 ; -.5, .5];
Ix = filter2(fx, IMG);
fy = [.5, .5 ; -.5, -.5];
Iy = filter2(fy, IMG);

```

Harris 算法可从单幅灰度图像中提取出角点。因此，若读入的图像为彩色图像(RGB)，须将其转化为灰度图像。代码中，`rgb2gray()`为图像类型转化函数，`filter2()`为滤波函数。

步骤 2：分别将 I_x 和 I_y 中各元素求平方，得矩阵 I_x^2 和 I_y^2 ，将 I_x 和 I_y 中相同坐标对应的各元素分别相乘得矩阵 I_{xy} 。分别对 I_x^2 、 I_y^2 、 I_{xy} 进行高斯滤波，即

$$I_x^2 = e^{-\frac{x^2 + y^2}{2\sigma^2}} \otimes I_x^2, \quad I_y^2 = e^{-\frac{x^2 + y^2}{2\sigma^2}} \otimes I_y^2, \quad I_{xy} = e^{-\frac{x^2 + y^2}{2\sigma^2}} \otimes I_{xy}$$

其中，“ \otimes ”为卷积算子，高斯滤波器窗口可取为 7×7 ，滤波参数 σ 可取为 1.52，以下为此步骤的相关代码。

```

Ix2 = Ix.^2;
Iy2 = Iy.^2;
Ixy = Ix.* Iy;
h= fspecial('gaussian',[7 7],1.53);
Ix2 = filter2(h,Ix2);
Iy2 = filter2(h,Iy2);
Ixy = filter2(h,Ixy);

```


代码中 `fspecial()` 为滤波模板构造函数。

步骤 3：求图像中坐标 (i, j) 处对应的矩阵 M

$$M = \begin{bmatrix} I_x^2(i, j) & I_x I_y(i, j) \\ I_x I_y(i, j) & I_y^2(i, j) \end{bmatrix}$$

当 $J = \det(M) - k \times \text{tr}(M)^2 > T$ 时，认为矩阵 M 的两个特征值均很大，则该像素点是特征点。实际算法中， k 取值介于 0.04 至 0.06 之间， T 值常取为图像范围内各像素点中 J 最大值的 0.01 倍。

在以每个特征点为中心的 3×3 窗口内进行非极大值抑制，仅当中心像素的 J 值大于此领域内其他所有像素的 J 值时，提取该中心像素作为“角点”。 以下为此步骤的相关代码。

```

%%%%%%%%%%%% 求取图像范围内 J 的最大值 Jmax %%%%%%%%%%%%%
Jmax=0;
for i = 1:h
    for j = 1:w
        M = [Ix2(i,j), Ixy(i,j); Ixy(i,j), Iy2(i,j)];
        J(i,j) = det(M)-0.06*(trace(M))^2;
        if J(i,j) > Jmax
            Jmax = J(i,j);
        end
    end
end
%%%%%%%%%%%% 非极大值抑制求取角点 %%%%%%%%%%%%%
for i = 2:h-1
    for j = 2:w-1
        if J(i,j) > 0.01* Jmax && J(i,j) > J(i-1,j-1) && J(i,j) > J(i-1,j) &&
            J(i,j) > J(i-1,j+1) && J(i,j) > J(i,j-1) && J(i,j) > J(i,j+1) &&
            J(i,j) > J(i+1,j-1) && J(i,j) > J(i+1,j) && J(i,j) > J(i+1,j+1)
            result(i,j) = 1;
        end;
    end;
end;
end;

```


程序中 `det()` 用于求取与输入矩阵对应的行列式的值；`trace()` 用于求取矩阵的“迹”；`result` 数组记录特征点的位置。

3.3.2 实验测试及分析

用上述算法对一张角点特征丰富的图像(240×180)进行测试，并利用 MATLAB 的 `tic-toc` 语句^[11]记录算法执行时间。实验结果如图 3.1。本次实



图 3.1 原图上 Harris 特征点提取算法实验结果

验中，检出 Harris 特征点 168 个，耗时 1.13 秒。将源图像缩小一倍后再次进行测试，实验结果如图 3.2 所示。图中标“+”符号处表示检测到了图像的角点。本次实验中，检出 Harris 特征点 107 个，耗时 0.71 秒。

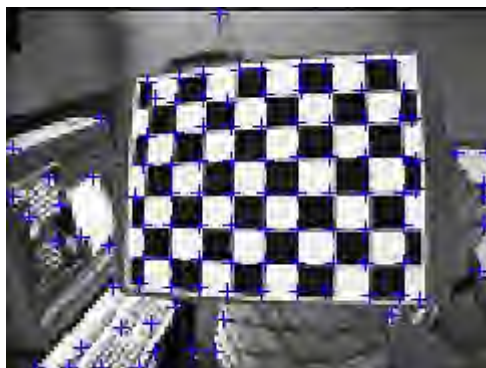


图 3.2 原图缩小后 Harris 特征点提取算法实验结果

两次实验的执行耗时均较短，说明 Harris 算法的实时性较好。但对比两次实验结果可以发现，当图像尺度缩小时，检测出的特征点数量明显减少，因此 Harris 算法对图像尺度变化较敏感。

3.4 SIFT 特征点提取算法的实现

SIFT 特征点是在图像不同尺度空间中提取出的，具有尺度不变性。本节将结合 MATLAB7.0 下编写的部分程序代码，详细分析 SIFT 特征点提取算法的实现步骤并通过实验测试算法的性能。

3.4.1 SIFT 算法实现步骤

下面用一幅校园图片(其灰度图如图 3.3 所示)为源图像来说明算法的详细实现过程。

步骤 1：构建灰度图像的高斯金字塔和 Dog 金字塔。

图 3.3 灰度图像 *IMG*

获取源图像的灰度图像 *IMG*，如图 3.3 所示。构建 *IMG* 在不同尺度空间下的高斯金字塔，高斯金字塔的阶数 N 取决于源图像的尺寸，一般取 $N = \lceil \log_2 \min(h, w) \rceil - 3$ 。金字塔的每阶具有 5 张尺度不同(对应高斯滤波函数的不同模糊参数 σ)的图像，下一阶的第一张图像由当前阶的第三张图像降采样得到。以下为此步骤的相关代码。

```
function gspyramid=buildpyramid(IMG , N, gs)
gspyramid=cell(N,5);
h=fspecial('gaussian', [7, 7], gs);
gspyramid{1,1} = imfilter(IMG,h,'conv');
basek=1;
for i=1:octavenum
    if i~=1
        IMG=dsample(IMG);
        basek=1.4^(2*(i-1));
        h=fspecial('gaussian',[7,7], gs*basek);
        gspyramid{i,1} = imfilter(IMG, h,'conv');
    end
    k=basek;
    for j=2: levelnum
        k=k*1.4;
        h=fspecial('gaussian', [7,7], gs*k);
        gspyramid{i,j}=imfilter(IMG, h, 'conv');
    end
end
end
```

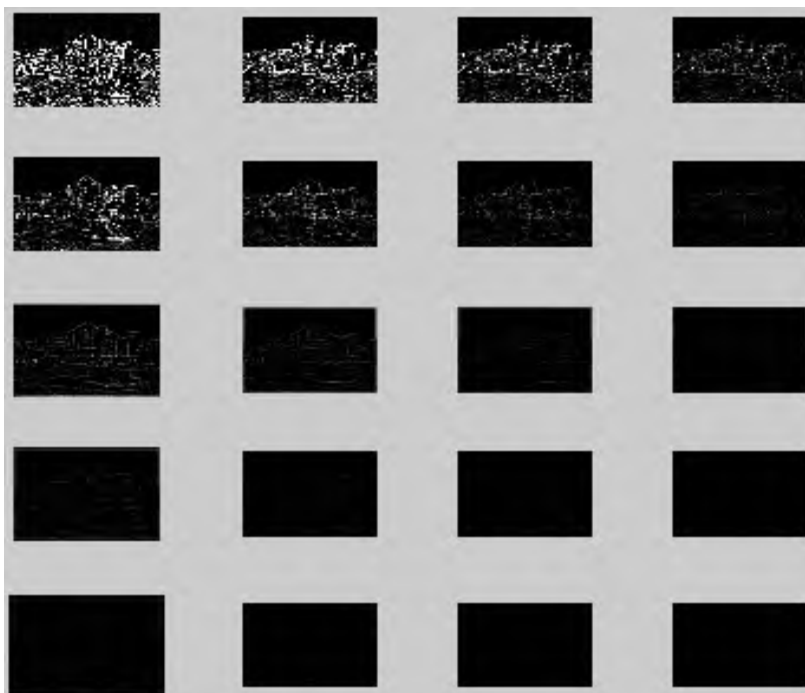

程序中 `dsample()` 为图像的降采样函数； gs 为高斯滤波函数的 σ 值，后面的实验中取 1.52。用上述算法对 *IMG* 构造高斯金字塔，如图 3.4。将高斯金字塔中相邻的两张图像对应的像素值相减可得到 Dog 金字塔，如图 3.5。

步骤 2：寻找 Dog 金字塔中的极值点。

在金字塔的每阶中，除顶层和底层外，每一层图像中的像素点都要和本层和两相邻层对应的 26 个像素点的像素值相比较，仅当该像素点的像素值最大或最小时，认为该点为极值点并记下该点的坐标值和对应的尺度。然而，这些极值点的位置并不是精确的，须采用三维二次拟合的方法修正这些极值点的坐标值及尺度值。按照 SIFT 算法原理中的方法，我们



图 3.4 灰度图像 *IMG* 的高斯金字塔

图 3.5 灰度图像 *IMG* 的 Dog 金字塔

将 Dog 函数泰勒展开并通过待修正极值点的一阶及二阶偏导数值求取它们离准确极值点位置的差距。实验中可近似认为：

$$\frac{\partial D}{\partial x} = 0.5 \times (D(x+1, y, \sigma) - D(x-1, y, \sigma))$$

$$\frac{\partial D}{\partial y} = 0.5 \times (D(x, y+1, \sigma) - D(x, y-1, \sigma))$$

$$\frac{\partial D}{\partial \sigma} = 0.5 \times (D(x, y, k\sigma) - D(x, y, \sigma/k))$$

$$\frac{\partial^2 D}{\partial x^2} = (D(x+1, y, \sigma) + D(x-1, y, \sigma) - 2 \times D(x, y, \sigma))$$

$$\frac{\partial^2 D}{\partial y^2} = (D(x, y+1, \sigma) + D(x, y-1, \sigma) - 2 \times D(x, y, \sigma))$$

$$\frac{\partial^2 D}{\partial \sigma^2} = (D(x, y, k\sigma) + D(x, y, \sigma/k) - 2 \times D(x, y, \sigma))$$

$$\frac{\partial^2 D}{\partial xy} = 0.25 \times (D(x+1, y+1, \sigma) + D(x-1, y-1, \sigma) + D(x-1, y+1, \sigma) + D(x+1, y-1, \sigma))$$

$$\frac{\partial^2 D}{\partial x \sigma} = 0.25 \times (D(x+1, y, \sigma+1) + D(x-1, y, \sigma-1) + D(x-1, y, \sigma+1) + D(x+1, y, \sigma-1))$$

$$\frac{\partial^2 D}{\partial y \sigma} = 0.25 \times (D(x, y+1, \sigma+1) + D(x, y-1, \sigma-1) + D(x, y-1, \sigma+1) + D(x, y+1, \sigma-1))$$

根据这个差距值修正原极值点并重新将其带入原泰勒展开式往复执行上述过程，直到这个差距调整到较小的范围内时，认为获得了极值点的准确位置。接着求取此极值点的 Dog 函数值，若函数值的绝对值大于 0.03 则接受该极值点。以下为此步骤的相关代码。

```
for r=1 : num
    D=max(r, 11);
    Da=zeros(3,1);
    Da(1,1)=(max(r,6)-max(r,5))*0.5;
    Da(2,1)=(max(r,8)-max(r,7))*0.5;
    Da(3,1)=(max(r,10)-max(r,9))*0.5;
    d=locate(r);
    %%%%%%%%% 根据距离值 d 修正极值点位置 %%%%%%%%%
    while (abs(d(1,1)) > 0.5) || (abs(d(2,1)) > 0.5) || (abs(d(3,1)) > 1.52 * sqrt(1.4) - 1.52)
        if d(1,1) > 0.5
            max(r,2)=max(r,2)+1;
        else if d(1,1) < - 0.5
            max(r,2) = max(r,2)-1;
        end
        if d(2,1) > 0.5
            max(r,1)=max(r,1)+1;
        else if d(2,1) < - 0.5
            max(r,1)=max(r,1)-1;
        end
        if d(3,1) > 1.52*sqrt(1.4) - 1.52
            max(r,3)= max(r,3)*gs ;
        else if d(3,1) < - (1.52*sqrt(1.4) - 1.52)
            max(r,3)= max(r,3) / gs ;
        end
        d=locate(r);
    end
end
```



```
if abs(D+0.5*Da'*dx)>0.03
    Tmax(w,:)=max(r,:);
    w=w+1;
end
end
```

程序中 num 为待修正极值点的个数, max 存储待修正极值点的位置和尺度等信息, $dx = locate()$ 用于求取待修正极值点与准确极值点间的差异, $Tmax$ 用于存储准确极值点的位置与尺度。

步骤 3: 去除边缘相应点

为去除边缘相应点, 应求取每一个极值点的 Hessian 矩阵。根据 H 是否满足 $\frac{tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}$ (通常取 r 取 10) 去除不满足此不等式的边缘相

应点。保留下来的极值点即为 SIFT 特征点, 为保证旋转不变性还应确定每个特征点的方向参数。围绕特征点选择一个窗口(矩形区域), 窗口内各采样点的梯度方向构成一个方向直方图, 根据直方图的峰值确定关键点方向。通过特征点的尺度可以计算出其所在图像对应的阶序和层序, 同时这个尺度值也决定了窗口的大小。这里为了保证不同尺度下的同一关键点方向包含相同的信息量, 窗口的大小必然不一样。一般情况下, 特征点所在图像的阶序越大(即越位于金字塔的上方), 窗口越大。

3.4.2 实验测试及分析

选取不同视角下拍摄的的照片进行测试, 第一张照片的尺度为 250×180 , 测试结果如图 3.6。本次实验中, 检出 SIFT 特征点 15 个, 耗时 12.12743 秒。第二张照片尺度为 125×90 , 比第一张照片的尺度小一倍, 测试结果如图 3.7。本次实验中, 检出 SIFT 特征点 17 个, 耗时 7.572 秒。



图 3.6 用 SIFT 算子处理第一张图片



图 3.7 用 SIFT 算子处理第二张图片

实验结果表明：两张图片重叠部分所检出的特征点位置、数量和方向基本相同。在第二张图像中，检测出的特征点数量并不因图像尺度缩小而减少。SIFT 算法能有效提取图像中的特征点并具有尺度不变性。但是算法运算量大，大约是 Harris 算法耗时的 10 倍。因此，其实时性较差。

3.5 Canny 特征区域提取算法的实现

Canny 特征区域提取算法^[12]可以提取图像中以轮廓线或边缘线的形式描述的特征区域。本节将结合 MATLAB7.0 下编写的部分程序代码，详细分析 Canny 特征区域提取算法的实现步骤并通过实例测试算法的性能。

3.5.1 Canny 算法实现步骤

步骤 1：计算梯度幅值图像

对灰度图像 *IMG*(如图 3.3)高斯滤波，并计算梯度幅值图像。以下为此步骤的相关代码，其中矩阵 *M* 和 *dir* 分别用于存储 *IMG* 中各像素的梯度幅值和梯度方向。

```
h=fspecial('gaussian',[7, 7],1.52);  
IMG=imfilter(IMG, h, 'conv');  
H1=[.5, .5; -.5, -.5];  
H2=[-.5, .5; -.5, .5];  
fai1=imfilter(IMG, H1, 'conv');  
fai2=imfilter(IMG, H2, 'conv');  
M=sqrt(fai1.^2+fai2.^2);  
dir=atan(fai1./fai2);
```

步骤 2：梯度非极大值抑制

对 *M* 中的像素值按照 *dir* 中的方向进行非极大值抑制。首先须将源图像中每个像素点的梯度方向归入四种类型，*M* 中的像素点应根据所属类型与梯度方向两侧的像素点相比较以完成非极大值抑制以下为此步骤的相关代码。

```
NMS=zeros(Isize);  
for i=2 : Isize(1) - 1  
    for j=2 : Isize(2) - 1  
        if (dir(i,j) >= -pi/2 && dir(i,j) < -pi/4) ||  
            (dir(i,j) >= pi/4 && dir(i,j) < pi/2)  
            if M(i,j) >= M(i,j-1) && M(i,j) >= M(i,j+1)  
                NMS(i,j)=M(i,j);  
            end  
        end  
    end  
end
```



```

if dir(i,j)>= -pi/4 && dir(i,j) < 0
    if M(i,j)>=M(i-1,j+1)&&M(i,j)>=M(i+1,j-1)
        NMS(i,j)=M(i,j);
    end
end
if dir(i,j)>=0 && dir(i,j) < pi/4
    if M(i,j)>=M(i-1,j-1)&&M(i,j)>=M(i+1,j+1)
        NMS(i,j)=M(i,j);
    end
end
if dir(i,j)<-pi/8 & dir(i,j)>=-3*pi/8
    if M(i,j)>=M(i-1,j-1)&&M(i,j)>=M(i+1,j+1)
        NMS(i,j)=M(i,j);
    end
end
end
end
end

```

程序中 *Isize* 记录源图像的尺寸，*NMS* 是经梯度非极大值抑制后的图像矩阵。

查看此时对源图像的处理效果，如图 3.8 所示。



图 3.8 梯度幅值图像的非极大值抑制效果

步骤 3：双阈值提取边缘点

设置两个阈值 $T_1 = 2T_2$ ，从 *NMS* 中提取出介于这两个值的点，即边缘

点。由这些边缘点组成的矩阵就是 Canny 特征区域图像。

3.5.2 实验测试及分析

实验中，用 Canny 算法提取 *IMG* 特征区域。实验结果如图 3.9。



图 3.9 灰度图像 *IMG* 中的 Canny 特征区域

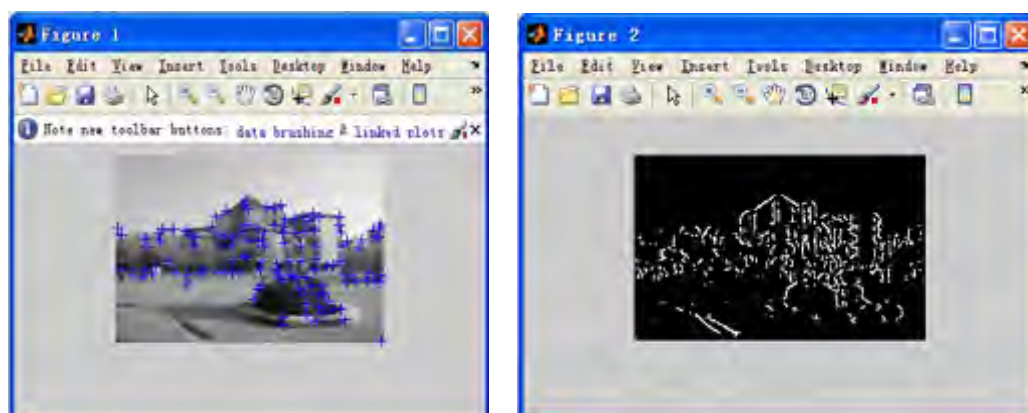
实验结果表明：Canny 算法可以有效提取目标图像中物体的特征区域。这些特征区域主要表现为物体的轮廓。用 Canny 特征描述图像特征有效规避了图像中颜色、光照强度等信息的影响。因此，在基于内容的图像检索中，可以 Canny 特征区域作为检索特征。

3.6 算法测试界面的设计

利用 MATLAB GUI 可方便的设计出基于图形界面的交互式的算法测试平台。实验中，用 MATLAB 7.0 的 GUI 编辑工具栏中的 Axes 控件显示待测图像，四个 Push Button 控件分别用作读入图像、和调用相关特征提取算法，Slider 控件用于调整 Canny 特征提取算法中的阈值，Edit Text 控件用于接收用户输入的数值作为高斯模糊系数 σ 传递给 SIFT 函数。如图 3.10 为该界面的运行效果，算法运行效果如图 3.11 所示。



图 3.10 算法测试界面



(a) Harris 算法运行效果

(b) Canny 算法运行效果

图 3.11 算法的运行效果

3.7 小结

本章主要结合 MATLAB 平台下的相关程序代码，分析了 Harris 特征点、SIFT 特征点及 Canny 特征区域提取算法的实现步骤。通过实验测试分析了三种算法的性能特点。

实验结果表明 Harris 算法可以在较短的时间内提取出图像中的大部分角点特征。但是当待测图像尺度发生变化时，提取出的特征点数量有明显的变化，即角点特征在图像尺度变化时不稳定。SIFT 算法克服了 Harris 算法的缺陷，具有尺度不变性，并通过求取每个特征点的方向参数保证了特征点的旋转不变性。但是 SIFT 算法的计算量大大超过 Harris 算法，实时性差。因此，在基于图像特征点的相关应用中，应根据实际情况，权衡两种算法的优缺点选择合适的算法。Canny 算法简单、高效，可方便的提取图像中的特征区域，输出以物体轮廓线形式描述的图像。在 Canny 轮廓图像中无多余色彩信息的干扰，可基于轮廓特征直接、直观的识别图像中的内容。正是由于 Canny 算法的这些特性，该算法的应用有必要成为图像识别、图像检索技术中的考虑因素。

第 4 章 Harris 特征点在图像拼接中的应用

4.1 概述

离散的图像序列包含了实际场景在空间和时间上的信息，单幅图像反映了实际场景在空间和时间上的局部信息。图像拼接技术可以将这些图像序列转化为大视角的高分辨率图像，大大减少了场景表示的数据量，压缩了场景的数据信息。近年来，图像拼接技术已在摄影测量学、计算机图形

学、数字图像处理、视频通信和计算机视觉等领域得到广泛的应用。其中，具体应用有虚拟场景的构建和虚拟漫游、医学图像分析、大型航空照片和卫星遥感图像的拼接、环境监测等。

图像拼接的质量主要依赖于图像特征的匹配^[15]精度，基于特征点的图像特征匹配方法是目前图像特征匹配技术中应用最广泛的方法。这类算法无需要利用图像中所有像素点的信息，仅需提取若干特征明显的点以实现匹配从而提高了算法的效率并减少了算法的运算量。

第三章的实验结果表明，Harris 算法计算量小但对图像尺度变化敏感；SIFT 特征点对图像尺度变化具有不变性，但 SIFT 特征点提取算法运算量大且在匹配时性能依赖于图像中的纹理。当基于图像中的特征点实现图像拼接时，若待拼接图像间仅仅是几何位置的变化、光照程度的变化则选择 Harris 特征点来完成特征匹配可以保证拼接的效率；若待拼接图像是在离拍摄对象远近不同的距离下拍摄的，则此时图像间存在尺度变换，选择 SIFT 特征点完成特征匹配可以保证匹配的精度从而有效完成图像拼接。下面将针对无明显尺度变化的图像，实现基于 Harris 特征点的图像拼接算法。

4.2 拼接算法实现

图像拼接算法主要包括如下四个步骤：(1)特征点的提取；(2)特征点的匹配；(3)图像的几何变换；(4)拼接接缝的融合。其中第一步的算法实现见 3.3 节。特征点的匹配，是从匹配结果中选出匹配度最高的特征点对。基于这个匹配结果进行图像间的几何变换即可实现初步的图像拼接。这时拼出的广域图像中往往存在较明显的接缝，需要对接缝进行融合处理。

4.2.1 Harris 特征点匹配

算法中采用了应用较为广泛的互相关(CC)系数匹配法进行特征点的匹配。基于互相关(CC)系数匹配法不直接利用特征点邻域内的像素值，而是基于特征点邻域像素灰度值的互相关系数为匹配原则进行匹配。首先，分别在参考图像和待匹配图像中以每个特征点为中心取大小 $(2N+1) \times (2N+1)$ 的相关窗口，然后以参考图像中的每一个特征点为参考点在待匹配图像中寻找对应的匹配点，匹配过程是基于特征点相关窗口之间的互相关系数：

$$CC = \frac{\sum_{x,y \in W} [I_1(x,y)I_2(x,y)]}{\sqrt{\sum_{x,y \in W} I_1^2(x,y)} \sqrt{\sum_{x,y \in W} I_2^2(x,y)}}$$

其中， W 是相关窗口的大小， I_1 和 I_2 分别为两张待匹配图像中特征点相关窗口内像素的灰度值， CC 是相关系数。对于每一个参考点，选取待匹配图像中与其 CC 值最大的点作为该参考点的匹配点。从所有的匹配点对中选出匹配程度最高(即 CC 值最大)的点作为两张图像的最佳匹配点对。下面为 MATLAB 下该匹配过程的相关代码。

```
max=0;
%%%%%%%%%%%%%% 特征点的匹配 %%%%%%%%%%%%%%%
for i=radius+1: size(result1,1)-radius
    for j=radius+1: size(result1,2)-radius
        if result1(i,j)==1
            temp1=sum(sum(I1(i-radius:i+radius, j-radius:j+radius)));
            temp1=sqrt(temp1);
            for ii=radius+1:size(result2,1)-radius
                for jj=radius+1:size(result2,2)-radius
                    if result2(ii,jj)==1
                        media=sum(I2(ii-radius:ii+radius,jj-radius:jj+radius))
```



```

temp2=sum(media);
I1= f(i-radius:i+radius,j-radius:j+radius);
I2= g(ii-radius:ii+radius,jj-radius:jj+radius);
C=I1.*I2;
temp3=sum(C(:));
temp2=sqrt(temp2);
CC=temp3/(temp1*temp2);
if CC>max
    max=CC;
    matchpts1= [i,j];
    matchpts2= [ii,jj];
end
end
end
end
end
end
end
end

```

代码中, *radius* 是相关窗口的半径; *max* 存储互相关系数的最大值; *result1* 和 *result2* 分别存储两张待匹配图像中的特征点; *matchpts1* 和 *matchpts2* 分别存储最佳匹配点对的坐标;

4.2.2 图像的几何变换

要实现图像的拼接必须基于最佳匹配点对, 对待拼接图像进行几何变换。假设有两张待拼接图像 S_1 和 S_2 , 如果最佳匹配点对在 S_1 和 S_2 中的坐标分别为 (x_1, y_1) 和 (x_2, y_2) , 则两图像间的水平方向坐标变换距离 $lx = x_1 - x_2$, 竖直方向坐标变换距离 $ly = y_1 - y_2$ 。若 $lx > 0$, 则将 S_2 沿水平正方向移动 lx 个像素单位; 若 $lx < 0$, 则将 S_1 沿水平正方向移动 $|lx|$ 个像素单位。若 $ly > 0$,

则将 S_2 沿竖直正方向移动 l_y 个像素单位；若 $l_y < 0$ ，则将 S_l 沿竖直正方向移动 $|l_y|$ 个像素单位。下面为本过程的相关代码。

```
lx = matchpts1(1,2) - matchpts2(1,2);
ly = matchpts1(1,1) - matchpts2(1,1);
M1= S1(1:size(S1,1),1:size(S1,2));
M2= S2(size(S2,1)-ly+1: size(S2,1)),size(S2,2)-lx+1: size(S2,2))
new=[M1,M2];
```

代码中，new 为由 S_l 和 S_2 拼接成的广域图像。

4.2.3 接缝融合

完成上述变换后即可得到一幅拼接的广域图像，但是这幅图像中往往残留着拼接时的接缝。这种情况主要是由于拍摄时阳光照度有时不均匀，造成待拼图像重叠部分中亮度的差异，而这种差异导致了接缝的出现。为消除接缝达到视觉一致性，须进行有效的图像融合。

接缝融合算法采用目前应用广泛的渐入渐出法，该方法是由 Szeliski 提出的。假如 f_1 、 f_2 是两幅待拼接图像，将图像 f_1 和 f_2 在同一二维空间进行叠加，则融合后的图像像素 f 可表示为：

$$f(x, y) = \begin{cases} f_1(x, y) & (x, y) \in f_1 \\ d_1 f_1(x, y) + d_2 f_2(x, y) & (x, y) \in f_1 \cap f_2 \\ f_2(x, y) & (x, y) \in f_2 \end{cases}$$

其中 d_1 、 d_2 表示权重值，二者取值一般与两张待拼接图像重叠区域的宽度有关，且 $d_1 + d_2 = 1$ ， $0 < d_1 < 1$ ， $0 < d_2 < 1$ 。在重叠区域中， d_1 由 1 渐变为 0， d_2 由 0 渐变为 1，由此实现了在重叠区域中由 f_1 慢慢过渡到 f_2 。

4.3 实验测试及分析

实验中用 Harris 算法提取两张有重叠区域且无明显尺度变换及旋转

变换的图像中的特征点，如图 4.1 和图 4.2。基于两图像的 Harris 特征点按照上述步骤进行拼接测试，并利用 `tic - toc` 语句记录算法执行时间。拼接效果如图 4.3。



图 4.1 待拼接图像 1 (685×500)



图 4.2 待拼接图像 2 (735×500)



图 4.3 图片的拼接效果

实验结果表明：在没有尺度变换或旋转变换的前提下，基于待拼接图像的 Harris 特征点进行匹配后再进行拼接可较精准的将待拼接图像拼接成广域图。采用渐入渐出法可有效消除拼接留下的接缝。在本次试验中，从待拼接图像 1 中提取出 1093 个特征点，提取过程耗时 8.62s；从待拼接图像 2 中提取出 823 个特征点，提取过程耗时 9.79s；特征点的匹配过程耗

时 78.25s;

第 5 章 Canny 特征在图像检索中的应用

5.1 问题引入

基于内容的图像检索方法主要通过特征提取算法提取出图像数据库中所有图像的尺寸、颜色分布等特征与检索目标的特征模板相匹配，并将匹配度介于阈值范围内的图像返还给检索者。本章将利用 Canny 特征区域检测算法实现对图像库中手枪图片的检索。

有效特征的提取是图像检索的前提。基于内容的图像检索方法大多以 RGB 图像或灰度图像的颜色分布直方图、纹理或光照明暗等作为检索特征。枪支图像的检索的难点在于：首先由于枪支本身色调单一且枪身整体的灰度变化不明显，故而系统在检索过程中很容易混淆与枪支特征模版色调、颜色分布或纹路相似的物体图像（如：黑木棍图像、树枝图像、阴影图像）；其次枪支形象本身不具有轴对称性，将枪支分别旋转 90 度、180 度或镜像翻转后，其姿态都有显著的差别。因此，如果仅提供单一姿态枪支的特征模板，则系统几乎无法辨认出其他姿态的枪支图像。正是由于枪支形象的这些特点，一般检索系统对枪支图像进行检索时，查准率和查全率均不理想，误检率也较大。

Canny 算法可有效排除图像中的色彩、明暗度、曝光程度等信息，提取图像中的特征区域，以轮廓线的形式表现图像特征。因此，若用 Canny 算法提取待检索图像的轮廓形象作为检索特征并基于此特征进行图像检索则可有效规避枪支形象以外的信息的干扰。本章将以手枪图像的检索为例，结合矩阵相似度匹配算法，提出了一种以图像 Canny 特征区域作为检索特征的图像检索方法，并通过实验测试验证了这种方法的高效性。

5.2 基于矩阵相似度的图像检索算法

在图像数据库中进行检索前，必须先形成检索目标作为对系统的“提问”。基于内容的图像检索中，这类“提问”就是目标图像的特征模板。例如，要从众多图像中检出手枪图像，须先告知系统手枪的形象特点，于是我们构造手枪图像的特征模板作为对手枪形象的描述。

特征模板可由若干张同类型图像(例如若干张不同造型手枪的图像)的检索特征构造得来，由于特征模板包含了这些图像“共有”的特征从而普遍地概括了这一类型图像的形象特点。高效的特征提取算法不但能够在较小的时间代价下求得图像数据库中各图像的特征，还能通过目标图像序列中的特征形成特征模板。基于矩阵相似度的特征提取算法计算量小，用于特征提取高效、简单。已有相关实验证明，该算法在人脸检索中具有出色的性能。因此，在本次工作中，利用该算法进行特征提取。

5.2.1 算法的实现

程永清、庄永明等于 1992 年基于代数理论推导出矩阵相似度的概念，由此可构造出了相似性判别函数^[13]。该函数可有效抽取同类图像的相似特征，这些特征包含了同类图像的共有信息。以这些特征可以构成一个向量中心作为该类型图像的特征模板。下面主要阐述该理论中的矩阵相似度、相似性判别函数以及向量间的差异。

(1) 矩阵相似度

设 $A_1, A_2, \dots, A_s (s \geq 2)$ 为 s 个 $m \times n$ 矩阵，若实数 $C(A_1, A_2, \dots, A_s)$ 定义为

$$C(A_1, A_2, \dots, A_s) = \frac{\text{tr}(\sum_{i \neq j}^s A_i A_j^T)}{(s-1)\text{tr}(\sum_{i=1}^s A_i A_i^T)} \quad (5-1)$$

其中, tr 表示矩阵的迹, T 表示矩阵的转置, $C(A_1, A_2, \dots, A_s)$ 为矩阵 A_1, A_2, \dots, A_s 的相似度。

(2) 相似性判别函数

$$\text{若 } Z(\alpha) \text{ 定义为: } Z(\alpha) = \frac{\alpha^T \left(\sum_{i \neq j}^s A_i A_j^T \right) \alpha}{\alpha^T \left(\sum_{i=1}^s A_i A_i^T \right) \alpha} \quad (5-2)$$

则称 $Z(\alpha)$ 为矩阵 A_1, A_2, \dots, A_s 相似性判别函数。对于相似性判别函数, 有性质: $|Z(\alpha)| \leq s-1$ 。设 A_1, A_2, \dots, A_s 为 s 个灰度图像, 若相似性判别函数 $Z(\alpha)$ 越大, 则表示图像 A_i ($i=1, 2, \dots, s$) 在向量 α 上的投影向量之间的差别越小, 反之, 则差别越大。

(3) 向量间的差异

若 $X = (x_1, x_2, \dots, x_n)$ 为 n 维向量, 则 $\|X\|$ 表示该向量的模值, 即

$$\|X\| = \left(\sum_{i=1}^n x_i^2 \right)^{1/2} \quad (5-3)$$

若设 b_1, b_2 为两个 n 维向量, 则 $\|b_1 - b_2\|$ 表示 b_1 与 b_2 之间的差异。如果 A_1, A_2 为两个灰度图矩阵, 则它们在向量 α 上的投影向量之间的差别可以用二者在向量 α 上的投影向量之差的模值表示。

相似性判别函数旨在找到一个使 $Z(\alpha)$ 值达到最大的向量 $\bar{\alpha}$, 即 A_1, A_2, \dots, A_s 在 $\bar{\alpha}$ 上的投影向量最为相似, 从而使得同类图像具有最小差异。寻找 $\bar{\alpha}$ 成为提取图像的特征向量和构建相应的特征模板的基础。

基于相似度矩阵的图像检索算法的具体实现步骤如下:

步骤 1: 导入若干张图像的灰度矩阵作为特征模板的构造样本集, 设为 $\{A_1, A_2, \dots, A_s\}$ 。

步骤 2: 对 $\sum_{i=1}^s A_i A_i^T$ 秩分解, 即 $\sum_{i=1}^s A_i A_i^T = P^T \Lambda P$

其中, P 为正交矩阵, $\Lambda = \text{diag}(k_1, k_2, \dots, k_n)$, k_i 是 $\sum_{i=1}^s A_i A_i^T$ 的特征根

步骤 3: 对 $\sum_{i \neq j}^s A_i A_j^T$ 进行合同变换, 即

$$B = (\bar{\Lambda} P) \left(\sum_{i \neq j}^s A_i A_j^T \right) (\bar{\Lambda} P)^T$$

这里 $\bar{\Lambda} = \text{diag}(\bar{k}_1, \bar{k}_2, \dots, \bar{k}_n)$ 定义为 $\bar{k}_i = \begin{cases} 1/\sqrt{k_i} & (k_i > 0) \\ 0 & (k_i = 0) \end{cases}$

步骤 4: 求 B 最大特征根对应的特征向量 γ 。

步骤 5: 求目标单位向量 $\bar{\alpha}$

$$\bar{\alpha} = (\bar{\Lambda} P)^T \gamma / \| (\bar{\Lambda} P)^T \gamma \|$$

步骤 6: 特征模板(向量中心)的构造。导入任意图像的灰度矩阵 K , 则其相似度特征即为矩阵 K 在 $\bar{\alpha}$ 上的投影向量。设该投影向量为 c , 则:

$$c = K^T \bar{\alpha}$$

同理, 步骤一中导入的每一幅样本图像的相似度特征可表征为:

$$x_i = A_i^T \bar{\alpha} \quad (i = 1, 2, \dots, s)$$

由此可构造该样本集的特征模板(向量中心)为: $x_c = \frac{1}{s} \sum_{i=1}^s x_i$ 。

步骤 7: 计算待检图像与特征模板的差异。设 K 为待检索图像, 则 K 的投影向量 c 与特征模板的差异为:

$$d = \| c - x_c \|$$

步骤 8: 图像检索。对于设定的阈值 T , 当差异 d 小于 T 时认为该待检图

像与特征模板相似，即确定其为要检索的图像。

下面为 MATLAB 下这几个步骤的相关代码。

```
function [objvector center] = matlabfunction(set)
for i=1:size(set), SUM1=SUM1+ set{i}*( set{i}.'); end
[P,V]=eig(SUM1);
PN=inv(P);
for x=1:dimension
    for y=1: dimension
        if V(x,y)>0
            VN(x,y)=1/sqrt(V(x,y));
        end
    end
end
for a=1:size(set)
    for b=1:length(set)
        if a~=b
            SUM2=SUM2+ set{a}*( set{b}.');
        end
    end
end
B=VN*PN*SUM2*((VN*PN).');
d=eig(B);
for r=1: dimension,if d(r)>tzmax,tzmax=d(r);maxcol=r;end,end
[Q,W]=eig(B);
tzmaxxl=zeros(dimension,1);
for c=1: dimension,tzmaxxl(c,1)=Q(c,maxcol);end
objvector=((((VN*PN).')*tzmaxxl)/norm(((VN*PN).')*tzmaxxl);
center=zeros(dimension,1);
for d=1:size(set)
    center=center+(( set{d}.')*objvector);
end
center=(1/length(set))*center;
```

代码中，set 为样本集；dimension 为特征向量的维度；objvector 为求得的目标单位向量；center 为最后构造出的特征模板(向量中心)。

5.3 实验测试与结果分析

本节将利用 5.2 节基于相似度矩阵的图像检索算法分别对手枪的原始图像和其 Canny 轮廓图像检索，并对实验结果进行分析。

5.3.1 基于原始图像样本的图像检索

由于手枪颜色可较明显的分为白色和黑色，二者灰度差异较大。因此，实验中分别基于 12 张黑枪样本图像(如图 5.1)和 12 张白枪样本图像(如图 5.2)建立他们的特征模板，并求出两组图像的单位向量 α 和特征模板 x_c 。

首先将读入的每一幅手枪图像转化为灰度图像。为了提高算法的运算速度，并简化冗杂的图像信息，本节采用灰度累加和矩阵替代源图像的灰度矩阵，具体方法如下。将这些读入的灰度图像的尺寸转化为 256×256 以统一图像规格，再以 16×16 的分割窗口将每一幅 256×256 的图像分割为 16×16 块。求取每一块窗口范围内的像素点灰度值之和，并将其依次存入一个新建的 16×16 的图像矩阵 M 的对应位置，称这个矩阵为“灰度累加和矩阵”。这样就可基于这些手枪样本图像的灰度累加和矩阵，计算其单位向量 α 和特征模板 x_c 。

完成特征模板的构造后，求取每一张待检索图像对应的 16×16 的灰度累加和矩阵。将其在黑枪和白枪的“单位向量”上的投影，并将得到的投影向量与第一步算出的黑枪与白枪的特征向量中心相比较，求取差异模值来判断该图像是否为手枪图像。



图 5.1 黑枪训练样本



图 5.2 白枪训练样本

若待检索图像的特征向量与黑枪特征向量中心的差距小于黑枪的差距阈值时，认为该图像为黑枪图像。否则，若其特征向量与白枪特征向量中心的差距小于白枪的差距阈值时，认为该图像为白枪图像。若都不满足，则认为该图像不是手枪图像。

实验中选取 398 张样本图像作为待检图像数据库 Q ，其中 199 张为手枪图像，199 张非手枪图像。使用查准率为 P 、查全率 R 和误检率 F 来评价检索结果。测试中发现，当待识别图像的特征向量与之前算出的黑枪向量中心及白枪向量中心的差距阈值分别大于等于 6×10^{-6} 和 3×10^{-6} 时，黑枪和白枪特征模板在 Q 中的误检率接近 50%，此时可认为算法失去检索能

力。因此，待识别图像的特征向量与黑枪向量中心和白枪向量中心的差距阈值只能在开区间（0， 6×10^{-6} ）和（0， 3×10^{-6} ）内取得。实验中，分别选取了六个阈值组合来进行图像检索，实验结果如表 5.1 所示。

表 5.1 不同阈值下的的检索结果

黑枪阈值	白枪阈值	a	B	P	R	F
1×10^{-6}	1×10^{-6}	119	137	87%	60%	9%
1×10^{-6}	2×10^{-6}	127	165	77%	64%	19%
3×10^{-6}	1×10^{-6}	139	183	76%	70%	22%
3×10^{-6}	2×10^{-6}	151	211	72%	76%	30%
5×10^{-6}	1×10^{-6}	143	213	67%	72%	35%
5×10^{-6}	2×10^{-6}	159	239	67%	80%	40%

从检索结果可以看出，基于原始灰度图像进行手枪图像检索，不可避免的受到图像中多余色彩信息的干扰。其中，第 3 个和第 4 个阈值组合的检索结果较好，但其查准率和查全率也刚大于 70%。

5.3.2 基于图像 Canny 特征区域的图像检索

单纯基于手枪图像的灰度矩阵进行图像检索不可避免的受到背景图像的干扰。图像中的色彩、光照明暗等因素也将影响图像检索的准确性。Canny 算法可有效提取图像中物体的轮廓线。可利用 Canny 算法提取出手枪样本的轮廓图像来构造手枪特征模板，然后将其用于检索。由于是以手枪的形状为基准，所以不再区分黑枪和白枪。

实验中用 Canny 算法提取出 12 张手枪样本的轮廓线图像（如图 5.3）并用这个新的样本集按照基于矩阵相似度的特征提取算法构造特征模板。可先将图像的尺寸适当的缩小，再用 Canny 算法进行轮廓提取。这样做可以筛除冗杂多余的信息并缩短程序执行时间、提高算法效率。本实验中将

读入的训练样本及待测图像统一缩放为 200×200 像素。

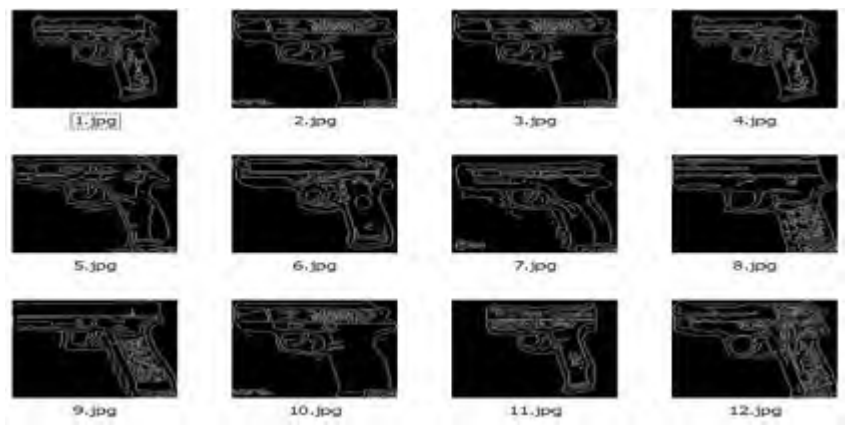


图 5.3 基于 Canny 轮廓线的手枪样本

实验中的待测图片还选取上节实验样本库 Q 中的 398 张样本图像。在特征匹配前，用 Canny 算法提取 Q 中每一张样本图像的特征区域图像。然后，用矩阵相似度算法提取这些特征区域图像的特征向量，并基于上一步中求出的特征模板进行匹配测试。实验中，选取了 12 个阈值用于图像检索，结果如表 5.2 所示。

表 5.2 不同阈值下的的检索结果

阈值	a	B	P	R	F
1.5	0	0	100%	0%	0%
2.0	8	8	100%	4%	0%
2.5	14	14	100%	7%	0%
3.0	32	32	100%	16%	0%
3.5	48	48	100%	24%	0%
4.0	80	82	98%	40%	1%
4.5	111	127	87%	56%	8%
5.0	149	171	87%	75%	11%
5.5	189	217	87%	95%	14%
6.0	191	265	72%	96%	37%
6.5	193	297	65%	97%	52%
7.0	197	318	62%	99%	61%

实验结果表明：当阈值设为 5.5 时，算法的查准率为 87%、查全率为 95%，而误检率仅为 14%，检索效果最好。

通过对比表 5.1 和 5.2，可以看出基于 Canny 轮廓图像的图像检索性能大大优于基于原始灰度图像的方法。该方法在对手枪图像的检索过程中有效的规避了图像中其他颜色信息的干扰，达到了理想的效果。

本节检索所用的模板中，手枪都是枪口朝左正方。如果待检索图像中手枪是倒放、斜放或反方向放置时，还用该模板检索，结果可能会不好。为解决这一问题，可以分别针对不同姿态的手枪图像(如图 5.4)按照 5.3.2 中的方法分别构造特征模板。在进行图像检索时，这些模板依次分别和待检图像匹配比较从而更全面的检测图像特征。



图 5.4 不同姿态的枪支图像

5.4 小结

在对以枪支为代表的一类色调单一物体的图像检索中，待检索图像自身区别度低、无规律的色彩信息常常导致检索系统的误检。在特征模板的构造过程中，参照样本的背景颜色、光照明暗、对比度等因素也常常作为干扰信息影响有效特征模板的生成。对于这一类图像的检索，若待检索目标的轮廓、形状或纹路有内在的统一性，且能明显区别于其他物体。则可以这些目标图像的轮廓图像作为特征模板的参照样本，以待检索图像的轮廓形象作为检索系统的识别对象。本章提出的基于 Canny 特征区域的图像检索方法正是这种基于图像内物体轮廓特征的检索手段。在特征模板的构造算法上，引入了矩阵相似度及相似性判别函数等概念，通过求取使参照样本矩阵相似性最大的单位向量，继而求取训练样本的向量中心作为特征

模板并与待检索图像矩阵的特征向量相比较以判断该图像是否为同类图像。

第 6 章 总结与展望

6.1 工作总结

本次毕业设计的研究工作从开题至今已历经五个月。在整个研究、学习、实践的过程中我碰到了不少困难也走了不少弯路。但今天，当全部工作接近尾声时，回顾整个历程，我收获颇丰。

本次工作主要围绕图像中特征点及特征区域提取算法的研究及其应用展开。学习研究之初，由于我对数字图像处理的一些方法和原理还不熟悉，因此在阅读研究参考论文中常常碰到一些困惑的地方。例如在对图像的高斯函数卷积运算的理解上，我开始没有正确理解图像的卷积操作与率波操作的关系导致了计算中的一些错误。经研究学习，我了解到卷积操作是对卷积模版的镜像矩阵做滤波操作，两种运算在卷积模版按中心对称的情形下等效。在特征点及特征区域的提取算法中涉及的高斯卷积核是一个中心对称的模版，故而可直接以卷积核作为滤波模版对待卷积矩阵进行滤波操作即可。此外，对参考论文中的一些概念及操作，我往往知其然而不知所以然。例如，在 SIFT 特征点的提取算法中，“尺度”是一个很重要的概念，但是论文中并未对这一概念作明确的解释。在查阅了 Lowe 的原文并核对了一些引用这一概念的论文后，我认识到这里的“尺度”不仅直观地表现为图像的尺寸，这种尺寸的变化也即为图像中景物由远即近的过程。在提取算法中，用高斯滤波函数的方差参数的变化将上述过程表现为由模糊到清晰的过程。该算法正是基于不同“尺度”下的图像提取出具有尺度不变性的特征。在研究核对了不同关于 SIFT 特征点提取算法的资料后，我

还发现了一些论文中表述不妥当的地方。例如，关于精确定位 SIFT 极值点的部分，我们应用的是三维二次函数的拟合方法。在拟合过程中，我们应根据极值点的偏差向量修正极值点的位置。但是，一些论文中将这个修正量笼统地表述为一个变量而并不指明其含义，这就不免造成初学者理解上的混淆。以上这些经验让我认识到在今后的科研学习工作中，要正确、全面地理解某种理论时应深入了解这种理论的背景原理，并大量查阅关于这一理论的不同资料。

在实验测试阶段，我选择了 MATLAB 作为实验平台。该平台封装了大量图像处理函数，可实时查看图像处理效果并直观地观察矩阵中元素值的变化，这给我的研究工作带来了很大的便利。在对 Canny 算法的性能测试中，我注意到该算法提取的特征区域规避了图像中多余的色彩信息，表现为图像中物体的轮廓线。而在很多图像检索应用中，正是由于待检索目标单一的色调和图像中多余色彩信息的干扰影响了检索的精度。我尝试着设计了一种基于图像 Canny 特征的检索方法，并通过实验测试证明了这种方法的有效性。这正是我这篇毕业论文的创新点。通过这次实验，我认识到一种技术的性能特性往往适用于其他相关应用中，在对这种技术进行研究测试时，应尽量挖掘这些特性。

6.2 进一步研究展望

(1) Harris 算法可快速准确地提取图像中的角点特征，但该算子对图像尺度的变化非常敏感，一般当图像尺度缩小一倍就几乎有 50% 的角点检测不到。SIFT 算法提取的特征点具有尺度不变性，且该算子对噪声、光照、旋转变换等具有较强的鲁棒性。但是，当该算子以 128 维向量描述一个特征点，该过程计算量大且在特征点匹配时，特征点间的区别度依赖与

源图像中的纹理丰富性。在今后的研究中，应侧重于设计鲁棒性强、特征点描述符简单且计算量小的特征点提取算法。

(2) 本次设计出的手枪图像检索方法是以 Canny 特征区域作为检索特征的。经测试，这种方法很有效。但是，由于参照样本大多是手枪的“特写图像”(手枪形象占主体地位的图像)，因此由此构造出的特征模板只适用于对类似的“手枪特写”图像进行检索而无法有效检索出“包含”手枪形象的图像。针对上述问题，须设计一种将图像中的物体形象分别“截取”的方法，并将“截取”后的物体图像矩阵作为检索对象通过我们设计的方法进行检索。

参考文献

- [1] 阮秋琦译. 数字图像处理[M]. 北京: 电子工业出版社, 2002.
- [2] 杨艳伟. 基于 SIFT 特征点的图像拼接技术研究 [D]. 硕士学位论文, 西安: 西安电子科技大学, 2005.
- [3] 杨占龙. 基于特征点的图像配准与拼接技术研究[D], 博士学位论文. 西安: 西安电子科技大学, 2008.
- [4] 刘智. TBIR 与 CBIR 结合检索 Web 图像的探讨[J]. 广西工学院学报, 2006, 17(2): 55 – 58.
- [5] 韦娜, 耿国华, 周明全. 基于内容的图像检索性能评价[J]. 中国图像图形学报, 2004, 9(1): 1271 – 1276.
- [6] C.Harris, M Stephens. A combined corner anda edge detector[J]. Forth Alvey Vision Conference, 1988, 12(5): 17 – 151.
- [7] D.Lowe.Distinctive. Image Features from Scale-Invariant Keypoints [J].International Journal of Computer Vision, 2004, 60(2): 91-110.
- [8] 王小睿, 吴信才, 李军. 模拟退火算法的改进策略在模板匹配上的应用[J].小型微计算机系统, 1997, 18(8): 32 – 37.
- [9] 赵向阳, 杜利民. 一种全自动稳健的图像拼接融合算法[J]. 中国图形图像学报,2004,9(4): 417 – 422.
- [10] 李静. 基于内容的图像检索技术研究现状综述[J]. 应用科技, 2010,7(2): 67 – 70.
- [11] 薛定宇. 高等应用数学问题的 MATLAB 求解[M]. 北京: 清华大学

出版社, 2008.

- [12] 陈杰等.一种尺度自适应 Canny 边缘检测方法[J]. 光电工程, 2008,35(2): 79~84.
- [13] 程永清, 庄永明, 杨静宇.基于矩阵相似度的图像特征抽取和识别[J]. 计算机研究与发展, 1992, 14(5): 42-48.
- [14] 张强. 精通 Matlab 图像处理[M]. 北京: 电子工业出版社, 2010.
- [15] Barbara Zitova, Jan Flusser. Image registration method: a survey [J]. Image and Vision Computing, 2003, 21(3): 977 – 1000.