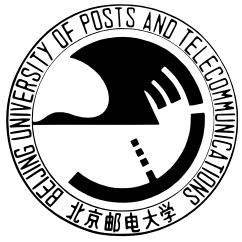
密级：公开　　保密期限：

北京郵電大學

硕士学位论文



题目：基于局部特征的图像重建算法研究

学　　号：2012110191

姓　　名：王继哲

专　　业：信号与信息处理

导　　师：李学明

学　　院：信息与通信工程学院

二〇一四年十二月

独创性（或创新性）声明

本人声明所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京邮电大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

本人签名：_____ 日期：_____

关于论文使用授权的说明

学位论文作者完全了解北京邮电大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属北京邮电大学。学校有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许学位论文被查阅和借阅；学校可以公布学位论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存、汇编学位论文。（保密的学位论文在解密后遵守此规定）

本学位论文不属于保密范围，适用本授权书。

本人签名：_____ 日期：_____

导师签名：_____ 日期：_____

基于局部特征的图像重建算法研究

摘 要

基于局部特征的图像重建算法是利用原始图像的局部特征信息，以大图像集为数据源，进行较为精确的图像重建工作，使重建后的图像与原始图像相似，并且图像质量达到人眼主观效果较好的程度。图像重建算法在拷贝检测、隐私保护、超高分辨率重建、场景还原等领域有着广泛的应用。本文在基于云的图像压缩中的应用场景下对重建算法系统框架进行研究，对其中的技术细节进行了完善。

本文首先介绍基于云的图像压缩这一新颖的应用场景，着重介绍客户端特征提取与数据压缩、服务器端数据解码后使用大规模图像数据集进行图像重建这一完整的系统流程，完整的介绍了系统中的各个模块及其对应的技术手段。其次从两个技术层面对目前图像重建系统加以阐述，包括传统的全景图拼接技术中的图像局部特征、局部特征匹配、2D 变换与特征配准以及图像融合等算法，此外，从相似图像搜索角度叙述了视觉词组、相似图像搜索、匹配特征块筛选等算法。

针对重建系统在大规模语料集场景下的应用特点，本文对上述几个环节进行完善，主要包括在匹配图像块筛选中提出阈值自适应阈值图像块筛选法、相似图像的分块查询、视觉词组的二维编码等，为重建提供了更有力的依据，从仿真结果来看重建效果达到了令人满意的程度。

关键词：图像重建　局部特征　图像分割　图像配准　图像融合　尺度不变特征变换　相似图像搜索

RESEARCH ON LOCAL FEATURES BASED IMAGE RECONSTRUCTION ALGORITHM

ABSTRACT

Local features based image reconstruction is an algorithm which uses local feature information of the original image,together with the large-scale image dataset,to perform accurate image reconstruction so that the reconstructed image is similar to the original image and achieves good visual quality.Firstly,we summarize the current architecture of the image reconstruction system,including the technologies of local descriptors,visual word group,partial duplicate image retrieval,feature matching and registration,patch filtering, and image fusion.Then we propose several technologies which include adaptive threshold validation,block-based similar query and 2-d visual words coding to optimize current system at the scene of large-scale corpus.The results demonstrate that the proposed methods provides stronger reconstruction evidence and thus improve the performance.

KEY WORDS: Image reconstruction local features image segmentation image registration image fusion sift partial-duplicate image retrieval

目 录

符号对照表

$(\cdot)^*$	复共轭
$(\cdot)^{\mathrm{T}}$	矩阵转置
$(\cdot)^{\mathrm{H}}$	矩阵共轭转置
$\mathbf{X}$	矩阵或向量
$\mathscr{A}$	集合
$\mathscr{A} \times \mathscr{B}$	集合 $\mathscr{A}$ 与集合 $\mathscr{B}$ 的 Cartesian 积，即 $\mathscr{A} \times \mathscr{B} = \{(a,b) : a \in \mathscr{A}, b \in \mathscr{B}\}$

第一章 绪论

本章主要介绍基于局部特征的图像重建系统在基于云的图像压缩方面的应用场景，该技术的研究背景、国内外的发展情况以及目前取得的研究成果，最后介绍了本文的主要研究内容和文章的组织结构。

1.1 论文课题的研究背景

随着数字化时代的不断发展，智能终端日益普及，终端应用的功能也日趋多样化，我们发现有一类应用服务规模迅速扩大，这一类型的应用采用相似的 CS 技术架构——智能终端使用传感器采集图像数据，并通过网络向服务器实时传输，由服务器来处理数据，将处理结果反馈给终端用户。而图像应用的爆发式增长给我们带来了一个全新的挑战：图像信息的传输占用了大量的带宽资源。目前的解决方案是在终端对原始图像进行下采样和压缩编码，产生的图像信息的损失大大降低了用户体验，而且传统的压缩编码算法占用了一定的 CPU 资源，压缩比不是很高，压缩后的图像数据量依然较大。

另一方面，走在大数据时代前沿的互联网拥有无比丰富的图像资源，图像张数数以亿计，图像样式种类五花八门，而且每天还不断有用户贡献着高质量高分辨率的图像。从信息的角度来说，我们拍摄的每一幅图像中所包含的部分或全部内容都可以在互联网上其它图像中找到。

以上两点观察启示我们打破传统的图像内逐像素压缩方法，采用一种全新的基于大数据集的外部图像压缩方法。在 2013 年 6 月有学者[1] 提出一种全新的压缩方式——基于云的图像编码。其核心思想是在客户端提取并编码发送少量的图像特征数据，并不传输图像数据本身，而在服务器端解码后利用特征数据在服务器的大图像数据集上匹配相似的图像，利用相似图像进行图像的重建。图1-1展示了这一客户端-服务器（Client-Service）数据应用模式。这种架构所运用的核心技术手段便是基于局部特征的图像重建算法，通过对原始图像的特征提取与重建，利用计算资源减少带宽损耗，从一个全新的维度进行数据压缩，为多媒体应用开启了一扇大门。

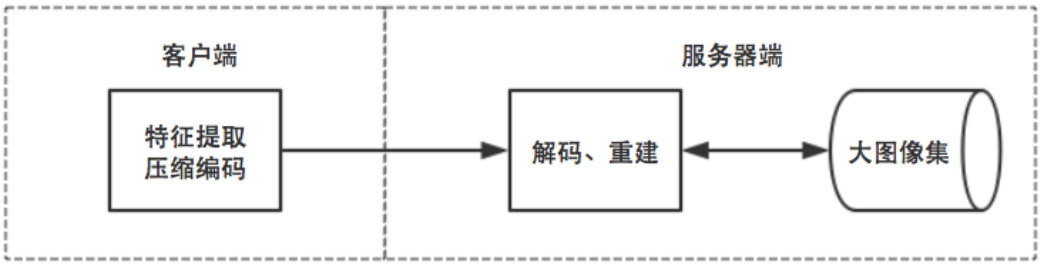


图 1-1　基于与的图像压缩模式

1.2　国内外研究现状

本文所探讨的基于局部特征的图像重建算法的脱胎于图像拼接技术和相似图像搜索技术，这两个技术相对而言较为成熟：图像拼接领域，SIFT 特征的强可区分性和不变性使之广泛应用在全景图拼接领域[2]；相似图像搜索在近年来发展较快，在搜索效率和搜索准确度两个层面进行探索，利用局部特征取代全局特征[3–5] 以及利用局部特征的空间编码[6] 进行更为精确的相似图像搜索，使用 min-hash 等技术对特征进行压缩[7] 提高搜索的速度。

在其上发展而来的基于局部特征的图像重建算法是近几年刚刚提出的一项新颖的技术手段，Philippe Weinzaepfel 等人于 2011 年首先提出了基于局部特征进行图像的重建[8]，重建方法首先是根据局部特征使用传统技术在大数据集上找到与原始图像视觉相似的图像块，通过特征匹配与配准将候选图像块转换到标准图像域，通过无缝拼接技术将其拼合在一起，最后使用平滑差值计算空白区域的像素值，以椭圆形图像块为基本单元，重建的结果如图1-3所示：

在这一自动化的图像重建系统中，虽然重建结果没有达到视觉满意程度，但是显示了基于特征进行重建的可能性，作者由此提出了图像的局部特征信息可能泄露用户隐私这一话题。

随后，Maryam Daneshi 和 Jiaqi Guo[9] 则在确实特征尺度信息的前提下，挖掘图像重建的最大可能。他们采用方形图像块作为重建的基本单元，利用贪婪算法逐步的学习每一个图像块的尺度，完成图像的亮度信息重建，最后采用用户指定颜色遮罩并用最优化算法来进行上色，重建结果如图1-2所示，重建结果保留了大部分的图像信息。

Lican Dai 等人提出将其应用在手机地标图像的实时分享[10]，采用基于搜索的重建技术搭建了 IMShare 系统，提出使用缩略图来指导服务器端的特征提取和图像融合，实验结果显示缩略图的使用大大提升了图像的重建结果，该系统不仅能够重

图 1-2 Daneshi 基于局部特征的图像重建结果

图 1-3 Weinzaepfel 基于局部特征的图像重建结果，从左向右依次是原始图像、重建图像、补全后的图像

建人眼视觉满意的图像，而且采用并行的处理手段，能在秒这一数量级上完成重建流程。Huanjing Yue 使用相似的技术手段进行超分辨率的图像生成，使用低分辨率（LR）的图像在大数据集上进行搜索，得到高分辨率（HR）的候选图像，再通过特征提配与图像重建的技术手段生成超分辨率图像（SR）。

综上所述，基于局部特征的图像重建算法是一项较新的领域，将其用在图像在客户端和服务器端压缩传输更是一个全新的领域，有着大量的技术难点需要攻克，本文主要是在这一应用场景下，对其中的各种技术手段做进一步的探索。

1.3 论文的主要研究内容与章节安排

1.3.1 主要研究内容

本课题以文献[1]提出的核心任务与技术框架为基础，重点研究在基于云的图像压缩应用场景下的基于局部特征的图像重建算法，进一步探索利用更丰富的图像特征信息在大语料集中进行高质量的图像重建，使重建图像质量达到人眼主观效果较好的程度。

1.3.2 章节安排

第二章 基于局部特征的图像重建算法概述

图像重建可以被概括的定义为这样一个基本问题：从一个退化版本的二维物体估算实际的二维物体[11]。退化过程的数学形式取决于图像重建算法实际的应用场景。

2.1 传统的图像重建算法

传统的图像图像重建算法所使用的场景一般是指图像修复（Image Restoration），原始“物体”由于经历了某种退化过程，不能直接由观测信息判断出来，为了消除退化过程的影响，必须根据观测到的数据进行重建来还原得到原始信息。在图像修复中，引起退化的原因叫做失真，其定义如下：

$$y = A(X) \bullet b \tag{2-1}$$

其中$A(\cdot)$是退化函数，可以看做是一个滤波器，b 表示的是噪声，$\bullet$ 表示的叠加方式。失真通常包含对 X 的卷积或者模糊，加性噪声或者乘性噪声。而图像修复的解决方案是通过对观测信息进行退化模型的数学建模，利用约束条件来推导出退化过程的逆过程，对观测信息进行逆过程得到原始图像。

另一类图像重建场景是超分辨率重建，在近年来得到飞速的发展，是炙手可热的研究领域，它的基本思想是通过多张连续的低分辨率图像序列得到一张高分辨率的图像。很多数字图像应用中都需要高分辨率的图像，高分辨率的图像能够提供更佳的视觉体验，提供更丰富的信息，比如高分辨率的医学图像能够让医生更好的进行病情诊断，高分辨率的卫星图像能够进行更准确的模式识别任务。从 1970 年代以来，CCD 和 CMOS 传感器被大规模的使用，获得了大量的数字图像，但是很多图像的分辨率较低，不能满足日益增长的业务需求，超分辨率重建是在这样的背景下诞生的[12]。

那么，我们如何通过多张低分辨率图像获得一个高分辨率图像呢？如果一个场景下有多张低分辨率图像，而且这些图像从不同的角度来“描述”这个场景，那么这些低分辨率的图像可以看做是该场景的子采样和子像素精度的位移。如果这些低分辨率图像是以整数像素为单位进行的位移，那么多张低分辨率图像没有提供任何“新的信息”，但是如果位移单位是子像素单位的，序列中的每一个图像不能够由其

他图像得出，换言之每个图像都提供了子像素精度的不同信息，我们可以利用这些信息重建一个高分辨率的图像。一般来说，SRR 算法分为基于重建和基于学习的两大类：基于重建的算法如频域重建法利用图像序列的交叠关系，凸集投影（POCS）等利用一些先验知识来约束求解过程，以达到增加细节信息的目的；基于学习的算法则使用多种机器学习的概率模型，包括基于流形学习、基于支持向量机和基于独立分量的超分辨率重建技术。基于学习的方法采用大量的高分辨率图像构造学习库来训练学习模型，在对低分辨率图像进行重建的过程中引入由学习模型获得的先验知识，进而得到图像的高频细节，获得较好的图像重建效果。

总体而言，超分辨率重建的整个流程包括三个基本环节：（1）低分辨图像的预处理，包括降噪和裁剪等基本图像数据处理。（2）配准过程，利用像素的空间信息估算低分辨率序列图像之间的运动矢量和空间位置关系。（3）完成重建，使用图像分割和融合等技术，利用多帧低分辨率图像的信息完成超分辨率重建。

2.2 基于局部特征的图像重建算法

从另一个角度来看，本文所采用的图像重建的部分流程可以看成是多幅图像的全景图拼接问题。与文献[2] 中的流程类似，主要包含以下几个环节：（1）使用具有不变性的特征来描述图像；（2）自动的找到图像之间的空间位置关系，进行图像配准；（3）图像融合，消除不同图像之间的光照差别，去除边缘噪声。与全景图拼接不同的是，本文提出的图像重建系统每幅小图（Patch）块大小不一，导致图像空间位置关系可能存在不准确，多幅图像之间有大量重叠，整体思路是先融合大块，后融合小块，分别介绍如下。

2.3 图像的局部特征

2.3.1 局部特征概述

图像的局部特征是计算机视觉领域一个基本问题，它能够反映图像某一局部的特性，对寻找图像对应的局部单元以及特征描述中有着重要作用。通常意义的局部特征包含两个方面，特征检测子（Detector）和特征描述子（Descriptor）。检测子能够检测出我们“感兴趣”的点或者局部区域，而一个好的局部特征描述子反映出图像的局部特性能够帮助找到图像与图像点集合对应关系，进而建立图像之间的空间对应关系。局部图像特征描述的核心问题是不变性（invariant）和可区分性

（discrimination）。

目前人们提出的众多图像局部特征中算子中，由 Lowe 提出的尺度不变特征变换（Scale Invariant Feature Transform，简称 SIFT）应用最为广泛。1999 年首次提出，至 2004 年得到完善[13] 的 SIFT 算子是图像局部特征研究领域的一项重大突破。SIFT 算子具有很强的可区分性，同时对尺度、旋转以及一定视角和光照变化等图像变化都具有不变性。在其之上衍生出来的 SURF（Speeded Up Robust Features）是对 SIFT 的改进版本，它利用 Haar 小波来近似 SIFT 方法中的梯度操作，同时利用积分图技术进行快速计算，SURF 的速度是 SIFT 的 3-7 倍。

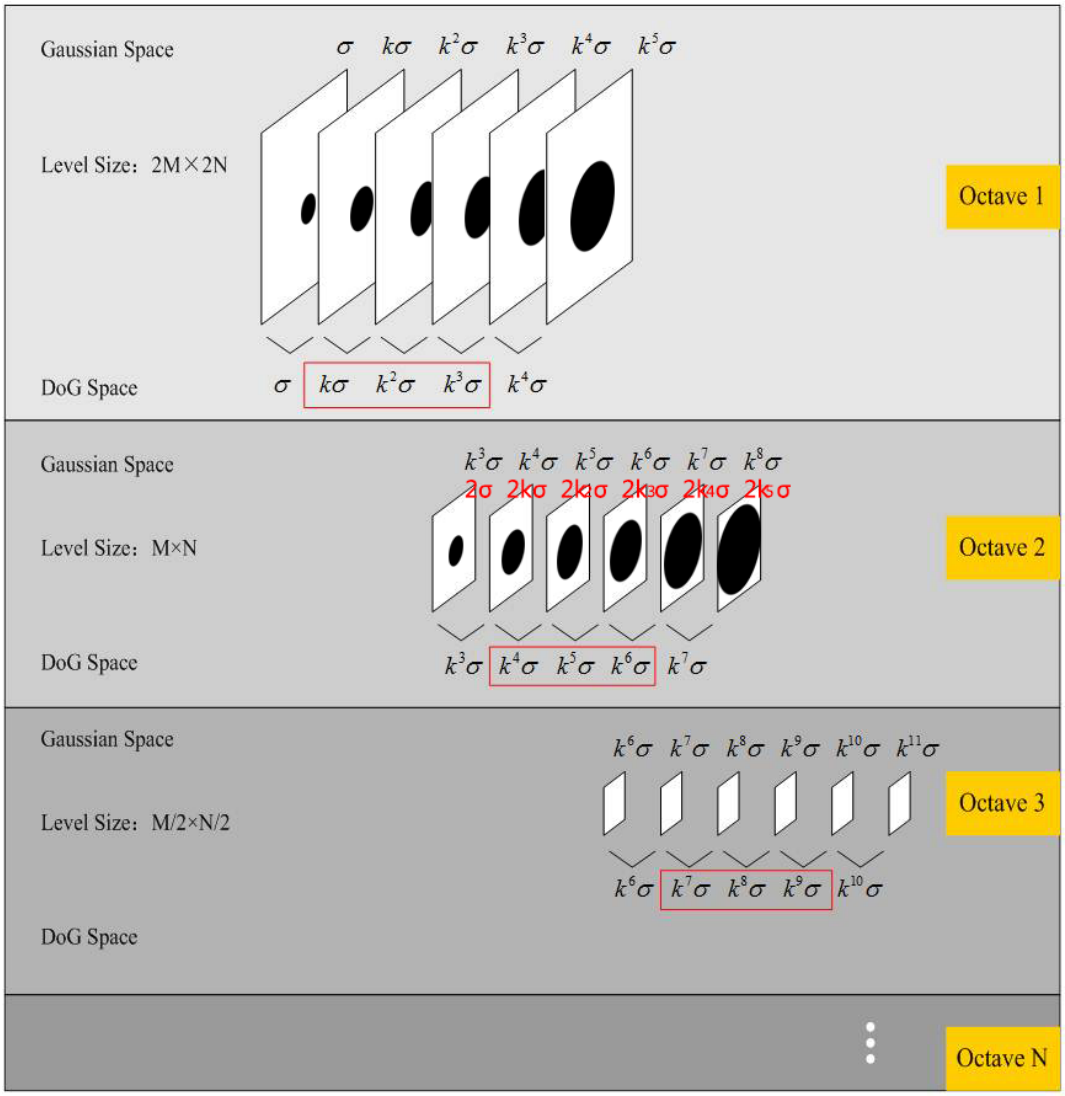
除此以外，常见的特征检测子包括 Harris 角点，ANMS 等，描述子还包括 DAISY，ASIFT，MROGH，BRIEF 等，分别适用于不同的图像应用场景下，本文提出的系统采用适用性最广泛的 SIFT 算子，下面我们对其进行简要的介绍。

2.3.2 SIFT

（1）尺度空间理论尺度空间理论目的是模拟图像数据的多尺度特征，尺度空间中各尺度图像的模糊程度逐渐变大，能够模拟人在距离目标由近到远时目标在视网膜上的形成过程。我们可以把两幅图像想象成是连续的，分别以它们作为底面作四棱锥，就像金字塔，那么每一个截面与原图像相似，那么两个金字塔中必然会有包含大小一致的物体的无穷个截面，但应用只能是离散的，所以我们只能构造有限层，层数越多当然越好，但处理时间会相应增加，层数太少不行，因为向下采样的截面中可能找不到尺寸大小一致的两个物体的图像。一个图像的尺度空间，$L(x,y,\sigma)$ 定义为一个变化尺度的高斯函数$G(x,y,\sigma)$ 与原图像$I(x,y)$ 的卷积。

$$L(x,y,\sigma) = G(x,y,\sigma) \otimes I(x,y) \tag{2-2}$$

下面这幅图反映了图像金字塔的情况：



图中的黑色圆盘是我加上去的，表示的是该图像所在的尺度的特征覆盖的范围，其特点是不同组同一层上的特征覆盖范围一样，同一组不同层上的特征覆盖范围逐步增大。关键点的尺度坐标就是按关键点所在的组和组内的层, 利用下面这个公式计算而来：

$$\sigma(o,s)=\sigma_0 2^{o+s/S},\quad o\in o_{\min}+[0,\ldots,O-1],\quad s\in[0,\ldots,S-1] \tag{2-3}$$

（2）SIFT 检测子 SIFT 算法有两个主要环节，一个是检测“感兴趣”的关键点，另一个是描述这个“关键点”。SIFT 关键点是精心选择的一组在高斯差分尺度空间（Difference of Gaussians scale space，DoG）上的极值点，该关键点包含三个关键信息，分别是（1）亚像素精度的（x，y）位置信息；（2）尺度大小，反映关键点局部的大

小，同时决定了特征的覆盖范围，对后文局部块的提取起到至关重要的作用；（3）所在高斯尺度空间上的主方向，该主方向是有一个高斯窗口函数计算得来，反映的是关键点所在局部的方向信息。其中差分高斯尺度空间表示为

$$D(x,\sigma(s,o)) \doteq G(x,\sigma(s+1,o)) - G(x,\sigma(s,o)). \tag{2-4}$$

关于尺度空间和描述子的具体讲解在 Lowe 的论文中[13] 已有详细的介绍，这里不再详述。

（3）SIFT 描述子 SIFT 描述子反映关键点局部的信息，是高斯尺度空间上某一局部和方向上的梯度信息，以直方图的形式对信息做统计，最终每一个描述子是一个 128 的特征。

2.4 特征匹配

提取图像的局部特征之后，我们希望找到两幅相似的图像间相一致的特征点，这一过程叫做特征匹配。匹配环节最重要的两个步骤是匹配策略的确定和高效的数据结构以及相应的匹配算法。

2.4.1 匹配策略

如果我们能够知道一个 SIFT 描述子和其它所有描述子的相似性的情况下，我们采取如下的匹配策略：

$$M(S_1,S_2) = \begin{cases} \text{true}, & \text{if} S_2 = S_m in, \frac{\text{Dis}(S,S_{min})}{\text{Dis}(S,\tilde{S}_{min})} > C \\ \text{false}, & \text{otherwise} \end{cases} \tag{2-5}$$

其中Dis$(\cdot,\cdot)$ 表示的是两个特征描述子之间的相似性度量，比如可以用欧氏距离表示，距离越大，相似性越小。S_{min} 和$\tilde{S}_{min}$ 分别表示的是与 S 距离最近和第二近的特征，。而 C 是一个阈值常数，通常取 1.5。上式表明，如果最匹配的特征的距离比其次匹配特征的距离的比值大于一定的程度，我们认为该特征有最佳匹配，最佳匹配就是最为相似的特征。

2.4.2 高效匹配算法

在上述的匹配策略中，我们有这样一个前提，能够比较当前特征点和每一个候选特征点的相似性，进而找到最为相似的一个候选特征。在解决这个最近邻问题时，

总体的时间复杂度是$O(n^2)$，在大多数的应用中使用并不现实。因此我们需要找到更为合适的索引结构来存储数据，进行快速的查找。解决这种高维数据查询检索的一些常见算法包括使用多维散列，局部敏感哈希（Locality Sensitive Hashing）以及多维搜索树等。本文使用多维搜索树中最为常见的 k-d 树（K-dimension tree），下面对其做简要介绍。

k-d 树是一种空间划分树，它把整个特征空间交替沿着垂直于坐标轴的超平面将空间进行分割，分割时尽量使得特征点的分布保持平衡。然后在特定的划分内进行相关搜索操作，有效减少搜索范围。

构建 k-d 树遵循如下的规则：

（1）随着树的深度增加，循环的选取坐标轴，作为分割超平面的法向量。对于维度为 3 的树来说来说，根节点选取 x 轴，根节点的孩子选取 y 轴，根节点的孙子选取 z 轴，根节点的曾孙子选取 x 轴，这样循环下去。

（2）每次均为所有对应实例的中位数的实例作为切分点，切分点作为父节点，左右两侧为划分的作为左右两子树。

对于 n 个实例的 k 维数据来说，建立 kd-tree 的时间复杂度为 O(k*n*logn)。

k-d 树的搜索算法如下[14]：

1. 在 k-d 树种找出包含目标点 x 的叶子节点：从根节点出发，递归访问 k-d 树。若 x 当前维的坐标小于切分点的坐标，则移动到左子结点，否则移动到右子节点。直到叶子节点。
2. 以当前叶子节点为“当前最近点”。
3. 回溯：递归的回退，对每个节点：
   - 如果该节点比当前最近节点离目标更近，则更新“当前最近点”
   - 当前最近点一定存在于该节点某个子节点对应的区域，检查该子节点的兄弟节点对应的区域是否有更近的点。
4. 回退到根节点，结束，“当前最近点”即为我们找到的最近邻。

2.5 图像配准

在得到两幅图像（在本文中是两个图像块）相匹配的特征点之后，我们有了相对应点的位置关系，如何利用这些位置关系将两幅图像重合的部分放置于正确的位

置上，是本节讨论的议题。我们首先简要介绍图像的 2D 几何变换，之后使用最为广泛的随机抽样一致算法进行图像的配准。

2.5.1 2D 几何变换

（1）旋转和平移变换, 也叫 2D 刚体运动即 2D 欧式变换（因其保持欧式距离），写作$x = Rx + t$ 或者写作

$$\begin{bmatrix} R & t \end{bmatrix} \bar{x} \tag{2-6}$$

其中

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{2-7}$$

是一个正交旋转矩阵，有$RR^T = I$ 和$|R| = 1$

（2）放缩旋转，也叫做相似变换，该变换可以表示为$\bar{x} = sRx + t$，其中 s 是一个任意的尺度因子。它也可以写作

$$x = \begin{bmatrix} sR & t \end{bmatrix} \bar{x} = \begin{bmatrix} a & -b & t_x \\ b & a & t_y \end{bmatrix} \bar{x} \tag{2-8}$$

其中我们不再需要$a^2 + b^2 = 1$。相似变换保持直线间的夹角。各种 2D 变换如下表所示：

变换	矩阵大小	自由度数	保持
平移	2×3	2	方向
刚性（欧式）	2×3	3	长度
相似	2×3	4	夹角
仿射	2×3	6	平行性
投影	2×3	8	直线性

表 2-1 2D 坐标变换

当我们使用 SIFT 算法得到匹配到的特征点后，有两种方法，一种是直接写出变换矩阵，另一种是使用随机抽样一致方式多次迭代找到最准确的变换矩阵。RANSAC 的方式求解得到的变换矩阵称为H，具体的算法将在后文” 基于空间信息的匹配搜索算法” 一节加以介绍，根据一对匹配的 SIFT 算子直接写出两个图像块的变换矩阵，我们称这个变换为H_0，具体求解方式如下：

结合一对匹配 SIFT 特征点$\tilde{S}$和S的位置、尺度和方向，我们可以得到两个图像块$P_{\tilde{S}}$和P_S的变换矩阵H_0：

$$H_0 = \begin{bmatrix} \frac{\tilde{s}_f}{s_f} R & T \end{bmatrix} \tag{2-9}$$

其中

$$R = \begin{bmatrix} \cos(\tilde{\theta} - \theta) & -\sin(\tilde{\theta} - \theta) \\ \sin(\tilde{\theta} - \theta) & \cos(\tilde{\theta} - \theta) \end{bmatrix} \tag{2-10}$$

$$T = \begin{bmatrix} \tilde{x}_f - x_f \\ \tilde{y}_f - y_f \end{bmatrix} \tag{2-11}$$

计算出的H_0和H都可以作为块的旋转矩阵，在实际的系统中，我们会同时计算两个矩阵，比较他们的准确程度，挑选使用准确度高的变换矩阵。

2.6 图像分割

本文主要采用的是基于图的图像分割算法

2.6.1 基于图的图像分割算法

1、背景介绍

主要参考的是这篇文章《Efficient Graph-Based Image Segmentation》Pedro F.Felzenszwalb

文章首先自己定义一种区域边界的度量方法，其度量方法是在基于图的图像表示法之上去定义的。在这种度量方法之上，我们衍生出来比较高效的图像分割算法。该算法是一种贪婪算法，并且分割结果满足全局属性。

通过上述逻辑关系可以看出，文章定义的度量方法需要比较准确，有一定的物理意义在里面，不然即使算法再高效，度量本身有问题，那么分割出来的图像区域也是不准确的。那么文章自然的可以分为两个部分 1）区域度量方法。2）高效分割算法

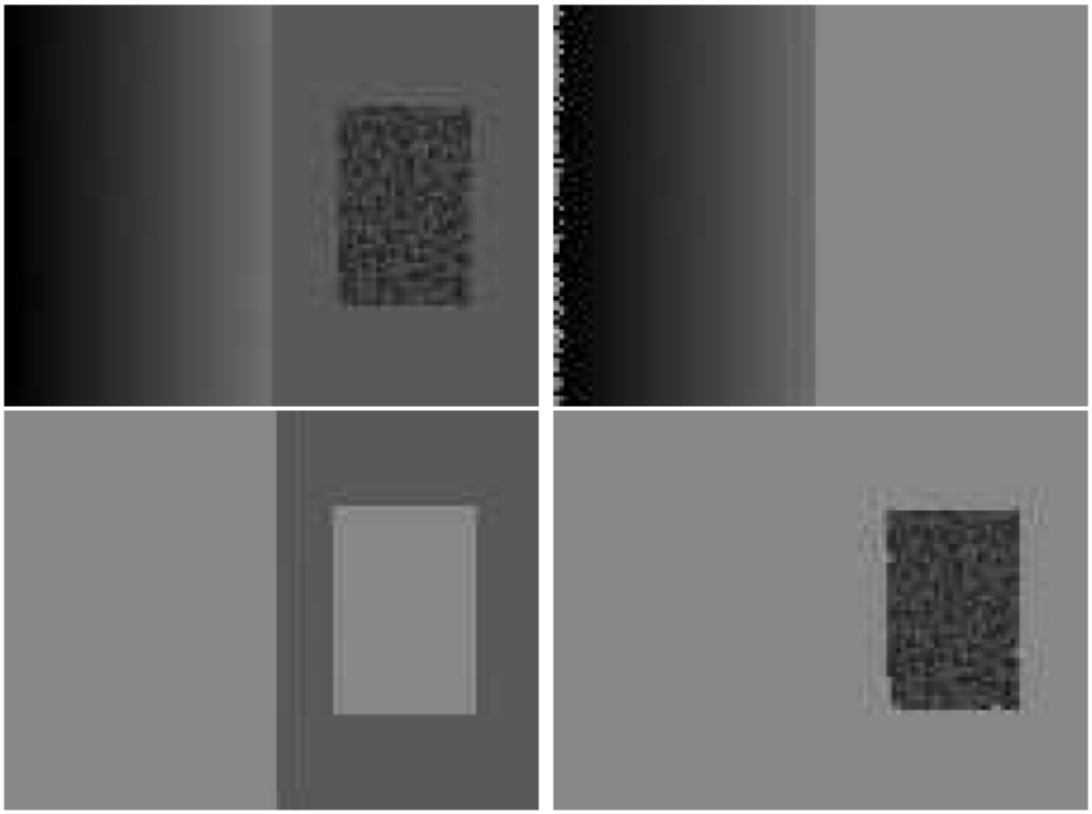
图像分割的在很多应用中非常重要，是很多高层应用的前提，比如识别、索引等，我们不具体举例。我们认为图像分割的方法有下面这样的特性：

- 能够捕捉到感知上比较重要的区域，这通常体现在图像的全局特性方面。这里有两个关键点，一方面要提供感知重要的精准属性，另一方面能够确定给定的分割技术是做什么的。我们认为应该有对分割结果属性的经确定以，这样的方法才能够更好的被理解，进而与其他的方法进行比较。

- 高效，接近图像像素点数量的线性时间复杂度。为了能够实际使用，我们认为分割方法应该与边缘检测或者其他 low-level 图像处理技术有着相似的时间复杂度，意味着时间复杂度是线性，而且常系数也比较小。比如每一秒能对几帧图像进行分割的算法就能够处理实时的视频数据。

然而，近几年的一些方法并不能够达成上述两方面要求，哪些方法太慢以致不能实践中使用。相比较而言，本文提到的方法已经因公在大尺度图像数据集应用上。有一些其他的方法可以比较快速的进行图像分割，但是这些方法不能捕捉感知上重要的非局部特性，下文会提到。总而言之，本文在保证效率的同时考虑到了图像全局属性上的感知重要区域。

首先我们来看一幅人造图像：



我们人眼会认为这幅图像有三个区域，这个例子能够解释什么是感知重要属性（conceptually important property）。首先，亮度的变化不应该单独的座位分割区域的衡量标准。比如图像中左侧渐变区域和右侧的高频噪声区域都有较大的亮度变化，但是我们他们应该被分割成多个区域。因此，假设一个区域有着接近恒定的或变化很小的亮度是不正确的。

第二个感知重要属性是有意义的区域不能单纯的依靠局部划分标准。还是在图上我们可以看到原因，渐变图像与常量区域的边界上的亮度差值比很多高频区域的差值要小，因此我们得出结论，为了分割一幅图像，我们需要引入一些适应性的或者非局部的衡量标准。

我们在下一节提出的衡量标准会比较两个属性：

- 边界的亮度差值
- 区域内部的邻居像素间的亮度差值

直观上，两个区域的边界上的亮度差值如果比比两个区域中至少一个区域的内部像素差值大的话，那么边界亮度差值会更多的影响我们的感知，这个时候我们说边界亮度差是感知重要的。

2、基于图的图像表示

好下面我们来进入正题，基于图的图像分割（Graph-Based Segmentation）。我们使用基于图的方法来做图像分割，令$G=(V,E)$表示一个无向图，点集$v_i \in V$，待分割的元素集合。边$(v_i,v_j) \in E$有一个相应的权重$w((v_i,v_j))$，是一个非负值，描述两个相邻元素v_i和v_j的不相似度。在图像分割，也就是本文的语境下，V 中的元素就是像素点，边就是它的两个像素点（这两个像素点是相邻的）不相似性的某种度量（例如亮度，颜色，运动，位置或者其他局部属性）。在文章的最后我们会讨论比较特殊的边集合和权重函数，不过这里的公式和不相似性度量的方法是独立的，我们可以按照自己的需求定制度量方案，这里讨论的是大框架。

在基于图的方法中，一个分割方案 S 是 V 的一个划分，每一个区域 (region or component)$C \in S$对应着图$G' = (V,E')$的一个连通区域，其中$E' \subseteq E$。有许多方法来衡量一个分割的好坏，大体上我们希望**一个区域内部的元素尽可能相似，不同区域之间的像素尽可能不同**。这意味着同一区域内，相邻两个点的有相对来说比较小的权值，不同区域的相邻两个点的边有大的权值。

3、成对的区域比较预测，内部不相似度与外部不相似度

这一节我们首先定义一个预测，D，来估计是否存在一个显著的证据表明有一个边界能将两个区域分割开。就像上文说的，就是对外部的不相似性与内部不相似进行比较，也就是比较 inter-component 和 within component 的差值。

我们定义内部不相似性为该区域最小生成树的最大边，$MST(C,E)$，即：

$$Int(C) = \max_{e \in MST(C,E)} w(e) \tag{2-12}$$

这个方法潜在的直觉是一个区域 C，它保持连通的最低要求是 Int(C) 这个 edge 所决定的。

定义两个区域的不同：区域$C_1, C_2 \subseteq V$，连接这两个区域的所有边的权值中，最小的那个权值。即，

$$Dif(C_1, C_2) = \min_{v_i \in C_1, v_j \in C_2, (v_i, v_j) \in E} w((v_i, v_j)) \tag{2-13}$$

如果两个区域没有连接的边，则令$Dif(C_1, C_2) = \infty$ 这个定义理论上可能会有问题，因为它只反映了（或者说只考虑到了）两个区域间权值最小的那条边。在实践中我们发现尽管有显著的局限，但这种度量方式结果颇佳。值得一提的是，改变这个衡量标准也是可以的，比如采用中位数或者其他的分位点，提升对异常值的鲁棒性，但这种改变会使问题编程 NP-hard 问题。因此一个小小的分割标准的改变会大大改变解决问题的难度。

区域比较预测法通过比较$Dif(C_1, C_2)$ 和$Int(C_1)$ 与$Int(C_2)$ 中较小的一个，来判断这两个区域是否有一个边界（换言之这两个区域是否有足够的理由保持两个区域）。

$$f(n) = \begin{cases} true, \text{if } \mathrm{D}if(C_1, C_2) > MInt(C_1, C_2) \\ \\ false, otherwise \end{cases} \tag{2-14}$$

我们引入了一个阈值函数来控制我们希望的外部不相似度与内部不相似度的相差程度。

$$MInt(C_1, C_2) = min(Int(C_1) + \tau(C_1), Int(C_2) + \tau(C_2)) \tag{2-15}$$

对于比较小的区域，$Int(C)$ 并不能够较好的反应局部特性，比如最极端的情况下，

当$|C|=1$时，$Int(C)=0$。因此我们需要一个跟区域大小相关的阈值函数

$$\tau(C)=\frac{k}{|C|}$$

其中$|C|$表示的是区域C的大小，k是一个常数。越是小的区域，我们越希望较大的外部不相似性。在实际中，我们可以调整k的取整来获得不同的效果。当k值很大时，算法倾向于分割出来较大的块，当k值较小时，算法倾向于更细的划分。

本节最后我们探讨一个比较有趣的话题，就是τ函数的选取，如果我们改变这个函数，不会对算法的大框架造成影响，而会对分割结果的倾向性有影响。比如我们可以让分割倾向于某一种形状A，令τ函数在区域不是形状A的时候较大即可。这种形状上的倾向可以比较简单，比如希望正方形的或者扁平状的，也可以比较复杂，是一种特殊的形状。

4、分割算法

本节讲解主要的算法部分，怎样利用上述的定义，在基于图的表示方法下，做出高效而准确的分割。算法的核心：

输入是一个图$G=(V,E)$，有n个点和m个边。输出是一个分割V，分割成$S=(C_1,...,C_2)$.

1. 对E进行排序，生成非递减的序列$\pi=(o_1,...,o_m)$

2. 从初始分割S^0开始，每一个点v_i自己就是一个区域

3. 对于每一个$q=1,...,m$重复步骤3

4. 通过S^{q-1}构建S^q，使用如下的方式：令v_i和v_j表示按顺序排列的第q条边的两个点，比如$o_q=(v_i,v_j)$。如果v_i和v_j在S^{q-1}中连个不同的区域下，并且$w(o_q)$比两个区域的内部不相似度都小，那么合并这连个区域，否则什么也不做。用公式来表达就是：令C_i^{q-1}是S^{q-1}的一个区域，它包含点v_i；令C_j^{q-1}是S^{q-1}的一个区域，它包含点v_j。如果$C_i^{q-1}\neq C_j^{q-1}$并且$w(o_q)\leq MInt(C_i^{q-1},C_j^{q-1})$，那么通过合并$C_i^{q-1}$和$C_j^{q-1}$我们得到了$S^q$；否则的话$S^q=S^{q-1}$

5. 返回$S=S^m$

分割结果如图所示：

2.7 图像融合

图像融合是指将多幅包含相关信息的图像处理成一幅图像的过程。相比于每一幅输入图像，输出图像往往包含了更丰富的信息。图像融合方法大体上可分为两类，一类是空间域融合，另一类是变换域融合。

经过图像配准之后，不同的图像经过 2D 变换，变换到正确的位置上。对于某些重合区域的像素来说，该位置上有两个或多个以上的像素，图像融合问题就是利用怎样的规则求得这些位置上的像素的值。常见的有取均值法，Brovey 方法，主成分分析法以及基于高频率波法，IHS 和基于曲波变换等技术。

在本文的图像重建系统中，图像融合是重建部分的最后一步，本文的图像融合有这样几个特点：

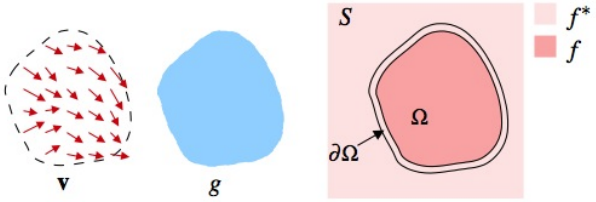
1. 待融合的图像块数很多，通常在几十到几百个；
2. 图像块的大小跨度很大，小的块有几十个像素，大的块有上万个像素；
3. 因为融合之前对每一个图像块进行了分割，所以图像块并不一定是完整的，融合的可能是块的一部分区域；
4. 有一张原图像的下采样图像作为参考图像，为图像融合提供了可靠的依据

通过观察上述四个特点我们发现，如果我们以下采样图像作为第一块拼合图像，采用图像块由大到小的方式依次拼接，那么图像融合的任务转变成逐一将小图像块贴合在一个背景图像上，是一个典型的图像无缝拼接任务，解决这个问题的经典方案是泊松图像编辑。

2.7.1 泊松图像编辑

泊松图像编辑是一种自动的“无缝融合”两张图像的技术。文献[15]首次提出。该方法所用的核心数学工具是带狄里克雷边界条件的泊松偏微分方程, 狄里克雷边界条件指定了在影响域内未知函数的拉普拉斯算子, 以及在区域边界上的未知函数值的拉普拉斯算子[16]。

算法的构思简洁巧妙。首先定义问题：我们改变的图像是S（背景图像），我们剪切粘贴的图像是g（前景）。两幅图像的位置关系如下图所示：



两幅图像融合的标准时：允许图像 B 改变颜色，但是仍然能够保留 B 的完整的“细节”。细节包括 B 中的边缘、角点、过度等等。而从图像中提取这些细节的多种方法中，都会使用到**图像梯度**。图像梯度是描述图像的一种数学表达，描述的是像素与相邻像素的相对变化（本质上是像素与其相邻像素的差值）。我们需要寻找的就是相对描述子，因为图像 A 与图像 B 之间的不统一主要是因为他们颜色上的绝对差。因此，更为严格的泊松图像编辑的目标是：允许改变绝对信息，即图像 B 的颜色，但是在粘贴之后尽可能的保留 B 的相对信息，即图像梯度。

我们将图像 B 的边缘像素固定，其像素值为图像 A 的像素值，然后求解其余的在选取内的像素值，约束条件是保持图像 B 的原始梯度。

设对 g 进行校正后得到的图像是 f，f 能够更好的与 S 融合。g 的边界与 S 的边界完全一致，匹配选取内部的像素，向内融合：

$$f_{(x,y)} = S_{(x,y)} \forall (x,y) \in \partial f^* \tag{2-16}$$

其中∂f^* 表示f^* 的边界

我们期望 H 内部像素的梯度值等于 B 内部像素的梯度值。一个点上图像梯度的定义是：该像素与所有像素的差值的和

$$|\nabla f^*_{(x,y)}| = 4f^*(x,y) - f^*(x-1,y) - f^*(x+1,y) - f^*(x,y-1) - f^*(x,y+1) \tag{2-17}$$

我们需要解决的问题是一个求最小值的问题：

$$\min_{f} \iint_{\Omega} |\nabla f|^2 \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega} \tag{2-18}$$

这个求最小的问题满足欧拉 -拉格朗日 (Euler-Lagrange) 等式：

$$\Delta f = 0 \text{ over } \Omega \text{ with } f|_{\partial}\Omega = f^*|_{\partial}\Omega \tag{2-19}$$

其中$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ 是拉普拉斯算子，公式 (2-19) 是满足 Dirichlet 边界条件的拉普拉斯等式。我们将公式 (2-18) 稍加修改，引入一个引导域v，得到如下结果：

$$\min_{f} \iint_{\Omega} |\nabla f - v|^2 \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega} \tag{2-20}$$

公式 (2-20) 的解是满足狄利克雷边界条件的泊松等式：

$$\Delta f = div\mathbf{v} \text{ over } \Omega \text{ with } f|\partial\Omega = f^*|\partial\Omega \tag{2-21}$$

其中$div\mathbf{v} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$ 是$\mathbf{v} = (u, v)$ 的散度。

上式等效于求

$$\Delta\tilde{f} = 0 \text{ over } \Omega, \tilde{f}|\partial\Omega = (f^* - g)|\partial\Omega \tag{2-22}$$

其中$f = g + \tilde{f}$，我们的问题转变成求差值的问题。

2.7.2 泊松图像编辑离散解

如果一个相邻像素是（1）边界像素，那么它的值是固定的；（2）超出选区边界，被排除。下面的差分方程总结了每一个像素点的所有情况

$$\begin{aligned} |N|f(x,y) - \sum_{(dx,dy)+(x,y)\in\Omega} f(x+dx, y+dy) - \sum_{(dx,dy)+(x,y)\in\partial\Omega} A(x+dx, y+dy) \\ = \sum_{(dx,dy)+(x,y)\in\Omega\bigcup\partial\Omega} f^*(x+dx, y+dy) - f^*(x,y) \end{aligned} \tag{2-23}$$

其中(x,y) 是 2D 网格上感兴趣的像素点的位置。N 是相邻像素的数量（包含边界像素，N<=4）。Ω 是 B 选区（不包含边界），$\partial\Omega$ 是边界，(dx,dy) 是可能的相邻像素点位置，包括 (-1, 0), (1, 0), (0, -1), (0, 1)。

等式左侧是计算未知像素点$f(x,y)$ 的空间梯度，计算的方式是将$f(x,y)$ 与每一个相邻的像素点做差值，并加和。每一个差值的形式都是$f(x,y) - other(x',y')$，其中(x',y') 是其他像素点的位置。

等式右边就是简单的计算f^* 图像在 (x,y) 处的梯度值，它与我们新的图像f 的像素值匹配。

对于 RGB 图像，这些公式会分别处理 R,G,B 三个通道。

下一步我们需要为 H 中的每一个点列出等式，注意到我们列出了一组线性方程，包含 k 个未知数，k 是我们需要求解的 H 中像素数量，最直接的方案是把所有的方程放到一个矩阵中，然后反转矩阵。然而，k 的值相当的大，对于 200X200 的选取而言，k 是 40,000。反转一个 40000X40000 的矩阵太庞大了。我们注意到矩阵是极为稀疏的，因为每一个点至多有 4 个相邻像素，（并且是正定的）。每一行至多有 5 个非零的元素，其余是 0。针对这样的特性，可以采用迭代矩阵求解算法。为了简便，我决定使用 Jacobi Method 来求解稀疏线性方程组。Jacobi Method 是梯度下降算法的一个特例。它的基本思路是：

- 以$Ax = b$ 的形式建立矩阵等式。A 是我上述定义的等式的矩阵，x 是我们待求解的值（本例中是 H 图像的像素值），b 是等式需要等于的值。如果你有一个稀疏矩阵，对它进行压缩是一个好主意（我的程序仅仅用数据结构存储哪些非零的条目）；
- 初始化 x，使之全为 0；
- 计算Ax 的积；
- 计算$b - Ax$ 的差值，这个差值衡量的是当前猜测的 x 的值和正确值之间的误差；
- 将差值$(b - Ax)$ 追加到 x 上。这就是我们让猜测向着正确方向前进的“梯度下降”的步骤；
- 重复步骤 3-5，直到 x 和$(b - Ax)$ 之间的差值足够小；

因为 A 是正定的，这个过程能够保证收敛到 x 的正确的解，并以指数速度收敛。

2.7.3 卷积近似解法

解等式 (2-22) 需要多次迭代，耗费大量时间，文献[17] 使用薄膜插值法（Membrane Interpolation）。f^* 的值可以通过近似方法得到，其思路是沿着区域边界平滑的拓展差值，直到展开到整个区域。差值可以写成卷积的形式：

$$\tilde{f} = \frac{G * \tilde{r}}{G * \tilde{R}} \tag{2-24}$$

其中

$$\begin{cases} \tilde{r}(x_i) = f^*(x_i) - g(x_i), & \forall x_i \in \partial\Omega \\ 0, & \text{其它} \end{cases} \tag{2-25}$$

而$\tilde{R}$是$\tilde{r}$的特征函数，$G(x_i, x_j)$是平移不变格林函数：

$$G(x_i, x_j) = G(\|x_i - x_j\|) = 2\pi \log \frac{1}{\|x_i - x_j\|} \tag{2-26}$$

卷积可以通过三个个滤波器进行快速的计算，这个计算的时间复杂度是O(n)，与像素数量呈线性关系。实验表明采用卷积近似解泊松方程得到的解的像素值和采用迭代求解差异不大，适用于本文的图像融合的场景下。

第三章 大规模近似重复图像搜索算法概述

随着多媒体业务的日益增长，近似重复图像搜索（Near Duplicate Image Retrieval）或部分重复图像搜索（Partial Duplicate Image Retrieval）技术得到了愈加广泛的应用。在我们的图像重建系统中的相似图像搜索环节，我们希望找到尽可能多的与用户拍摄图像相似的图像，将其作为后续重建环节的候选图像。因此我们面临的三个技术难点是：（1）相似搜索是在图像的局部进行的，而不是整幅图像，所以使用全局特征进行相似图像搜索的传统方案并不适用，是否有能表述局部特性的图像表示方法；（2）图像的局部特征信息较少，如何充分利用特征之间的几何位置关系进行图像局部匹配来提高搜索精度；（3）云端图像数据库是 Web 规模的（Web-Scale），图像数据量极大，对算法的时空复杂度限制较大。如何在使用图像局部特征和其空间位置关系的同时尽量不增加搜索算法的复杂度，是本系统需要解决的难题。

本章首先介绍传统的图像搜索算法，再介绍利用局部特征的空间信息的相似图像搜索算法，最后针对本论文的应用场景，提出一种结合多种技术的新的相似图像搜索技术。

3.1 基于局部特征的相似图像搜索算法

最常见的基于局部特征的相似搜索算法包含两个环节。第一步，从图像中提取局部特征，图像的感兴趣区域可以通过自动的特征点检测或者均匀取样获得，最常见的局部特征描述子包括梯度方向直方图（histograms of oriented gradient，HOG）和 SIFT、SURF 等。从一幅图像抽取的特征集合叫做视觉词袋（Bag of Visual Words）。在第二步中，我们需要定义两个视觉词袋之间的相似性，第一类是直接比较两个视觉词袋的相似性，例如投票方法；第二类是通过视觉词袋计算一个特征签名（signature，通常是一个向量），进而比较两个签名之间的相似度。两种方式都需要对数据库中的所有图像与请求的图像比较相似度并排序[4]。

3.1.1 视觉词袋模型

视觉词袋（Bag of Visual Words）模型是图像表示中最为经典的一种表示方法。它经常被用来进行图像分类和相似性搜索领域。它来自文档检索基于关键字查询的

方法中词袋（Bag of Words）的表示方法，其基本思想是：（1）统计语料库中的所有单词，生成单词表；（2）对于每一篇文档，统计每一个单词出现的频次，用由这些单词出现的次数生成直方图，用直方图来表示这篇文档。这种直方图的表示就是词袋表示。视觉词袋类似于 BoW 模型，算法的基本思路如下：

（1）离线部分：

- 提取特征：根据使用场景与实际业务的不同，可以选择不同的特征，文献[18]对视觉词袋模型进行深入的分析，综合比较了各种特征检测器、描述子等。在这一步，我们综合考虑特征的时空复杂度、鲁棒性、可区分性等。

- 生成视觉词码表：统计图像数据库中出现的所有特征，去除冗余的特征（比如几乎每一篇文档中都会出现的特征，类似于文档中的停用词）组成视觉词码表（Codebook）。如果提取的图像特征过多，一般需要对特征进行量化，利用聚类算法先把相近的单词归为一类（类似于文档检索里的找词根），利用聚类的结果生成视觉词码表。

- 利用视觉词码表量化所有的图像特征

- 利用词频表示的词袋模型来表示数据库中的每一幅图像

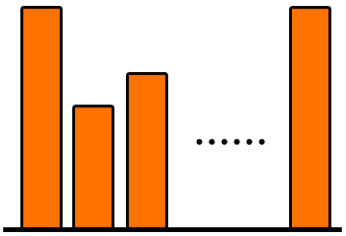
（2）在线部分：

- 提取请求图像的局部特征；

- 利用视觉词码表量化该图像的图像特征；

- 利用词频表示的词袋模型来表示请求图像；

- 利用词频表示做进一步的处理，例如分类，相似性比较等。



3.1.2 局部特征的聚类

生成视觉词码表是将局部特征进行聚类的过程，最经典的聚类算法是 K 均值聚类（K-MEANS），在实际中，我们通常在云端图像语料集中随机的选取一部分的特征，对这部分特征使用 K-MEANS 进行聚类，确定所有类中心后，再对语料集中的所有特征进行处理，找到每一个特征距离最近的类中心，完成对所有局部特征的量化。后续的请求图像在提取特征后按照同样的方式进行量化。

然而，在本文提出的系统中，图像局部特征数量极大（对于上百万图像素的图像来说，每幅图像的 sift 数量平均在 3000 以上，100K 的图像集上，有 300M 以上个 128D 的特征。），传统的 K-MEANS 不能满足我们的性能要求，最近的一些研究针对图像据图特征提出了许多快速聚类的方法，文献[19–21] 等提出了近似 K 聚类（Approximate K-MEANS，AKM）和结构化 K 聚类（Hierarchy K-MEANS，HKM）的方案，。

近似 K-MEANS 是传统 K-MEANS 的一种替代，传统的 K-MEANS 的时间花销主要是在计算特征点最近邻的类中心上，每一次迭代，我们需要计算每一个特征点，计算它和每一个类中心的距离，所以每次迭代的时间复杂度是$O(NK)$，其中 N 是特征点数量，K 是类中心的数量。在改进的版本中，每一次迭代之前，我们使用随机 k-d 森林来构建类中心来加快速度。在常规的 k-d 树中，我们需要决定每一次划分是在哪一个维的哪一个点上，通常我们选择方差最大的一个维度作为划分维度，以该维度上中值点作为划分点，在同一维度上比划分点小的点落在 k-d 数当前节点的左侧，大的落在右侧。在随机 k-d 森林中，每一棵 k-d 树都是构建在所有的类中心上，不过在构建时采用划分策略有所不同，划分维度是方差较大的几个维度中随机选择的，划分点也是随机的在中值附近选择一个点。所有的 k-d 树组成了 k-d 森林，这个森林构建了一个互相交叠的特征划分空间。因为量化的存在，一个在划分边缘的特征很可能找到错误的最近邻类中心，而互相交叠的划分方式则大大减轻了这一影响，增强了高维计算的鲁棒性。

计算一个特征点所属类的过程如下：对随机森林中的每一棵树，递归的下滤到其叶子节点，计算它到可区分边界的距离，将所有的距离记录在一个优先队列中。迭代的选择最近的划分，持续的将隐藏节点加入到优先级队列中，当迭代次数达到指定数值的时候，搜索停止。

K-MEANS 的时间复杂度是$O(NK+N)=O(NK)$，AKM 算法的时间复杂度是$O(Nlog(K)$。实验表明，AKM 在大幅降低时间复杂度的同时，保证了正确率

（与 K-MEANS 划分的类中中心相比不一致的几率 <1%）。————————————TO Do Here————————————

3.1.3 相似性度量

3.2 改进的相似搜索算法

文献[4]对近期的大规模相似图像搜索技术做了总结，提到了 Partial-Duplicate Image Retrieval via Saliency-Guided Visual Matching[22] 技术，通过视觉显著性（saliency）模型进行比较，消除背景中的噪声。这种方法使得索引和匹配都集中在显著性区域，更能够符合用户的预期。显著值和空间约束都能够被用来进行相似性度量，并且能够高效的进行二级索引，对于大规模的 partial duplicate search 非常有利，但是内存开销比较大。

Web-Scale Image Retrieval Using Compact Tensor Aggregation of Visual Descriptors[23] 描述了目前存在的各种视觉描述子的概况，介绍了相关的索引技术，包括哈希、词袋以及基于树的表示方法。（hashing, bag-of-words, and tree-based representation）引出内存开销问题并提出一种生成高度压缩签名（highly compact signatures）的方法，包括张量聚合，PCA，kernel PCA 等一些列算法。它改进了 Fisher Vector 族描述子，提高它的可区分性，以及特征签名的大小（feature discrimina- tive power and the size of feature signature）。

对于相似性视频搜索，它的特点是特征维数特别大，有研究提出了稀疏投影方式进行特征降维，并且使用数据挖掘的知识使用一些 metadata 来共同进行搜索[24]。使用机器学习技术，学习稀疏投影矩阵（sparse projection matrices）。这种学习方法可以选择性的使用外部信息，比如 WikiPedia 上的知识和 Google 搜索结果中的摘要，创建一个语义相关的投影矩阵，生成一个压缩签名，以满足手机媒体检索的诸多限制。手机内存空间小，计算资源有限，传统的将高维特征映射到低维的投影矩阵在手机内存是放不下的。而我们的稀疏投影矩阵是能够在手机上使用的。

下面我们简单介绍各种性能改进的方法。

3.2.1 sim-hash

文本的去重算法中常见的有余弦夹角算法、欧式距离、Jaccard 相似度、最长公共子串、编辑距离等，但是只适合于小数据集。simhash 传统的用来判断两篇文章的相似度，将两篇文章映射到低维空间上，并且保持它们互相之间的相似度，但是

它很难应用在图像比较上，因为图像的特征是用实数来表示的，尽管可以将其量化，但是两幅相似图像量化后的特征集合交叠的比率仍旧很小，远远小于文档，因为两幅图像不相似的区域的噪声特征非常大。但是如果使用视觉词组，那么如果两个相似区域的视觉词组会非常相同，我们就可以使用 simhash 了。所以，min-hash 的使用场景是特征比较多，相似度比较显著的情况下。

3.2.2 最小哈希的相似性比较

最小哈希方法是一种广泛应用在相似性查找领域的算法。

在生成视觉词带之后，比较两幅图像或者两篇文章的相似度问题转化为比较两个只包含 0，1 元素的集合的相似度，集合的相似度是 Jaccard 相似度。

$$JS(A,B)=\frac{|A\cap B|}{|A\cup B|}$$

我们首先定义一组随机哈希函数$f_j:\mathscr{V}\rightarrow R$, 每一个哈希函数是独立的，将一个视觉词映射为一个实数。两个不同的视觉词X_a 和X_b，哈希函数需要满足两点：（1）$f_j(X_a)\neq f_j(X_b)$；（2）$P(f_j(X_a))<(P(f_j(X_b))=0.5$。

注意到函数f_j 能够反映出视觉词集合的一个顺序排列情况，我们定义 min-hash 就是这个排列中排在最前面的视觉词，因此有

$$m(\mathscr{A}_i,f_j)=arg\min_{X\in A_i} f_j(X)$$

根据上述定义，我们会发现以下这个事实：

$$Pr(m(A,f_j)=(B,f_j))=\frac{|A\cap B|}{|A\cup B|}=sim_s(A,B)$$

如果 r 是随机变量，当$m(A,f_j)=(B,f_j)$ 时值为 1，其它情况下值为 0 的，那么 r 可认为是 J(A,B) 的无偏估计。因此我们可以使用 min-hash 函数将一幅图片或者一篇文章转化为一个数（对该文章中的每一个单词 id 使用 hash 函数后得到一个新的 id 序列，这个序列中的第一个出现 1 的行号，就是 min-hash 的值），当我们使用 k 个 hash 函数，得到 k 个值，将原本的高维向量映射到了低维。min-hash 在压缩原始集合的情况下，保证了集合的相似度没有被破坏。文献[7] 中提出将词频 -逆文档频率（term frequency–inverse document frequency，简称 TF-IDF）融合在传统的 min-hash 算法中，实验表明能够提高搜索的准确率。

3.2.3 LSH

使用 min-hash 对数据降维后，可以使用 LSH 缩小查找范围，其基本思路是将相似的集合聚集到一起，减小查找范围，避免比较不相似的集合。

对每一列 c（即每个集合）我们都计算出了 n 行 min-hash 值，我们把这 n 个值均分成 b 组，每组包含相邻的 r=n/b 行。对于每一列，把其每组的 r 个数都算一个 hash 值出来，把此列的编号记录到 hash 值对应的 bucket 里。如果两列被放到了同一个 bucket 里，说明它们至少有一组 (r 个) 数的 hash 值相同，此时可认为它们有较大可能相似度较高（称为一对 candidate）。最后在比较时只对落在同一个 bucket 里的集合两两计算，而不是全部的两两比较。

3.3 基于空间信息的匹配搜索算法

使用视觉词袋模型来表示图像并比较视觉词袋之间的相似性做法比较成熟、最为普及的做法

3.3.1 随机抽样一致算法

随机抽样一致 RANdom SAmple Consensus（RANSAC）是一种空间匹配算法。该算法将数据分成两类，局内点 (inlier) 和局外点（outlier）它可以从一组包含局外点的观测数据集中，通过迭代方式估计数学模型的参数。

这是一种不确定的算法，有一定的概率得出一个正确的或者说是可接受的合理结果；一般情况下，迭代次数的增加可以提升结果的准确性。该算法由 Fischler 和 Bolles 于 1981 年提出，在图像检索中，RANSAC 可以作为检索后的后续处理，对图像的中目标进行空间一致验证。

RANSAC 算法对数据集做了三个假设：

- 数据由局内点组成，局内点的数据的分布符合某一特定的概率模型；
- 与局内点相对的是局外点，他们不能够适应该模型;
- 局内点与局外点之外的数据属于噪声

RANSAC 有以下几个步骤：

- 随机选择数据集合的一个子集

- 使用选择的自己拟合一个数学模型
- 确定该模型下局外点的个数
- 重复步骤 1 3 若干次，以最好的一次结果最为最终拟合出来的数学模型

RANSAC 算法迭代次数的选取取决于我们期望的准确率与样本数量。设 p 为任意给定对应点合法的概率，即

$$p = \frac{\text{局内点的数量}}{\text{数据集全部数据的数量}}$$

而 P 是经过 S 次试验后成功的总体概率。设我们需要 k 个随机样本来估计模型，那么在一次试验中，该 k 个样本都是局内点的可能性为p^k。因此，S 次试验失败的可能性是

$$1 - P = (1 - p^k)^S$$

两边去对数，得到最少需要的试验次数是

$$S = \frac{log(1-P)}{log(1-p^k)}$$

随着 k 的增大，需要的最少试验次数增多，在实际中，我们应该尽可能的选择小的 k 值。在模型确定以及最大迭代次数允许的情况下，RANSAC 总是能找到最优解。对于含有较大误差的数据集，RANSAC 的效果远优于直接的最小二乘法。

当对两幅图像进行匹配的时候，所以相互匹配的局部特征作为数据全集，我们要估算的模型是一个变换矩阵 H，能够将图像I 投影到图像I'。每次迭代过程中，随机的选择四对匹配的特征点，根据这四个特征点的位置信息解得变换 H，利用 H 计算其它匹配对的位置信息中有哪些属于局外点，记录局外点的个数。局外点的个数越少，变换矩阵 H 越准确。反复迭代多次得到一个相对准确的透视变换模型。

上述提到的用四对匹配点拟合出的变换矩阵叫做单应矩阵（Homography）, 最简单的求解单应性矩阵的算法叫做直接线性变换法（Direct Linear Transform,DLT）[25]，其具体算法如下：

假设我们相匹配的一对点分别是x 和x'，单应性矩阵是H，那么有如下等式：

$$c\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = H\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{3-1}$$

其中 c 是一个非零常数，$(u\ v\ 1)^{\mathrm{T}}$ 代表x'，$(x\ y\ 1)^{\mathrm{T}}$ 代表x，而 $H=\begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix}$

将公式 (3-1) 展开，分别用第一行和第二行除以第三行，得到

$$-h_1x-h_2y-h_3+(h_7x+h_8y+h_9)u=0 \tag{3-2}$$

$$-h_4x-h_5y-h_6+(h_7x+h_8y+h_9)u=0 \tag{3-3}$$

公式 (3-2) 和 (3-3) 可以写成矩阵的形式：

$$A_ih=0 \tag{3-4}$$

其中 $A_i=\begin{pmatrix} -x & -y & -1 & 0 & 0 & 0 & ux & uy & u \\ 0 & 0 & 0 & -x & -y & -1 & vx & vy & v \end{pmatrix}$，而 $h=(h_1\ h_2\ h_3\ h_4\ h_5\ h_6\ h_7\ h_8\ h_9)^{\mathrm{T}}$。因为每一对匹配的点可以提供两个等式，对于解决 8 自由度的矩阵 H，只需要四对（任意三点不能共线）匹配的特征点。

DLT 算法依赖于坐标系的原点和尺度，所以该算法并不稳定，在实际中更多的使用多个匹配点得到更多的方程，将求单应矩阵的问题转化为求解最小二乘的问题，用矩阵奇异值分解（Singular value decomposition，SVD）的方法来求解等式$Ah=0$。

3.3.2 视觉词组

图像搜索与文本搜索的一个显著区别是图像是二维的，包含大量的空间关系信息，而 BoW 的一个被人诟病的问题便是没有利用任何的图像空间信息。随着图像业务需求的提高，有更多的学者提出利用图像局部特征的空间位置信息进行更加精确的相似图像搜索[3,5,6,19,26]。

文献[3] 深入研究 SIFT 描述子。提出了一个非常优雅的方法：生成 SIFT 组，嵌入几何信息，最终将一组 SIFT 压缩到一个 64 比特的二维签名中，叫做 Nested-SIFT。它的优点是 Nested—SIFT 使用 SIFT 描述子的嵌套关系，很自然的将不同尺度的局部关键点组合在一起，生成一个特征签名。嵌入空间信息的 Nested—SIFT 可区分性更强。使用 SimHash 进行压缩后，在视觉搜索中效率更高。实验结果表明这种方法

提高搜索的准确度，减少了内存消耗，提高搜索速度。其缺点是生成 Nested-SIFT 会有一定的计算消耗。

文献[26] 采用较为复杂的空间编码，对图像 2D 空间进行了不同维度的划分，区分了局部特征的水平与垂直方向，并且加入了扇形区域的编码，增加了视觉词组的旋转不变性。该算法较为复杂，适用于全局图像的相似查询，实验结果表明加入空间信息验证后，能够提升准确率，两幅不相关的图像可能会有相似的特征集合，但是相似特征集合在空间位置上依然保持相似的概率极低。

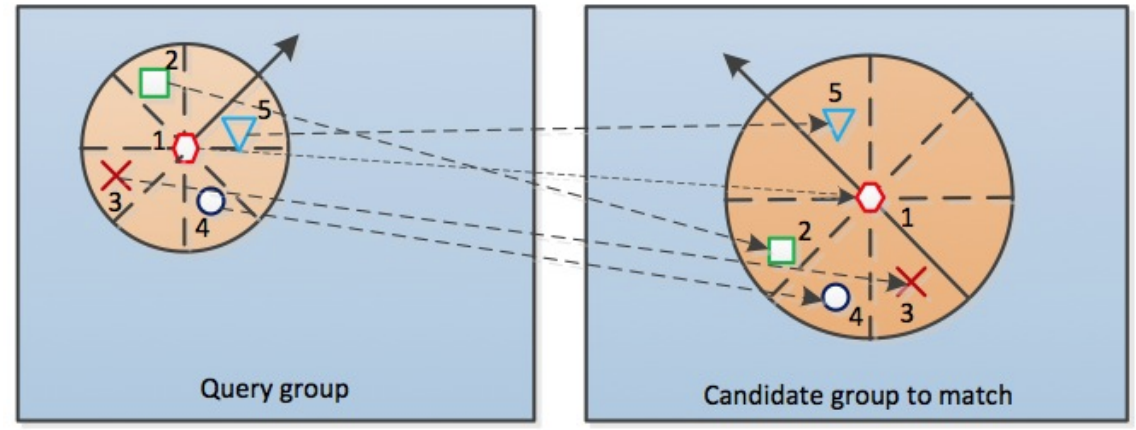
因为本文系统需要查询的是具有局部相似的图像块，相比于其他相似性匹配算法，我们需要的图像块粒度更细，即云端图像与请求图像全局相似度可能很低，但是局部相似度非常高，这幅图像也会被加入到候选图像中。文献[10] 提出了简洁的做法将一个图像块内的局部特征编码成视觉词组。本位在该文基础上稍作改进，在保证算法效率的同时，增加了对算子尺度的编码，进一步增强其准确度。

对于每一个图像块而言，中心位置有一个视觉词，该视觉词的尺度与其覆盖范围（影响范围）成正比，在该范围内，有若干视觉词，我们将范围内的所有视觉词看做一个视觉词组（Visual Words Group）。我们希望将词组内的每一个视觉词分配一个编码 x，表征这个词在视觉词组的相对关系，这样我们可以采用如下的规则进行匹配：

$$E_m(G_x, G_y) = E_v(G_x, G_y) - E_r(G_x, G_y) \tag{3-5}$$

其中$E_v(G_x, G_y)$ 是能够匹配上的视觉词，这里匹配上定义为两个视觉词相同，并且含有相同的编码 x。其中$E_r(G_x, G_y)$ 是错误匹配的视觉词，错误匹配是指两个视觉词相同，但是含有不同的编码 x。

那么怎样编码视觉词，能够体现视觉词的相对关系呢？我们从两个维度对视觉词进行编码，一个是它与中心视觉词的相对方向，另一个是相对尺度大小。视觉词组的中心词的主方向作为基准方向，沿着顺时针或者逆时针方向，将整个区域分成 n 个子区域。接下来对于每一个子区域，根据 sift 算子量化前的尺度信息对比值大小的不同，将子区域分成 r 个维度，这样一个视觉词共有 n*r 个子区域，如图所示：


2
5
1
3
4
Query group
5
1
2
4
3
Candidate group to match

附录 A 不定型（0/0）极限的计算

定理 A.1（L'Hospital 法则） 若

1. 当 $x \to a$ 时，函数 $f(x)$ 和 $g(x)$ 都趋于零；
2. 在点 a 某去心邻域内，$f'(x)$ 和 $g'(x)$ 都存在，且 $g'(x) \neq 0$；
3. $\lim\limits_{x\to a} \dfrac{f'(x)}{g'(x)}$ 存在（或为无穷大），

那么

$$\lim_{x\to a} \frac{f(x)}{g(x)} = \lim_{x\to a} \frac{f'(x)}{g'(x)}. \tag{A-1}$$

证明： 以下只证明两函数 $f(x)$ 和 $g(x)$ 在 $x = a$ 为光滑函数的情形。由于 $f(a) = g(a) = 0$，原极限可以重写为

$$\lim_{x\to a} \frac{f(x) - f(a)}{g(x) - g(a)}.$$

对分子分母同时除以 $(x - a)$，得到

$$\lim_{x\to a} \frac{\dfrac{f(x) - f(a)}{x - a}}{\dfrac{g(x) - g(a)}{x - a}} = \frac{\lim\limits_{x\to a} \dfrac{f(x) - f(a)}{x - a}}{\lim\limits_{x\to a} \dfrac{g(x) - g(a)}{x - a}}.$$

分子分母各得一差商极限，即函数 $f(x)$ 和 $g(x)$ 分别在 $x = a$ 处的导数

$$\lim_{x\to a} \frac{f(x)}{g(x)} = \frac{f'(a)}{g'(a)}.$$

由光滑函数的导函数必为一光滑函数，故 (A-1) 得证。 □

参考文献

[1] Yue H, Sun X, Yang J, et al. Cloud-based Image Coding for Mobile Devices-Toward Thousands to One Compression [J], 2013.

[2] Brown M, Lowe D G. Automatic Panoramic Image Stitching using Invariant Features [J]. International Journal of Computer Vision, 74 (1), 2006: 59–73.

[3] Xu P, Zhang L, Yang K, et al. Nested-SIFT for Efficient Image Matching and Retrieval [J], 2013.

[4] POLICY N. Web-Scale Near-Duplicate Search: Techniques and Applications [J]. IEEE MultiMedia, 2013.

[5] Wu Z, Ke Q, Isard M, et al. Bundling features for large scale partial-duplicate web image search [C]. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, 2009: 25–32.

[6] Zhou W, Lu Y, Li H, et al. Spatial coding for large scale partial-duplicate web image search [J], 2010: 511–520.

[7] Chum O, Philbin J, Zisserman A. Near Duplicate Image Detection: min-Hash and tf-idf Weighting. [J], 2008.

[8] Weinzaepfel P, Jegou H, Perez P. Reconstructing an image from its local descriptors [C]. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, 2011: 337–344.

[9] Daneshi M, Guo J. Image reconstruction based on local feature descriptors [J], 2011.

[10] Dai L, Yue H, Sun X, et al. IMShare: instantly sharing your mobile landmark images by search-based reconstruction [J], 2012: 579–588.

[11] Demoment G. Image reconstruction and restoration: Overview of common estimation structures and problems [J]. Acoustics, Speech and Signal Processing, IEEE Transactions on, 37 (12), 1989: 2024–2036.

[12] Park S C, Park M K, Kang M G. Super-resolution image reconstruction: a technical overview [J]. Signal Processing Magazine, IEEE, 20 (3), 2003: 21–36.

[13] Lowe D G. Distinctive image features from scale-invariant keypoints [J]. International Journal of Computer Vision, 60 (2), 2004: 91–110.

[14] 李航. 统计学习方法. 2012.

[15] Pérez P, Gangnet M, Blake A. Poisson image editing [J]. ACM Transactions on Graphics (TOG), 22 (3), 2003: 313–318.

[16] 张建桥, 王长元. 基于泊松方程的数字图像无缝拼合 [J]. 现代电子技术, 33 (017), 2010: 139–141.

[17] Farbman Z, Fattal R, Lischinski D. Convolution pyramids [C]. In the 2011 SIGGRAPH Asia Conference, New York, New York, USA, 2011: 1.

[18] Zhang J, Marszalek M, Lazebnik S, et al. Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study [C]. In Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on, 2006: 13.

[19] Philbin J, Chum O, Isard M, et al. Object retrieval with large vocabularies and fast spatial matching [C]. In Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on, 2007: 1–8.

[20] Muja M, Lowe D G. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. [J], 2009: 331–340.

[21] Wang J, Hua X-S, Li S, et al. Optimized KD-Tree for Scalable Search [J], 2010.

[22] Li L, Jiang S, Zha Z-J, et al. Partial-Duplicate Image Retrieval via Saliency-Guided Visual Matching [J]. MultiMedia, IEEE, 20 (3), 2013: 13–23.

[23] Negrel R, Picard D, Gosselin P. Web scale image retrieval using compact tensor aggregation of visual descriptors [J], 2013.

[24] Wu G-L, Kuo Y-H, Chiu T-H, et al. Scalable mobile video retrieval with sparse projection learning and Pseudo label mining [J], 2013.

[25] Dubrofsky E. Homography estimation [J], 2009.

[26] Zhou W, Li H, Lu Y, et al. Sift match verification by geometric coding for large-scale partial-duplicate web image search [J]. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP), 9 (1), 2013: 4.

致　谢

感谢 Donald Ervin Knuth.

攻读学位期间发表的学术论文目录

期刊论文

[1] **Zhang San**, Newton I, Hawking S W, et al. An extended brief history of time [J]. Journal of Galaxy, 1234 (4), 2079: 567–890.（SCI 收录，检索号：786FZ）.

会议论文

[1] McClane J, McClane L, Gennero H, et al. Transcript in Die hard [C]. In Proc. HDDD 100th Super Technology Conference (STC 2046), Eta Cygni, Cygnus, September 21–24, 2046: 123–456.（EI 源刊）.