

分类号 \_\_\_\_\_  
UDC \_\_\_\_\_

密级 \_\_\_\_\_  
编号 \_\_\_\_\_

# 中国科学院研究生院

## 硕士学位论文

基于图像特征的拼接算法研究

张霖

指导教师 焦建彬 教授

中国科学院研究生院

申请学位级别 工学硕士 学科专业名称 计算机应用技术

论文提交日期 2011年4月 论文答辩日期 2011年5月

培养单位 中国科学院研究生院

学位授予单位 中国科学院研究生院

答辩委员会主席 \_\_\_\_\_

## 中国科学院研究生院直属院系

### 研究生学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明或致谢。

作者签名：

日 期：

## 中国科学院研究生院直属院系

### 学位论文授权使用声明

本人完全了解并同意遵守中国科学院有关保存和使用学位论文的规定，即中国科学院有权保留送交学位论文的副本，允许该论文被查阅，可以公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存、汇编本学位论文。

涉密的学位论文在解密后适用本声明。

作者签名：

日 期：

导师签名：

日 期：



## 摘 要

普通数码相机所拍摄的图像具有大视场图像分辨率低、分辨率高的图像视场小的特点。为了解决视场和分辨率的矛盾，人们提出了图像拼接技术，将反映同一场景的多幅有重叠区域的图像拼成一幅高分辨率大视野范围的图像。该技术现已广泛应用在全景监控、虚拟现实、遥感图像处理、医学图像处理等领域。

本文研究基于图像特征点的自动拼接算法，该算法可以实现多幅有序图像的自动拼接，最终得到环绕全景图，可用于全景监控系统。本文重点研究了基于特征点的全景图生成的各个阶段的技术，并针对其中的两个阶段提出了改进。

在特征匹配阶段，本文提出了改进的最小费用  $K$  流算法（MKP 算法）。该算法根据已有的最小费用  $K$  流算法的思想，将图论中的匹配知识同图像匹配知识相结合，把图像特征点匹配建模成最小费用最大流问题，并用改进的对偶算法进行求解，可以自动获得有较高准确率的图像特征点匹配集。实验表明，在不需人工干预的情况下，和基于  $k-d$  tree 的最近邻匹配算法相比，该算法可以提高匹配准确率约 5%。

在变换估计阶段，本文提出了基于类内散布矩阵的改进的 RANSAC 算法，用于对图像的特征点匹配集进行提纯。改进的 RANSAC 算法用类内散布矩阵进行初选，改变了原 RANSAC 算法中匹配点对的随机选取方式，有效减少了原 RANSAC 算法的迭代次数，提高了计算效率。实验表明，在保证准确率的条件下，本文算法的计算效率比原算法高出约 20%。

**关键词：**全景图像拼接，改进最小费用  $K$  流，对偶算法，类内散布矩阵，RANSAC

## **Automatic Panoramic Image Stitching Algorithm Based on Features**

**Zhang Lin** (Computer Application)

Directed by: **Jiao Jianbin** (Professor)

In some situations, we need panoramic images with high-resolution and greater view. But the images taken by general digital camera usually can not meet the need of the practical application. Image stitching technology is used to generate a panoramic image from multiple small images. The technology is widely used in panoramic video surveillance, virtual reality, remote sensing image processing, medical image processing and other fields.

In this paper, we discuss the automatic image stitching algorithm based on SIFT features. The algorithm can generate a complete and high resolution panoramic image from a sequence of small images taken by one or more cameras. The paper gives a detail introduction of every step in the process of generating a panoramic image and proposes two improved algorithms.

In feature matching stage, an improved Minimum Cost K Flow algorithm is proposed in the paper. The algorithm models image matching problem as Minimum Cost Maximum Flow problem. The improved MKP algorithm does not need to adjust the number of matching feature, but it can get feature pairs as more as possible. Experiment results show that the improved MKP algorithm can increase the accuracy of matching about 5 percent without the manual work.

In image transformation stage, an improved RANSAC algorithm with within-class scatter matrix is proposed in the paper for deleting the erroneous matching pairs. In the improved algorithm, the dispersal of the feature points is used in guided sampling to reduce the number of iteration. Experiment results show that the improved algorithm can increase the registration speed by some 20 percent with the same accuracy and robustness comparing to the original RANSAC algorithm.

**Key Words:** Panoramic image stitching, improved MKP algorithm, within-class scatter matrix, improved RANSAC

## 目 录

基于图像特征的拼接算法研究 .....	i
摘 要 .....	I
图目录 .....	V
表目录 .....	VI
第一章 绪论 .....	1
1.1 图像拼接技术的研究背景及应用领域 .....	1
1.1.1 研究背景和意义 .....	1
1.1.2 应用领域 .....	2
1.2 图像拼接技术的研究现状 .....	4
1.2.1 直接方法 .....	4
1.2.2 基于特征的方法 .....	4
1.3 论文的研究内容 .....	7
1.4 论文的组织结构 .....	8
第二章 图像拼接的基础知识 .....	10
2.1 图像获取 .....	10
2.2 图像特征提取与匹配 .....	11
2.2.1 图像特征提取 .....	11
2.2.2 图像特征匹配 .....	12
2.3 图像变换与融合 .....	14
2.3.1 图像变换模型 .....	14
2.3.2 图像融合 .....	15
2.4 参考面选取与投影 .....	17
2.4.1 平面投影模型 .....	17
2.4.2 柱面投影模型 .....	18
2.4.3 球面投影模型 .....	19
2.5 小结 .....	20
第三章 基于改进的最小费用 K 流的特征点匹配算法 .....	21
3.1 图论基础知识 .....	21
3.1.1 图和二部图 .....	21
3.1.2 网络流 .....	22
3.1.3 二部图的匹配 .....	23
3.2 选取度量函数 .....	24
3.3 改进的最小费用 K 流匹配算法 .....	24

3.3.1	构造最小费用 K 流模型 .....	24
3.3.2	利用改进对偶算法求解模型 .....	26
3.3.3	改进的最小费用 K 流匹配算法流程 .....	29
3.4	实验对比与分析 .....	30
3.5	小结 .....	31
第四章	基于改进 RANSAC 的图像拼接算法 .....	32
4.1	RANSAC 算法 .....	32
4.1.1	RANSAC 算法的基本思想 .....	32
4.1.2	用于图像拼接的 RANSAC 算法 .....	34
4.1.3	算法计算量分析 .....	35
4.2	改进 RANSAC 算法 .....	36
4.2.1	类内散布矩阵 .....	37
4.2.2	利用类内散布矩阵改进 RANSAC 算法 .....	37
4.2.3	运算时间分析 .....	38
4.3	实验对比与分析 .....	39
4.3.1	准确性 .....	40
4.3.2	计算效率 .....	41
4.3.3	多幅图像拼接实验结果 .....	42
4.4	小结 .....	44
第五章	总结与展望 .....	45
参考文献	.....	46
附录 1	.....	50
致 谢	.....	51

## 图目录

图 1.1 鱼眼镜头拍摄图像效果 .....	2
图 1.2 航拍图像拼接效果图 .....	2
图 1.3 视网膜拼接图 .....	3
图 1.4 全景拼接图（约 180 度） .....	3
图 1.5 图像拼接流程图 .....	8
图 2.1 几种变换的示例图 .....	14
图 2.2 融合前后对比效果图 .....	16
图 2.3 平面投影示意图 .....	18
图 2.4 柱面投影示意图 .....	18
图 2.5 球面投影 .....	19
图 3.1 SIFT 特征示例图 .....	21
图 3.2 图示例 .....	22
图 3.3 二部图和完全二部图 .....	22
图 3.4 匹配和最大匹配 .....	23
图 3.5 图像对特征点的赋权完全二部图 .....	25
图 3.6 容量网络模型 .....	25
图 3.7 增加附加源点的网络 .....	27
图 3.8 匹配结果 .....	31
图 4.1 RANSAC 直线拟合示例图 .....	32
图 4.2 直线拟合示例图 .....	36
图 4.3 改进的 RANSAC 算法流程 .....	39
图 4.4 实验结果比较图 1 .....	40
图 4.5 实验结果比较图 2 .....	40
图 4.6 迭代次数对比 .....	41
图 4.7 时间对比 .....	42
图 4.8 原图像 .....	42
图 4.9 柱面投影 .....	43
图 4.10 特征点提取 .....	43
图 4.11 特征点匹配 .....	44
图 4.12 最终效果图 .....	44

## 表目录

表 3.1 三种方法的 Precision 比较.....	31
表 4.1 随机采样次数随内点比例变化而变化的情况 .....	35

## 第一章 绪论

### 1.1 图像拼接技术的研究背景及应用领域

图像拼接是计算机图形学、计算机视觉、图像处理领域的重要研究内容，并且已经应用于社会生活的各个方面。图像拼接技术研究的是如何对多幅有部分重叠区域的窄视角图像进行无缝拼接，得到宽视角图像来满足人们的各种研究需要。这项技术为视场扩展、图像降噪、空间解析、模糊消除等提供了方法和可能性，现已应用于许多领域，比如：全景图像的合成、遥感图像的拼合、基于图像的绘制技术、物体的三维重建（虚拟现实技术、医学图像分析、遥感技术处理）等等。

#### 1.1.1 研究背景和意义

本课题受到下列项目的资助：

1. “天目 360 度智能监控系统产业化开发”，哈尔滨产业技术创新与育成专项资金项目，2011.01-2013.12，在研。
2. “基于多源数据的飞行器进近威胁目标检测跟踪及行为预测”，国家自然科学基金重点项目（课题编号：61039003），2011.01-2014.12，在研。

在现实生活中，人们常常需要用到宽视角的图像，比如获取某城市的地形图、得到某区域 360° 的环绕全景图像、对某地的 360 度监控图像等。这些图像都是超越人眼视野范围的高分辨率大场景范围图像。然而，场景范围和分辨率存在着矛盾，一般来说场景范围越大，分辨率越低；分辨率越高，场景范围越小。在使用普通数码相机进行拍摄时，人们必须调节相机的焦距才能拍摄较宽场景，这时获取的图片的分辨率相对较低，常满足不了要求。另外，对于一些超大尺寸拍摄目标，由于拍摄距离和普通数码相机的视野范围的限制，没有办法用一张照片拍摄出完整目标，这一点在航空图像拍摄时尤为突出。为了得到满足要求的大视角范围的图像，人们发明了广角镜头和扫描式相机。这些设备可以部分解决视角不足这个问题，但是它们价钱昂贵，使用复杂，而且其图像边缘常会难以避免的产生扭曲变形（如图 1.1 所示），在一些场合无法使用。于是为了在保证高质量高分辨率的条件下获取大视野范围的场景图像，人们提出了图像拼接技术。图像拼

接技术就是将一组重叠的局部图像拼接合成一幅大型的无缝高分辨率图像，它使得人们可以通过手持或者支架固定照相机获取图片，然后对其使用图像拼接算法，获得高分辨率大视野范围的场景图像。



图 1.1 鱼镜头拍摄图像效果

### 1.1.2 应用领域

目前图像拼接技术可以广泛应用于诸多领域，归纳起来有以下四个方面：

#### (1)碎片（多源）图像的组合

在航空航天和宇宙探测、以及地质和海洋勘测等领域，图像拼接技术将探测器获取的小范围高分辨率照片拼接，组成大视野范围的高分辨率图像，如图 1.2，这样可以用低精度的普通数码相机来获取高精度的图片，以此降低光学设备的成本和拍摄数据传输的复杂性；在刑侦领域，可将犯罪现场的照片、证物扫描图片进行拼接，得到完整现场图片方便破案。



图 1.2 航拍图像拼接效果图<sup>[1]</sup>

#### (2)医学图像处理

在医学领域，现有的显微镜等医疗设备存在视野狭小的问题；另外，一种手段获得的图像信息可能会有所缺失，图像拼接技术可将 X 射线断层扫描技术、

数字减影血管造影术、核磁共振、X光片等技术获得的图像组合，得到病人的完整信息，提高检查的准确性；利用图像拼接中的匹配技术，可将病人的图像样本与解剖图谱比较，用于安全监控来实施对病人的治疗情况进行跟踪；此外在核医学领域，可结合记录解剖身体结构的核磁共振图像和检测身体功能和代谢活动的正电子发射断层扫描技术获得的图片，将多方面图片拼为一个整体进行研究。如图 1.3，是应用于视网膜激光手术领域的完整视网膜拼接图。

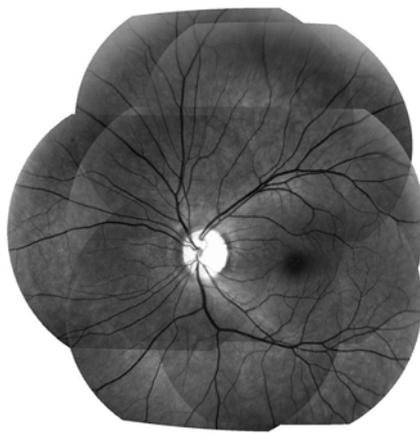


图 1.3 视网膜拼接图<sup>[2]</sup>

### (3)全景图像的合成

全景摄影一般是指拍摄普通照相机无法拍摄的大视野范围照片的摄影技术。一般得到两种广角照片：一种是全景照片（panorama），也就是  $180^\circ$  到  $360^\circ$  的照片；另一种是超过普通照片视场范围的广角照片。全景图片可通过全景照相机完成，也可以通过拼接技术完成。目前许多软件和数码相机都提供了图像拼接功能，让拍摄者可以将多幅照片拼成宽视场范围的照片，用普通镜头获得乃至超过广角镜头所能拍摄照片的视场范围，同时有着更好的视觉效果。另外，全景拼接可以用在视频监控领域，将多个摄像头取到的场景拼成一幅全景场景，从而更有效率地进行视频监控。



图 1.4 全景拼接图（约  $180^\circ$ ）

### (4)虚拟现实

传统的计算机图形学基于几何模型, 用户需要构造出场景的几何模型和光照及表面属性等的建模来完成场景的绘制, 场景复杂时过程非常繁琐。近年来提出了基于图像的场景绘制技术 **IBR(Image-based Rendering)**, 从拍摄的多幅具有重叠关系的照片直接绘制场景, 使用图像拼接技术生成全景图, 用全景图代替复杂的三维场景建模和绘制, 来表示实景。

可以说, 图像拼接技术涉及计算机视觉、图像处理、计算机图形学等多个学科的知识, 具有十分广阔的应用前景, 深入研究这项技术有着重要的意义。

## 1.2 图像拼接技术的研究现状

根据目前的研究重点, 图像拼接技术可以分成图像配准和图像融合两大主要部分。图像配准是根据几何运动模型将同一场景拍摄的不同图像进行对齐, 找出有重叠区域的两幅或多幅图像间的变换关系。图像融合是指将已经配准处理的图像再经过一定的处理, 提取各自的信息, 最后综合成同一图像, 实现相邻图像间的自然过渡。图像拼接的方法按配准方法的不同, 分为直接方法和基于特征的方法。

### 1.2.1 直接方法

直接的方法一般直接利用图像的信息进行配准, 大致分为基于变换域、基于灰度、基于区域三类。基于变换域的方法提出较早, Anuta<sup>[3]</sup>在 1970 年就使用快速傅里叶技术估计图像间的平移量; B.Reddy 等<sup>[4]</sup>使用了快速傅里叶变换进行旋转和尺度变化的估计; 后来, 基于小波的配准方法提出, Moigne 等<sup>[5]</sup>在 1996 年使用小波分解的分量中取得特征点, 之后迭代求得旋转参数; 基于灰度的方法方面, Szeliski 等<sup>[6]</sup>在 1997 年提出了直接像素亮度差优化的方法; Viola 等<sup>[7]</sup>在 1995 年提出引入互信息进行配准, 开启了基于信息理论的配准研究, 其中 Plum 等<sup>[8]</sup>在 2000 年结合梯度信息改善了它的极值性能; Dae-Hyun 等<sup>[9]</sup>在 2003 年使用了基于区域的方法。直接的方法一般直接用图像本身信息计算, 对于光照敏感, 且不适于比较复杂的运动模型, 计算量也比较大。而基于特征的方法使用的是图像中的显著特征, 对图像的噪声、畸变有一定的鲁棒性, 因此, 随着特征检测和匹配技术的发展, 基于特征的配准方法成为研究的热点。

### 1.2.2 基于特征的方法

基于特征的方法分为四步<sup>[10]</sup>, 特征检测、特征匹配、估计变换模型及图像融

合。

特征检测方面, Moravec<sup>[11]</sup>在 1981 年发表的一篇关于立体图像的利用角点进行配准的论文, 是基于特征的图像配准的技术的起源。Harris 和 Stephens<sup>[12]</sup>对其进行了改进, 在 1992 提出了 Harris 角点检测, 并将其应用于运动物体跟踪和三维重建, 从此 Harris 角点检测在图像配准的其它方面得到了广泛应用。在此基础上 Mikolajczyk<sup>[13]</sup>在 1992 年研究出 Harris-Affine 特征检测, 具有仿射不变的特性。赵向阳<sup>[14]</sup>在 2004 年将 Harris 角点检测运用到图像拼接中, 取得了不错的效果。一些检测算法将特征检测推广到多尺度空间, Lowe 等<sup>[15]</sup>在 1999 年将图像 DoG(Difference of Gaussian)空间中的极值点, 并进一步得到了具有尺度不变性和旋转不变性的 SIFT(Scale Invariant Feature Transform)特征点。Brown 等<sup>[16]</sup>在 2003 发表了一篇关于将 SIFT 特征检测技术应用于全景图像自动拼接的论文, 实验中所采用的图片是用同一架相机在等焦距上围绕固定轴旋转一圈所拍摄的图片的配准和拼接。

对特征检测后, 可以直接进行特征匹配, Cho 等<sup>[17]</sup>直接用特征周围像素的相似性进行比较, 但这种方法不能保证稳定, 因此比较好的做法是对每个特征要赋予一定的描述字。Schmid 等<sup>[18]</sup>在 1997 年采用高斯微分不变量, 对 Harris 角点构造了基于尺度空间和不变量理论的图像特征描述算子 SIFT 技术, 并将其应用于图像检索。Mikolajczyk 等<sup>[19][20]</sup>使用在特征的领域内进行方向直方图的统计来抽取特征的方向作为特征的描述。Lowe 等<sup>[21]</sup>在 2004 年用 DoG 空间中的极值点作为特征点, 对特征点的邻域寻找梯度直方图的峰值, 构造了基于尺度和旋转不变性的描述算子 SIFT, 并将其用于目标识别。Mikolajczyk 等<sup>[22]</sup>在 2005 年对 10 种通用的具有代表性的描述子, 针对不同的场景、不同的形变、不同的光照变化、不同的模糊程度和不同分辨率等情况, 采用正确匹配率作为评价标准进行了性能评价, 实验结果表明: 在以上各种情况下, 基于 SIFT 描述子的性能最好, 具有较高的辨析力, 且不依赖于具体的特征检测手段。

进行特征检测并对特征进行描述后, 便可进行两幅图特征集间的特征匹配。对于高维的描述子, 不易直接计算出最相近的描述子。因此匹配时需要使用不同的索引策略, 其中很多方法是基于在高维空间找到最近邻。Samet 等<sup>[23]</sup>在 1989 年提出了 k-d tree 结构; Arya 等<sup>[24]</sup>在 1993 年提出使用优先搜索(Priority research)进行 k-d 树的搜索, 该算法在执行过程中利用一个优先队列保存搜索过程中遇到的次近邻兄弟节点, 在搜索到叶子结点后先从优先队列里的节点出发继续搜索,

另外,作者还提出增量距离计算的思想进行加速搜索。Beis 等<sup>[25]</sup>在 1997 年提出了 BBF 算法,给 k-d tree 算法使用了改良的搜索顺序;Nister 等<sup>[26]</sup>在 2006 年使用了 metric tree,是对在每个等级的每个水平的少数的原型比较其特征描述符。何新英等<sup>[27]</sup>在 2009 年提出了基于网络流的特征优化匹配算法,即最小费用 K 流算法,将特征点匹配问题转化成为图论中的 K 匹配的赋权图问题,用图论中的经典算法最小费用 K 流求解,在精度方面取得不错的匹配效果,但其 K 值需要手动选取,不适合自动图像拼接算法。

找到初始匹配集之后,就要选取合适的模型来估计模型的参数。Szeliski<sup>[28]</sup>在 1996 年提出将正交投影变换模型用于拼接,这个模型有 8 个参数,可以表征平移旋转缩放等相机的多种运动,适用于许多情况,成为了经典算法。对于估计参数方面,由经过了匹配后仍可能存在错误点,于是需要使用从含有错误数据的集合中拟合模型的方法,Stewart<sup>[29]</sup>在 1999 年将 Rousseeuw<sup>[30]</sup>在 1984 年提出的 LMedS (Least Median of Squares) 算法运用到图像拼接中,这个方法是将这个问题作为最小化问题,采用多次迭代,比较残差的中值,选有最小均值的对应矩阵作为运动估计;Fischler 等<sup>[31]</sup>在 1981 年提出了 RANSAC 算法,也是采用迭代的方法,先选取初始匹配集中的一个子集来计算运动估计,再计算所有匹配的剩余残差,迭代多次,最后选取有最多相符合的内点对的集合来最后估计模型参数,与 LMedS 算法相比,RANSAC 在估计模型参数的同时,可以取得最终的匹配内点集。RANSAC 又有许多改进版本,Chum and Matas 在 2005 年提出了 PROSAC<sup>[32]</sup>,初始化就添加更有可能的匹配对,因此能够加速找到最优内点集的过程;D.R. Myatt 提出了 NAPSAC<sup>[33]</sup>,同样采用有引导的采样,减少迭代次数。本文将根据图像拼接自身的特点,提出一种新的用于拼接的改进 RANSAC 的方法,可以有效减少迭代次数,提高计算效率。

求出图像间的变换矩阵,下一步就可以对图像进行变换。变换即将图像投影到同一坐标系下,在计算过程中,很可能遇到算出的坐标非整数的情况,使得图像出现空洞,这是需要采用灰度级插补技术<sup>[34]</sup>来避免这种失真。

分析现有图像技术,我们可以发现图像拼接相对于其他的图像处理技术有其自身的特点,首先源图像的获取条件不同决定了有些拼接算法的针对性很强,一些特定条件下的算法可能完全不适合其他场景;其次图像内容的多样性和多变化性使得拼接技术多样;最后拼接技术有多个环节,具有复杂性的特点。这些特点使得目前还很难出现一种拼接技术可以适用于所有场景,现有的两种拼接方法各

有不足:

- ❖ 直接的方法中比较典型的是基于频域的方法, 它利用互相关函数对两幅图像进行相似性计算。这种方法很难处理存在小比例缩放和小角度旋转的图像, 通常计算复杂性较高, 同时对噪声也比较敏感;
- ❖ 基于特征的方法需要提取图像的几何或内容特征, 它依赖于图像特征, 若特征选取有误, 或特征间的匹配有误, 得到的结果会受较大影响;

总的来说, 基于特征的方法更有效一些, 而对于基于特征的方法来说, 其研究方向是要找到更优的特征检测及描述方法, 更准确的特征点匹配方法, 更优化的变化估计方法, 来减少拼接系统的处理时间, 使得系统具有自动性和自适应性, 使其可处理更复杂的照相机模型, 本文主要研究基于特征的方法。

### 1.3 论文的研究内容

论文主要研究了基于图像特征点的拼接算法, 基于特征点的图像拼接主要包括特征提取、特征匹配、变换估计、图像融合四个部分, 其基本流程如图 1.5 所示。本文主要针对其中两个部分: 特征匹配和变换模型参数估计做了创新与改进。

特征匹配方面, 针对最小费用  $K$  流算法 (MKP 算法) 的局限性与不足, 我们提出了改进的最小费用  $K$  流算法。此算法将 SIFT 特征点作为图像的兴趣点, 采用新的度量函数来度量匹配度, 建立最小费用  $K$  流模型, 通过改进的对偶算法求解最小费用  $K$  流, 自动求出有较小错误率的匹配点集。改进 MKP 算法实现了匹配个数的自动选取, 改进了原 MKP 算法中的匹配个数需要手动选取的缺点, 提高了 MKP 算法在图像拼接上的可用性, 而且能获得比常用匹配算法更高的准确率。

变换估计方面, 提出了一种针对图像拼接的改进 RANSAC 算法。该算法利用了图像拼接中的特征点的分布情况对随机样本集进行预处理, 改进了 RANSAC 算法中点对选取的随机性, 在获得同样的计算精度的条件下, 有效减少了变换估计的迭代次数, 提高了变换参数的计算速度。

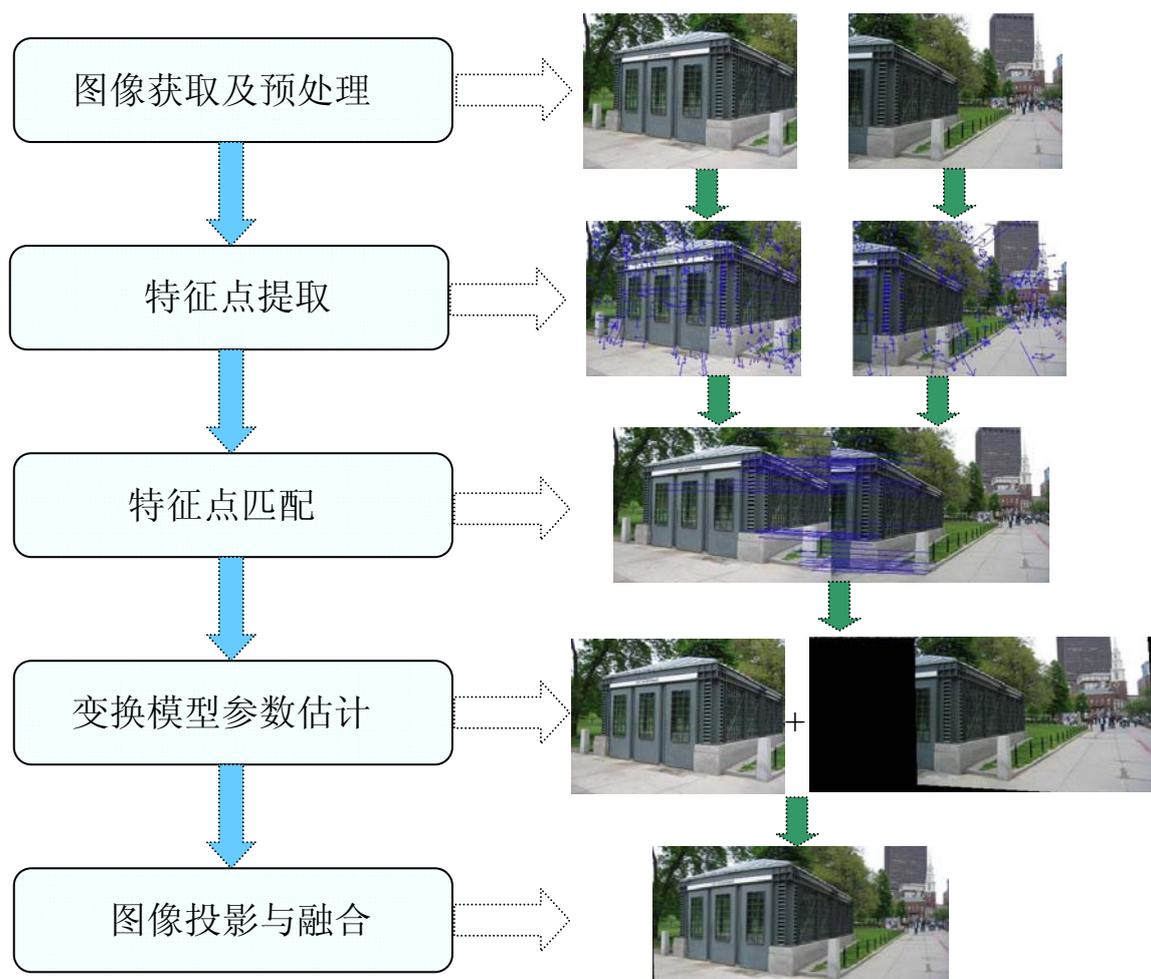


图 1.5 图像拼接流程图

#### 1.4 论文的组织结构

第一章，绪论。主要论述了图像拼接算法的研究背景和意义，分析了国内外的研究现状和现有算法的不足，总结了研究的难点和论文的研究方向，介绍了本文的主要研究内容和贡献。

第二章，图像拼接基础知识介绍。按照图像拼接的基本步骤分别介绍了图像获取，特征点的提取与匹配，图像变换与融合，全景图像投影模型几个方面的基础知识。

第三章，基于最小费用最大流的特征点匹配算法。据已有的最小费用  $K$  流算法的思想，将图论中的匹配知识同图像匹配知识相结合，把图像特征点匹配建模成最小费用最大流问题，并用改进的对偶算法进行求解。将该算法在公共测试集合上进行了测试和比较，实验表明这种改进方法是可行的、有效的，在不需人

工干预的情况下，和基于  $k-d$  tree 的最近邻匹配算法相比，该算法可以提高匹配准确率约 5%。

第四章，基于改进 RANSAC 的图像拼接算法。先介绍了 RANSAC 算法和类内散布矩阵，然后提出了基于类内散度矩阵的改进 RANSAC 算法，对图像的特征点匹配集提纯之后进行拼接。将该算法在公共测试集合上进行了测试和比较，用实验表明改进 RANSAC 算法在保证拼接效果不变的情况下，提高了图像拼接的速度约 20%；最后给出了应用本文拼接算法的多幅图像拼接效果图。

## 第二章 图像拼接的基础知识

图像拼接技术是将有部分重叠区域的窄视角图像进行无缝拼接,得到一幅宽视角图像。基于特征点的图像拼接技术因其较好的效果和鲁棒性,现已成为拼接领域中的热门技术。基于特征点的图像拼接通常由这几个阶段组成:图像获取、特征点提取与匹配、图像变换估计与融合。如果是多幅图像或全景图像拼接,有时还要加入参考平面选取与投影这一步。本章将对各阶段涉及的基本知识进行介绍。

### 2.1 图像获取

拼接的第一步是获取图像,现在用于拼接的图像基本上是用数码设备直接获取的数码照片或视频图片,一般来说,图像获取的设备分两类<sup>[35]</sup>,一类是数码相机,一类是视频采集卡和摄像头的组合。

#### 1) 照相机拍摄

一般来说使用照相机进行图像获取的方式有三种情况:

- ❖ 旋转照相机:这种情况是指将三脚架固定在一个位置。拍摄时相机绕三脚架的垂直轴旋转,每次旋转一定的角度拍一张照片,要求相邻照片间有一定比例的重叠区域。这种拍摄方法不需要更多相机参数,但是采集的图片不在一个平面上,需要投影到同一平面才能进行拼接,这个平面通常是柱面或球面。
- ❖ 平移照相机:这种拍摄一般是将三脚架固定在某个滑轨的滑车上,让相机平行于成像平面移动拍摄。
- ❖ 手持照相机:这是最简单的一种图像的获取方式,但是也是最难拼接的,因为在这种情况下的拍摄过程中,相机的运动情况最复杂,一般是平移运动、镜头缩放、水平扫动、垂直扫动、旋转运动等的综合。但是目前基本上要求获取图像时站在原地绕转拍摄或者沿照相机的光轴垂直方向走动拍摄,这样可以使获得的图像基本符合变换模型,从而可以实现拼接。

## 2) 采集卡获取

是指用在现实的工程应用中，使用采集卡和摄像头配合来采集视频图像。这种采集方式大致分两种情况，一种是多个摄像机位置固定，每个摄像机的采集场景相对位置不变，这种可以仅在初始化时计算变换模型参数，给后面的视频帧直接使用，但是在摄像头安装时需要其光轴交汇，否则拼接效果一般，对硬件摆放有较高要求；第二种是摄像机位置变化，随监控目标移动，这种需要进行实时变换参数计算，对于匹配的速度要求很高。

在图像采集后，由于图像采集过程中的各种不可控因素，使得图像中的属性发生变化，可能有失真、变形、对比度降低等畸变，有时会在拼接前加一步预处理，来完成待拼接图像的畸变校正的工作，提高预处理提高图像的质量，有助于拼接过程的进行。

## 2.2 图像特征提取与匹配

由于待拼接的两幅图片存在空间变换和灰度变换，在将它们拼接起来之前，要先对他们进行空间位置的对准，找到它们重叠部分的相应位置，确定它们之间的变换关系。基于特征的图像拼接方法中，首先就是要对图像进行特征提取和匹配，通过得到特征点匹配点集，估计两幅图像间的变换关系。

### 2.2.1 图像特征提取

特征点的提取一般分为特征检测和特征描述。

#### 2.2.1.1 特征检测

通常来说，我们希望用于拼接的图像特征有如下特性<sup>[36]</sup>：

- ❖ 在两幅图中都出现，而且具备丰富的信息可用于后续匹配。
- ❖ 有较高的定位精度，可以比较均匀地分布在图像的各个部分。
- ❖ 希望特征易于检测，而且检测算法对图像的噪声、几何形变、灰度变化具有一定的适应性和不变性。
- ❖ 有一定的物理意义，在图中清晰可辨。

图中常见的特征如点、线、面可部分满足上面的要求。现有常用检测子有 SIFT 检测子、Harris-Affine 检测子、MSER 检测子、EBR 检测子等，其中被广泛应用的是 SIFT 检测子，本文中也采用了这种检测子，第三章会详细的介绍它的提取过程。

### 2.2.1.2 特征描述

特征描述是用特征描述子来表达图像特征。得到表达图像特征的描述子后，可使用算法寻找这些描述子之间的匹配关系，从而求得图像间的特征匹配集。

特征描述子一般具有如下性质<sup>[36]</sup>：

- ❖ 特征描述子对所选取的图像几何形变具有不变性。
- ❖ 若图像的特征受到噪声影响或者图像的几何形变与理论上的几何模型略有不同，特征描述子应该有稳定性，即可得到接近原始特征的描述。
- ❖ 特征描述子间有很强的相互辨析能力。
- ❖ 如果特征描述子的形式是向量，这个向量的每个分量从功能上应是独立的。

常用描述子有 SIFT 特征描述子、互信息特征、矩特征。Mikolajczyk 在 2005 年对 10 种通用的局部不变量描述子进行了实验比较<sup>[22]</sup>，实验表明 SIFT 描述子性能最佳、分辨能力最好，而且其性能不依赖于具体的特征检测算法。

### 2.2.2 图像特征匹配

特征匹配就是要找到两幅待拼接图像的重叠区域中，表述场景中同一个点的特征点对，这些点对可以用来计算变换。经过特征点提取，我们已经得到了两幅图像的特征点的特征向量，特征点匹配就是要判定特征点的相似程度，建立两个图像的特征点之间的对应关系。

要判断特征点的相似程度，首先要选取相似性度量，然后再用匹配算法进行匹配集求解。相似性度量是衡量点匹配与否的标准。常用相似性度量有三种：距离测度、相似度、概率测度。

#### 2.2.2.1 距离测度

##### (1) 均方根误差

两图像的匹配均方根误差即灰度矢量  $x$  与矢量  $y$  之间的模的平方根。

$$x - y = (x_1 - y_1, x_2 - y_2, \dots, x_N - y_N)$$

$$S = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2 + \dots + |x_N - y_N|^2} = \sqrt{\sum_{i=1}^N |x_i - y_i|^2} \quad (2.6)$$

其中  $x_i, y_i, i=1, 2, \dots, N$  表示对应位置处的灰度值， $S$  是  $N$  维空间点  $x$  和  $y$  的距离。

##### (2) 差绝对值和误差

两图像匹配的差绝对值和 (Absolute Difference) 是指灰度矢量  $x$  与矢量  $y$  之差各分量的绝对值之和，即：

$$S = |x_1 - y_1| + |x_2 - y_2| + \dots + |x_N - y_N| = \sum_{i=1}^N |x_i - y_i| \quad (2.7)$$

### (3) Mahalanobis 距离

$$d_M(x, y) = (x - y)^T \sum_y^{-1} (x - y) \quad (2.8)$$

即马氏距离，其中，假设基准模板  $Y$  是具有协方差矩阵  $\sum_y$  的正态分布。这种距离测度考虑了基准模板特征的离散程度，其获得分类能力优于前面两种分类，但它的缺点是夸大了变化微小的变量的作用。

### (4) Hausdorff 距离

点  $p_1, p_2$  分别为图像  $I_1, I_2$  中的一个特征点， $a, b$  分别是他们的描述子， $p_1$  点到图  $I_2$  中的任一点的距离记为： $\|a - b\|$ ； $p_1$  点到  $I_2$  的最小距离记为： $d_R(a) = \min_{b \in R} \|a - b\|$ ；图像  $I_1, I_2$  之间的 Hausdorff 距离记为： $H(I_1, I_2) = \max_{a \in I_1} d_{I_2}(a)$ ，即  $I_1$  中所有的点到  $I_2$  中所有点的距离的最大值。

### (5) 欧氏距离

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2.9)$$

## 2.2.2.2 相似度

定义相似测度  $S(x, y)$  为：

$$S(x, y) = \frac{x^T y}{\|x\| \cdot \|y\|}, \quad (\|x\| = (x^T x)^{\frac{1}{2}}, \|y\| = (y^T y)^{\frac{1}{2}}) \quad (2.10)$$

$S(x, y)$  可表征模板与子图像间的相似程度， $S(x, y)$  越大表示越相似。

## 2.2.2.3 概率测度

后验概率常被用来评估参考图与待配准图之间的相似程度。后验概率值越大，代表越匹配。根据贝叶斯法则，后验概率分布可表达为：

$$p(d | I_L, I_R) \propto p(d) p(I_L, I_R | d) \quad (2.11)$$

其中， $I_L, I_R$  为待匹配图像对； $d$  表示视差； $p(d)$  可以体现视差分布特性； $p(I_L, I_R | d)$  可以体现为给定视差  $d$  时，图像对  $I_L, I_R$  之间的相似程度。由此可见，在确立后验概率表达式时，我们可以加入一些图像内在的约束条件，以有效减少匹配的歧义性。

## 2.3 图像变换与融合

经过上一步特征点匹配，我们得到了两幅待拼接图像的匹配点对集，下一步就要根据选取的图像变换模型来估计变换参数，将待配准图像变换到参考图像的坐标系下，最后用图像融合算法形成视觉上平滑的高分辨率拼接图。

### 2.3.1 图像变换模型

变换模型的类型主要有四种<sup>[10]</sup>：刚体变换(Rigid transform)、仿射变换(Affine transform)、透视变换(Projective transform)和非线性变换(Nonlinear transform)。模型选取的依据一般是图像获取的方法和应用要求的场景、准确度等。图 2.4 给出了这几种变换的示例图。

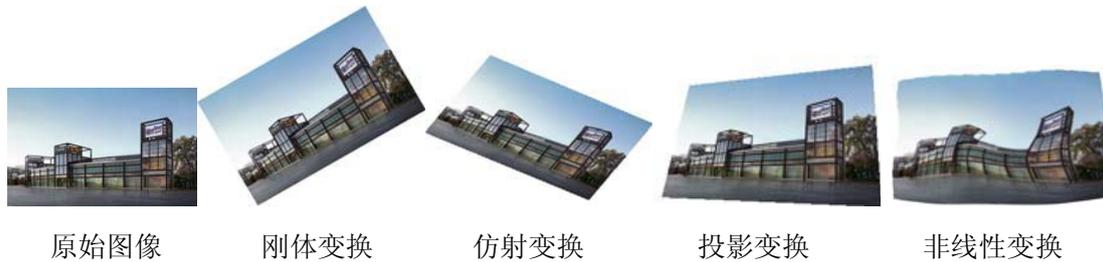


图 2.1 几种变换的示例图<sup>[36]</sup>

#### 2.3.1.1 刚体变换

刚体变换是指经过变换后，图像两点间的距离保持不变的变换。刚体变换可分解为平移、旋转和反转（镜像）三种运动。图像  $I$  上的点  $(x, y)$  到图像  $I'$  上的点  $(x', y')$  的变换公式为：

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \varphi & \pm \sin \varphi \\ \sin \varphi & \mp \cos \varphi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (2.12)$$

其中  $\varphi$  代表了旋转角度， $\begin{bmatrix} t_x \\ t_y \end{bmatrix}$  代表了平移量。

#### 2.3.1.2 仿射变换

仿射变换是指经过变换后，图像上的直线和平行关系不改变的变换。仿射变换可以分解为线性（矩阵）变换和平移变换。图像  $I$  上的点  $(x, y)$  经仿射变换到图像  $I'$  上的点  $(x', y')$  的变换公式用齐次坐标表示为：

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2.13)$$

由上式可知，仿射变换有六个自由度，我们最少可以用三对不共线的匹配点求解这个变换。

### 2.3.1.3 透视变换

投影变换是指经过变换后，图像上的直线关系保持，平行关系基本不保持，但交比不变的变换。图像  $I$  上的点  $(x, y)$  经投影变换到图像  $I'$  上的点  $(x', y')$  的变换公式用齐次坐标表示为：

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2.14)$$

投影变换可以表述照相机的平移、垂直扫动、旋转、镜头缩放等运动，有更广泛的应用价值。由上式可知，投影变换有八个自由度，我们使用至少四对点来确定它的参数。

### 2.3.1.4 非线性变换

非线性变换是指经过变换后，图像上的直线关系不再保持的变换。图像  $I$  上的点  $(x, y)$  经变换到图像  $I'$  上的点  $(x', y')$  的变换公式如下

$$(x', y') = F(x, y) \quad (2.15)$$

其中， $F$  表示任意一种映射函数。典型的非线性变换是多项式变换，典型的多项式变换公式如下：

$$\begin{aligned} x' &= a_{00} + a_{10}x + a_{01}y + a_{20}x^2 + a_{11}xy + a_{02}y^2 + \dots \\ y' &= b_{00} + b_{10}x + b_{01}y + b_{20}x^2 + b_{11}xy + b_{02}y^2 + \dots \end{aligned} \quad (2.16)$$

## 2.3.2 图像融合

最后就是要将一幅变换到另一幅的视角上，进行拼接。由于几何变形、动态场景、或光照变化的存在，相邻图像间常常会有亮度和色彩不连续的问题，图像融合技术正是要消除这些问题，最终得到一幅无缝平滑过渡的高质量图像。

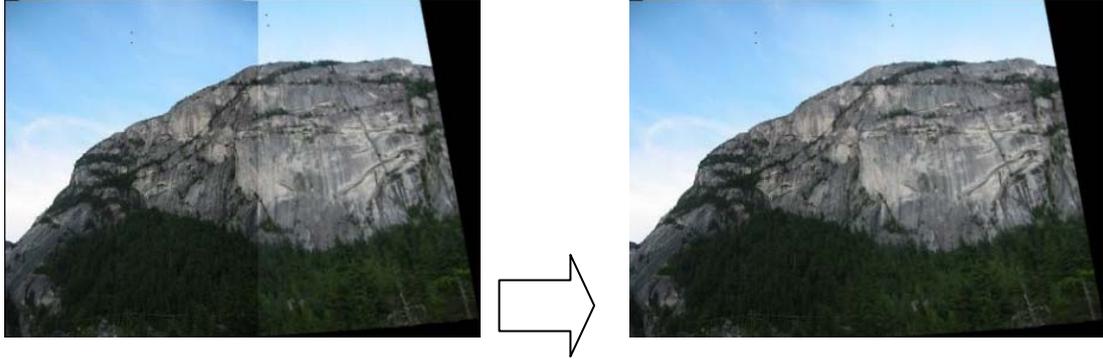


图 2.2 融合前后对比效果图

常用的融合方法有直接平均法、线性加权法、多分辨率法<sup>[49]</sup>。

### 2.3.2.1 直接平均法

直接平均法是在两图重叠区域，将对应像素点的灰度值进行叠加后求平均，各像素点的值由下面的公式确定：

$$I(x, y) = \begin{cases} I_1(x, y) & (x, y) \in R_1 \\ \frac{1}{2}(I_1(x, y) + I_2(x, y)) & (x, y) \in R_2 \\ I_2(x, y) & (x, y) \in R_3 \end{cases} \quad (2.17)$$

其中， $I_1(x, y)$  和  $I_2(x, y)$  分别表示参考图像和配准图像在点  $(x, y)$  处的像素值； $I(x, y)$  表示融合图像在点  $(x, y)$  处的像素值； $R_1$  表示参考图像的非重叠区域， $R_3$  表示配准图像的非重叠区域； $R_2$  表示两图重叠区域。

这种拼接方法简单但是效果不好，融合部分会有明显带状感。

### 2.3.2.2 加权平均法

加权平均法原理同直接平均法类似，但在处理重叠区域时不再是简单的叠加平均，而是加权叠加平均。这个方法相当于使图像的重叠部分由一幅图渐渐过渡到另一幅图，实现方式是将两幅图像重叠区域的像素值按一定得权重相加再平均其基本公式如下：

$$I(x, y) = f_1(x)I_1(x, y) + f_2(x)I_2(x, y) \quad (2.18)$$

其中  $f_1(x)$  和  $f_2(x)$  分别是参考图像和配准图像的权重函数，可以使用线性函数和高斯函数等。

### 2.3.2.3 多分辨率法

多分辨率法<sup>[37]</sup>先将图像分解成一系列具有不同分辨率的子带图像，然后在各层子空间上进行拼合，最后用重构算法合成出原分辨率下重叠区域的图像。这个方法可以有效的保持细节信息不再融合时被平滑掉，还使得图像低频信息可以很好实现平滑过渡，拼接后图像背景比较连贯。

## 2.4 参考面选取与投影

在多幅图像拼接尤其是全景拼接的时候，要根据全景图不同目的和相机的不同运动模式来选择拼接图的二维曲面形式。例如，当我们需要 360 度环绕全景图时，图像的获取方式是相机环绕拍摄。获取的图像的投影平面之间有一定的夹角，如果我们直接对图像进行变换拼接，会使图像产生严重拉伸变形，破坏对象的视觉一致性和空间约束关系。因此通常在拼接前，我们会根据摄像机的运动方式选择合适的投影平面，将取得的反映各自投影平面的图像统一投影到同一参考面上。例如，构造柱面全景图时先将各图像投影到柱面上，再进行拼接，这样就能得到视觉一致的全景图。常用投影模型有平面投影模型、柱面投影模型、球面投影模型。

### 2.4.1 平面投影模型

平面投影模型通常先求得待拼接图像间的变换关系，再选定参考平面来将图像逐个投影到此平面，最后再进行图像融合。

平面投影模型选取的参考平面可以是空间中的任意平面，一般会选择一幅参考图像所在的平面。如图 2.6 我们可以发现，平面投影会使直线得到保持，符合正常的视角习惯，但是进行投影的图像会发生拉伸形变，这种投影模型的视角只能无限逼近 90 度，越接近 90 度，失真越严重。因此平面模型通常适用于平行于拍摄平面运动的相机所拍摄的图片。

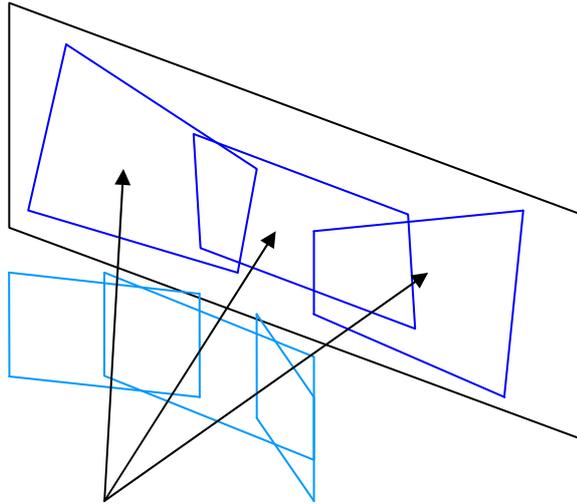


图 2.3 平面投影示意图

#### 2.4.2 柱面投影模型

柱面投影模型<sup>[38][39]</sup>是将待拼接图像投影到参考柱面上，再对图像进行拼接。  
图  $I$  上的一点  $P(x, y)$  到柱面坐标系上的点  $Q(x', y')$  的投影如图 2.7:

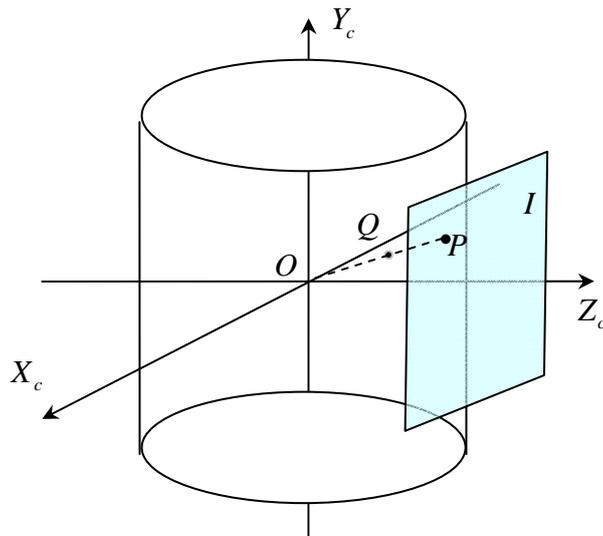


图 2.4 柱面投影示意图

图中参考柱面是以照相机坐标系的原点  $O$  为圆柱体的中心，以摄像机的像素焦距  $f$  为圆柱体的半径。则点  $Q(x', y')$  的坐标可由下列公式获得:

$$\left\{ \begin{array}{l} \hat{x} = f \cdot \arctan\left(\frac{x - \frac{W}{2}}{f}\right) + f \cdot \arctan\left(\frac{W}{2f}\right) \\ \hat{y} = \frac{f \cdot \left(y - \frac{H}{2}\right)}{\sqrt{\left(x - \frac{W}{2}\right)^2 + f^2}} + \frac{H}{2} \end{array} \right. \quad (2.19)$$

其中  $f$  为像素焦距， $W$  为图像  $I$  的宽度， $H$  为图像  $I$  的高度。

柱面投影的特点是，景物在垂直方向上不发生形变，也就是说，图像  $I$  上的位于同一垂直线上的两个像素点在参考柱面上的投影点有相同的横坐标。柱面投影适用于 360 度浏览的全景图。本文在生成全景图时，就是使用了柱面投影模型。

### 2.4.3 球面投影模型

球面投影模型是将待拼接图像投影到球面上，再对图像进行拼接。进行球面投影的图像拍摄时要求照相机坐标系与世界坐标系只存在旋转变换关系，即要求世界坐标系的原点取在摄像机光心位置，同摄像机坐标系原点重合。

图  $I$  上的一点  $P(x, y)$  到柱面坐标系上的点  $Q(x', y')$  的投影如图 2.8:

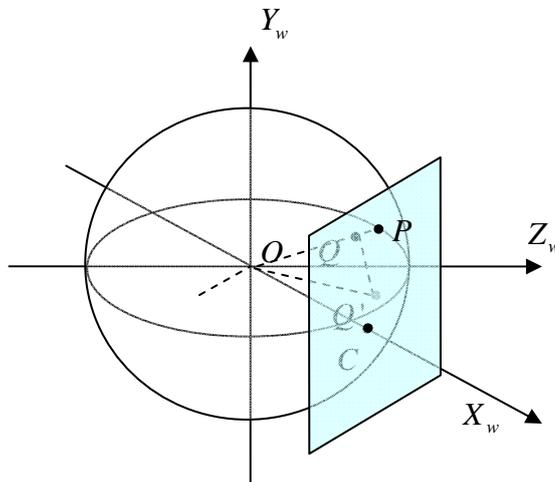


图 2.5 球面投影

图中，世界坐标系  $X_w Y_w Z_w$  与摄像机坐标系原点重合；球面以原点  $O$  为球心，摄像机的像素焦距  $f$  为半径；图像  $I$  以同光轴交汇的点  $C$  为坐标原点，图像上的一点  $P(x, y)$  对应球面上的点  $P'$ ，则球面点  $P'(x_b, y_b)$  在球面坐标系下的经纬坐标

为：

$$\begin{aligned}x_b &= f \times \arctan\left(\frac{x}{f}\right) \\y_b &= f \times \arctan\left(\frac{y}{\sqrt{x^2 + f^2}}\right)\end{aligned}\tag{2.20}$$

球面投影可以构建出漫游效果理想的全景图，但是图像投影后的形变比较严重，不经过光心的直线都会被映射成曲线，实现比较复杂。

## 2.5 小结

本章介绍了图像拼接各阶段的基础知识。首先介绍了图像获取的方式和成像原理；然后介绍了特征点提取与匹配的相关内容；之后介绍了图像变换的基础知识和相关变换模型；最后介绍了多幅或者全景图像拼接时的参考平面的选择。

### 第三章 基于改进的最小费用 K 流的特征点匹配算法

根据前面的介绍,基于特征的图像拼接算法在成功获取图像并进行预处理之后,首先要进行图像特征的提取与匹配。本文使用了 SIFT 特征点进行图像特征的检测与描述,具体实现方法参见文献<sup>[21]</sup>。图 3.1 是在一幅图像上检测并描述的 SIFT 特征。

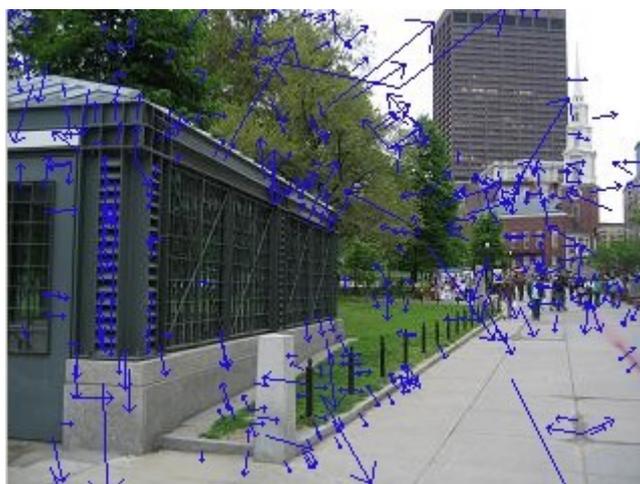


图 3.1 SIFT 特征示例图

图中,我们用箭头代表图像的特征点,箭头的尾端代表特征点在图像上的坐标位置;箭头的方向代表特征点的主方向或者辅方向;箭头的长度与特征点所在的尺度成正比,尺度越大,箭头越长。在实现时我们获得的是图像上 SIFT 特征点的 128 维描述子。下一步就是要建立图像特征点之间的关联,也就是图像特征点的匹配。特征匹配的目的在于尽量减少误差的同时获得足够多的匹配点。图论中也有匹配的问题,根据 MKP 算法<sup>[27]</sup>的思想,结合图论的知识,我们将图像特征点匹配建模成最小费用最大流问题,再使用图论中的经典算法求解。

#### 3.1 图论基础知识

##### 3.1.1 图和二部图

一个图  $G$  定义<sup>[40]</sup>为一个偶对  $(V, E)$ ,记作  $G = (V, E)$ ,其中:

- ❖  $V$  是一个集合,其中的元素称为顶点;
- ❖  $E$  是无序积  $V \times V$  中的一个子集合,其元素称为边;集合  $V \times V$  中的元

素在  $E$  中出现不止一次。

分别用  $V(G)$  和  $E(G)$  表示图  $G$  的顶点集合与边的集合。

例如：

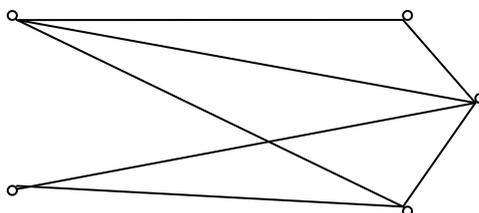


图 3.2 图示例

二部图定义为：设  $V_1$  和  $V_2$  是  $G$  的顶点子集，使得  $V_1 \cup V_2 = V(G), V_1 \cap V_2 = \emptyset$ ，且  $G$  的每一条边的一个端点在  $V_1$  中，另一个端点在  $V_2$  中，则称  $G$  为二部图 (bipartite graph)，记做  $G = (V_1, V_2; E)$ 。其中若  $V_1$  中的顶点与  $V_2$  中的每一个顶点都邻接，则称为完全二部图。如下图 3.3(a) 是一个二部图，图 3.3(b) 是一个完全二部图。

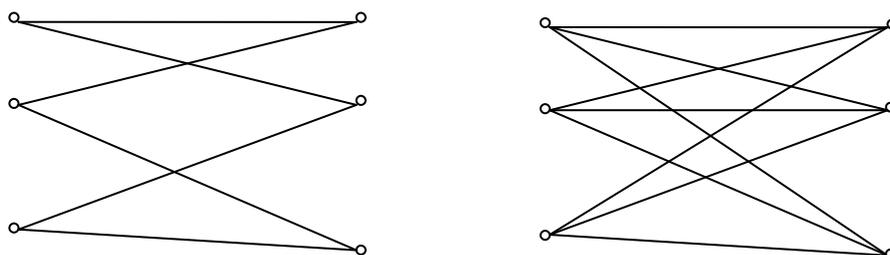


图 3.3 二部图和完全二部图

### 3.1.2 网络流

网络和网络流定义<sup>[41]</sup>：设有向连通图  $G = (V, E)$ ， $G$  的每条边  $(v_i, v_j)$  上有非负数  $c_{ij}$  作为边的容量，仅有一个入次为 0 的点  $v_s$  叫做发点 (源)，一个出次为 0 的点  $v_t$  叫做收点 (汇)，其余点为中间点，这样的网络  $G$  叫做容量网络，常被记做  $G = (V, E, C)$ 。

对任一图  $G$  中的边  $(v_i, v_j)$  有流量  $f_{ij}$ ，称集合  $f = \{f_{ij}\}$  为网络  $G$  上的一个流。可行流就是满足下列条件的流  $f$ ：

- ❖ 容量限制条件：对  $G$  中每条边  $(v_i, v_j)$ ，有  $0 \leq f_{ij} \leq c_{ij}$
- ❖ 平衡条件：对中间点  $v_i$ ，有  $\sum_j f_{ij} = \sum_k f_{ki}$  (也就是说中间点的输入量同输出量相同)；对收、发点  $v_t, v_s$ ，有  $\sum_s f_{st} = \sum_j f_{jt} = W$  (即从  $v_s$  出发的流总量等于  $v_t$  点输入的量。)  $W$  为网络流的总流量。

最大流问题其实就是在容量网络中，寻找最大的可行流。

有些问题，除了考虑流的容量，还考虑流的费用最小。这就引申出了最小费用流问题：

若已知容量网络  $G = (V, E, C)$ ，每条边  $(v_i, v_j)$  除了已给出容量  $c_{ij}$  外，还给出了单位流量的费用  $d_{ij} (\geq 0)$ ，记为  $G = (V, E, C, d)$ 。求  $G$  的一个可行流  $f = \{f_{ij}\}$ ，使得  $W(f) = v$ ，且总费用最小。

$$d(f) = \sum_{(v_i, v_j) \in E} d_{ij} f_{ij} \quad (3.3)$$

当要求  $f$  是最大流时，这问题就为最小费用最大流问题。

### 3.1.3 二部图的匹配

在图论中的二部图的匹配的定义如下<sup>[40][41]</sup>：

二部图  $G = (V_1, V_2; E)$ ，设  $M$  是  $E(G)$  的一个子集，如果  $M$  中任意两条边在  $G$  中都没有公共端点，则称  $M$  是  $G$  的一个匹配。 $M$  中的一条边的两个端点叫做在  $M$  下是配对的。如果不存在另一个  $M'$  使得  $|M'| > |M|$  ( $|M|$  表示集合  $M$  中边的个数)，则称  $M$  为最大匹配。

给出一个例子如下图，图  $G$  是一个图，图 3.4 (a) 中由边  $e_1$  和  $e_2$  组成的集合  $M$  可以叫做  $G$  的一个匹配。图 3.4 (b) 中由边  $e_1, e_2, e_3$  组成的集合  $M'$  可以叫做图  $G$  的一个最大匹配。

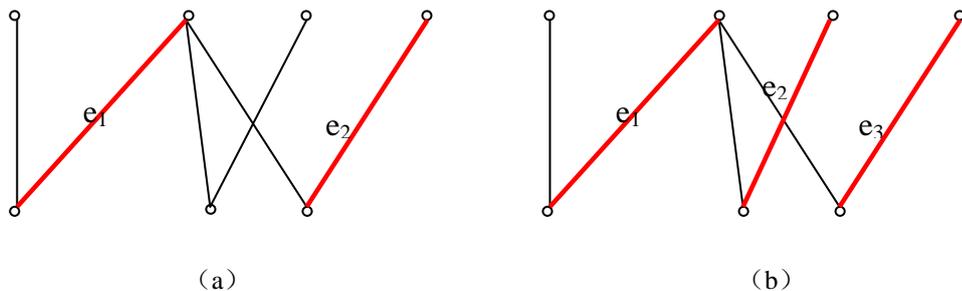


图 3.4 匹配和最大匹配

求解图论的最大匹配问题的一种常用方法是将其建模成容量网络，然后用网络最大流方法。常在二部图中增加两个新点  $v_s$ 、 $v_t$  分别作为发点、收点，并使用有向边将它们同原二部图中的顶点相连，令它们的容量全都是 1，当这个网络的流达到最大时，如果边  $e_i$  上的流量是 1，那么这条边就是匹配集中的一个元素。

如果给图的每条边增加一个可以称之为“权”的数量指标，要求使权组成的目标函数最小的问题就成了赋权二部图的匹配问题。求解时，我们在容量网络的

每条弧上再规定一个单位流量的费用，那么这个问题就变成了最小费用流问题。即要在这个网络中求得一个可行流，使得流量等于 $\nu$ ，而且总费用最小，其中 $\nu$ 可以是所要求的二部图的最大匹配数，也可以是某个给定的匹配数值。

分析图论中的匹配问题，我们发现，图像中的匹配问题和它非常相似。如果我们把待匹配图像对的特征点集 $P_1$ 和 $P_2$ 看成是赋权完全二部图的两个顶点子集，把两图待匹配的特征点之间可能存在的匹配关系构成图的边，把特征点之间的匹配度量函数的值 $w$ 看作是这些边的权，那么，图像中的匹配问题就相当于要求解这个赋权图的一个匹配，这个匹配的边的个数需要在实际中确定，且要求边的权值和最小。

### 3.2 选取度量函数

通常来说，相似性度量函数是综合块匹配和特征匹配两方面的情况决定<sup>[42]</sup>。本文中用到的 SIFT 特征点用 128 维描述符描述，信息量丰富，为了减少计算的复杂度，通常使用欧式距离来度量 SIFT 特征点对之间的相似性。而由于欧式距离可能带来由于特征向量模值的数量级不同而造成的误差，因此使用改进的匹配度量函数<sup>[27]</sup>：

$$w(p_i^1, p_j^2) = \frac{\sum_{d=1}^D (x_d - y_d)^2}{\sqrt{\sum_{d=1}^D x_d^2} \sqrt{\sum_{d=1}^D y_d^2}} \quad (3.2)$$

其中， $p_i^1 = (x_1, x_2, \dots, x_D)$  是待配准图像特征点集合 $P_1$ 中的第 $i$ 个特征点描述子， $p_j^2 = (y_1, y_2, \dots, y_D)$  是 $P_2$ 集合中的第 $j$ 个特征点描述子， $P_1 \& P_2$  是两幅待配准图像特征点的集合。

### 3.3 改进的最小费用 K 流匹配算法

#### 3.3.1 构造最小费用 K 流模型

首先，构造待匹配图像对的赋权完全二部图<sup>[27]</sup>： $G = (P^1, P^2, E, w)$ ，

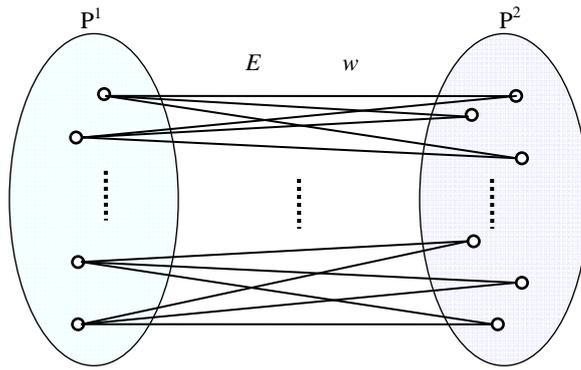


图 3.5 图像对特征点的赋权完全二部图

其中, 完全二部图的顶点集为  $P^1 \cup P^2$ ,  $P^1 = \{p_0^1, p_1^1, \dots, p_i^1, \dots, p_N^1\}$ ,  $P^2 = \{p_0^2, p_1^2, \dots, p_j^2, \dots, p_M^2\}$ , 分别代表待匹配图像  $I_1$  和  $I_2$  中的特征点的集合; 完全二部图的边的集合为:  $E = \{(i, j) | i \in P^1, j \in P^2\}$ , 边上的权为  $w: w(i, j)$ , 代表一幅图中的特征点有可能跟另一幅图中的任一点相匹配, 其匹配代价为上一节介绍的匹配度量函数决定。求解两图像的  $K (K \leq \min(M, N))$  条匹配的问题转化为赋权匹配问题如下:

在赋权的完全二部图中, 寻找包含  $K (K \leq \min(M, N))$  条边的匹配  $M = \{e_1, e_2, e_3, e_4, \dots, e_K\}, e_i \in E, i = 1, 2, \dots, K$ , 使得这  $K$  条边的权值和最小, 即目标函数  $W(M) = \sum_{i=1}^K w(e_i)$  可以取得最小值。

这样, 就把图像的特征点的匹配问题转化成了求解赋权图的匹配问题,  $K$  匹配问题通常可构造扩展的最小费用最大流模型来求<sup>[27]</sup>。

首先构造赋权二部图的容量网络形式, 它是由完全二部图和两个端点(源点:  $s$ , 收点:  $t$ )组成。

构造的网络  $N = (V, A, C, w)$  如图 3.6

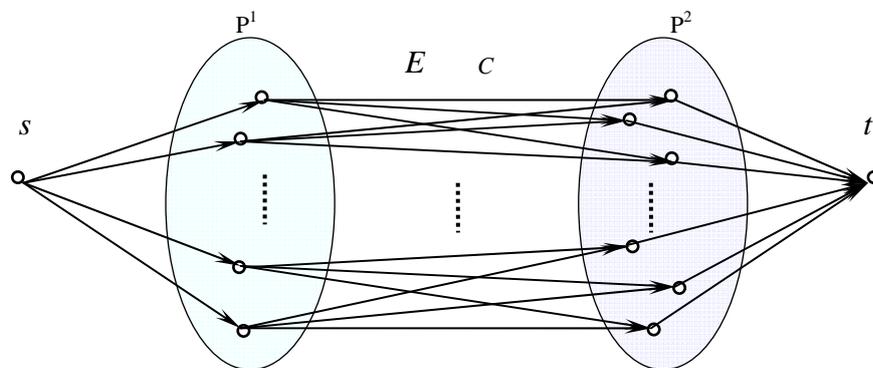


图 3.6 容量网络模型

其中顶点的集合  $V = \{s, t\} \cup P^1 \cup P^2$ ；弧的集合  $A = A_1 \cup A_2 \cup A_3$ ，

$A_1 = \{(s, i) | i \in P^1\}$ ， $A_2 = \{(i, j) | i \in P^1, j \in P^2\}$ ， $A_3 = \{(j, t) | j \in P^2\}$ ；网络上的费

用  $c_{ij} = \begin{cases} 0 & (i, j) \in A_1 \cup A_3 \\ w(i, j) & (i, j) \in A_2 \end{cases}$ ，其中  $w(i, j)$  为赋权图的边的权重；每条弧上的流

的容量为单位 1。具体来说，就是从源点  $s$  出发到  $P^1$  部分的每个顶点都有一条弧，弧的容量为单位 1，费用为 0；从  $P^2$  部分的每个顶点到收点  $t$  都有一条弧，弧的容量为单位 1，费用为 0；从  $P^1$  到  $P^2$  正是上面所介绍的赋权的完全二部图，其中每条弧的容量为单位 1，费用为二部图上每条边的权值，即前面定义的匹配度量函数值。

现在问题的求解目标就是，从源点  $s$  到收点  $t$  找到一个可行流，使得流的流量  $W(f) = K$ ，且总费用  $d(f) = \sum_{(v_i, v_j) \in E} c_{ij} f_{ij}$  最小。

用线性规划数学模型表示为：

$$\min \sum_{(i, j) \in A} C_{i, j} x_{ij} = \min \sum_{(i, j) \in A_2} C_{i, j} x_{ij} \quad (3.4)$$

$$\begin{cases} \sum_{j \in O(i)} x_{ij} - \sum_{j \in I(i)} x_{ji} = \begin{cases} k & i = s \\ 0 & i \in P^1 \cup P^2 \\ -k & i = t \end{cases}, & O(i) = \{j \in V | (i, j) \in A\} \\ & I(i) = \{j \in V | (j, i) \in A\} \\ 0 \leq x_{ij} \leq 1 \end{cases} \quad (3.5)$$

其中，公式 3.5 是对公式 3.4 中  $x_{ij}$  的约束，其含义是代表上面介绍过的网络可行流的平衡条件。根据流的方向不同可以定义不同方向的匹配。

求解网络中的最小费用  $K$  流，可以通过改进模型，将问题转化为最小费用最大流，再利用图论中的经典算法求解。

### 3.3.2 利用改进对偶算法求解模型

#### 3.3.2.1 最小费用最大流模型

将最小费用  $K$  流模型转化为最小费用最大流模型的方法是：

在网络中增加附加源点  $s'$ ，让  $s'$  连接到原始的源点  $s$ ，这条弧  $(s', s)$  上的容积和费用分别设为  $K$  和 0，增加此源点的目的就是为了将网络上的最大流约束为  $K$ 。这样，新网络上的最小费用最大流就正好等于原来网络上的最小费用  $K$  流。新网络如下所示：

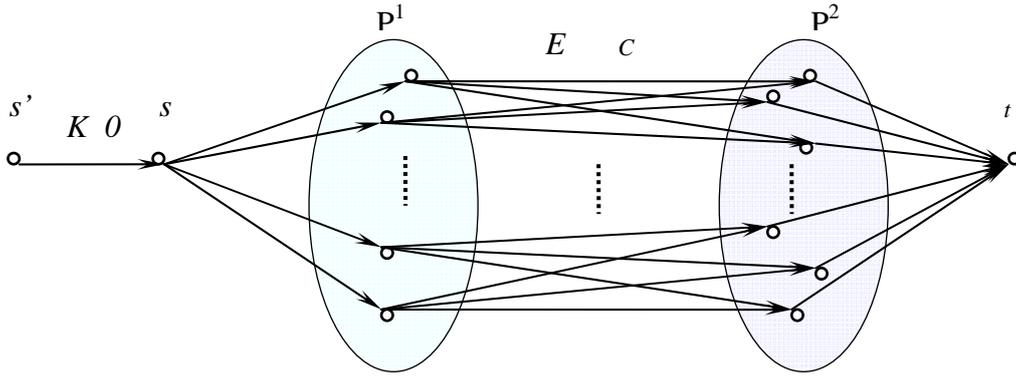


图 3.7 增加附加源点的网络

使用原 MKP 算法的求解方法<sup>[27]</sup>求解这个模型，需要提前人工输入  $K$  的值，但是实际应用中，我们无法提前获知  $K$  的值，我们希望的是输入两幅图像自动获得这两幅图像的匹配点对，这些点对的数目在保证精确度的情况下尽可能得多。因此，改进算法的目标是：在求解的时候找到可以自动获得较优匹配数目  $K_b$  的方法。

### 3.3.2.2 基于对偶算法的模型求解方法

求解网络最小费用流的算法有很多，比如标号算法、原始算法、对偶算法等等，在本文的问题中，我们需要在运算过程中确定  $K$  的值，因此要找到一个每一步调整得到的结果都是全局最优解的解法，对偶算法基本满足这个要求。

#### 1) 对偶算法的基本思路

先找到一个流量为  $W(f^{(0)}) < K$  的最小费用流  $f^{(0)}$ ，然后寻找从  $s$  到  $t$  的可增广链  $\mu$ （增广路径的概念参看图论<sup>[40][43]</sup>），用最大流方法将  $f^{(0)}$  调整到  $f^{(1)}$ ，使得  $f^{(1)}$  的流量为  $W(f^{(0)}) + \theta$  流量下的最小费用流，不断进行直到  $W(f^{(k)}) = K$ ，即求得最大流为止。

#### 2) 长度网络的构造

长度网络是求解过程中的辅助网络，在网络中  $N$  求解关于  $f$  的最小费用可增广链等价于在其长度网络  $L(f)$  中求从  $s$  到  $t$  的最短路径。

对网络  $N = (V, A, C, w)$  构造长度网络  $L(f)$  的方法<sup>[41]</sup>是：保持原网络各点，每条边用两条方向相反的有向边代替，各边的权重  $l_{ij}$  按如下规则定：

$$\diamond \text{ 当边 } (v_i, v_j) \in E, \text{ 令 } l_{ij} = \begin{cases} w_{ij} & f_{ij} < c_{ij} \\ +\infty & f_{ij} = c_{ij} \end{cases}, \text{ } +\infty \text{ 含义是当边的流量已经等于容}$$

量，那么这条边已经饱和，无法再增大流量，否则要花费很高的代价，

因此权为  $+\infty$  的边可以从网络中剔除。

❖ 当边  $(v_j, v_i) \in E$  为原网络  $N$  中  $(v_i, v_j)$  的反向边，那么，令

$$l_{ij} = \begin{cases} -w_{ij} & f_{ij} > 0 \\ +\infty & f_{ij} = 1 \end{cases}, \text{ 此处 } +\infty \text{ 的含义是此边流量已减少到 } 0, \text{ 无法再减少,}$$

也要去掉权为  $+\infty$  的边。

### 3) 应用对偶算法求解

对偶算法的基本步骤是：

- a) 取零流为初始可行流，即  $f^{(0)} = \{0\}$ ；
- b) 如果有  $f^{(k-1)}$ ，流量为  $W(f^{(k-1)}) < K$ ，构造长度网络  $L(f^{(k-1)})$ ；
- c) 在长度网络  $L(f^{(k-1)})$  中求从  $s$  到  $t$  的最短路。如果不存在最短路，则网络中的流已为最大流，停止；否则到 d)；
- d) 在原网络  $N = (V, A, C, w)$  中与这条最短路相应的可增广链  $\mu$  上，作  $f^{(k)} = f_{\mu}^{(k)} \theta$ ，即用最大流算法进行调整。其中  $\theta$  为路径上可增加流量的最大值，调整后  $f^{(k)}$  的流量为  $W(f^{(k-1)}) + \theta$ ，若  $W(f^{(k-1)}) + \theta = K$  则停止，否则让  $f^{(k)}$  代替  $f^{(k-1)}$  返回 b)；

对本文中构造的网络来说，由于每条边的容量都是 1，因此每次调整都是使得流量增加一个单位，即  $W(f^{(k)}) = W(f^{(k-1)}) + 1$ ，而且这个时候获得的是当前匹配数目的最优解，有着这个数目下的最小的费用和  $C_k$ ，那么增加一条边所造成的总费用的增量为  $C_k - C_{k-1}$ ，我们可以从增加一条边所造成的总费用增量入手，寻找合适的阈值使得若增量大于此阈值，则说明匹配个数的再度增加会大大降低匹配的精度，应该在这个匹配个数上停止，选取这个个数作为  $K$ 。

我们通过多幅图像实验发现，每一次调整后，总费用的增量  $d_z^k (d_z^k = C_k - C_{k-1})$  比前一次的增量  $d_z^{k-1}$  要大，而且刚开始第二次调整费用是第一次调整的费用增量（即流量为 1 时的总费用）的两倍左右，然后以第二次的增量为基础渐渐的增长。根据最近邻的思想，距离太大的是匹配对的可能性较小，则如果调整时费用增加过大，就可能加入了错误匹配对。通过比对数据集的一共 105 幅图片，我们发现当费用的增长超过了第一次调整费用增量的若干倍时，加入匹配集合中的错误匹配就会迅速增加，使得误匹配率大幅升高，因此，我们将阈值设为：

$$T = \xi d_z^1 = \xi(C_1 - C_0) = \xi w_1 \quad (3.6)$$

其中,  $w_1$  是第一次调整后的费用, 即流为 1 时的最小费用;  $\xi$  为调节因子, 经过实验我们发现应该取 1.5 到 2.5 之间的值。

求得匹配集后, 这其中还是有可能存在不满足要求的匹配。于是引入检验算法, 利用弧上的匹配度量比值和双向约束的方法来检验每条匹配, 去掉风险性高的匹配。

### 3.3.3 改进的最小费用 K 流匹配算法流程

---

算法: 改进的最小费用K流匹配算法

---

输入: 两幅待匹配图像

1. 提取两幅图像的SIFT特征点集合  $P^1$  和  $P^2$ 。
2. 计算各点之间的匹配度量函数  $w(p_i^1, p_j^2)$ , 作为容量网络的权值。
3. 预处理。为了缩小搜索空间, 先删除一部分不可能匹配的弧, 比较弧上所连接的两点, 如果它们的主方向比值超过设定的阈值, 则将这条弧从网络上删除。
4. 利用改进算法求解
  - (1) 取零流为初始可行流, 即  $f^{(0)} = \{0\}$ ,  $K$  初值设为无穷大。
  - (2) 如果有  $f^{(k-1)}$ , 流量为  $W(f^{(k-1)}) < K$ , 构造长度网络  $L(f^{(k-1)})$ 。
  - (3) 在长度网络  $L(f^{(k-1)})$  中求从  $s$  到  $t$  的最短路。如果不存在最短路, 则网络中的流已为最大流, 停止; 否则到 (4)。
  - (4) 求本次调整的费用增量, 如果费用增量超过阈值 (公式 3.6), 则图中的流量已为合适的流量, 设置  $K = W(f^{(k-1)})$ , 停止; 否则到 (5)。
  - (5) 在原网络  $N = (V, A, C, w)$  中与这条最短路相应的可增广链  $\mu$  上, 作  $f^{(k)} = f_{\mu}^{(k)} \theta$ , 即用最大流算法进行调整。其中  $\theta$  为路径上可增加流量的最大值, 调整后得  $f^{(k)}$  的流量为  $W(f^{(k-1)}) + \theta$ , 然后让  $f^{(k)}$  代替  $f^{(k-1)}$  返回 (2)。
5. 此时图中流量为 1 的边即为一个匹配, 边连接的两点为匹配集中的一对点。
6. 用检验算法来检查第 5 步中生成的匹配集, 去掉风险性高的匹配点对。

输出: 较为精确的全局最优匹配集

---

算法中的在长度网络中求解一条最短路径这一步, 是通过 Bellman-Ford 算法求解的, 这个算法的算法时间复杂度为  $O(|V| \cdot |E|)$ 。因此以上算法的时间复杂

度最多为  $O(K \cdot |V| \cdot |E|)$ ,  $K \leq \min(M, N)$ 。

### 3.4 实验对比与分析

为了验证本文提出算法的有效性，在测试集合上进行了实验测试。

测试集合：ICCV2005计算机视觉大赛<sup>[44]</sup>的测试集Test4(28张图片)、Final5(38张图片)和SDL实验室测试集合(39张图片)。测试集合中的图片综合了各种情形，如：不同视角拍摄、不同光照和明暗变换、相似结构多的情形等。

实验条件：2.83GHz，3GB，VC++6.0，Windows XP。

在这三个测试集上，将改进的MKP算法、MKP算法和Lowe算法<sup>[21]</sup>进行了比较，结果如图3.8。图中的三组图为一组，左上角是Lowe算法的匹配结果，右上是原MKP算法，下面是改进的MKP算法。

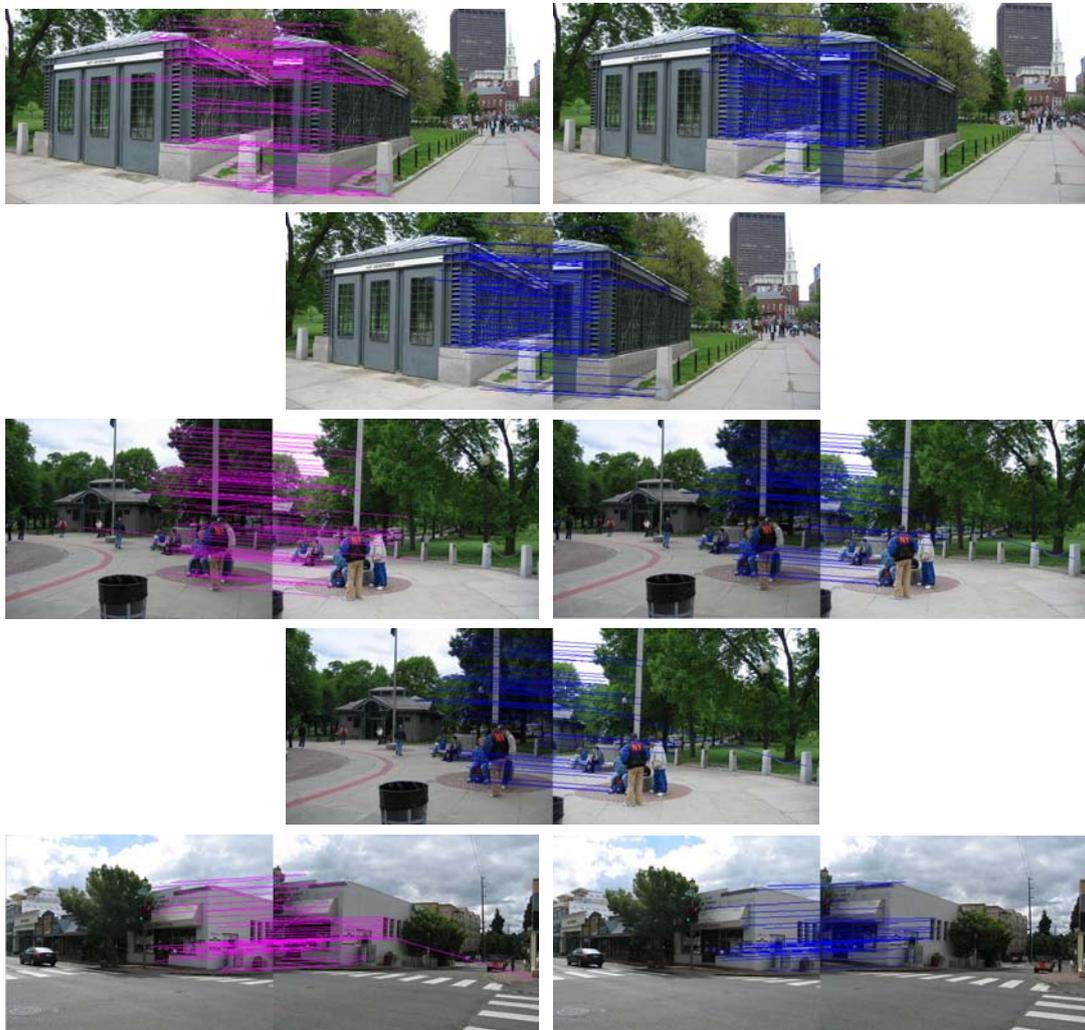




图 3.8 匹配结果

结果显示，原MKP算法有最优的正确率，但是算法在实验中需要凭人工干预来使之取到有最优正确率的  $K$  值；而改进的自动匹配算法可以自动取得比较合适的  $K$  值，但不能保证选取的  $K$  值是有最优的，因此正确率有所下降，但准确率仍优于Lowe的算法。

此外，还使用了Information Retrieval Systems的衡量方法中的Precision来量化实验结果。Precision的定义如下，简称为P：

$$Precision = \frac{\text{正确匹配的数目}}{\text{正确匹配的数目} + \text{错误匹配的数目}}$$

其中匹配的数目指的是匹配集中匹配特征点对的数目。

表 3.1 三种方法的 Precision 比较

测试集	图像数目	P of Lowe's approach	P of MKP	P of improved MKP
Test4	28	0.70	0.85	0.75
Final5	38	0.71	0.87	0.77
Our image sets	39	0.79	0.89	0.82
Total	105	0.73	0.87	0.78

### 3.5 小结

本章介绍了基于改进的最小费用  $K$  流的特征点匹配算法。根据已有的 MKP 算法的思想，分析了图论中的图匹配和图像匹配的相似性，把图像特征点匹配建模成网络流中的最小费用最大流问题，并用改进的对偶算法进行求解，实验表明改进后的 MKP 算法是可行的，可自动获得较精确的匹配点集。

## 第四章 基于改进 RANSAC 的图像拼接算法

在特征点提取、匹配后，我们得到了两幅图之间的匹配点集，接下来需要根据这个匹配点集估算出变换参数，并对变换后的图像进行重采样和融合，生成拼接后的图像。

### 4.1 RANSAC 算法

在视觉领域，一个核心问题是模型参数的鲁棒估计，因为在给定数据集中常常有不符模型的外点。外点是整体与假设模型不太符合的数据<sup>[45]</sup>，一般来说，外点分为大误差数据和完全错误数据。如何在含有大量外点的数据集中估计出正确的模型参数，是人们研究的重点。

RANSAC(Random Sample Consensus)算法，即随机抽样一致性算法，由 Fischle 和 Bones 在 1981 年首次提出的<sup>[31]</sup>，是一种鲁棒的拟合数据的算法，可以从包含多个外点的数据集中筛选内点并准确估计模型参数，有很强的容错能力，可以处理有超过 50% 错误率的数据。现已在图像拼接领域广泛应用。

#### 4.1.1 RANSAC 算法的基本思想

RANSAC 算法的基本思想是重复  $N$  次最小样本数抽样，对每次抽样获得的模型参数进行全数据检验，区分内点外点，最后选取有内点个数最多的一次抽样，并且从这些内点中重新估计模型的参数值。下面用直线拟合为例介绍这种算法。

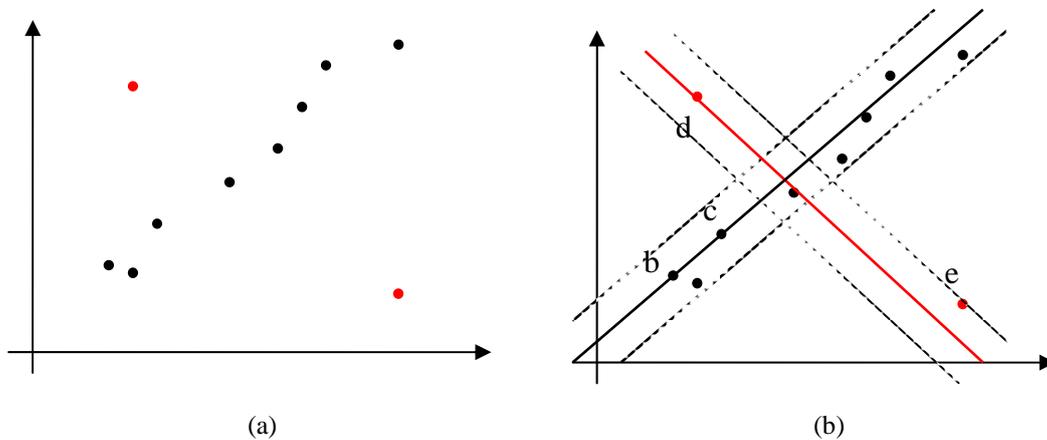


图 4.1 RANSAC 直线拟合示例图

数据集合如图 4.1 所示，我们要通过这个数据集合拟合一条直线。这个集合中有 8 个属于内点，2 个属于外点。用 RANSAC 拟合的过程是：

Step1 选取两个点，生成一条直线

Step2 通过一定的距离阈值在其它的几个点中寻找这条直线的内点，得到一个内点集，这些内点也叫做这条直线的支撑。

Step3 重复前两步数次，最后有最大支撑集的直线是样本点集的最好拟合。如图 4.1 所示，选取点 b、c 求得的直线 bc 有最多个数的内点，它是这个集合的最好估计，而 d、e 两个外点会被剔除掉。

Step4 用得到的 8 个内点来进行最小二乘拟合，得到最接近理想解的直线。

其中有 3 个量需要确定<sup>[14]</sup>：内点与外点的距离阈值  $t$ ，采样次数  $N$ ，内点的个数。

(1) 距离阈值：假设提取特征点的误差符合均值为 0，标准差为  $\sigma$  的高斯分布，这时高斯变量的平方和是  $d_v^2 = d(x_i; Hx_i)^2$ ，符合自由度为  $m$  的  $\chi_m^2$  分布。则符合  $\chi_m^2$  分布的随机变量的概率小于积分的上限，即：

$$F_m(k^2) = \int_0^{k^2} \chi_m^2(\xi) d\xi < k^2 \quad (4.1)$$

则可得阈值为：

$$\begin{cases} \text{内点} & d_v^2 < t^2 \\ \text{外点} & d_v^2 \geq t^2 \end{cases} \quad \text{其中, } t^2 = F_m^{-1}(\alpha)\sigma^2 \quad (4.2)$$

其中， $\alpha$  表示符合内点条件的特征点对是内点的先验概率，一般取 0.95。则  $F_m^{-1}(0.95)\sigma^2 = 5.99\sigma^2$ ，对提取的特征点计算出  $\sigma$  就可以完全确定阈值。

(2) 随机采样的次数  $M$ ：理想的情况是采尽所有的组合，但这样次数非常巨大。则要求在一定得置信概率下，这  $M$  组抽样中至少有一组的数据全是内点。设  $P$  为置信概率， $p_1$  为正确数据率，即点集中任何一对匹配点是内点的概率，计算模型参数需要的最小数据量为  $m$ ，则采到  $M$  次时有：

$$(1 - p_1^m)^M = 1 - P \quad (4.3)$$

由此确定采样次数：

$$M = \log(1 - P) / \log(1 - (p_1^m)) \quad (4.4)$$

(3) 内点个数：如果总的特征点对为  $n$  对，则内点数为  $p_1 n$ 。实验中有时会多次迭代直到内点的数目基本趋于一致。

RANSAC 算法充分利用了数据集中的所有数据，根据可接受的误差范围将

它们分成内点和外点。它利用了内点数据比较准确的特点来剔除不准确的测量数据，因此，得到的结果是一种最优化的结果<sup>[31]</sup>。

#### 4.1.2 用于图像拼接的 RANSAC 算法

前面已经介绍了投影变换模型，它可以描述相邻图像间的变换关系<sup>[50]</sup>。

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (4.5)$$

$(x \ y \ 1)^T$  和  $(x' \ y' \ 1)^T$  是两幅相邻图像  $I$  和  $I'$  上的一对匹配点  $m$  和  $m'$  的齐次坐标。则每对匹配点可以产生两个方程：

$$\begin{cases} x'(h_{31}x + h_{32}y + 1) = h_{11}x + h_{12}y + h_{13} \\ y'(h_{31}x + h_{32}y + 1) = h_{21}x + h_{22}y + h_{23} \end{cases} \quad (4.6)$$

代入 4 对不共线的匹配点，可得到 8 个方程：

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1'x_1 & -x_1'y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y_1'x_1 & -y_1'y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2'x_2 & -x_2'y_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -y_2'x_2 & -y_2'y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3'x_3 & -x_3'y_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -y_3'x_3 & -y_3'y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4'x_4 & -x_4'y_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -y_4'x_4 & -y_4'y_4 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} x_1' \\ y_1' \\ x_2' \\ y_2' \\ x_3' \\ y_3' \\ x_4' \\ y_4' \end{bmatrix} \quad (4.7)$$

这样，配准问题转化为了求解 8 参数投影变换矩阵  $H$  的问题，理论上来说，在相邻两幅图像中选取 4 对匹配点就可以确定一个变换矩阵，但是受噪声、特征点描述子、模型误差、错误匹配等因素的影响，常常需要选取尽量多的匹配点坐标，然后利用 LM 优化算法求解参数。估计这个 8 参数模型的 RANSAC 算法如下：

---

#### 算法 4.1: RANSAC 算法

---

输入：匹配点集+投影变换模型

1. 随机选取四对粗匹配点。
2. 用这四个点计算投影变换模型的 8 个参数，得到变换矩阵  $H$ 。
3. 检验其它点对是否符合这个矩阵  $H$ ，小于距离阈值的为内点，大于距离阈值的为外点。

4. 重复上述步骤，采样并检验  $M$  次，得到这  $M$  次中的有最多支撑点的  $H$  的内点集作为最终内点集。
5. 依据最终内点集用 LM 优化算法对  $H$  进行优化估计。

输出：变换矩阵  $H$

#### 4.1.3 算法计算量分析

在上一节已经分析到，在一定置信概率下，求得满足要求的最小抽样数  $M$  如下<sup>[46]</sup>：

$$M = \log(1 - P) / \log(1 - (p_1)^m) \quad (4.8)$$

其中， $P$  为置信概率； $p_1$  为正确数据率，即点集中任何一对匹配点是内点的概率； $m$  为计算模型参数需要的最小数据量。

进行图像配准时， $m=4$ ，随着内点比例的减少， $P$  不同时，理论上需要的随机采样次数  $M$  的变化情况如下表：

表 4.1 随机采样次数随内点比例变化而变化的情况

	集合中内点的比例 (%)									
	95	90	80	70	60	50	40	30	20	
P=0.95	2	3	6	11	22	47	116	368	1870	
P=0.99	3	5	9	17	34	72	178	576	2876	

可以看到，随着置信概率和外点比例升高，理论上的次数升高的比较快，这会导致 RANSAC 算法效率的下降。而据实验数据看，实际上的次数比理论还要高。

设样本集一共有  $N$  个点；从原始数据中取得一组抽样所需时间为  $t_s$ ；由一组抽样计算模型的时间为  $t_g$ ；用一个数据检验模型参数所用的时间为  $t_e$ ；则 RANSAC 算法的计算时间  $T$  为<sup>[46]</sup>：

$$T = M(t_s + t_g) + M(N - 4)t_e \quad (4.9)$$

分析上述式子可以看出，RANSAC 的计算时间分为两部分：一部分是  $M$  组抽样和估计模型参数组的时间；另一部分是在全数据集上检验这  $M$  个模型参数组的时间。由于我们要保证一定的置信率，在内点率和模型固定的情况下， $M$  的值是不能减少的；另外，当全数据集非常大时，对于模型的全数据集检验是个非常耗费时间的过程，因此，如果要提高算法的运行效率，我们需要在减少参与全数据集检验的模型参数组个数上入手。

## 4.2 改进 RANSAC 算法

对于图像拼接问题来说，要减少参与检验的模型参数组的个数，我们可以考虑在进行全数据检验前，先对用来拟合模型参数组的四个点本身进行评价检验，或者说，进行指导性的选点。

假设某次抽取的样本集中都是内点，他们估计出的模型仍可能没有足够多的内点支撑，仍以图像拟合直观来解释这个问题。比如我们某次抽取的样本集中的点  $s_1$  和  $s_2$ ，这两个点决定的直线如图，则支撑这条直线的内点数比较少，甚至还可能把实际上的外点包括在内，因此，我们应该分析应用 RANSAC 的问题本身的性质，以此来进行有指导性的抽样，或者说找到判定抽样好坏的标准。

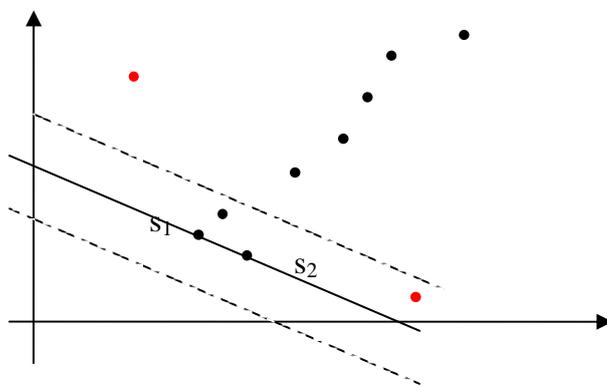


图 4.2 直线拟合示例图

针对图像拼接这个问题来说，通常情况下，一个像素点会同其相邻的像素点有着相同或相似的变换，这是由图像的连续性决定的。由于图片可能存在一些细微的局部形变，两组相距较远的区域的点所拟合的变换矩阵可能会相差较大。也就是说，我们在两个局部区域分别取点集，它们拟合出来的变换矩阵  $H$  会有所不同。符合要求的全图的变换模型，必然是综合了广泛区域的变换因素，才可能有足够多的内点支撑。由此分析，如果我们选取的是非常集中的几个点，那么拟合出来的变换矩阵  $H$  反映的是局部的变换，这个  $H$  应该是一个弱估计，也就是有较少内点支撑的变换矩阵。与此相对，如果四个点分布较广泛，那么它们拟合出来的矩阵  $H$ ，可以反映全局的变换，会获得比较多的内点的支撑。

由上面的推断，我们提出了用于图像拼接的改进的 RANSAC 算法，这个算法对于随机抽取的样本提前检验，能通过检验的样本才用来拟合模型以及参与之后的全数据检验。

### 4.2.1 类内散布矩阵

类内散布矩阵可以用来描述样本集的分散程度<sup>[47]</sup>。

在同一类模式点集  $\{x\}$  内，各样本间的距离的均方值可以写成：

$$\begin{aligned}\overline{D^2} &= E\{\|x_i - x_j\|^2\} \\ &= E\{(x_i - x_j)'(x_i - x_j)\} \\ &= E\{x_i'x_i + x_j'x_j\} - E\{x_i'x_j + x_j'x_i\}\end{aligned}\quad (4.10)$$

其中， $x_i$  和  $x_j$  是属于同一类别的相互独立的模式样本向量，利用它们之间的互相独立的性质，上式为：

$$\begin{aligned}\overline{D^2} &= 2E\{x'x\} - 2E\{x'\}E\{x\} \\ &= 2tr[E\{x'x\} - mm'] \\ &= 2tr[R - mm'] \\ &= 2tr[C]\end{aligned}\quad (4.11)$$

其中， $R$  是此类模式分布的自相关矩阵； $m$  是此类模式的均值向量； $C$  为协方差矩阵。那么，对于属于同一类的模式样本，类内散布矩阵可以表示各样本围绕它们的均值周围的散布情况，这里即此分布的协方差矩阵。

也就是说对于同一类内模式点集  $\{a^{(i)}\}_{i=1,2,\dots,K}$ ，其类内散布矩阵可表示为：

$$S = \sum_{i=1}^K \{(a^{(i)} - m)(a^{(i)} - m)^T\} \quad (4.12)$$

其中  $m = \frac{1}{K} \sum_{i=1}^K a^{(i)}$ 。

### 4.2.2 利用类内散布矩阵改进 RANSAC 算法

在 RANSAC 算法中，随机选取四个点直接进行模型参数的估计，然后进行全数据检验，上面我们已经分析过，应该先对选取的样本集进行提前检验。当样本集中的四个点足够分散时，才会得到好的参数估计，过于集中的样本点集有很大的概率获得较差的参数估计<sup>[48]</sup>。

为了度量样本集中的四个点的分散程度，我们使用归一化的所有特征点的类内散布矩阵  $T_h$ ：

$$T_h = \frac{1}{k} \sum_{i=1}^k \{(x^{(i)} - m)(x^{(i)} - m)^T\} \quad (4.13)$$

其中  $\{x^{(i)}\}, i=1,2,3,\dots,k$  参考图像的参与匹配的点的集合;  $m = \frac{1}{k} \sum_{i=1}^k \{x^{(i)}\}$ ;  $T_h$  是一个  $2 \times 2$  的矩阵, 对角线上的两个元素分别代表了这些点在  $x$  轴和  $y$  轴上的分散度。

让  $T_h = \begin{bmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{bmatrix}$  代表所有点的类内散布矩阵,  $T_h' = \begin{bmatrix} a_{00}' & a_{01}' \\ a_{10}' & a_{11}' \end{bmatrix}$  代表某次抽取

的样本中的四个点的类内散布矩阵, 元素  $(a_{00}, a_{11})$  和  $(a_{00}', a_{11}')$  分别代表了所有点和四个点的散布情况。如果下面的式子为真, 则我们认为这次抽样的样本集属于“好”样本集, 它们可以用来估计变换矩阵, 进而进行全数据集检验, 如果结果是假, 我们就重新抽取样本集。为了获得与原 RANSAC 相同的置信度, 我们可以增加抽样次数  $M$ 。

$$a_{00}' \geq \varepsilon a_{00} \parallel a_{11}' \geq \varepsilon a_{11} \quad (4.14)$$

其中  $\varepsilon$  是尺度影响因子, 它是为了调节一些分布较远的外点的影响, 经过多次实验我们发现取 0.125-0.3 可以获得较好的效果。

改进的 RANSAC 的流程如图 4.3 所示。

流程中加粗的框内是加入的预检验部分。由于图像匹配中的点, 大部分都在两幅图像相重叠的区域里, 因此取得过于集中的点的概率很大, 通过这个方法可以过滤掉大量的错误模型参数。

### 4.2.3 运算时间分析

上述改进的 RANSAC 算法的计算时间为:

$$\begin{aligned} T_i &= t_t + a t_m + m(t_s + t_g + t_m + (N-4)t_e) \\ &= t_t + (a+m)t_m + m(t_s + t_g + (N-4)t_e) \end{aligned} \quad (4.15)$$

其中  $t_t$  是计算点集中所有匹配点的类内散布矩阵  $T_h$  上的时间;  $a$  是被扔掉的样本的数目;  $t_m$  是计算某次样本集的散布矩阵  $T_h'$  上的时间;  $m$  是进行全数据检验的参数估计的数目, 即迭代的次数;  $t_s$  是抽样时间, 即随机抽取四个点的时间;  $t_g$  是生成一个假设估计上的时间;  $N$  是匹配点集中匹配点对的数目;  $t_e$  是用来对全数据集进行假设检验的时间; 采样总次数  $M = m + a$ 。

普通 RANSAC 的计算时间为:

$$T_n = M(t_s + t_g + (N-4)t_e) \quad (4.16)$$

比较两个运算时间, 可以发现, 两个时间的差如下所示:

$$T_n - T_i = (M - m)(t_s + t_g + (N - 4)t_e) - (t_i + Mt_m) \quad (4.17)$$

其中,  $t_i$  和  $t_m$  的值都非常小, 而进行一次全数据检验的时间  $(N - 4)t_e$  的值很大, 尤其是当参与模型拟合的点对数目较多时。因此, 此算法可以有效减少参数估计时间, 提高效率。

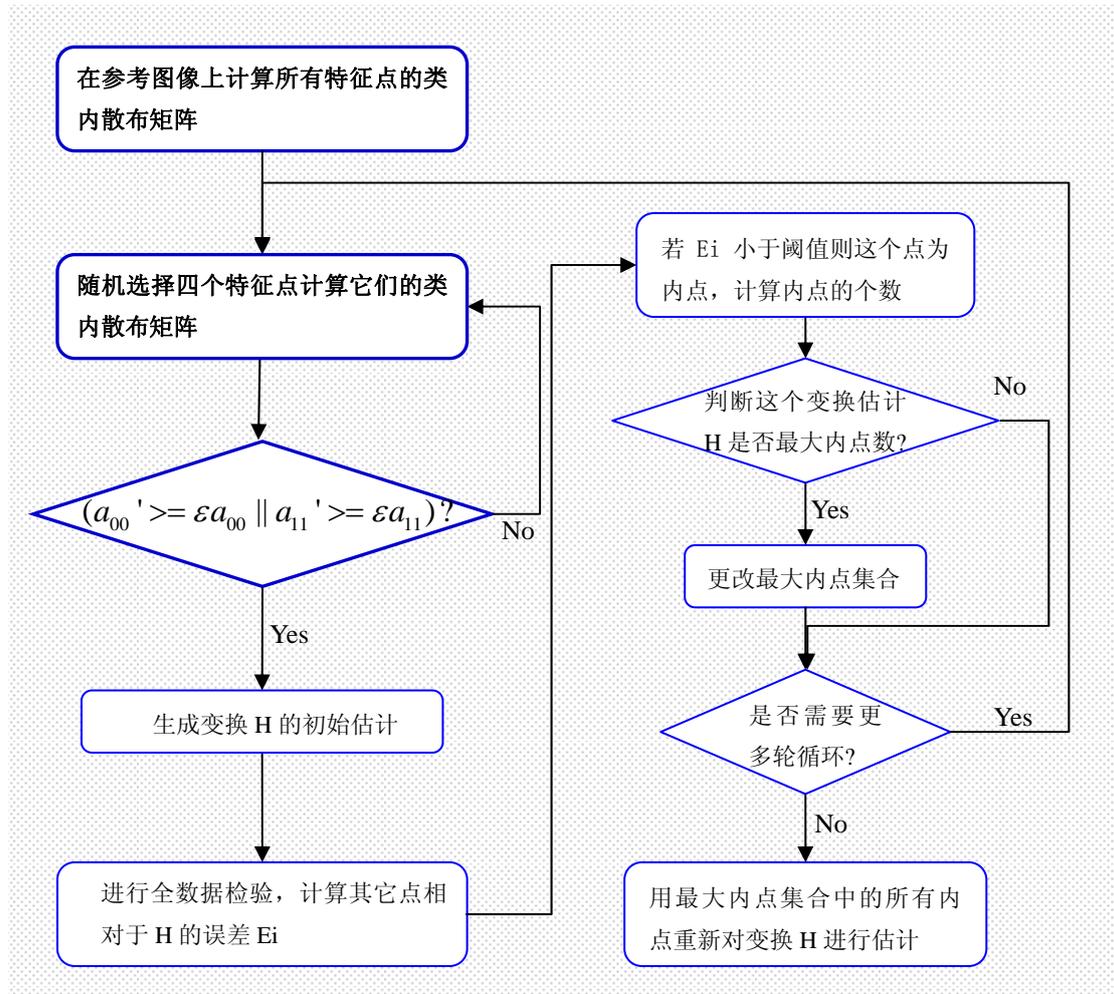


图 4.3 改进的 RANSAC 算法流程

### 4.3 实验对比与分析

我们在20对小尺寸照片（320\*240）和8对大尺寸照片（800\*600）上比较了我们的算法和原始RANSAC算法的准确性和计算效率。这些图片是随机从ICCV2005 Computer Vision Contest数据集和我们自己的数据集中取出。每一个算法在每一幅图像上重复40次以使结果具备统计上的代表性。

实验条件：2.83GHz，3GB，VC++6.0，Windows XP。

### 4.3.1 准确性

为了验证其在错误率高的情况下的效果，我们故意放大了粗匹配时的阈值，使得错误率较高。我们对比了在每幅图上使用两种算法所得到的变换矩阵  $H$ ，结果显示我们的算法可以在有效减少迭代次数的情况下，得到同原 RANSAC 算法同样的计算质量，所得的变换矩阵各对应元素之间的差不超过 0.02，考虑到原矩阵元素的数量级，可以认为所得矩阵近似相等。



图 4.4 实验结果比较图 1

一组实验结果比较如图4.4，在粗匹配之后，得到了135对匹配点，其中有100个正确匹配。使用原始的RANSAC算法，最后所得内点数的平均值是98.5，平均的迭代次数是24.05，计算的结果矩阵为  $H_1 = \begin{bmatrix} 1.316 & -0.042 & -249.254 \\ 0.106 & 1.268 & -73.690 \\ 0.001 & 0.000 & 1.000 \end{bmatrix}$ 。使用我们的算法，

最后所得内点数平均值为99.38，平均的迭代次数是9.55，计算的结果矩阵为

$H_2 = \begin{bmatrix} 1.317 & -0.042 & -249.270 \\ 0.105 & 1.269 & -73.672 \\ 0.001 & 0.000 & 1.000 \end{bmatrix}$ 。4.4(a)和图4.4(b)显示了拼接结果，可以看到拼接效果都

在人眼可接受的范围内。对比这两个矩阵，  $H_1 - H_2 = \begin{bmatrix} -0.001 & 0.000 & -0.016 \\ 0.001 & -0.001 & -0.018 \\ 0.000 & 0.000 & 0.000 \end{bmatrix}$ ，近似相

同，而最终迭代次数由原来的24.05下降到9.55



图 4.5 实验结果比较图 2

第二组实验结果比较如图4.5, 在粗匹配之后, 得到了62对匹配点, 其中有46个正确匹配。使用原始的RANSAC算法, 最后所得内点数的平均值是42.5, 平均的迭代次数是20.05, 计算的结果矩阵为

$$H_1 = \begin{bmatrix} 1.193 & -0.026 & -184.779 \\ 0.089 & 1.118 & -9.118 \\ 0.001 & 0.000 & 1.000 \end{bmatrix}。$$

使用我们的算法, 最后所得内点数平均值为43.5, 平均的迭代次数是14.3, 计算的结果矩阵为

$$H_2 = \begin{bmatrix} 1.193 & -0.026 & -184.785 \\ 0.088 & 1.119 & -9.129 \\ 0.001 & 0.000 & 1.000 \end{bmatrix}。$$

图4.5(a)和图4.5(b)显示了拼接结果, 可以看到拼接效果都在人眼可接受的范围内。对比这两个矩阵,  $H_1 - H_2 = \begin{bmatrix} 0.000 & 0.000 & -0.006 \\ 0.001 & -0.001 & -0.010 \\ 0.000 & 0.000 & 0.000 \end{bmatrix}$ , 近

似相同, 而最终迭代次数由原来的20.05下降到14.3。

### 4.3.2 计算效率

在最后获得同样的计算精度的情况下, 我们比较了算法的计算效率。我们从减少迭代次数和缩短计算时间两个方面来比较算法。

我们在20幅小尺寸图像(320\*240)上计算了迭代的次数如图4.6, 其中横轴为不同的图像编号, 纵轴为迭代次数, 即参与全数据检验的模型参数组的个数。由于数量级不同, 我们将20幅图像的结果分两部分用两个不同刻度数量级的坐标系进行了展示:

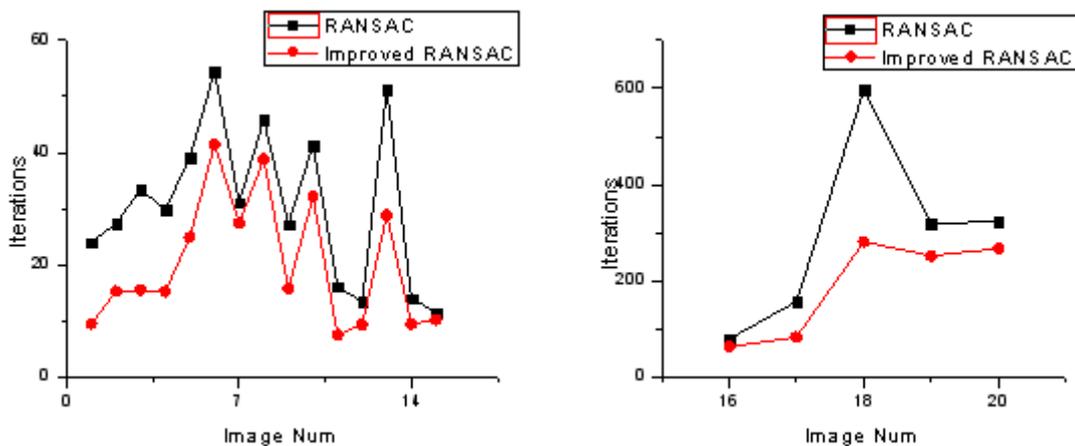


图 4.6 迭代次数对比

对不同图减少的迭代次数比率不同, 这是由于特征点的分布情况不同造成的。比对所有的实验图片我们发现, 如果特征点在图片中的分布过于集中, 则算法的提高性不是很好, 这是由于特征点过于集中时, 局部变换的影响比较微弱造成的。同样, 尺度因子  $\varepsilon$  将影响最终的效果。

我们在 8 幅较大尺寸图片（800\*600）上比较了运行时间如下图，其中横轴代表图片的编号，纵轴代表运行的时间：

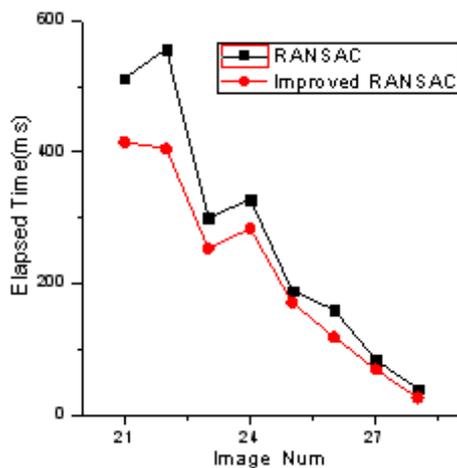


图 4.7 时间对比

选取尺寸较大的图片进行运行时间的比较是由于其所用时间较长，结果不易受到其他因素的干扰。可以看到，在同样的准确度的条件下，改进的 RANSAC 算法降低了约 19.72% 的运行时间。

### 4.3.3 多幅图像拼接实验结果

本节使用两组图片（图 4.8）来展示本文算法的各个关键步骤，得到拼接后的全景图像（180 度左右）。



图 4.8 原图像

第一步：投影到柱面。



图 4.9 柱面投影

第二步：特征点提取，选用 SIFT 特征点。

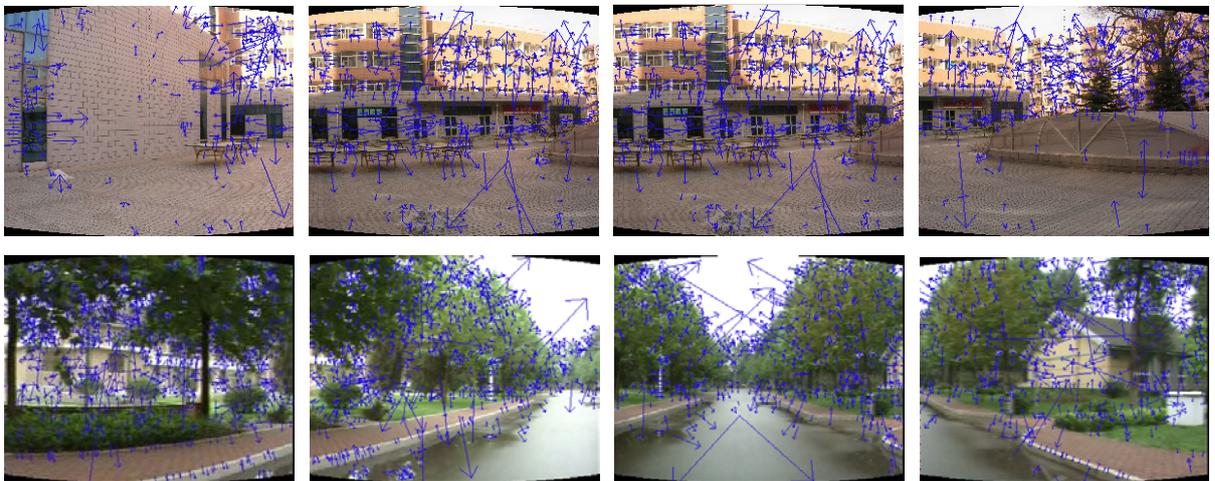
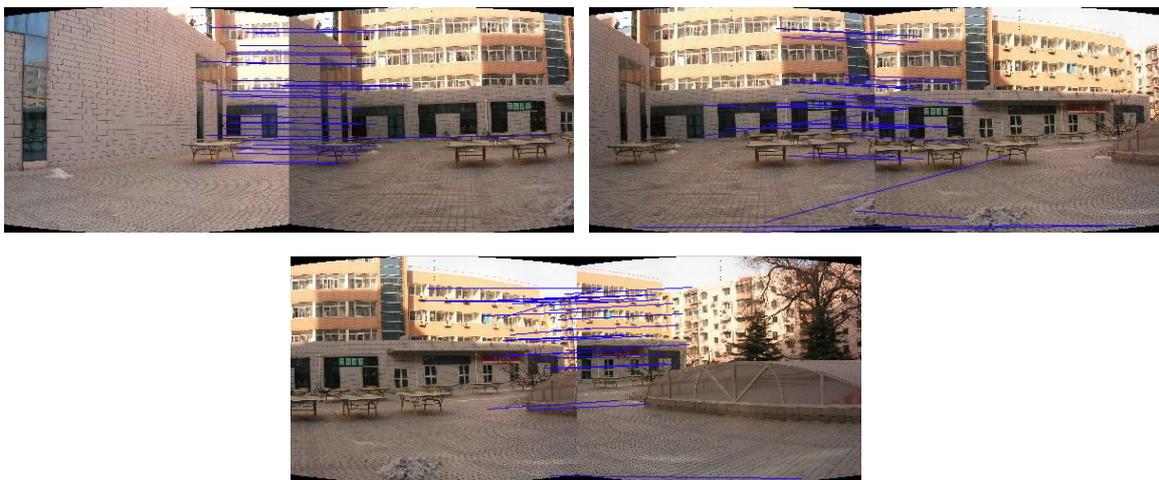


图 4.10 特征点提取

第三步：特征匹配，使用改进的 MKP 算法。



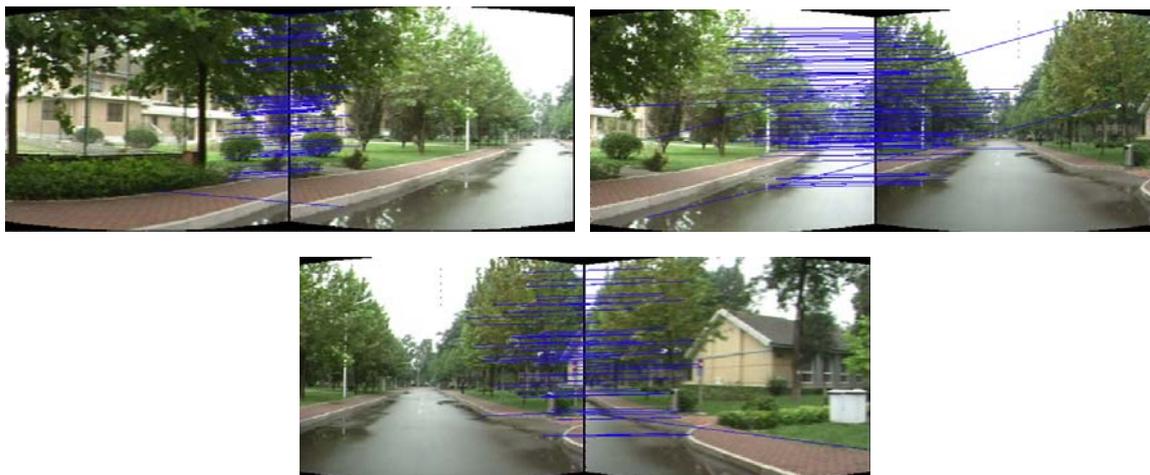


图 4.11 特征点匹配

第四步：图像变换估计和融合。使用改进 RANSAC 算法估计变换模型；然后用简单的最近邻插值法和加权平均融合技术生成最终图像。



图 4.12 最终效果图

#### 4.4 小结

本章介绍了基于改进 RANSAC 的图像拼接算法。首先介绍了 RANSAC 算法和类内散布矩阵，然后提出了新的可应用于图像拼接的改进 RANSAC 算法，并在公共测试集合上进行了测试证明了其可行性；最后给出了应用本文拼接算法的多幅图的拼接效果图。

## 第五章 总结与展望

本文研究了基于图像特征的自动图像拼接算法,针对算法中的特征匹配和变换模型估计这两个部分做了创新与改进:

在特征匹配方面,针对最小费用 K 流算法 (MKP 算法) 的局限性与不足,本文提出了改进的最小费用 K 流算法。此算法将 SIFT 特征点作为图像的兴趣点,采用新的度量函数来度量匹配度,建立最小费用 K 流模型。通过改进的对偶算法求解,自动求出具有较小错误率的匹配点集。改进的 MKP 算法实现了匹配个数的自动选取,改进了原 MKP 算法中的匹配个数需要手动选取的缺点,提高了 MKP 算法的可用性。实验证明,在不需人工干预的情况下,和基于 k-d tree 的最近邻匹配算法相比,该算法可以提高匹配准确率约 5%。

在变换估计方面,本文提出了一种针对图像拼接的改进 RANSAC 算法。该算法利用图像拼接中的特征点的分布情况对随机样本集进行预处理,改进了 RANSAC 算法中点对选取的随机性,有效减少了变换估计的迭代次数,提高了变换估计的计算速度。实验表明,在保证计算准确率的条件下,本文算法的计算效率比原算法高出约 20%。

实验证明本文的拼接算法适用于采集设备环绕拍摄的全景图像拼接,可以取得不错的拼接效果。但是,由于拼接算法各个步骤的复杂性,系统还有尚待改善之处:

(1) 本文实现的是针对一个或多个相机环绕光心拍摄的图像进行柱面全景图拼接。如果采集设备光轴有上下倾斜,将对拼接效果产生很大影响。这个可以在拼接模型和投影平面上进行改善,比如使用球面投影模型。根据输入图像实现投影模型的自动选取是今后的努力方向。

(2) 由于时间和应用场合的原因,算法的融合方法采用了线性加权融合,无法解决由于目标移动产生的鬼影等问题。

## 参考文献

- [1] Hanche Huang. The Study of Aerial Images Stitching Based on SIFT Algorithm [D]. Master Thesis from National Sun Yat-Sen University, Taiwan, China, 1998.
- [2] Ali Can, Charles V.Stewart, Badrinath Roysam. Robust Hierarchical Algorithm for Constructing a Mosaic from Images of the Curved Human Retina[C]. Conf. on Computer Vision and Pattern Recognition, 1999.
- [3] Aunta P.E. Spatial registration of multispectral and multitemporal digital imagery using fast Fourier transforms techniques. IEEE Transactions on Geoscience Electronics, 1970. PP. 8(4). 353-368.
- [4] B.Reddy, B.Chaterji. A FFT Based Technique for Translation, Rotation, and Scale Invariant Image Registration [J]. IEEE Trans. Pattern Analysis and Machine Intelligence, 1996. Vol.5(8), PP.1266-1271.
- [5] Moigne J.L, Crompt R.F. The Use of Wavelets for Remote Sensing Image Registration and Fusion. SPIE, 1996. PP.535-544.
- [6] Szeliski R, Heung-Yeung Shum. Creating Full View Panoramic Image Mosaics and Environment Maps. SIGGRAPH 97 Conference Proceedings, 1997. PP 3(1):251-258
- [7] Viola P.A and Wells W.M Alignment by Maximization of Mutual Information. Proceedings of the Fifth International Conference on Computer Vision. June, 1995. PP. 16-23.
- [8] Pluim J.P.W., Maintz J.B.A., Viergever M.A. Image registration by maximization of combined mutual information and gradient information. IEEE Transactions on Medical Imaging, 2000. PP. 19(8), 809-814.
- [9] Dae-Hyun Kim, Yong-In Yoon, Jong-Soo Choi. An efficient method to build panoramic image mosaics [J]. Pattern Recognition Letters, 2003. 24 2421-2429
- [10] Barbara Zitova, Jan Flusser. Image registration methods: a survey [J]. Image and Vision Computing, 2003. PP. 977-1000.
- [11] Hans P. Moravec. Rover visual obstacle avoidance. In Proceedings of the 7th International Conference on Artificial Intelligence, 1981. PP. 785-790.

- [12] C.Harris, M.Stephens. A Combined Corner and Edge Detector[C]. The 4th Vision Conference, 1988. PP.147-151.
- [13] K.Mikolajczyk, C.Semhid. An Affine Invariant Interest Point Detector[C]. 7th European Conference on Computer vision, 2002. PP.128-142.
- [14] 赵向阳, 杜利民. 一种全自动稳健的图像拼接融合算法 [J]. 中国图像图形学报, (2004)04-0417-06, 1006-8961.
- [15] D.Lowe. Object Recognition from Local Scale-Invariant Features[C].7th International Conference on Computer vision, 1999. PP.1150-1157.
- [16] M.Brown, D.G Lowe. Recognizing Panoramas [C]. Proceedings of IEEE International Conference on Computer Vision, 2003. PP.1218-1225.
- [17] S.H.Cho, Y.K.Chung. Automatic Image Mosaic System Using Image Feature Detection and Taylor Series [C]. 7th International Conference on digital Image Computing Techniques and Applications, 2003. PP.549-556.
- [18] Schmid.C, Mohr.R. Local gray value invariants for image retrieval [J]. IEEE Trans. on Pattern Analysis and Machine Intelligence, 1997. PP. 19(5):530-534.
- [19] K.Mikolajczyk, C.Schmid. An Affine Invariant Interest Point Detector[C].7th European Conference on Computer vision, 2002. PP.128-142.
- [20] K.Mikolajczyk, C.Schmid. A Performance Evaluation of Local Descriptors[C]. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. PP.257-263.
- [21] David G.Lowe. Distinctive Image Features from Scale-Invariant Keypoints [J]. International Journal of Computer Vision, 2004. vol.2. no.60, PP.91-110.
- [22] Krystian Mikolajczyk, Cordelia Schmid. A Performance Evaluation of Local Descriptors [J]. IEEE Trans. Pattern Analysis and Machine Intelligence, vol.27, no.10, 2005, PP.1615-1630.
- [23] Hanan Samet. The Design and Analysis of Spatial Data Structures [D]. Addison-Wesley, 1989.
- [24] Sunil Arya, David M.Mount. Approximate Nearest Neighbor Queries in Fixed Dimensions. 4th Annual ACM-SIAM Symposium on Discrete Algorithms, 1993. PP.271-280.
- [25] Jeffrey S. Beis and David G. Lowe. Shape Indexing Using Approximate

- Nearest-Neighbour Search in High-Dimensional Spaces. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997. PP. 1000–1006.
- [26] David Nistér, Henrik Stewénus. Scalable recognition with a vocabulary tree. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, PP. 2161–2168.
- [27] Xinying He, Qixiang Ye, Guihong Zhou, Jianbing Jiao. Image Feature Matching with Network Flow: A Global Optimization Method [C]. Proceedings of SPIE in Visual Communications and Image Processing, 2009, Vol.72570F.
- [28] Richard Szeliski. Video mosaics for virtual environments [J]. IEEE Computer Graphics and Applications, 1996. PP. 16(2):20~30.
- [29] Charles V. Stewart. Robust Parameter Estimation in Computer vision. SIAM Reviews 41(3), 1999. PP. 513-537.
- [30] Peter J. Rousseeuw. Least median of squares regression [J]. Journal of the American Statistical Association, 1984. PP. 79, 871–880.
- [31] Fischler, M. A, Bolles, R. C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM, 1981. PP. 24(6), 381–395.
- [32] Ondrej Chum, Jiri Matas. Matching with PROSAC-Progressive Sample Consensus[C]. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. Volume 1.
- [33] D.R. Myatt, P.H.S Torr, S.J. Nasuto, J.M. Bishop, Craddock. NAPSAC: High noise, high dimensional robust estimation - it's in the bag. In Proceedings of the 13th British Machine Vision Conference, 2002, PP. 458–467.
- [34] Rafael C. Gonzalez, Richard E. Woods. Digital Image Processing (Second Edition) [M]. Prentice Hall, 2002.
- [35] 何新英. 基于特征点的图像匹配技术 [D]. 河北农业大学硕士学位论文, 2009.
- [36] 严磊. 基于特征匹配的全自动图像拼接算法研究 [D]. 中国科技大学硕士学位论文, 2009.
- [37] Peter.J. Burt, Edward H. Adelson. A Multiresolution Spline With Application to Image Mosaics [J]. ACM Transactions on Graphics, 1983. PP. 2(4):217-236.

- [38]Halfhill Tom R. See you around [J]. Byte Magazine, 1995, 3(5).PP:85-90.
- [39]A.Krishnan, N.Ahuja. Panoramic image acquisition [C]. IEEE Conference on Computer Vision and Pattern Recognition, 1996. PP. 379-384.
- [40]王朝瑞. 图论 [M]. 北京理工大学出版社, 2001.
- [41]胡运权, 郭耀煌. 运筹学教程 [M]. 清华大学出版社, 2003.
- [42]Alexander Toshev, Jianbo Shi, Kostas Daniilidis. Image Matching via Saliency Region Correspondences [C]. In Conference on Computer Vision and Pattern Recognition. 2007. PP.1-8.
- [43]刘振宏, 蔡茂诚, 译. 组合最优化:算法和复杂性 [M]. 北京:清华大学出版社, 1988.
- [44]R.Szeliski.Iccv 2005 computer vision contest [DB/OL].  
<http://research.microsoft.com/iccv2005/Contest/>.
- [45]陈付幸, 王润生. 基于预检验的快速随机抽样一致性算法 [J]. 软件学报, 100-9825/2005/16(08)1431.
- [46]Sunglok Choi, Taemin Kim, Wonpil Yu. Performance Evaluation of Ransac Family [C]. BMVC, 2009.
- [47]蔡元龙. 模式识别 [M]. 西北电讯工程学院出版社, 1986.
- [48]Lin Zhang, Zhihua Liu, Jianbin Jiao. An improved RANSAC algorithm using within-class scatter matrix for fast image stitching [C]. IS&T/SPIE Symposium on Electronic imaging, 2011. Volume 7870.
- [49]游磊. 图像拼接的核心算法研究 [D]. 重庆大学硕士学位论文, 2009.
- [50]赵辉. 基于点特征的图像配准算法研究 [D]. 山东大学硕士学位论文, 2006.

## 附录 1

- ❖ **Lin Zhang**, Zhihua Liu, Jianbin Jiao. An improved RANSAC algorithm using within-class scatter matrix for fast image stitching[C]. Proceedings of SPIE-IS and T Electronic Imaging - Image Processing: Algorithms and Systems IX, 2011. Volume 7870,pp. 787017-787017-10 (2011).

## 致 谢

回首三年硕士研究生学习生活，我经历了艰辛，付出了努力，在学业和生活上得到很多锻炼、成长与感悟。在毕业论文完成之际，由衷感谢这三年来曾经给予我帮助的老师、同学、家人和朋友。

首先，我要感谢恩师焦建彬教授，本课题的研究工作是在他悉心指导下完成。焦建彬教授从理论和实验方面给予了大量的、极其有益的建议和具体的指导，并在论文的撰写和审稿中倾注了大量的心血，他诲人不倦的精神和对我的谆谆教导，使我学到了不少可贵的东西，让我受益匪浅。此外，他还在生活和思想上关心我、照顾我，再次表示万分的感谢！恩师严谨的治学态度、积极的进取精神、平易近人的工作作风将使我终身受益。

其次，衷心感谢叶齐祥老师在三年的学习与生活中对我的极大帮助和支持，他渊博的专业知识和精益求精的动手能力让我惊叹和佩服，在理论学习和项目开发过程中给我提供了耐心的引导和帮助。

再次，特别要感谢我的课题组长刘志花师姐，在我两年的研究生涯和论文撰写时期，她给了我无私的帮助，她严谨的研究态度和丰富的知识极大的影响了我，让我在百思不得其解之时茅塞顿开。感谢实验室的其他同学，他们的关心让我时刻感受到团体的温暖，以愉快的心迎接每一天。

此外，我要虔诚感谢亲爱的家人和挚友，他们对我的无条件的爱陪我度过了太多的难关，给了我无限动力，感谢他们始终理解我和支持我！

最后，再次对在我的学习、生活、工作上给予指导和帮助的老师、同学们、朋友们致以衷心的感谢。对此，我铭记终生，必将以勤奋的工作、认真的生活回报大家！