



浙江工业大学

硕士学位论文

基于多视图的三维重建及其模型重拓扑

作者姓名	潘志颖
指导教师	张剑华 副教授
学科专业	软件工程
学位类型	工学硕士
培养类别	全日制学术型硕士
所在学院	计算机科学与技术学院

提交日期：2020 年 06 月

3D Multi-View Reconstruction and Model Re-topology

Dissertation Submitted to
Zhejiang University of Technology
in partial fulfillment of the requirement
for the degree of
Master of Engineering



by

Zhi-ying PAN

Dissertation Supervisor: Associate Prof. Jian-hua ZHANG

June., 2020

浙江工业大学学位论文原创性声明

本人郑重声明：所提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研究成果。除文中已经加以标注引用的内容外，本论文不包含其他个人或集体已经发表或撰写过的研究成果，也不含为获得浙江工业大学或其它教育机构的学位证书而使用过的材料。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本人承担本声明的法律责任。

作者签名：

日期： 2020 年 6 月

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权浙江工业大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

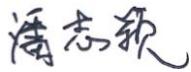
本学位论文属于 1、保密，在一年解密后适用本授权书。

2、保密，在二年解密后适用本授权书。

3、保密，在三年解密后适用本授权书。

4、不保密。

(请在以上相应方框内打“√”)

作者签名：

日期： 2020 年 6 月

导师签名：

日期： 2020 年 6 月

中图分类号 TP391

学校代码 10337

UDC 004

密级 公开

研究生类别 全日制学术型硕士



浙江工业大学

工学硕士学位论文

基于多视图的三维重建及其模型重拓扑

3D Multi-View Reconstruction and Model Re-topology

作者姓名 潘志颖

第一导师 张剑华

学位类型 学术型硕士

学科专业 软件工程

培养单位 计算机科学与技术

研究方向 计算机视觉

学院

答辩日期： 2020 年 5 月 31 日

基于多视图的三维重建及其模型重拓扑

摘要

随着计算机视觉技术的发展，三维重建技术成为一个热门的研究方向，该技术也越来越多的应用出现的各个领域，例如增强现实、历史建筑模型建模等方面。

在三维重建方面，现在的技术还难以生成高质量的模型，容易出现面片缺失的情况。另外在实际的生产环节中，三维重建生成的模型都存在着数据量大、冗余面多或贴图杂乱难以编辑的问题，不能直接用于增强现实应用、3D 游戏等对模型质量要求高的行业。本文主要通过提升特征匹配效果来提升三维重建效果，以及通过模型的重拓扑算法将生成的三维网格转为半规则网格。本文的主要工作和成果如下：

1. 针对已有三维重建方法效果不理想的问题，本文在增量式稀疏重建中的图像特征匹配阶段进行优化工作。本文方法结合了基于网格统计的特征匹配方法和重新匹配算法，提升特征匹配准确性和匹配数量。基于网格统计的匹配方法可以将错误匹配的特征点对排除，重新匹配算法则是通过随机采样一致性算法来验证候选匹配队列中是否存在潜在的匹配特征点，从而增加特征点匹配数目，提高特征匹配的效果，实现减少三维重建模型面片缺失的情况。

2. 针对难以处理由三维扫描或多视图重建生成的模型纹理的问题，本文提出了一种基于机器学习的网格模型重拓扑算法。它能够将任意带纹理坐标的模型转化为半规则网格，本文会保留其纹理并且对模型中的环境光遮蔽进行重新处理。通过使用神经网络寻找奇异点位置，计算并优化方向场以生成半规则网格。实验表明，本方法能够显著地减少模型面数，生成规整的网格模型，明显改善模型纹理难编辑的问题，并且有较好的重拓扑效果。

关键词：增量式，三维重建，特征点匹配，从运动恢复结构，模型重拓扑

3D MULTI-VIEW RECONSTRUCTION AND MODEL RE-TOLOPOGY

ABSTRACT

With the development of computer vision technology, three-dimensional reconstruction technology has become a hot research direction, and it has been applied in more and more fields, such as augmented reality, historical building model modeling and so on.

In the aspect of three-dimensional reconstruction, it is still difficult to generate high-quality models with the current technology. In addition, in the actual production process, the models generated by three-dimensional reconstruction have many problems such as too much data, redundant polygons or confused maps, which are difficult to edit. They cannot be directly used in augmented reality applications, 3D games and other industries that require high quality models. This paper mainly improves the effect of three-dimensional reconstruction by improving the effect of feature matching and converts the generated three-dimensional mesh to semi-regular mesh by the model's re-topological algorithm. The main work of this paper are listed as follows:

1. To solve the problem of poor 3-D reconstruction, we optimize the image feature matching phase in incremental sparse reconstruction. Combining the feature matching method based on grid statistics and the re-matching algorithm, the matching method based on grid statistics can exclude the mismatched feature pairs, and the re-matching algorithm verifies the candidate matching by random sampling consistency algorithm, which is realized by verifying whether there are potential matching feature points in the queue, and this method can increase the number of feature point matches, improve the effect of feature matching, and reduce the missing patches of the three-dimensional reconstruction model.

2. To solve the problem that textures are difficult to process in the models generated by three-dimensional scanning or multi-view reconstruction, we propose a machine-learning based mesh re-topology algorithm. It can convert any model with texture to a semi-regular mesh, and we will retain its texture and re-process the ambient light shading on model. First, we use a neural network to find the singularity of the model, then we calculate and optimize the direction fields to generate the semi-regular mesh. Experiments show that this method can significantly reduce the

number of polygons in the model, generate semi-regular models. Our method can significantly reduce the difficulty of editing model texture, and have a good topological effect.

KEY WORDS: Incremental, 3-D reconstruction, feature point matching, struct from motion, model re-topology

目 录

摘 要	I
ABSTRACT	III
插图清单	VII
附表清单	IX
第一章 绪论	1
1.1 引言	1
1.2 国内外研究现状	2
1.2.1 增量式从运动恢复结构	3
1.2.2 全局式从运动恢复结构	6
1.2.3 混合式从运动恢复结构	8
1.2.4 层次式从运动恢复结构	9
1.2.5 重网格化研究现状	10
1.2.6 环境光遮蔽研究现状	12
1.3 研究内容	13
1.4 本文的主要工作及组织结构	14
第二章 三维重建理论基础	15
2.1 相机与图像	15
2.1.1 图像像素坐标系	15
2.1.2 像平面坐标系	15
2.1.3 相机坐标系	16
2.1.4 世界坐标系	17
2.2 图像畸变矫正	18
2.3 特征点提取	19
2.4 对极几何	21
2.5 三角化	24
2.6 曲面的表示方式	25
2.7 多边形网格的格式	25
2.8 四边形网格的重网格化	26
2.9 本章小结	28
第三章 基于匹配优化的从运动恢复结构	29
3.1 引言	29
3.2 图像预处理	29

3.3 特征提取和匹配.....	31
3.3.1 特征提取.....	31
3.3.2 基于统计的特征匹配算法.....	32
3.4 增量式稀疏重建.....	38
3.5 实验结果与分析.....	40
3.6 本章小结.....	45
第四章 基于机器学习的网格重拓扑和光照调整.....	46
4.1 引言.....	46
4.2 寻找端点.....	47
4.3 重网格化.....	48
4.4 修改光照.....	52
4.5 实验结果与分析.....	53
4.6 本章小结.....	59
第五章 结论与展望.....	60
5.1 结 论.....	60
5.2 展 望.....	61
参考文献.....	62
致 谢.....	65
作者简介.....	66
1 作者简历.....	66
2 攻读硕士学位期间发表的学术论文.....	66

插图清单

图 1-1 多视图三维重建过程 ^[8]	1
图 1-2 兔子模型	2
图 1-3 网络数据集重建软件 ^[10]	4
图 1-4 VisualSFM 流程图	4
图 1-5 COLMAP 流程图	5
图 1-6 不同场景下的动态物体误匹配 ^[16]	7
图 1-7 混合式重建流程图	8
图 1-8 总体框架流程图	13
图 2-1 图像像素坐标系	15
图 2-2 像平面坐标系	16
图 2-3 相机坐标系	17
图 2-4 世界坐标系与相机坐标系	18
图 2-5 枕形畸变和桶形畸变 ^[52]	19
图 2-6 极值点检测 ^[53]	20
图 2-7 对极几何示意图	21
图 2-8 三角化示意图	23
图 2-9 深度的不确定性	24
图 2-10 各向同性	27
图 2-11 不规则网格(左)、介半规则网格(中)、规则网格(右) ^[54]	27
图 3-1 前景分割效果图	31
图 3-2 特征提取	32
图 3-3 基于统计的特征匹配算法流程图	33
图 3-4 基于统计的特征匹配效果图	34
图 3-5 重匹配算法流程图	36
图 3-6 匹配算法流程图	38
图 3-7 稀疏重建流程图	39
图 3-8 西瓜原始数据集	41
图 3-9 西瓜前景图	41
图 3-10 西瓜重建效果图	41
图 3-11 青蛙图片上的特征匹配算法比较	42
图 3-12 雕塑图像上的特征匹配算法比较	43

图 3-13 青蛙和人脸重建结果对比	44
图 3-14 兔子和西瓜的重建效果对比	45
图 4-1 系统流程图	46
图 4-2 奇异点位置	47
图 4-3 对模型文件进行展开的结果	47
图 4-4 图分片结果	48
图 4-5 方向场示意图	49
图 4-6 内部能量项转化	50
图 4-7 外部能量项	51
图 4-8 位置场示意图	52
图 4-9 2D 纹理过滤器 ^[51]	53
图 4-10 环境光的调整	53
图 4-11 模型 rockerarm	54
图 4-12 模型 fertility	55
图 4-13 模型 buddha	55
图 4-14 模型 fandisk	56
图 4-15 人头	56
图 4-16 兔子	57
图 4-17 大卫	57
图 4-18 青蛙	57
图 4-19 寺庙	57
图 4-20 西瓜	58
图 4-21 移除环境光的结果	58

表格清单

表 2-1 网格模型文件格式	26
表 3-1 特征匹配数目比较	44

第一章 绪论

1.1 引言

随着计算机技术的发展，三维重建系统有了很大的进展，它通常被用于很多领域，例如历史文物古迹的保存^[1]^[2]^[3]、增强现实^[4]、虚拟现实^[5]、医学图像^[6]等方面。三维重建是通过分析计算三维空间中物体的不同性质，例如颜色、纹理、反射等信息，通过计算机视觉技术将物体的形状颜色恢复出来，并在计算机中以约定好的方式进行模拟渲染的过程。

通常情况下，三维模型的获取主要分为三种方式，其一是通过人工手动建模的方式，利用 AutoCAD、3DMAX 等三维建模软件进行建模，这种方式需要富有经验的专业工作人员进行操作，并且需要耗费大量的人力物力等资源，一般适用于需要进行精细化显示的领域，例如游戏动画中的人物场景建模等。

第二种方法是通过红外深度相机、结构光或者激光等形式来进行^[7]，属于主动式三维建模方法，它一般都是通过将光线主动投影到物体上，通过计算光的反射光线的形变或者飞行时间来确定物体离相机的距离，这种形式一般都是会生成不具有颜色信息的模型，如果是和 RGB 相机结合，可以将颜色信息映射到生成的三维模型上，使其具有颜色。

第三种方法则是本文采用的方法，基于计算机视觉的被动式的三维重建系统，这种方法主要是通过计算不同视角拍摄物体图像的三维空间位置来重新恢复物体深度的，由于采用的是自然光线，可以直接获取到物体的纹理。基于多视图的三维重建是利用多幅图像之间的匹配关系和几何约束来确定不同的相机的空间位姿，然后通过相机的位姿来计算出被拍摄物体的空间点位置，之后进行稠密重建确定各个面片的位置和朝向。

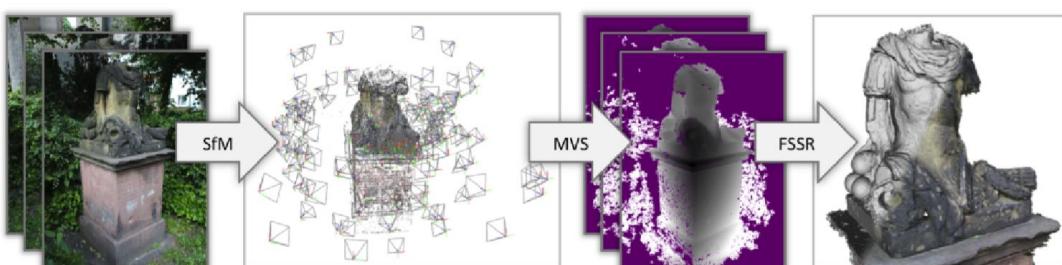
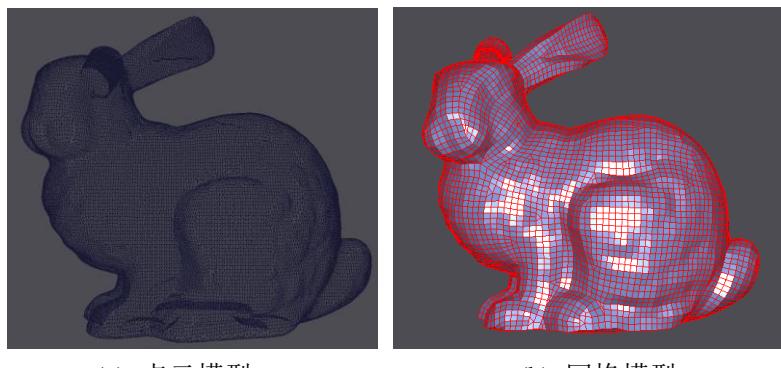


图 1-1 多视图三维重建过程^[8]

Figure 1-1. Multi-view 3D reconstruction process

基于计算机视觉的三维重建系统在受到物体外观、环境光线、相机模型、系统优化等因素的影响下，产生不精确的模型是不可避免的。为了生成更加准确的，更具有鲁棒性的三维模型，本文采用了结合基于网格统计方法图像匹配方法和改进的增量式重建方法，可以生成具有更多注册三维点云、更长的跟踪和更加具有鲁棒性的稀疏重建结果。

在生成三维重建模型之后，还需要对模型进行优化和微调工作。基于计算机视觉的三维重建方法生成网格模型一般都是有过多的复杂面片和杂乱的纹理贴图，然而这些方法生成的模型都存在着数据量大、冗余面多或贴图杂乱难以编辑的问题，不能直接用于手机增强现实应用、3D 游戏等对模型质量要求高的行业。



(a) 点云模型

(b) 网格模型

图 1-2 兔子模型

Figure 1-2. Bunny model

在计算机视觉中，100 多张高分辨率的图像通常能够生成高达 100 多万面的模型，大约需要占用几十兆的空间，这对网络带宽、模型渲染速度有着很大的挑战。由于建模生成的三维模型其 UV 展开比较混乱，设计人员通常难以直接确定 UV 展开图中每个分片对应于模型上的哪个位置，也就不能够对模型进行二次创作，所以需要能够将模型转化为可以生成规则 UV 展开图的半规则网格模型。针对上述问题，本文提出了一种基于机器学习的三维模型自动重拓扑和纹理重映射方法。

1.2 国内外研究现状

三维重建主要分为主动式三维重建和被动式三维重建，主动式三维重建主要采用将结构光或者红外光投影到物体上，以获取物体的深度信息，从而恢复物体三维结构，被动式三维重建则通过利用图像的信息恢复出物体三维结构。被动式三维重建主要采用多张图像之间的相同物体的视差估计出不具备真实尺度的深度信息，其主要的算法为从运动恢复结构算法。

从运动恢复结构（Struct-from-motion）是一种利用图片中的颜色、纹理、形状等特征信息恢复出图片空间位置和朝向的一种技术。它能够将一系列无序的图

片转为具有确定位姿的三维空间点云。它的大致流程如下：首先利用图像的纹理信息提取特征点，特征点是一种在图像视角发生巨大变化时仍然保持不变的点，能够在图像匹配过程中发挥巨大的作用。提取到特征点信息后，每两幅图像进行特征点匹配，计算出特征点之间的对应关系，多张图像之间的匹配关系连起来形成一个跟踪关系，每个跟踪关系实际上代表了一个三维空间中的点。第三步是选取其中两张图像作为初始图像，利用五点法求出两幅图像的相对的旋转平移关系，之后采用捆集调整进行优化，由于之前计算出了两幅图像的相对位姿关系，故可以将图像上对应的特征点对进行三角化，利用对极约束关系，计算出对应特征点对在三维空间中的坐标位置。第四步是从剩下的图像中选择和当前图像匹配关系最好的一幅图像，再重复第三步的方法计算出新增加图像的位姿，并添加新的三维点到空间中。最后是将图像上和之前已经使用的图像上的能匹配上的特征点进行三角化，完成重建算法后，再进行一次全局的光束平差法。

如上所述是一般的从运动恢复结构的算法流程。根据优化的方法的不同，从运动恢复结构可以分为如下几种方法，一是增量式重建，增量式重建就如同上述算法，从一系列图像中选择较好的初始化图像对作为起点，完成对应特征点的重建，之后不停的从剩下的图像中选择合适的图像，将其加入到重建流程中来，使用光束平差法优化三维位置。该种方法的初始图像对选择策略和之后的图像选择策略会很大的影响到整体的重建效果。

最早的大规模从网络数据集中重建物体开始于 Pollefeys 等人^[9]提出的图像自标定算法，利用曲线和平行线等外部信息恢复出相机的内参，这为之后的从互联网数据集重建提供了基础。

1.2.1 增量式从运动恢复结构

Snavely 等人^[10]提出了一个可以让人们交互的浏览大型无序的网络数据集的软件，他们的系统基于图像进行建模，自动计算出每张图像的三维位置，在他们的图像浏览器中，用户可以知道他们浏览图像相对于场景的位置。首先对于每张图像，提取 SIFT 特征点，通过最近点对特征点的描述子进行匹配，然后使用 RANSAC 估计出每个图像对的基础矩阵，在 RANSAC 优化的过程中，使用八点法计算出候选的基础矩阵。最后，根据恢复出来的基础矩阵排除错误的匹配结果，如果剩下的匹配数目太少，就将这个图像对抛弃。在找到每对图像的特征匹配关系后，形成一个轨迹，每个轨迹包含了各个图像中匹配上的特征点。如果在一个轨迹中，图像有超过 1 个图像点，则说明该特征点是错误匹配。

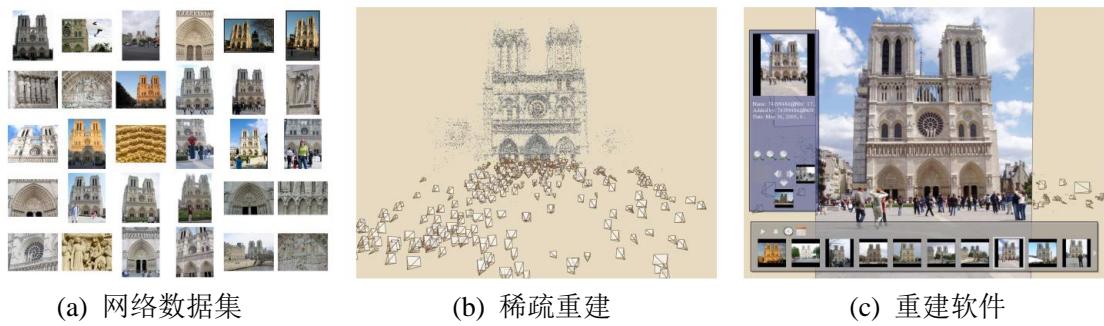
图 1-3 网络数据集重建软件^[10]

Figure 1-3. Network dataset reconstruction software

之后，通过最小化重投影误差的方法为每个轨迹恢复出一个三维空间点，最小化重投影误差是三维空间点到每幅图像的投影和图像上特征点的距离的最小和，可以通过 Levenberg-Marquardt 法求解。由于该方法只是找到了一个局部极小值点，故需要提供一个比较好的初始化参数。增量式求解的方法就是在求解的过程中，迭代的添加新的匹配图像对和特征点。为了避免退化的情况发生，初始的图像对选取策略为选择特征匹配数目多，基线长，并且特征点不是都在同一平面上。

接下来，选择新的图像添加到重建过程中，找到剩下图像中特征匹配数目最多的图像，通过直接线性变换（DLT）初始化外参。然后通过外参和图片的 EXIF 标记估计出相机的焦距。最后，将新图像中观察到的轨迹添加到优化过程中来，不断重复这一过程直到没有剩余的图像。为了最小化目标函数，他们使用了稀疏光束平差法来最小化。

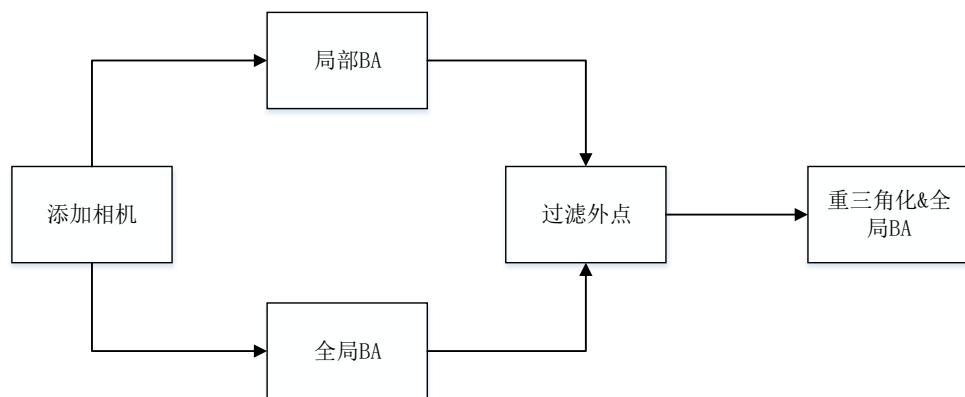


图 1-4 VisualSFM 流程图
Figure 1-4. VisualSFM flowchart

该增量式的方法在碰到大数目的图像时，运行速度会变慢，运行效率为 $O(n^4)$ ；另外对于复杂的没有足够纹理特征的场景，很难实现重建，难以有较好的效果。Wu^[11]提出了 VisualSFM，在光束平差法中引入了预条件共轭梯度法，将传统的 SfM 算法的运行速度提升到了 $O(n)$ 的效率，这为大尺度的场景重建提供了良好

的基础。针对大场景下的特征点匹配耗时长的问题, VisualSFM 采用抢占式的特征点匹配算法, 在减少了 95% 匹配数目的情况下, 仍然可以实现较好的重建效果。为了减少匹配的耗时, VisualSFM 使用少量的大尺度特征来进行匹配, 由于大尺度的特征能够覆盖更大的场景范围且能表示出场景的特点, 故能够和其他场景中大尺度的特征进行匹配, 另外在由于尺度金字塔上, 大尺度的特征往往能够提取的数目较少, 所有匹配特征所需要花费的时间也较少。

由于增量式的从运动恢复结构算法是对相机姿态进行累计估计的, 所以存在漂移误差的问题。另外由于初始的位姿估计不准确的原因, 有些正确的匹配可能因此无法匹配上。为了解决上述问题, 他们引入了重三角化步骤, 在增量式 SfM 过程中定期对无法三角化的匹配上的特征对进行重新三角化, 之后再用全局光束平差法和点云过滤来移除错误的点。实验证明, 他们的算法可以在 4 千张带回环的图像中, 减少漂移带来的问题。

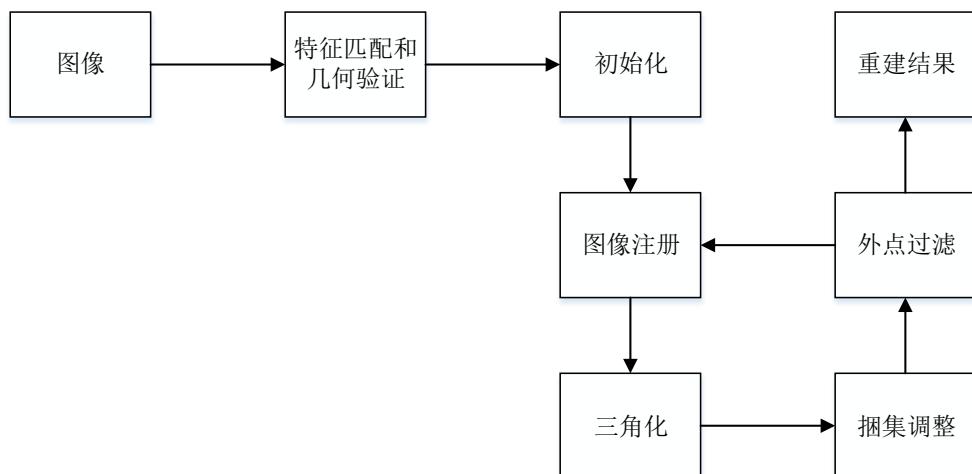


图 1-5 COLMAP 流程图
Figure 1-5. COLMAP flowchart

一般来说, SfM 过程中产生错误匹配或者漂移会导致图像无法正确的匹配从而生成错误的三角化点云。为了提高重建过程中的准确性和完整性, Johannes^[12]等人提出了 COLMAP, 他们引入了多模型的几何验证策略来增强场景图的效果, 提升初始化和三角化的鲁棒性。多模型的几何验证策略是采用计算基础矩阵、单应性矩阵和本质矩阵来判断图像上的场景是一般的平移旋转、纯旋转还是平面, 在初始化的时候, 选择非纯旋转非全景图的通过几何验证的初始图像对, 从而提高后续图像注册的准确性和鲁棒性。

其次, 通过提出了一个下一帧的选择策略。通过迭代进行光束平差法、重三角化和离群点滤除提高系统的完整性和正确性。计算图像位姿 PNP 问题的准确性依赖于图像上的点的数目和他们的分布, 所以下一个图像的选择是可以和当前地图点匹配数目多并且点分布均匀的图像, 通过计算多尺度下特征点分布的均匀程度和特征点数目的综合分数来确定下一个视图的选择。

三角化的过程中，为了减少对极约束的误差，使用 RANSAC 来排除错误的匹配。好的三角化需要满足如下约束：足够大的三角化角度，深度值为正以及重投影误差足够小。为了消除累计误差，在图像注册和三角化之后执行光束平差法，并滤除重投影误差较大的三维点，与 VisualSFM 类似，也执行重新三角化来消除漂移误差，并且在 BA 的前后都执行重三角化步骤，解决之前图像位姿计算不准确导致的无法形成轨迹的问题。

针对光束平差法耗时长的问题，COLMAP 将空间位置近似的图像划分为一个小组，每个小组代表了一个虚拟的相机位置，在优化的时候，对整个小组的位置来进行优化，而到实际的相机位置的时候，通过小组内的位置转化关系来计算实际的重投影误差。这对于重投影误差来说影响较小，但却明显的提升了整体计算优化的速度。

为了在一天内建成一个城市大小的模型，Agarwal 等人^[13]将重建过程并行化，设计了一个分布式并行重建系统，实现了用 500 个计算核心和 15 万张图片在一天内重建出罗马，同时他们还比较了不同的算法在并行条件下的表现。他们在图像匹配阶段先使用词汇树找到最相近的匹配图像，然后再对这些图像进行详细的特征匹配，这可以显著减少特征匹配的数目。在光束平差法中，他们根据问题的规模，选择截断或者精确步进的 LM 算法，系统的底层实现则采用了分布式计算引擎。

通常来说，SFM 的流程都采用固定的阈值来估计模型，Moulon 等人^[14]则采用自适应阈值来估计每个模型，不同的数据集中的噪声都不相同，采取自适应的阈值可以使其达到更加高的精度。在计算单应性矩阵和基础矩阵的时候，他们将 RANSAC 替换为 AC-RANSAC(自适应阈值的 RANSAC)，自适应阈值可以更好的区分内点和外点，并且优先选择符合基础矩阵的匹配特征点，这为相机位姿的计算提供了更好更稳定的匹配估计。对于不同的视图也会有不同的置信度，每个产生更大投影误差的三角化点也会被丢弃。随着加入相机数目的变化，自适应位姿估计的阈值也在不停的变化。

增量式的从结构到运动算法普遍会遇到估计位姿累计漂移的问题和大场景下的计算时间较长的问题^[15]，上述算法在一定程度上解决了这些问题。

1.2.2 全局式从运动恢复结构

前面的工作主要都是基于序列化增量式从运动恢复结构，从两张图的重建开始，添加一张张图像，最后完成完整的重建结果，但是增量式方法会存在累计漂移以及无法闭环的问题，以及当初始图像选择不好时，重建质量也会存在问题。为了解决上述问题，人们提出了全局式的从运动恢复结构，大多数的流程如下：首先计算每张图的全局旋转，然后计算其平移和场景的结构。对于较小的基线来说，这样可以计算出较为准确的旋转。全局的方法将所有的视图都放入到优化中来，使得全局的误差最小，能最大的降低增量式方法漂移的误差。但在全局式方

法中，如果场景中存在较多的误匹配或者存在重复的场景时，最终会容易产生错误模型。另外，由于全局式方法是基于结构和重投影误差，故需要的运行时间普遍会比较长。

Enqvist 等人^[16]提出一个基于旋转一致性和凸优化的非连续方法，该方法对于短基线、场景中的移动对象和回环的情况能生成更加鲁邦的结果。一般的错误匹配都会被 RANSAC 排除在外，但场景中的移动物体仍会生成大量的匹配点对，这会和全局的场景运动相冲突。如下图的拖拉机是同一个物体，但在不同的场景下，它们仍可以找到对应的匹配关系。

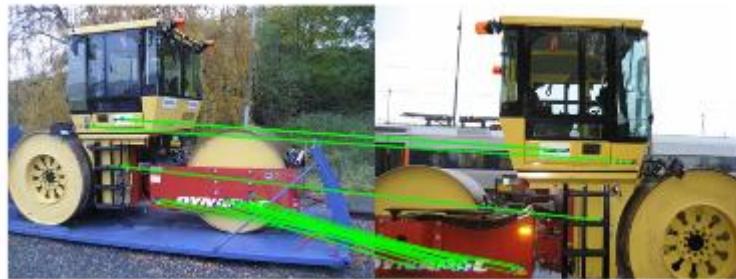


图 1-6 不同场景下的动态物体误匹配^[16]

Figure 1-6. Mismatching of dynamic objects in different scenes

该方法的主要贡献是提高了异常值的处理能力，对于基线较短的情况，虽然它的极线几何估计是不准确的，但旋转的计算结果是可靠的。其次是提出一个基于环和一致性的旋转估计模型。其算法的主要流程如下：1. 使用 SIFT 特征，并匹配图像 2. 估计每两幅图像之间的相对旋转关系，使用 RANSAC 和光束平差法来求解。3. 检测和移除旋转中的较大误差 4. 使用剩下来的相对旋转来估计相机的朝向。5. 根据上一步的结果进行重建。6. 使用光束平差法来提升全局重建的效果。

Venu 等人^[17]提出了线性估计和直接估计方法来全局估计相机的运动，由于 N 张图像有最多 $N*(N-1)$ 个相对位置关系，在对旋转平移误差进行建模的情况下，构建高度冗余的方程组，利用最小二乘法对参数进行求解。之后他们又提出了通过李群来表示运动，并线性的求解所有可能的相对运动^[18]。

Mica 等人^[19]提出了将求解相机旋转问题转化为齐次线性方程组，然后使用特征值分解和半定规划来恢复相机的旋转，并给出了从图像特征点匹配关系直接恢复相机位置的推导公式。

Moulon 等人^[20]提升了现有计算全局旋转方法的有效性，提出了一个有效的三焦张量估计方法，可以从中提取出稳定精确的平移估计。为了降低整体的计算时间，他们先在局部坐标系上计算场景结构，然后再将多个小场景合并为全局一致的大场景，由此他们实现了更高的精度和更短的计算速度。

Jiang 等人^[21]提出了一个鲁棒的线性的相机位姿计算方法，首先计算相机的

旋转，然后通过最小化相机中心和基线之间的几何距离来求相机的平移。这种方法能产生更鲁邦的结果，并且不会产生退化的结果。在恢复相机的姿态后，就可以据此估计出场景点的位置。

全局式的从运动恢复结构存在两个主要问题，由于本质矩阵只能知道相对平移的方向所以对平移做平均比较困难，另一个是当特征匹配失败时，计算出的本质矩阵存在误差。Cui 等人^[22]在每张图像上计算稀疏深度图，深度图可以将其转化为相似变换，并计算出相对平移的尺度，同时，深度图也可以用来过滤本质矩阵，从而使得上述两个优化问题得到稳定的解。

Cui 等人^[23]提出了一个相机看到相同场景点的线性位置约束，该方法可以处理共线和弱相机关系，并基于 L1 范式求解线性方程。他们是将场景点到其对应视线的欧几里德距离作为最小化目标函数，并且这对于最终的非线性光束平差法是一个很好初始化，另外，由于不是重投影误差，其存在直接线性解，减少了计算时间。

全局式的从运动恢复结构一般都会存在对图像特征点匹配出现错误时鲁棒性差的问题，Wilson^[24]等人将求解旋转的问题简化为一维子问题以删除匹配上的异常值。和其他的将不准确的场景图作为输入不同，Sweeney^[25]等人通过回环约束来将有噪声的视图恢复成具有一致性的视图，使得任意的 SFM 算法都可以提升效果。

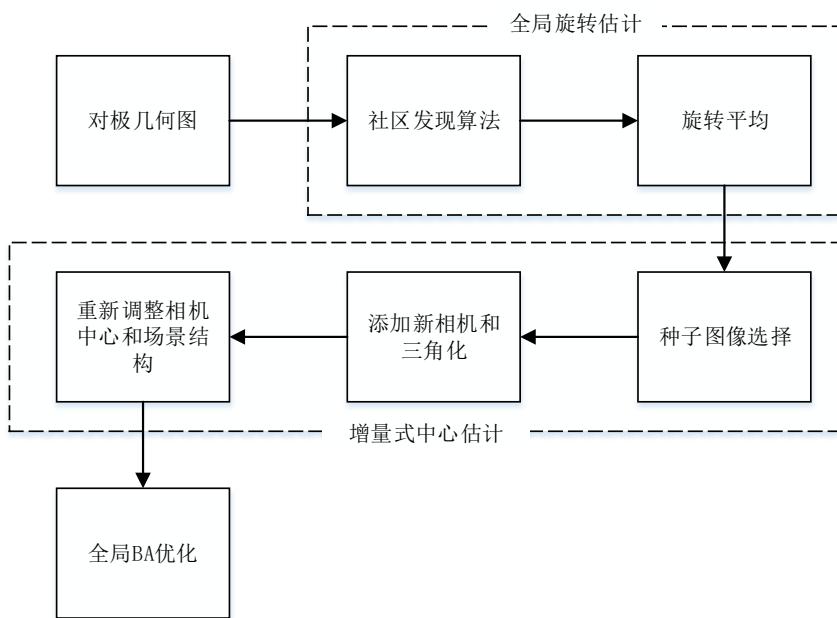


图 1-7 混合式重建流程图

Figure 1-7. Mixed reconstruction flowchart

1.2.3 混合式从运动恢复结构

从运动恢复结构按照计算相机位姿方法的不同有增量式和全局式，增量式的方法其鲁棒性和准确性都不错，但由于需要大量的捆集调整，其运行时间普遍较

长，全局式方法则是将所有的相机位姿同时考虑，根据极线几何约束来进行统一优化，但一般对外点都比较敏感。Cui 等人^[26]提出了一个自适应的旋转平均方法来估计相机的全局旋转，之后用增量式的方法计算相机的中心。在计算效率方面比其他的要快，同时也可以达到和其他方法相似的准确度和鲁棒性。

如图 1-7，系统的输入是几何场景图，由于全局的平均旋转求解对场景图的结构和匹配的准确性比较敏感，他们使用了基于复杂网络分析的自动分组算法，对每个组进行旋转平均，然后将他们融合到统一的坐标系中。在得到每个相机的旋转后，以增量式的方法估计相机的中心，并对场景进行重建。在选择添加新的相机的时候，寻找特征匹配数目更多、基线更长和相机位姿准确的新视图添加到重建过程中来。然后基于场景中的对应关系和图像间的特征跟踪来迭代的估计相机的中心，并且最后通过捆集调整来优化整个模型的重建效果。

为了将 SFM 应用在大型城市数据集上，Zhu 等人^[27]通过将重叠的相机进行聚类把大型的 SFM 转化为多个更小的子集来求解，并提出了一个混合公式将局部的增量式重建过程融合到全局的运动平均框架中去，实现了重建整个几百万图像的数据集。由于单个计算机内存的限制，他们将每个集群的大小限制为 100。相机聚类算法分为两个阶段，首先根据相机集群的大小限制来对相机进行划分，超过的分成两个平衡的子图，其次利用相机间的完整性约束对子图进行扩展。在生产跟踪阶段，用层次式相机聚合树来避免一次性载入所有的特征和匹配结果。对于每个相机聚簇，都并行的执行局部增量式的重建，然后通过基于 RANSAC 的过滤器和重复局部捆集调整移除错误匹配关系，局部的增量式重建采用了鲁棒的 N 视图位姿估计方法，这相对于基本矩阵和三焦张量的方法可以产生更加准确和鲁邦的结果。在完成平均平移算法和三角化后，再进行全局的捆集调整优化建模效果。

1.2.4 层次式从运动恢复结构

增量式的重建主要有如下几个问题：初始化视图的选择会影响到最终模型的重建质量，每一帧之间的位姿估计都存在误差导致的漂移问题，捆集调整来调整整体模型的效果导致的重建时间长的问题。为了解决这些问题，Farenzena^[28]等人提出了层次式的重建方法，他们根据图像间的重叠程度，以图像为叶子节点，构成一颗自下而上的聚簇树，重建过程将从叶子节点开始，自下而上的进行。由于重建过程被分解为一个个更小的问题，相比于增量式的重建方法，其运行速度也会更快。由于其重建的过程和初始视图的选择无关，其鲁棒性也更强。

图像间的聚类根据两张图像之间的特征匹配数目和匹配特征点覆盖的区域面积来决定。图像之间通过层次来进行聚类，从最底层的叶子节点开始，每一次合并两个距离最小的聚簇，两个聚簇间的距离由聚簇间最小的距离决定。在选择图像的时候，也需同时满足如下两个要求：有足够的共同特征点，基线足够长。为了降低捆集调整的时间，他们选择每个聚簇中 K 个视图——活动视图，而不

是调整全部的视图，选择的 K 个视图是新加入的、单张图像或者来自于另一个聚簇。锚定视图则是和活动视图距离最近的 K 个视图，它们不会再捆集调整中被优化，只是用于计算三维点的重投影误差。虽然整个过程的时间复杂度没有降低，但需要参与计算的图像数目降低了，同时整个过程可以并行化执行。

为了进一步提高运行效率，Gherardi 等人^[29]提出了一个新的聚类策略可以使整个树更加平衡，将聚簇中所有距离的和作为该聚簇的度量，故使得总体的生成树的高度更加小，降低了算法的时间复杂度。

Toldo^[30]等人也采取了采用分层的方法来完成重建流程，并且提出了一个在没有辅助信息的情况下自动标定图像的方法。

1.2.5 重网格化研究现状

三维网格模型一直都是各个领域内的重要显示工具，它可以被广泛的应用在各种场所，例如历史文化遗迹的展示、工业生产现场、游戏娱乐产业等。随着计算机视觉技术的发展，越来越多的三维建模技术得到长足的进步，对应的商业化应用也得到了人们的重视。但单纯从计算机视觉算法中获取到的三维模型往往没办法直接应用在商业化的应用中，获取到的模型一般都会存在过多的杂乱的面，其模型体积也会格外的庞大，不适用于网络的快速传输，故需要对其进行平滑、参数化、重网格化、简化、修补等操作，使其成为更完善的模型。

其中重网格化技术是将生成和原模型相似的，并且其连接性具有规律的一种技术。重网格化技术能用于许多不同的领域，例如模型压缩，表面平滑化，模型编辑和规整化等。

参数化方法可以分为全局参数化和局部参数化。全局参数化方法是把所有的顶点都考虑在内，找到一个全局最优的解，使得模型的代价函数最小。

2003 年，Alliez^[31]等人提出了一种交互式的重网格方法，他们首先将输入网格进行参数化，生成多个二维的参数地图，并计算几何质量；然后对参数地图进行分析过滤，生成控制地图，控制地图是用来决定重采样过程中的采样频率的；之后根据控制地图的采样频率来进行采样，三角化，生成一个新的三维网格模型。他们的算法可以生成多种不同类型的网格模型，均匀网格模型、规则四边形网格模型、半规则三角形模型和自适应曲率模型。在生成三维网格模型的过程中，他们为用户提供了交互式的方式，可以控制不同区域的采样频率。

2004 年，Alliez 等人^[32]提出了通过跟踪主曲率线的方法对网格进行重构，对于输入的三角网格，他们先构建曲率张量域，对它进行平滑操作，在网格上找出脐点位置；在网格曲面上沿着主曲率方向进行跟踪，局部密度由主曲率方向来确定，在脐点的周围用点采样的方式来确定；最终的网格通过对主曲率线的降采样来生成。最终生成一个各向异性的网格，同时具备三角形和四边形网格，四边形网格存在于主曲率线附近，三角形网格则位于脐点周围，且是各向同性的。这样生成的模型更符合原模型的特征。

之后，Marinov 等人^[33]改进了上面的方法，他们不再依赖于全局的参数化方法，可以应用于任意亏格的网格模型中。同样的，他们首先在原来的模型上计算连续的线性曲率张量域，然后建立一个遵循主曲率方向的曲线网络，并进行采样，他们定义了局部参数化方法，并跟踪最大曲率线、最小曲率线、边界和特征线；在不同的网格区域中，由用户指定的近似估计和曲率共同决定该区域的局部密度；对于各向同性的平坦区域，他们通过计算置信度来可靠的集成主曲率线，这样可以生成更稳定的结果。

针对任意拓扑结构的网络，Khodakovsky 等人^[34]提出了一个低失真的全局平滑参数化方法，根据形状质量和参数畸变来优化面片的布局结构。参数化的过程分为两个步骤，首先将原始的网格分割为面片，然后对于每一个面片，计算到参数域的映射函数，最后在参数域中对曲面进行采样生成新的网格模型。

KaLberer 等人^[35]提出了一个可以在任意 2 维流形上自动计算全局参数化的方法，其参数线可以通过给定的场来引导，例如主曲率场。参数线是全局连续的，并可以将其转化为四边形网格模型。该算法可以将给定的场转化为单向量场，然后通过霍奇分解生成一个可积的向量场。

Ray 等人^[36]为任意拓扑的三角形网格模型提出了一种全局参数化方法，给定输入网格上的两个正交的分段线性向量场，通过最小化目标函数来和输入向量场对齐。他们的方法既可以生成保角参数，也可以生成保面积参数，如果引入更多的奇异点，也可以生成等距参数。

Knöppel^[37]等人通过求解全局最优方向场来生成网格模型。针对线性参数系统表现不好的问题，Aksoylu^[38]等人使用多层预处理器来简化网格，构建多层次网格结构。

对于模型规整化中的畸变问题，Levi^[39]等人提出了一种新的模型畸变的度量，通过优化这一度量能够减小模型的畸变。全局参数化方法的优点是模型畸变小模型质量高，但通常所需的计算时间也会比较长。

局部方法则是直接改变网格的拓扑结构，如边的合并、分割和交换操作。如 Kobbelt^[40]等人通过模拟收缩包装过程来逼近输入网格模型，他们将顶点均匀的分布在曲面上，然后通过投影的方式来模拟收缩的过程。局部参数化方法计算速度快，但有可能造成模型特征缺失、奇异点多的情况。

参数化是将纹理映射到模型表面，它通常包含两个步骤：最小化畸变和分割优化。曲线的畸变代表着表面映射的质量。Khodakovsky^[41]等人提出了一个低畸变的全局平滑参数化方法，在块与块之间用了一个显式的转化函数来保证全局的平滑。Liu^[42]等人同时使用了局部和全局的参数化方法，用局部操作保证畸变的最小值，然后把所有的三角形合并成为一个控制地图。Poranne^[43]等人将分割和畸变计算联合优化，使得整个计算过程更快。

在以重网格化为目的的参数化中，网格的分割操作的重要性要低许多，更多

考虑的是奇异点的放置位置，它代表了每个块中的起止点，奇异点的位置越接近于人们期待的位置，模型的质量通常会越好。Ben-Chen^[44]等人提出了一种自动计算奇异点位置的方法，然后将网格沿着曲率方向同时过奇异点进行分割。Campen^[45]等人根据主方向连续地优化顶点位置，在网格上自动生成四边形网格。

以上的这些方法是基于参数化的方法，他们都是将网格模型映射到二维的参数域上，然后对二维的参数域进行重新参数化，取其整数坐标作为网格的顶点，最后重新将二维平面上的映射回到原来的模型上，进行重采样生成最终的重网格化结果。基于参数化的方法在碰到亏格大于 0 的网格模型时，需要对模型进行分割和拼接，容易出现扭曲的问题。

除了参数化方法以外，还可以通过构建方向场的形式来生成四边形网格。

Ray 等人^[46]设计了一个 N 对称的方向场，他们通过引入转向数的概念，对 N 对称方向场的拓扑结构进行全面的研究，并给出了构建 N 对称方向场的算法，在用户定义了一系列的奇异点后，他们的算法可以构造出满足所有约束的平滑方向场，并且不会生成其他的奇异点。Jakob^[47]等人对方向场进行旋转，使得旋转代价最小，然后根据旋转后的方向确定规整网格边的方向。

Bommes 等人^[48]提出了新的三角形网格转化为四边形网格的方法，在构建平滑的对称交叉场并满足稀疏的方向约束之后，网格被低失真地展开。然后计算一个无缝的全局平滑参数，其等参线和交叉场的方向一致。和其他的方法相比，该方法可以自动确定四边形中的奇异点的位置数目和类型。交叉场和参数化的过程都被当做混合整数的问题来求解，并且用自适应的贪婪算法来有效的求解，最终生成的网格有较好的质量。

1.2.6 环境光遮蔽研究现状

环境光遮蔽是一种用来描述场景中物体之间漫反射光线遮挡关系的渲染技术。在计算机图形学中，传统方法采用光线追踪的方法，每一个像素的环境光是通过追踪穿过该像素的每个光线得到的，并且每个光线需要与场景中的物体测试相交性，这需要大量的计算资源，很难达到实时计算的效果。2008 年，Mittring^[49]提出了一种实时的环境光遮蔽计算方法——屏幕空间环境光遮蔽，它使用了深度和位置信息来估计遮蔽，将三维空间中的计算转化到屏幕二维空间，大大地提高了计算效率。Bavoil^[50]等人提出的 HBAO 则不需要依赖于场景的预算计算，使之能够用于动态场景下的计算。Holden^[51]等人提出的 NNAO 技术使用了机器学习的方法估计场景的遮蔽值，他们根据 SSAO 的模型训练了一个四层的神经网络，训练后的实验表明计算速度比 SSAO 和 HBAO 更快，并且有较高的精确度。在本文，我们使用了 NNAO 来估计重拓扑模型的环境光遮蔽。

1.3 研究内容

本文的主要研究内容是如何进一步优化增量式三维重建，以及对重建结果的网格进行重拓扑和调整光照。在常见的三维物体重建过程中，一般需要围绕目标物体进行全方面多角度的拍摄，然后采用基于多视图的三维重建算法对拍摄到的图像进行处理，利用几何约束和特征匹配关系从图像中找到对应的特征匹配关系从而恢复出物体和相机之间的空间坐标关系，在这一过程中，错误的特征匹配和部分图像较少的匹配数目都容易导致最终生成模型的不完整。

在完成对物体的三维重建算法后，生成的网格模型大多比较杂乱无章，同时为了实现更加高精度的三维重建结果，三维重建的网格也普遍具有更多的顶点和网格数目，这限制了重建后网格的应用，重建拍摄过程中的光照也会对此有一定影响。

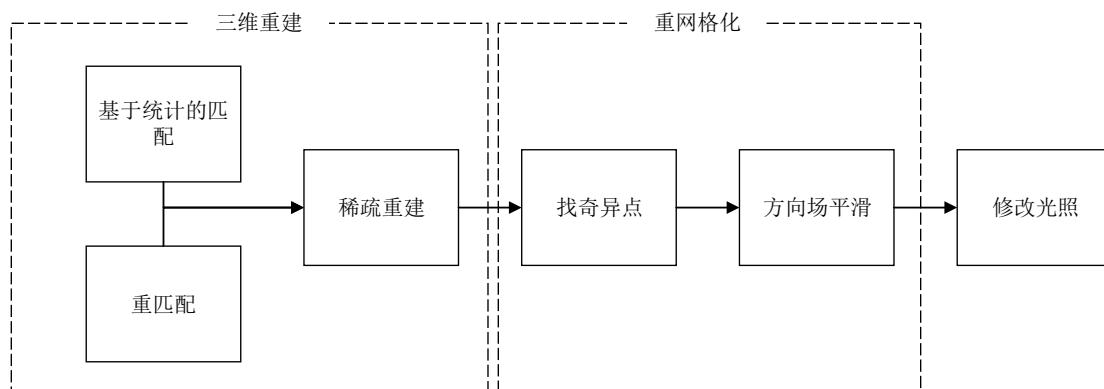


图 1-8 总体框架流程图
Figure 1-8. Framework flowchart

为了解决上述问题，本文开展了如下的研究工作：

(1) 基于特征点匹配优化的三维重建算法。数据预处理，确定前景，剔除背景，确保对单个物体的重建结果不包含背景元素。结合基于网格统计的特征点匹配算法和重匹配算法，对特征匹配结果进行过滤验证，并对其中的匹配失败结果进行重新匹配，寻找候选匹配队列中的次优结果，并对其进行验证，优化特征匹配结果，生成更加完整的三维重建模型。

(2) 基于神经网络的重网格化和光照调整。利用神经网络寻找网格中的奇异点位置。根据指定的奇异点位置，构造光滑能量项，对模型的四旋转方向场进行平滑，生成半规则网格。利用环境光遮蔽技术，去除模型自带的环境光，使其可以应用在不同的场景中。

1.4 本文的主要工作及组织结构

本文主要有五章内容，其主要内容如下：

第一章 绪论。本章主要介绍了本文的研究意义和背景，阐述了三维重建技术和重网格化技术在业界的重要意义。概述了三维重建技术在各个领域内的应用和研究意义。之后，简单的介绍了从运动恢复结构技术、重网格技术和环境光遮蔽在国内外的研究现状。最后简述了本文的主要研究内容和本文的组织架构。

第二章 三维重建和重网格化的理论基础。本章首先介绍了在计算机视觉中二维图像坐标系、三维相机坐标系和世界坐标系之间的坐标转化关系，推导出了从世界坐标系到图像中坐标的转化关系的公式。之后介绍了相机的畸变模型，纠正图像在拍摄中的出现的弯曲现象。然后介绍了 SIFT 特征点提取算法、对极几何和三角化技术，完成从图像特征点到最终计算出三维空间点坐标的计算推导。在重网格化方面，依次介绍了曲面的表示方式、多边形网格的存储方式和重网格化的基本概念。

第三章 基于匹配优化的从运动恢复结构。第三章首先简要的介绍了本算法的大致流程，然后分步介绍了图像预处理、基于网格统计的匹配算法和重匹配算法、基于匹配优化的增量式 SFM 算法，之后对本算法进行对比实验，验证本算法的精度和有效性。

第四章 基于机器学习的网格重拓扑和光照修改。本章首先阐述了本算法的应用背景和算法的整体流程。然后按照算法的流程，首先介绍了通过机器学习的方法寻找网格中的奇异点，然后介绍了构建平滑能量项来优化网格位置域和方向域，生成半规则网格的流程，之后介绍了利用阴影遮蔽算法调整模型阴影的过程。最后和其他重网格化算法进行对比实验。

第五章 总结和展望。本章首先对本文的工作进行全面的总结，并分析了其优缺点，并对未来的工作内容作了展望。

第二章 三维重建理论基础

2.1 相机与图像

本节主要介绍计算机视觉在二维图像坐标系和三维空间坐标系中的理论基础知识，包括了图像坐标系，三维空间坐标系以及相机针孔模型。图像是光线经过物体反射通过相机镜头映射在底片上形成的图案，而在这一过程中，光线会发生折射而造成图像发生形变，所以需要对图像进行去畸变工作。在计算机视觉中，也需要计算从图像中某一个特征点到三维空间坐标系中的空间点的转化关系，为了更好的理解这部分内容，下面需要对这一部分知识进行介绍。

2.1.1 图像像素坐标系

在二维图像中，以图像左上角为原点，其水平向右横向坐标系为 X 轴，垂直于横向坐标系向下的是纵向坐标系为 Y 轴，每一个像素是一个单位的偏移量。为了在计算机中显示彩色图像，将彩色图像分为 RGB 三通道，即由红绿蓝三原色组成，每一个通道的亮度取值各不相同即可显示为不同的颜色，其取值范围一般是 0 到 255，值越高代表该点的亮度越高。在 OpenCV 中，如果需要获取图像中某一个点的颜色值，可以按照如上图像坐标系的规定取值。

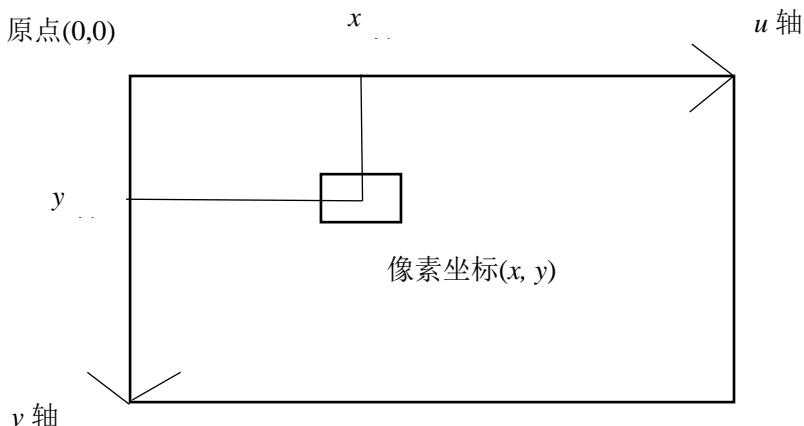


图 2-1 图像像素坐标系
Figure 2-1. Image pixel coordinate

2.1.2 像平面坐标系

像平面坐标系是相机的实际成像的坐标系，它和代表了实际成像的位置。它的坐标系和图像像素坐标系相差了一个平移和一个缩放尺度， X 轴和 u 轴平行， Y 轴则和 v 轴平行，图像的光心是图像的中心，也是像平面坐标系的坐标原点，

其单位为毫米。

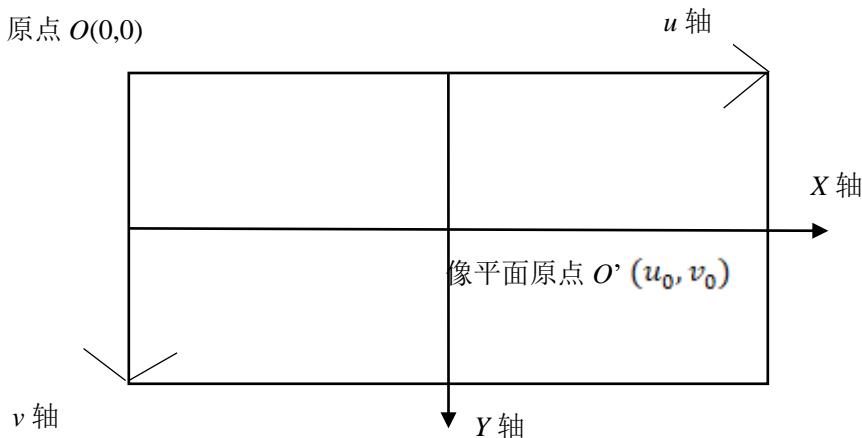


图 2-2 像平面坐标系

Figure 2-2. Image planar coordinate

假设图像中每个像素在 X 轴上的实际尺寸是 dx 毫米, 像素在 Y 轴上的实际尺寸是 dy 毫米, (u_0, v_0) 是图像光心在图像像素坐标系下的坐标, 那么我们可以得到像平面坐标系和图像像素坐标系之间的转化关系:

$$\begin{aligned} u &= \frac{x}{dx} + u_0 \\ v &= \frac{y}{dy} + v_0 \end{aligned} \quad (2-1)$$

如果用矩阵来表示, 则表示为下式的形式:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2-2)$$

2.1.3 相机坐标系

相机坐标系是以相机的光心为原点, 相机坐标系的 X_c 轴与像平面坐标系的 X 轴平行, Y_c 轴与像平面坐标系的 Y 轴平行, 坐标系是以米为单位。相机坐标系是三维空间下的坐标系, 其 Z_c 轴是与 X_c 轴和 Y_c 轴形成的平面垂直的朝向相机外部的方向, 其与光轴相重合。

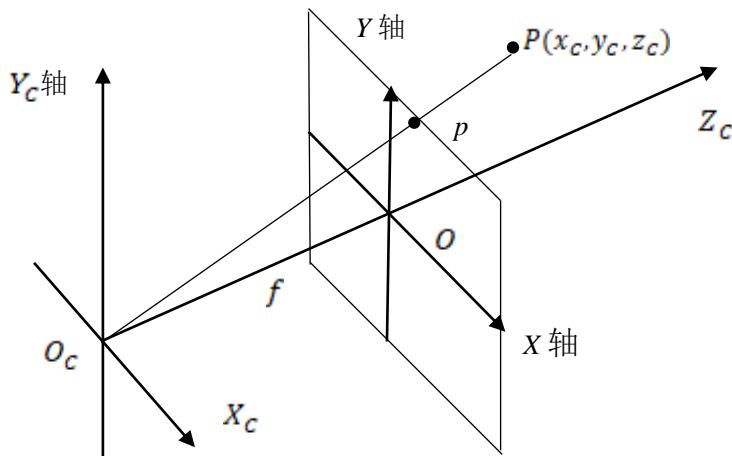


图 2-3 相机坐标系
Figure 2-3. Camera coordinate

从相机坐标系到像平面坐标系的转化，我们可以用一个小孔成像模型来模拟，假设 $P(x_c, y_c, z_c)$ 是三维空间中的一个点，它通过相机光心 O_c 映射到相机的成像平面 $X - O - Y$ 于点 $p(x, y, z)$ 。在图 2-3 中为了更好的显示，我们将成像平面显示到了光心的前面，镜像地显示该平面。成像平面到 $X - O - Y$ 相机坐标系的 $X_c - O_c - Y_c$ 平面的距离我们用焦距 f 来表示。

根据三角形的相似定理，我们可以得到如下公式：

$$\begin{cases} \frac{x}{f} = \frac{X_c}{Z_c} \\ \frac{y}{f} = \frac{Y_c}{Z_c} \end{cases} \quad (2-3)$$

如果用矩阵的形式来表示的话，可以得到如下：

$$Z_c \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (2-4)$$

2.1.4 世界坐标系

世界坐标系是一般是在整个项目中保持绝对不动的坐标系，和相机坐标系之间存在一个刚体变换的关系。在欧式空间中，大小形状都保持不变的物体本文将其称为刚体，而刚体变换是指刚体从一个位姿到另外一个位姿的变换，刚体变换是由旋转和平移组成。

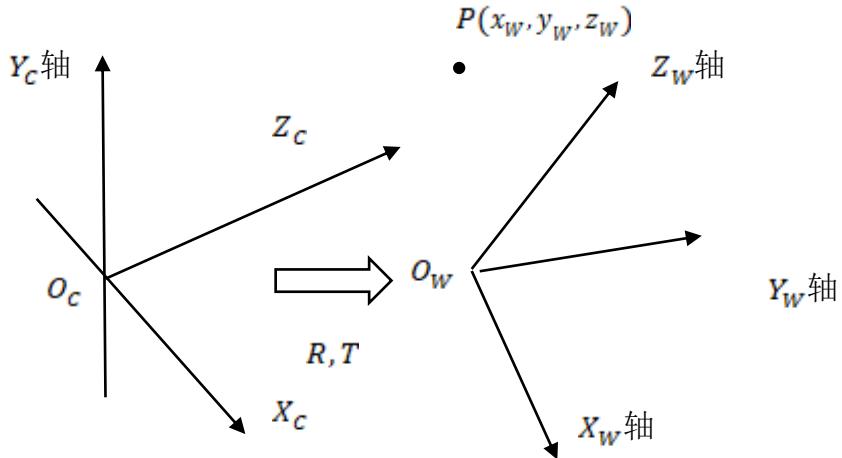


图 2-4 世界坐标系与相机坐标系

Figure 2-4. World coordinate system and camera coordinate

在世界坐标系下有一点 $P(x_w, y_w, z_w)$ ，假设从世界坐标系到相机坐标系下的转化是旋转 R 和平移 T 组成，那么根据坐标系之间的转化关系，我们可以得到点 P 在相机坐标系下的表示为

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2-5)$$

其中旋转矩阵 R 是一个 $3*3$ 的正交矩阵，平移矩阵 T 是 $3*1$ 的矩阵。

综合上述公式，我们可以推导出从世界坐标系到相机像素坐标系的转化关系，如下所示：

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (2-6)$$

2.2 图像畸变矫正

在相机中，我们一般会用凸透镜将光线汇聚到感光元件上，然而由于透镜本身的固有的特性和组装工艺的问题，光线透过在凸透镜的边缘使得直线产生弯曲的现象叫做图像的畸变。在上面的针孔模型中，直线不会再通过光心的时候发生变化，故可以用上述的公式很好的描述图像的变化。在实际的应用中，相机本身的畸变是不可以忽视的，需要对相机的畸变进行矫正，恢复图像本身的状态。

图像的畸变主要有两种，径向畸变和切向畸变。

径向畸变也分为两种，桶形畸变和枕形畸变，它主要是由于镜头的不同位置对光的放大率不同造成的，中心点的畸变一般都是 0，即光线穿过光心不会发生弯曲，具体如下所示：

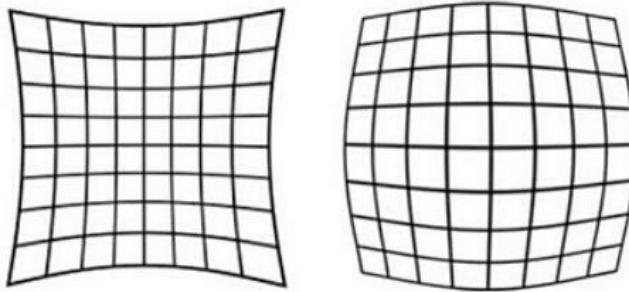


图 2-5 枕形畸变(左)和桶形畸变(右)^[52]

Figure 2-5. Pillow and bucket distortion

桶形畸变和枕形畸变我们可以通过泰勒级数的前三项展开式来模拟：

$$\begin{cases} x_0 = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y_0 = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{cases} \quad (2-7)$$

其中 (x_0, y_0) 是经过畸变矫正后点在图像上的位置， (x, y) 是原始图像中点的位置。

切向畸变则是由镜头和相机的感光元件在组装过程中安装倾斜造成的，镜头与相机的成像平面不平行导致显示的物体是倾斜的。

切向畸变可以通过如下的公式来纠正：

$$\begin{aligned} x_0 &= x + [2p_1y + p_2(r^2 + 2x^2)] \\ y_0 &= y + [2p_2x + p_1(r^2 + 2y^2)] \end{aligned} \quad (2-8)$$

其中 p_1 和 p_2 是切向畸变的畸变参数。

当需要同时纠正两种畸变，则可以通过同时结合上述两个公式来完成：

$$\begin{cases} x_0 = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [2p_1y + p_2(r^2 + 2x^2)] \\ y_0 = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + [2p_2x + p_1(r^2 + 2y^2)] \end{cases} \quad (2-9)$$

2.3 特征点提取

在计算机视觉中，特征点一般都是图像上梯度变化较大的点或者是图像中物体的边缘，通过图像中特征点的匹配，能够对物体进行识别或对场景的位置进行定位。

在视觉三维重建或者同时定位与建图(Simultaneous localization and mapping，即 SLAM)中，特征点是其中很重要的一部分，为了保证在不同的视角下都能够匹配到同一个特征点，设计的特征点提取方法需要满足旋转不变性和尺度不变

性。旋转不变性是指在特征点发生旋转后仍能够识别和匹配上，尺度不变性则是在和物体距离不同的情况下完成匹配。

其中比较经典的特征点提取方法有 SIFT、SURF、ORB 等，下面主要介绍下 SIFT 特征点。

SIFT(Scale-invariant feature transform)，即尺度不变特征变换^[53]，其主要特点是在不同尺度的图像金字塔中查找关键点，其提取出的特征点可以在光照、仿射变换、视角变换和噪音等因素的影响下仍然可以提供鲁邦的匹配效果。

算法主要分为以下几个步骤：

首先构建尺度空间，检测极值点。构建尺度空间的目的是为了模仿在不同距离下，物体成像大小的不同，使得在不同的尺度空间下仍然可以检出相同的特征点。

实验证明，高斯卷积核可以实现尺度变换，构建尺度金字塔，尺度空间定义为高斯卷积核和输入图像的卷积结果，其公式如下：

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2-10)$$

其中，高斯卷积核的定义如下：

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (2-11)$$

为了能够在尺度空间中稳定地检出特征点，SIFT 设计了高斯差分尺度空间，它是由两个相邻的尺度空间结果进行差分得到的：

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \quad (2-12)$$

在构建完尺度空间后，检测尺度空间极值点。极值点的定义是在某个点的周围 8 个点和上下相邻尺度空间中相邻的 9*2 个点中的最大值或最小值。

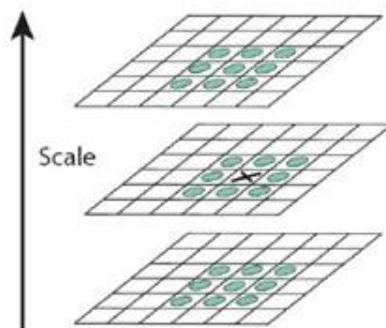


图 2-6 极值点检测^[53]

Figure 2-6. Extreme point detection

由于上述的结果会对噪声和边缘比较敏感，后续需要继续对极值点进行筛选，排除掉不稳定和错误的检测结果。

为了保证特征点的旋转不变性，SIFT 将梯度方向的最高值的方向作为主方向，辅方向是有主方向 80% 高度的地方。以关键点为中心的所有领域中的点都需

要计算梯度方向。

每一个关键点的梯度计算方式为：

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (2-13)$$

梯度方向的计算如下：

$$\theta(x, y) = \tan^{-1} \left[\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \right] \quad (2-14)$$

得到所有点的梯度方向之后，对所有梯度方向进行统计，并以每 10 度为单位来生成灰度直方图，SIFT 将灰度直方图中的梯度方向最多的方向设为主方向，达到主方向高度 80% 的第二高峰为辅方向。

生成关键点的描述子可以在后续的特征匹配中发挥重要作用，描述子包含了关键点和周围领域内点的信息。描述子是以关键点为中心，16*16 的窗口内，以 4*4 为一个小块计算 8 个方向上的梯度直方图，这样就形成了 128 维的描述子。再对这 128 维向量进行归一化操作后就可以排除光照对其的影响。特征匹配的过程一般都是以特征点描述子之间的欧氏距离来判定两个描述子之间是否可以配对上。

2.4 对极几何

对极几何是两个相机和空间点之间构成的特殊几何关系，通过特征匹配关系可以从中计算出两个相机的相对位置关系和它们朝向的变化。

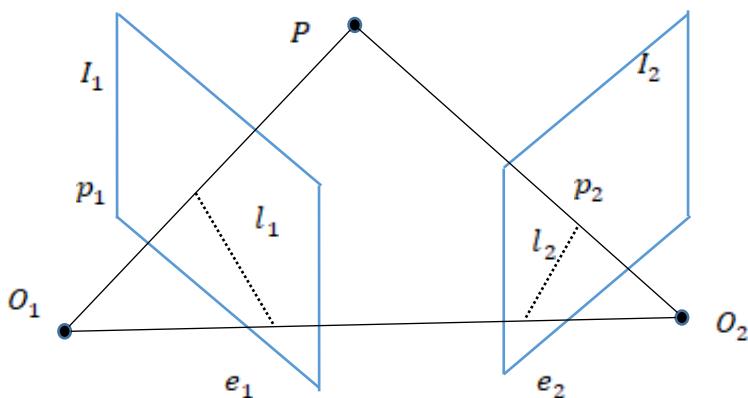


图 2-7 对极几何示意图
Figure 2-7. Diagram of polar geometry

在上图中， O_1 和 O_2 是图像 I_1 和 I_2 的相机中心，点 P 是空间中的一个点，点 P 在图像 I_1 和 I_2 的投影分别是 p_1 和 p_2 ，两个相机之间的连线 O_1O_2 称为基线，基线和图像 I_1 和 I_2 的交点是对极点 e_1 和 e_2 ，它们也是另外一个相机在对应平面的成像

点, PO_1O_2 构成的平面是对极平面, 点 p_1 、 p_2 、 e_3 、 e_4 也在该对极平面上, 对极平面与图像之间的交线称为对极线, 即直线 l_1 和 l_2 。

如上所述, 两个对应的关键点之间的约束关系称为对极约束, 对极约束指的是在不知道空间点 P 的位置的时候, 我们可以从几何关系确定点 P 在第二幅图像中的投影位置一定在对极线 l_2 上。在知道多对特征点之间的匹配关系后, 便可据此推算出两个相机之间的相对运动关系。如果用公式来表示对极约束, 则可以写为下式:

$$p_2^T K^{-T} t \wedge R K^{-1} p_1 = 0 \quad (2-15)$$

其中 p_1 、 p_2 是空间点 P 在图像 I_1 和 I_2 上的投影位置的像素坐标, K 是相机的内参矩阵, $t \wedge R$ 是两个相机之间的转化关系。从该公式中, 我们可以得到它的几何意义是点 P 、 O_1 、 O_2 三点共面。

若我们将 p_1 、 p_2 的坐标转化到归一化平面上, 则

$$\begin{aligned} x_1 &= K^{-1} p_1 \\ x_2 &= K^{-1} p_2 \end{aligned} \quad (2-16)$$

设 $E = t \wedge R$ 为本质矩阵, $F = K^{-T} E K^{-1}$ 为基础矩阵, 那么上式可以记为:

$$x_2^T E x_1 = p_2^T F p_1 = 0 \quad (2-17)$$

本质矩阵 E 和基础矩阵 F 之间只相差一个相机内参 K , 故我们一般求解本质矩阵 E 。本质矩阵一般采用八点法来求解, 通过多个点之间的匹配关系列出线性方程组, 设其中一个匹配点的坐标是 $p_1 = [x_1, y_1]^T$, 另一个点的坐标是 $p_2 = [x_2, y_2]^T$, 那么根据上面的公式, 可以得到下式

$$(x_1, y_1, 1) \begin{pmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = 0 \quad (2-18)$$

若我们将矩阵展开可以得到线性方程组的形式,

$$[x_1 x_2, x_1 y_2, x_1, y_1 x_2, y_1 y_2, y_1, x_2, y_2, 1] \bullet e = 0 \quad (2-19)$$

并将其他所有的点对都带入到上面的公式中, 那么可以得到如下的线性方程组:

$$\begin{pmatrix} x_1^1 x_2^1 & x_1^1 y_2^1 & x_1^1 & y_1^1 x_2^1 & y_1^1 y_2^1 & y_1^1 & x_2^1 & y_2^1 & 1 \\ x_1^2 x_2^2 & x_1^2 y_2^2 & x_1^2 & y_1^2 x_2^2 & y_1^2 y_2^2 & y_1^2 & x_2^2 & y_2^2 & 1 \\ \vdots & \vdots \\ x_1^8 x_2^8 & x_1^8 y_2^8 & x_1^8 & y_1^8 x_2^8 & y_1^8 y_2^8 & y_1^8 & x_2^8 & y_2^8 & 1 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \\ e_8 \\ e_9 \end{pmatrix} = 0 \quad (2-20)$$

在理想情况下，本质矩阵 E 是齐次矩阵，可以直接用线性算法求解。在现实的情况下，由于匹配精准度和噪声的影响，基础矩阵的秩不为 8，那么可以通过最小二乘法来求解。

从本质矩阵 E 恢复相机间的旋转 R 和平移 t ，可以通过奇异值分解的方法来求解，本质矩阵的 SVD 分解可以写为如下的形式：

$$E = U \Sigma V^T \quad (2-21)$$

我们将本质矩阵分解为了一个正交矩阵 U ，一个对角矩阵 $\Sigma=diag(\sigma, \sigma, 0)$ 和一个正交矩阵 V^T 的乘积。本质矩阵的分解可能存在四个解：

$$\begin{aligned} t_1^{\wedge} &= UR_Z\left(\frac{\pi}{2}\right) \Sigma U^T, R_1 = UR_Z^T\left(\frac{\pi}{2}\right) V^T \\ t_2^{\wedge} &= UR_Z\left(-\frac{\pi}{2}\right) \Sigma U^T, R_2 = UR_Z^T\left(-\frac{\pi}{2}\right) V^T \end{aligned} \quad (2-22)$$

判断正确的解只需要将解带入公式中，判断三维空间点的深度值是否都为正，只有点的深度值为正，矩阵的分解结果才是正确的。

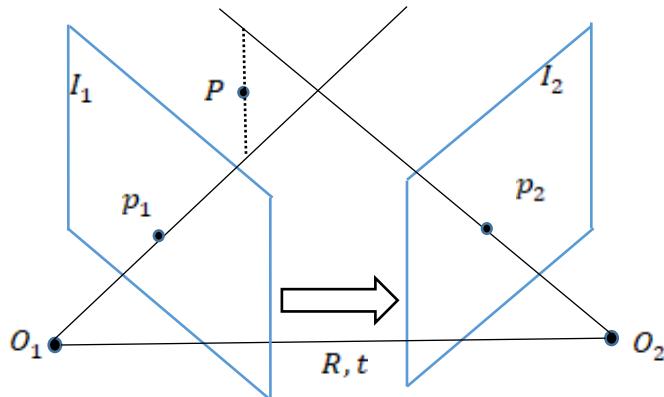


图 2-8 三角化示意图
Diagram 2-8. Triangulation

2.5 三角化

三角化是在已知两个相机坐标系之间的转化关系的情况下，两张图像在不同的位置看到同一个三维空间点，利用匹配上的特征点对和三角几何关系，求解三维空间点深度值得过程。

如图 2-8 所示，与对极几何的图相似，点 O_1 和 O_2 是图像 I_1 、 I_2 的相机中心，点 P 是空间中的一个点，点 P 在图像 I_1 、 I_2 的投影分别是 p_1 、 p_2 ，设 $p_1 = (x_1, y_1, 1)$ ， $p_2 = (x_2, y_2, 1)$ ，相机中心 O_1 到 O_2 之间的转化关系是旋转矩阵 R 和平移向量 t ，那么根据对极几何中的公式可以得到：

$$s_1 x_1 = s_2 x_2 + t \quad (2-23)$$

若对上式的左右两边都左乘以 x_1^\wedge ：

$$s_1 x_1^\wedge x_1 = 0 = s_2 x_1^\wedge R x_2 + x_1^\wedge t \quad (2-24)$$

等式右侧为 0，故可以求得

$$s_2 = -x_1^\wedge t x_2^{-1} R^{-1} x_1^{\wedge-1} \quad (2-25)$$

将 s_2 的结果带入到上式中，可以求得 s_1 ：

$$s_1 = -x_1^\wedge t x_2^{-1} R^{-1} x_1^{\wedge-1} R x_2 x_1^{-1} + t x_1^{-1} \quad (2-26)$$

由于存在误差的问题，该等式不一定刚好存在零解，在这种情况下，可以通过最小二乘法或者中点法求解。中点法是将 $O_1 P$ 和 $O_2 P$ 之间的垂线的中点作为三维空间坐标点。

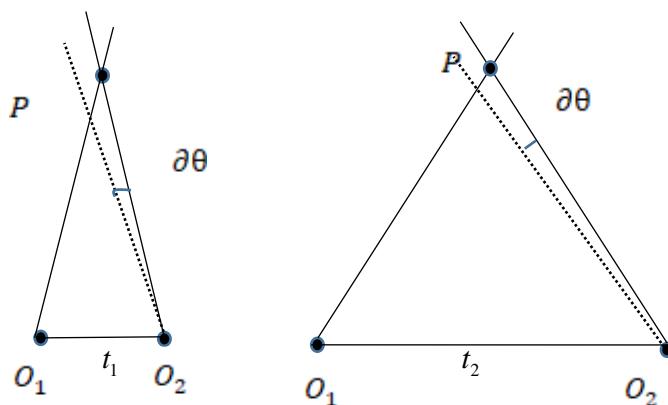


图 2-9 深度的不确定性
Figure 2-9. Depth uncertainty

为了减少三角化中存在的误差，我们在计算的时候，通常会选择较大平移的图像，以增加深度计算的准确度。

从图 2-9 可以看出，在 θ 相同的时候，长基线比短基线的深度误差要小。

2.6 曲面的表示方式

在几何网格的处理中，网格模型的处理主要针对多边形几何对象，本章主要介绍几何对象的表示方法，多边形几何网格对象是由二维曲面组成。

在计算机图形学中，曲面被定义为嵌入到 IR^3 中的可定向 2D 流形。对于几何网格的处理来说，首先需要确定其网格曲面的表示方式。

曲面的表示主要有两种类型：

(1)参数表示(Parametric representations)，参数曲面被定义为将二维参数域 $\Omega \subset IR^2$ 映射到三维表面 $S = f(\Omega) \subset IR^3$ 的向量值参数化函数 $f: \Omega \rightarrow S$ 。若一个函数的值域是一个线性空间或一个线性空间的一个子集，那么这个函数被称为向量值函数。

(2)隐式表示(Implicit representations)，曲面的隐式表示被定义为标量值函数的零集 $F: IR^3 \rightarrow IR$ 。例如， $S = \{x \in IR^3 \mid F(x) = 0\}$ 。

以平面上的单位圆为例，它的参数表示如下：

$$f: [0, 2\pi] \rightarrow IR^3, t \rightarrow \begin{pmatrix} \cos t \\ \sin t \end{pmatrix} \quad (2-27)$$

它的隐式表示如下：

$$F: IR^3 \rightarrow IR, (x, y) \rightarrow \sqrt{x^2 + y^2} - 1 \quad (2-28)$$

参数表示和隐式表示都有其独特的优点和缺点，因此对于不同的几何问题，都应该选择更适合的那一个。根据不同的需求采用不同的方法。当需要对模型进行渲染的时候，除了空间点的坐标和面的信息，还需要保存各个面的法向量信息。

2.7 多边形网格的格式

多边形网格模型在三维重建和网格处理过程中，其存储方式主要有三种，obj 文件、ply 文件、off 文件。在本文中，本文主要采用了 obj 格式和 ply 格式来存储。下面简单介绍下这两种文件的存储格式。

Obj 文件是一个通用的网格模型存储格式，它支持直线、多边形、法线、曲线等多种类型。

Obj 文件有四种主要的数据类型。

顶点(Geometric vertices)用字母 v 表示，其后跟着每个顶点的三维空间坐标，格式为 $v \ x_n \ y_n \ z_n$ 。

纹理坐标(Texture vertices)用 vt 来表示，三个坐标的含义是对应的纹理贴图的对应坐标，在为三角面片绘制纹理时，需要据此找到对应纹理图片， u 和 v 是二维纹理的横纵坐标， w 是三维纹理的坐标。格式为 $vt \ u_n \ v_n \ w_n$ 。

顶点法向量(Vertex normals)用 vn 来表示，顶点的法向量是每个顶点的朝向，在渲染的时候，需要根据每一个面片的三个顶点的法向量求平均，来确定该面片的朝向信息。格式为 $vn \quad x_n \quad y_n \quad z_n$ 。

面(Face)用字母 f 来表示，面是绘制模型的关键，每一个三角形面片对应三个顶点坐标、三个纹理坐标、三个顶点法向量。如果是四边形面，那么对应的数据项会多一个。格式为 $f \quad v/vt/vn \quad v/vt/vn \quad v/vt/vn$ 。

Ply 格式(Polygon file format)是另外一种多边形文件格式，ply 文件有两种保存方式，ASCII 码或者二进制的形式，如果是以 ASCII 码的格式保存，其文件格式规定如下，如表 2-1 所示：

表 2-1 网格模型文件格式
Table 2-1. Grid model file format

-
1. ply
 2. format ascii 1.0
 3. comment ***
 4. element vertex n
 5. property uint8 x
 6. property uint8 y
 7. property uint8 z
 8. element face m
 9. property list uint8 int32 vertex_index
 10. end_header
 11. x₁ y₁ z₁
 12. x₂ y₂ z₂
 13.
 14. x_n y_n z_n
 15. 4 v₀ v₁ v₂ v₃
 16.
 17. 4 v_i v_j v_k v_n
-

文件的首行表示这是个 ply 文件；其次 *format* 表示该文件是通过 ASCII 码或者二进制保存，二进制用 *binary_little_endian* 或者 *binary_big_endian* 表示；*comment* 代表这行的内容是注释；*element vertex n* 是说明在 *header* 结束后面有 n 个顶点坐标；*property uint8 x* 说明了顶点的属性格式 x 是 *uint8* 类型的，*property uint8 y* 和 *property uint8 z* 也是类似的，其他类型有 *float32*、*float64* 等；*element face m* 代表这个模型存在 m 个面；*property list uint8 int32 vertex_index* 在这个面的列表里面是各个顶点的索引下标，每一个整数代表组成这个面的一个顶点；*end_header* 则是头部的结束标志；接下来的 n 行对应着 n 个顶点的坐标，从 0 开始编号；最后的 m 行是 m 个面是由那几个顶点组成的。

2.8 四边形网格的重网格化

重网格化技术是以一个三维网格作为输入，生成和输入网格形状近似，并且

满足一些质量要求的新网格。这种技术可以用来降低网格的复杂程度，并提升网格的质量，使其可以用在不同的应用程序中，例如动画、游戏、增强现实等。网格的质量指的是像规则程度、和特征的对齐程度、网格的大小形状等评价标准，质量越高的网格，越容易应用于行业的开发中。

从局部的角度来看，可以从面的类型、形状、密度分布、朝向和旋转四个角度来判断网格的质量。

网格按照每个面包含的顶点数目来区分，最常见的主要有两种，三角形网格和四边形网格。本文的研究内容主要针对四边形网格，四边形网格可以更广泛的应用在不同的系统中。四边形网格在理论上都可以通过生成对角顶点的边来对四边形网格进行剖分，三角形网格也可以采用三角形重心和各个顶点相连接的方式，只要抛弃原来的边，就可以生成四边形网格。

网格元素的形状。各向异性和各向同性指的是四边形网格在各个方向是否是均匀的，如果四边形趋近于正方形，那么它是各向同性的，各向同性的网格分布均匀，更方便处理。各向异性的网格，它的形状更和网格特征相契合，能同输入网格达到更相似的外形。

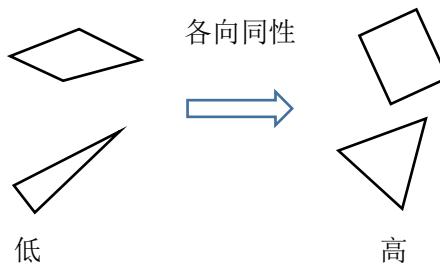


图 2-10 各向同性

Figure 2-10. Isotropy

网格元素的密度：如果网格以均匀的密度分布表面中，那么它的特征容易变得模糊。在非均匀的网格中，曲率高的区域会有更多更小的面片，它可以保证网格和输入网格的形状更相似。网格的对齐和朝向：在重网格化的过程中，尖锐的特征可能受到影响无法保证其原来的状态，为了减少这种情况的发生，网格元素应该和特征相对齐，其中各向异性就发挥了重要作用。

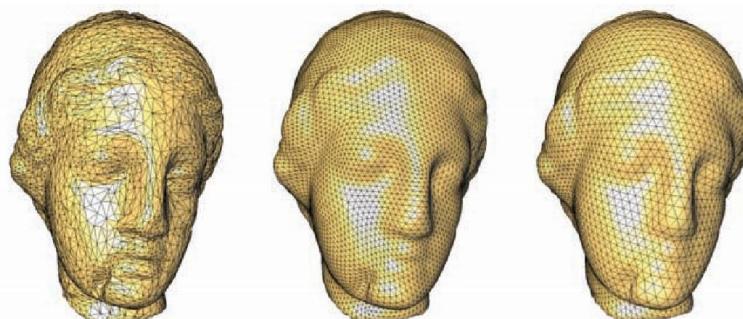


图 2-11 不规则网格(左)、价半规则网格(中)、规则网格(右)^[54]

Figure 2-11. Irregular grids (left), semiregular grids (middle), regular grids (right)

从全局的角度来看，网格可以根据其顶点的价来分为规则网格、价半规则网格和不规则网格。顶点的价是和该顶点相连的边的数目，四边形网格中，顶点的价为 4，价不为 4 的顶点称为奇异点或者不规则顶点。

不规则网格(Irregular meshes)是连通性不存在任何规律的网格，价半规则网格(Semiregular meshes)是对粗糙网格进行常规细分得到的，其不规则顶点的数目比较少，并且保持恒定。规则网格(Regular mesh)的全部顶点都是规则的，它可以更好的用于渲染过程。

2.9 本章小结

本章主要对三维重建和网格重拓扑中需要的一些基础知识进行简单的介绍，依次简要的介绍了三维空间中物体到图像的坐标转化关系，图像去畸变的流程，三维重建中恢复空间点深度的算法，以及重网格化中一些常见的基础概念。

第三章 基于匹配优化的从运动恢复结构

3.1 引言

本章主要介绍基于匹配优化的增量式从运动恢复结构算法。本算法主要针对特征匹配进行了优化，提高特征匹配的效果，从而实现更好的增量式稀疏重建。主要算法流程如下：首先获取图像数据集，对图像进行预处理，获取图像中的焦距等信息，并对图像进行去畸变工作，如果只对单一物体进行重建，则需要额外去除图像中的背景，消除来自背景的干扰；之后从各个图像中提取特征点，并采用本文提出的特征点优化匹配算法进行特征点匹配；将具有相同的特征点的图像连起来，生成特征跟踪；从图像数据集中选择合适的下一帧图像，进行增量式稀疏重建，并采用捆集调整优化相关相机和三维空间点的位姿，不断迭代最终生成稀疏重建的结果。

3.2 图像预处理

在正式进行重建算法前，需要对图像进行预处理工作，并构建场景，方便之后的稀疏重建模块进行处理。其主要工作内容为从图像数据集的文件夹中读取图片内容，提取 EXIF 信息，获取焦距等相机拍摄信息，对图像做畸变处理。在工业应用中，通常情况下需要展现每一个物体的对应形象，那么我们需要对某一个物体进行单独的重建，因此去除环境对其干扰也格外重要，本文需要采用前景提取算法对物体进行提取，避免在重建完成之后，仍保有其他不相关的噪声信息。

构建场景的工作首先从读取文件夹中的内容开始，在读取到该文件夹下所有的文件名后，对所有的图像文件都进行重命名，按照字典顺序将其重命名为从 0 开始的整数字符。

在读取到每一个文件的格式时，如果是 JPEG 格式的文件，本文采用了 libjpeg 的库来对 jpeg 格式的文件进行读取。Libjpeg 库是一个开源的、完全用 C 语言编写的 JPEG 格式的库，该库中包含了读取 JPEG 格式文件中的 EXIF 信息(可交换图像文件格式，Exchangeable image file format)，EXIF 信息中保存了各种相机拍摄时候的数据，例如焦距、相机类型、分辨率、CCD(电荷耦合器件，Charge coupled device)尺寸等内容。

其中的相机焦距和分辨率信息是后续的稀疏三维重建过程需要的摄像机参数，本文通过库中的接口进行读取。如果碰到其他格式的图像或者没有读取到对

应的 EXIF 信息，那么本文全部都采用默认的 35mm 的焦距，尽管这是存在误差的，但在整个稀疏重建过程中是在可允许的范围内。

在将原图片保存到对应的场景文件的同时，本文会生成对应的低分辨率图像，这将用于之后的特征提取和匹配的阶段。低分辨率图像采用线性差值的方法来生成，目标图像生成的分辨率大小为 50，为了保证生成的图像仍然和原图像的比例一致，设原图像的比例为 $r_o = w_o / h_o$ ，低分辨率图像的比例为 $r_t = w_t / h_t$ ，如果 $r_o > r_t$ ，那么生成图像的宽高如下：

$$\begin{cases} w_t = \lceil h_t * r_o \rceil \\ h_t = h_t \end{cases} \quad (3-1)$$

如果 $r_o < r_t$ ，那么生成图像的宽高如下：

$$\begin{cases} w_t = w_t \\ h_t = \lceil w_t / r_o \rceil \end{cases} \quad (3-2)$$

如果图像的焦距和畸变参数都是已知的，本文可以用第二章中的相机畸变模型对图像进行去畸变，生成更准确的图像。

如果是为了对某个单独的物体进行建模，那么需要对图像做一些额外的处理工作，在实验室环境下，我们准备了两种背景颜色分别是黑色和白色，来避免背景对重建的干扰。

为了排除图像中噪声的影响，首先采用核为 3 的高斯滤波器对输入图像 I 进行平滑，生成平滑后的图像 I_g ；然后对图像 I_g 进行锐化，本文采用 Laplacian 高通滤波算子来进行锐化操作，锐化后的图像为 I_L ，其核如下所示：

$$H = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (3-3)$$

锐化后的图像 I_L 是 I_g 和算子 H 进行卷积操作生成的。初始的图像前景可由大津算法进行二值化分割生成，大津法是通过将背景和前景两类之间的方差最大化来确定合适的阈值 t ，由于实验中的背景大半部分为黑色或者白色，其可以很好在目标区域中实现分割。据此可以得到初始分割掩码图片：

$$I_{mask} = \begin{cases} 0 & I_L(x, y) \geq t \\ 255 & I_L(x, y) < t \end{cases} \quad (3-4)$$

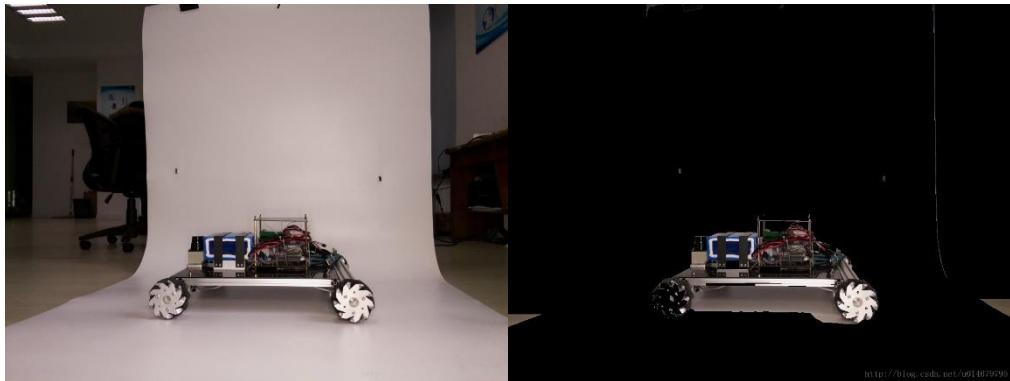
然后对掩码图片进行核大小为 20 的腐蚀和膨胀操作，进一步去除之前没有滤除掉的噪声和部分错误分割的结果。

由于大津法生成的分割结果不是很理想，需要进一步采用其他分割方法进行进一步的处理。本文采用了 GrabCut 算法，将之前得到的掩码图像的前景和背景标记为可能的前景和背景输入到 GrabCut 算法中，最终生成较为可靠的分割结

果。

最后是生成提取出的前景目标，将原图像 I 和掩码图像 I_{mask} 做逐像素的与操作，生成目标物体的前景分割结果 I_{obj} ：

$$I_{obj} = I \oplus I_{mask} \quad (3-5)$$



(a) 原图

(b) 分割结果

图 3-1 前景分割效果图

Figure 3-1. Fore-ground split effect chart

3.3 特征提取和匹配

3.3.1 特征提取

特征点提取是在重建过程中一个非常重要的模块，特征点是在用来描述图像一部分区域内容的一种描述符，由多维的二进制向量或者实数向量构成，它一般是通过提取该部分区域的梯度直方图来获得的。

在基于视觉的三维重建算法中，特征点提取的质量和位置的准确性显得尤为重要，特征点提取位置的准确性会影响到对应位置重建生成模型的质量。特征点的质量主要体现在匹配的准确性，能够与不同的特征点具备区分度，对于同一位位置的特征能够轻易的匹配上，在不同的特征之间存在明显的差异，无法完成匹配。此外还需要有在各种情况下的不变性，例如旋转不变性、尺度不变性、光度不变性等，并且具备对图像噪声、模糊、图像压缩等因素的抗干扰能力。

对于三维重建算法来说，在数据集较小的情况下，特征点提取的速度并不格外重要，大多数的重建算法都是采用离线处理的办法，不需要进行实时重建。当需要对大场景进行重建时，例如重建整座城市，需要考虑到特征提取的时间问题，在本文中，主要针对单个物体进行重建。

在比较了多种描述符的重建结果后，本文中采用的特征提取方法主要是 SIFT 和 SURF 两种特征提取算法，SIFT 是一种具备旋转不变性、尺度不变性的特征提取方法，SURF(Speeded up robust features，加速鲁邦特征)是一种基于 SIFT 改

进的快速特征提取方法，它是基于积分图像进行卷积运算，采用了 Hessain 响应来确定特征点的位置。

如下图，是 SIFT 和 SURF 特征提取的结果，左边是 SIFT，右边是 SURF，可以看出 SIFT 特征点提取的分布更加均匀，特征点的数目更加多。SURF 特征点的位置集中于目标物体等具备明显梯度的位置之上。



(a) SIFT 特征

(b) SURF 特征

图 3-2 特征提取

Figure 3-2. Feature extraction

两种提取算法相结合可以得到数目更加多、更加鲁棒的特征点，在整个重建的流程中，也更容易恢复出更多的三维空间点。

3.3.2 基于统计的特征匹配算法

三维重建中的三维点云的位置是通过多幅图像之间的特征匹配后的结果进行三角化得到，如果提取到的特征点没有匹配上对应的正确的特征点，会使得之后的重建结果产生较大的误差。

为了提高三维重建过程中的稳定性，本文采用了基于网格统计的特征匹配算法。假设不存在物体之间发生遮挡的情况时，在相机运动的过程中，随着视角的变化，对应的特征点在图像上的位置也会发生对应的改变，但他们之间的相对位置不会发生改变。

该方法正是基于这样的空间一致性提出的，假设在图像空间中存在较多匹配上的特征点，本文便可以通过统计对应特征点的邻域中是否存在足够多的特征点来支持这一匹配，也可以由此判断出该特征匹配的准确性。

算法的流程图如下：

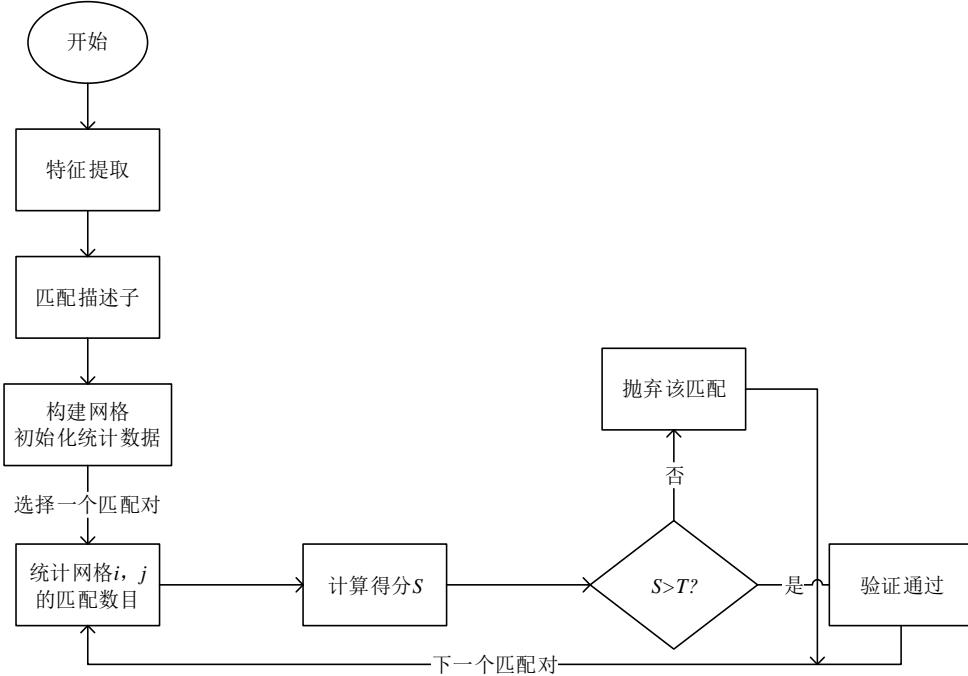


图 3-3 基于统计的特征匹配算法流程图

Figure 3-3. Flowchart of statistical-based feature matching algorithm

首先本文从图片中提取 SIFT 和 SURF 特征点，并计算描述子。为了尽可能的提高验证的正确性，在提取特征点的时候，应当尽量的增多特征点提取的数目。在本文的实验中，在一张 1000*1000 分辨率的图片中，本文设计的算法提取了大约 10000 个特征点。

对特征点进行匹配，使用 *NORM_L2* 汉明距离来衡量，根据 Csurka 等人论文中的建议，L2 正则化后的描述符通常都有助于提高匹配的质量^[55]。

在图像中构建 $m \times n$ 的网格，本文统计不同网格中的对应的特征匹配数目。根据图像的分辨率，对网格进行划分，划分网格的数目 m 和 n 的计算方式如下所示：

$$\begin{aligned} m &= l_{img} * l_{target} / k \\ n &= r_{img} * r_{target} / k \end{aligned} \quad (3-6)$$

其中， m 是横向的格子数目， n 是纵向的格子的数目， l_{img} 、 r_{img} 是输入图像的列数和行数， l_{target} 、 r_{target} 是目标图像的列数和行数， k 是比例系数，一般设置为 20。

初始化一个二维数组 $grid$ ，用来保存格子 i 和格子 j 中的特征匹配数目，选择其中匹配上最大的数目的格子为候选格子。

接下来，计算一个得分函数 S ，评估某一个特征点附近的匹配程度。

$$S_{i,j} = \sum N_{i+k, j+l} \quad k, l \in \{-1, 0, 1\} \quad (3-7)$$

其中， i, j 是某一个匹配上特征点分别在图像 I_1 和图像 I_2 所在的格子编号， $N_{i,j}$ 表

示在图像 I_1 的第 i 个格子和图像 I_2 的第 j 个格子上能够匹配上的特征点数目。

如果这个匹配附近的特征点的数目得分大于阈值 $T_{i,j}$ ，那么就认为这个匹配是正确的，阈值 $T_{i,j}$ 的计算公式如下：

$$T_{i,j} = k_t \times \sqrt{\frac{\sum N_{i+k}^f}{\sum N_{i+k,j+l}^{pair}}} \quad k, l \in \{-1, 0, 1\} \quad (3-8)$$

其中， k_t 是阈值系数， N_{i+k}^f 表示在图像 I_1 的第 $i+k$ 个格子中的特征点的数目， $N_{i+k,j+l}^{pair}$ 的取值范围是 {0,1}，如果对应格子中存在匹配那么取值为 1，否则等于 0。该公式表示了格子 i 和格子 j 能够匹配上的特征数目占图像 I_1 提取到的特征点数目的比例大小。

上述是没有考虑图像旋转和缩放的情形。在公开的网络重建数据集中，图像通常是由不同人在不同的距离下拍摄的，所以构建尺度空间是非常有必要。如果考虑图像的缩放的情况，本文将修改图像 I_2 中的格子大小和数目，其修改后的格子数目分别为

$$\begin{aligned} m_{scale} &= k_{scale} \times m \\ n_{scale} &= k_{scale} \times n \end{aligned} \quad (3-9)$$

其中尺度变换参数 k_{scale} 的取值范围为 $k_{scale} \in \{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}, 1, 2, 3, 4\}$ 。

如果考虑图像旋转的情况，9 个对应的格子，那么就存在 8 种可能的旋转情况，分别对应从旋转 0° 、 45° 到 315° ，其中每次相隔 45° 的角度偏移。在本文中保持图像 I_1 的格子不动，修改图像的 I_2 中对应格子的编号。

上面的每一种旋转和缩放的情况都需要遍历一遍，然后选择其中匹配上数目最多的情况作为最终的结果。

如图 3-4 所示为使用该算法的特征匹配效果。

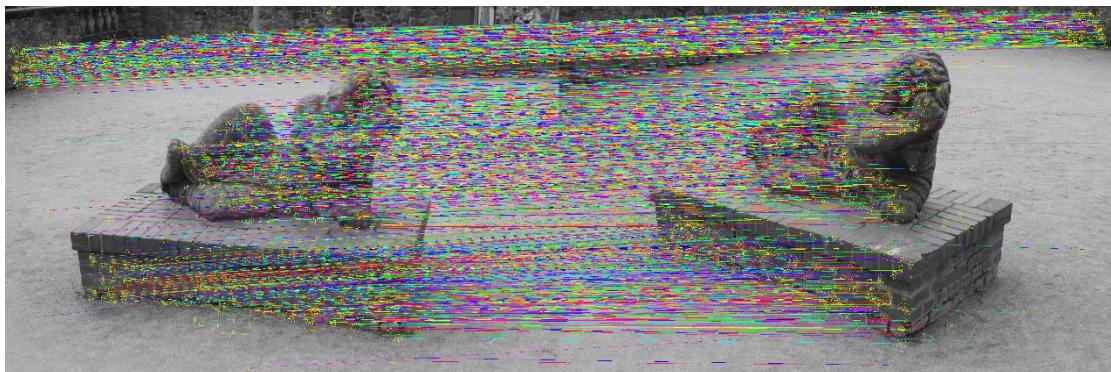


图 3-4 基于统计的特征匹配效果图

Figure 3-4. Statistical-based feature matching effect chart

从图 3-4 中可以看到，本算法可以生成大量匹配上的特征点对，并且绝大部分的特征匹配都是准确的。

基于统计的特征匹配方法能在很大程度上保存正确的匹配点对，但也会存在一些匹配点对，其位于正确的网格附近，并且附近存在足够多的匹配支持它，但它的匹配结果却是错误，这就需要通过其他方法对其进行验证。本文是通过增加基于迭代 RANSAC 的几何一致性验证、双向匹配一致性验证来实现。

双向一致性验证是两幅图像在完成匹配的过程之后，用图像 I_1 作为右图与图像 I_2 作为左图再进行一次上述的匹配过程，在匹配的过程中，保存匹配上的对应特征点的下标，然后对两次的匹配结果进行对比，如果两个都是对面图像中最符合的匹配特征点，那么说明这个匹配结果是正确的。即需要满足下面的两个要求：

$$\begin{aligned} \text{match2_1}[\text{match1_2}_i] &= i \\ \text{match1_2}[\text{match2_1}_j] &= j \end{aligned} \quad (3-10)$$

其中， match2_1 和 match1_2 分别保存了从图像 I_1 到图像 I_2 之间和图像 I_2 到图像 I_1 之间的匹配结果的映射关系。

在完成双向一致性验证之后，需要对剩下的匹配结果的数目 N 进行验证，判断其是否大于预先设定的阈值 $\text{MIN_MATCH_THRES}=8$ 。

基于迭代 RANSAC 的几何一致性验证是通过迭代计算基础矩阵的 RANSAC 算法，RANSAC(Random sample consensus，随机采样一致性算法)是在一组包含错误数据的数据集上，随机抽取一部分数据将其作为初始值，之后用这些初始的特征匹配点对使用 8 点法估计两个相机之间的基础矩阵，并通过 SVD 分解求解两个相机之间的位姿转化关系。

接下来对匹配中的每一个点对，计算 Sampson 距离作为误差，如果误差小于设定的阈值 $t=0.0015$ ，那么这个匹配是符合计算出来的基础矩阵的，Sampson 距离的计算公式如下

$$d(x_1, x_2) = \frac{(x_2^T F x_1)^2}{(F x_1)_x^2 + (F x_1)_y^2 + (x_2^T F)_x^2 + (x_2^T F)_y^2} \quad (3-11)$$

在完成一组的基础矩阵验证后，统计内点数目，之后进行迭代，找到内点数目最多的一次迭代结果作为最终的验证结果。如果通过验证后的匹配点对数目小于 8，那么两幅图像 I_1 和 I_2 便不满足匹配的要求。

通过上述验证成功的特征匹配点对基本可以保证匹配的准确性，但该方法对于三维重建来说仍然存在着一些问题，在图像中特征点提取数目的比较少的位置时，同时也是特征匹配数目较少的区域，倘若该区域存在错误和正确的特征匹配点对，该方法容易将正确的特征匹配对当作错误的。

从上面的匹配结果也可以明显的看出，SIFT 特征点在地面提取的特征点基本都没有保留下。另外，当视角发生改变的时候，一张图像 I_1 上特征可能对应于另外一张图像 I_2 上边缘的特征点，由于在图像 I_2 中周围的网格数目较少，该算法

的结果也容易变得更差。

为了解决上述问题，本文采取融合其他特征匹配方法的方式，缓解基于统计的特征匹配方法中的缺陷。

本文采取了基于多个候选匹配结果的重匹配算法来重新获得被错误消除的正确匹配。该算法的流程如下：

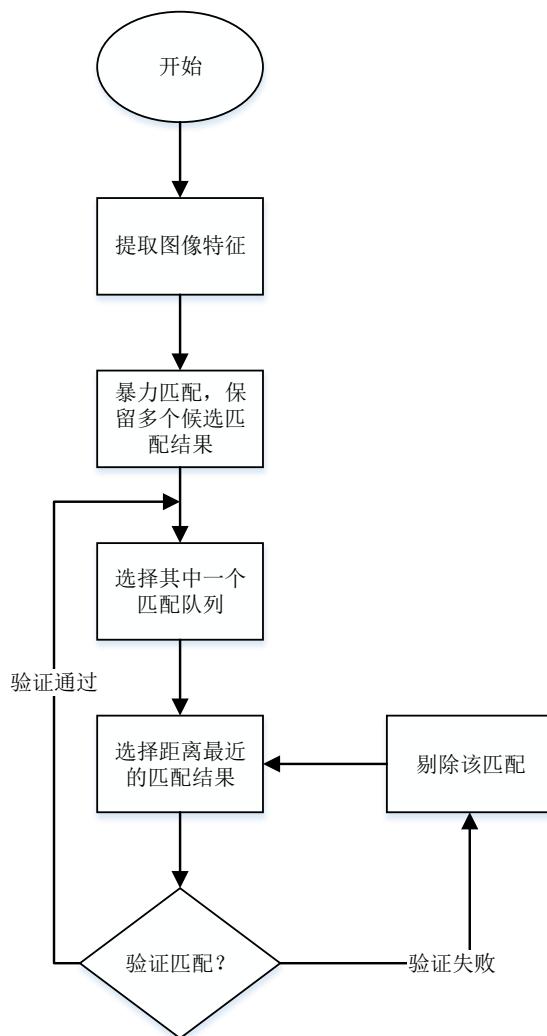


图 3-5 重匹配算法流程图

Figure 3-5. Statistical-based feature matching effect chart

首先也是提取特征点，然后采用暴力匹配的方法，对图像 I_1 和 I_2 中提取的每一个特征点都进行匹配，对于每一个特征点都保存一个候选的优先队列 L ，队列 L 中为该特征点在另外一幅图像中汉明距离最近的 n 个候选特征点，这 n 个特征点按照距离的远近从小到大进行排列，本文中的 n 的个数为 2。

进入候选队列的特征有两种情况：

- (1) 如果最近的特征点距离明显小于次近的特征点距离，那么将这两个特征点都放入到候选队列中。两者之间的比值 $ratio$ 为

$$ratio = \frac{L1_{i,j}}{L1_{i,k}} \quad (3-12)$$

$L1_{i,j}$ 代表特征点 i 和最近的特征点 j 的距离, $L1_{i,k}$ 代表特征点 i 和次近的特征点 k 之间的距离。

- (2) 如果最近的特征点距离和次近的特征点距离之间的比例没有明显的差异, 那么在候选队列中还需要放入第三近的特征点下标, 直到队列满, 或者距离的比例存在明显的变化。

之后, 选择其中一个特征点 f_{left} 的匹配结果队列 $L_{f_{left}}$, 从优先队列中选择距离最小的候选特征点 f_{right}^1 进行验证, 验证采用几何一致性验证。

由于在上述的匹配过程中, 已经生成了大量的验证通过的匹配结果, 故不需要对已经完成匹配的结果再做一次匹配和验证。在上述图像的匹配中缺少的特征点相对来说是少数的, 故本文直接采取了上述生成的基础矩阵结果来对这次优先队列中的结果进行验证, 如果满足 Sampson 距离阈值的约束, 那么便可以将其加入到最终的匹配结果中。

如果该队列中最近的候选特征点不满足上述的几何一致性约束, 那么就从候选队列中选取下一个次近的特征点进行几何一致性验证。

循环上述步骤, 直到所有的候选匹配队列中的都被验证。在结束一个重新匹配后, 对剩下的所有匹配都进行重匹配过程。

如上为本文的重新匹配算法。本文可以将其结合到基于统计的特征匹配算法中, 对其中的特征匹配失败的匹配对进行重新匹配, 以获得更多可靠的匹配结果。结合基于统计的特征匹配方法和重匹配算法的流程图如图 3-6 所示。

在提取到图像特征之后, 分为两个部分执行:

其一是执行基于统计的特征匹配算法, 通过最近邻算法得到匹配结果, 并通过网格统计验证、双向验证以及几何一致性验证来判断最近邻匹配结果是否可靠;

其二是通过暴力匹配, 生成候选队列保存距离最近的特征点匹配结果, 在得到基于统计的特征匹配算法的失败结果后, 对失败的匹配进行重匹配算法, 最终将两个匹配结果进行融合, 得到最后的特征点匹配结果。

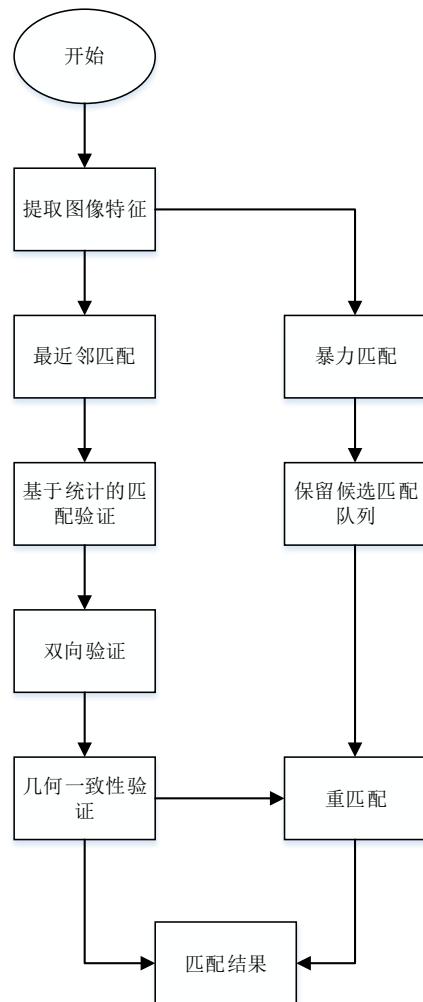


图 3-6 匹配算法流程图
Figure 3-6. Flowchart of matching algorithm

3.4 增量式稀疏重建

到目前为止本章已经介绍了基于统计的特征匹配的主要流程，接下来介绍如何将上述的匹配方法用于稀疏重建中。

稀疏重建的主要流程如下：

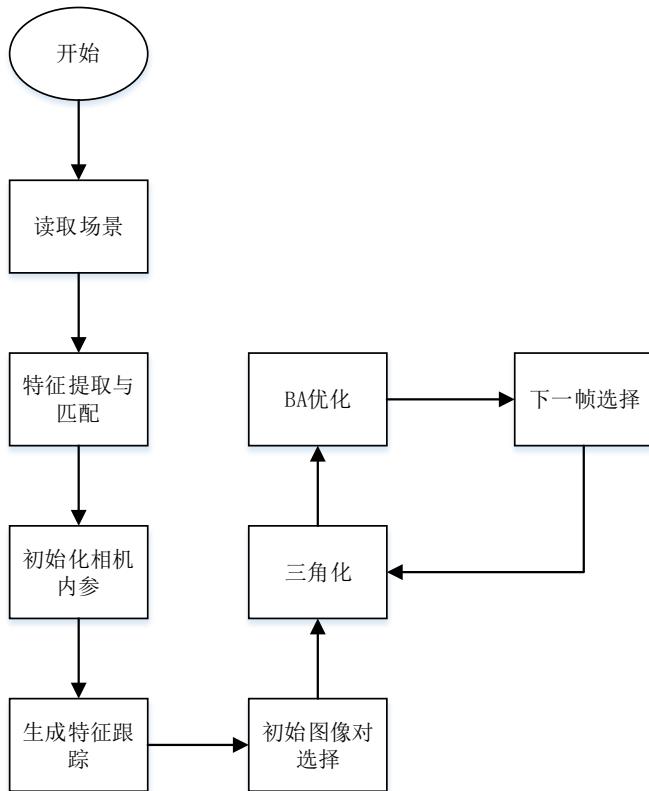


图 3-7 稀疏重建流程图

Figure 3-7. Flowchart of sparse reconstruction

首先从目标文件夹中读取 3.2 章节处理好的图像内容，采用 3.3 章节的算法对每一幅图像进行特征提取和匹配工作，获取不同图像之间的匹配关系。初始化相机内参采用的也是 3.2 中提及的方法。

跟踪是将特征提取和匹配步骤中生成的特征匹配对联合起来，确定相同特征点在每一幅图像中对应的特征点位置。检测是否存在一个跟踪对应图像上多个特征点位置，如果有则将其删除。

初始图像对的选择策略，是首先对所有的匹配对的数目进行筛选，按照图像间的特征点匹配数目将视图对按照从大到小进行排列。之后对匹配上的图像对进行筛选，排除匹配数目过少的，本文设置的匹配数目最小阈值为 8。

用 RANSAC 算法估计两幅图像之间的单应性矩阵，并返回其中符合单应性矩阵的匹配特征点对的数目，计算符合数目的百分比：

$$percent = \frac{N_{inlier}}{N_{match}} \quad (3-13)$$

其中， N_{inlier} 是符合单应性矩阵的匹配数目， N_{match} 是两幅图像之间的所有匹配数目。 $percent$ 的阈值若大于 80%，则认为该变换是平面与平面之间的映射，不符合增量式重建的需求。

接下来，通过相机之间的匹配关系恢复相机间的位姿关系，并将图像间的匹配上的特征点进行三角化，并计算平均夹角 α

$$\bar{\alpha} = \text{avg}(\text{arc cos}(\frac{\overrightarrow{O_1P_i} \bullet \overrightarrow{O_2P_i}}{\|\overrightarrow{O_1P_i}\| \|\overrightarrow{O_2P_i}\|})) \quad (3-14)$$

其中， P_i 是三角化后的特征点的三维空间坐标， O_1 和 O_2 分别是图像 I_1 和 I_2 的相机坐标中心。

初始图像对的夹角需要满足最小的三角化阈值 $t_{angle} = 1^\circ$ 。

如果上述的条件都满足，则选择图像间的特征点匹配数目最多的两幅视图，如果没有满足这些条件的视图对，则计算得分函数 S ， S 由三部分的乘积组成，和匹配的数目、三角化角度以及单应性内点比例相关联。

$$\begin{aligned} S_{match} &= \frac{1+e^{\frac{6^\circ(20-N_{match})}{200}}}{2}-1 \\ S_{angle} &= \frac{1+e^{\frac{6^\circ(1-angle)}{8}}}{2}-1 \\ S_{h-inliner} &= \frac{1+e^{\frac{6^\circ(percent-0.7)}{0.4}}}{2}-1 \end{aligned} \quad (3-15)$$

其中，匹配数目越接近 200，三角化角度越接近 8° ，单应性矩阵内点比例越接近 40%，其得分越高。

得分函数 S 是三项的乘积：

$$S = S_{match} * S_{angle} * S_{h-inliner} \quad (3-16)$$

选择完初始图像对后，对其中的特征点进行三角化，并将其全部放入到捆绑调整中进行优化位姿。

下一帧的选择策略是按照跟踪的数目来进行排序。在完成一部分视图的重建后，对所有的图像对进行捆绑优化，并对重投影误差较高的跟踪进行排除。

3.5 实验结果与分析

本文是在 MVE(Multi-View Environment)算法^[8]的基础上，对其中稀疏重建部分进行修改优化完成的，本文进行了多组对比实验，验证本文算法的可靠性和鲁棒性，下面主要展示四组实验数据集下的运行结果。

首先本文采用了自己拍摄的西瓜数据集，对其做图像预处理工作，如下为西瓜数据集的原图。

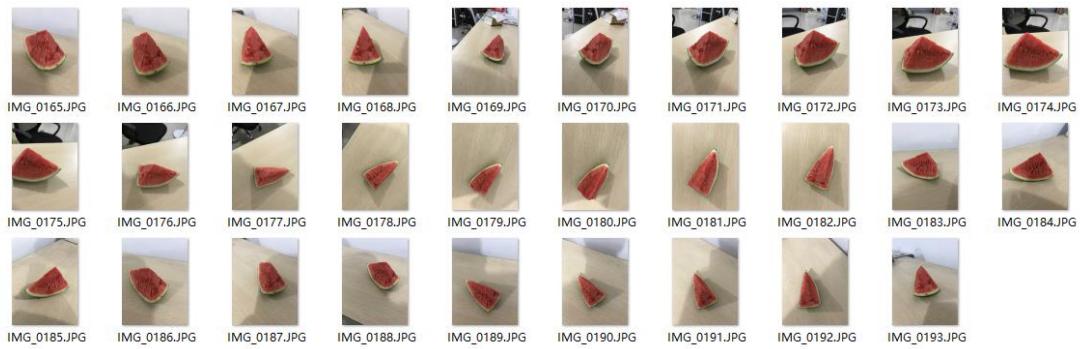


图 3-8 西瓜原始数据集

Figure 3-8. Watermelon original dataset

下图为经过预处理后的西瓜数据集，为了重建出单个西瓜，我们将其中的背景部分剔除。

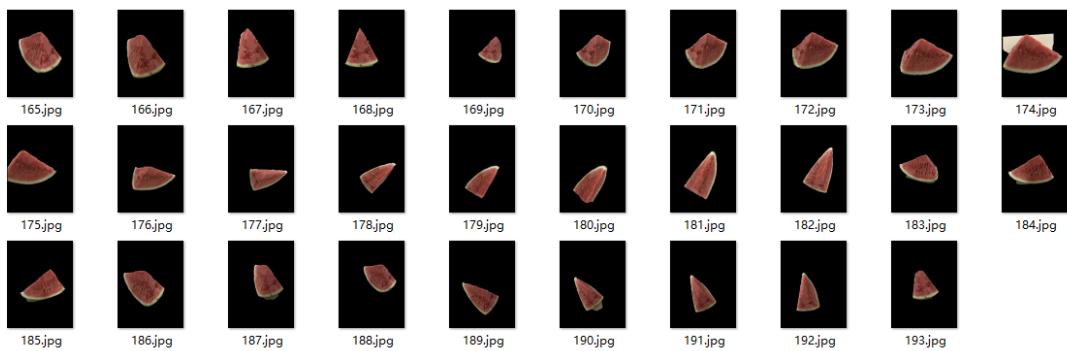


图 3-9 西瓜前景图

Figure 3-9. Watermelon foreground

可以看到基本实现将数据集中的背景剔除，只保留了前景内容,如图 3-10 所示，本算法可以将西瓜完整的重建出来。



图 3-10 西瓜重建效果图

Figure 3-10. Watermelon reconstruction result

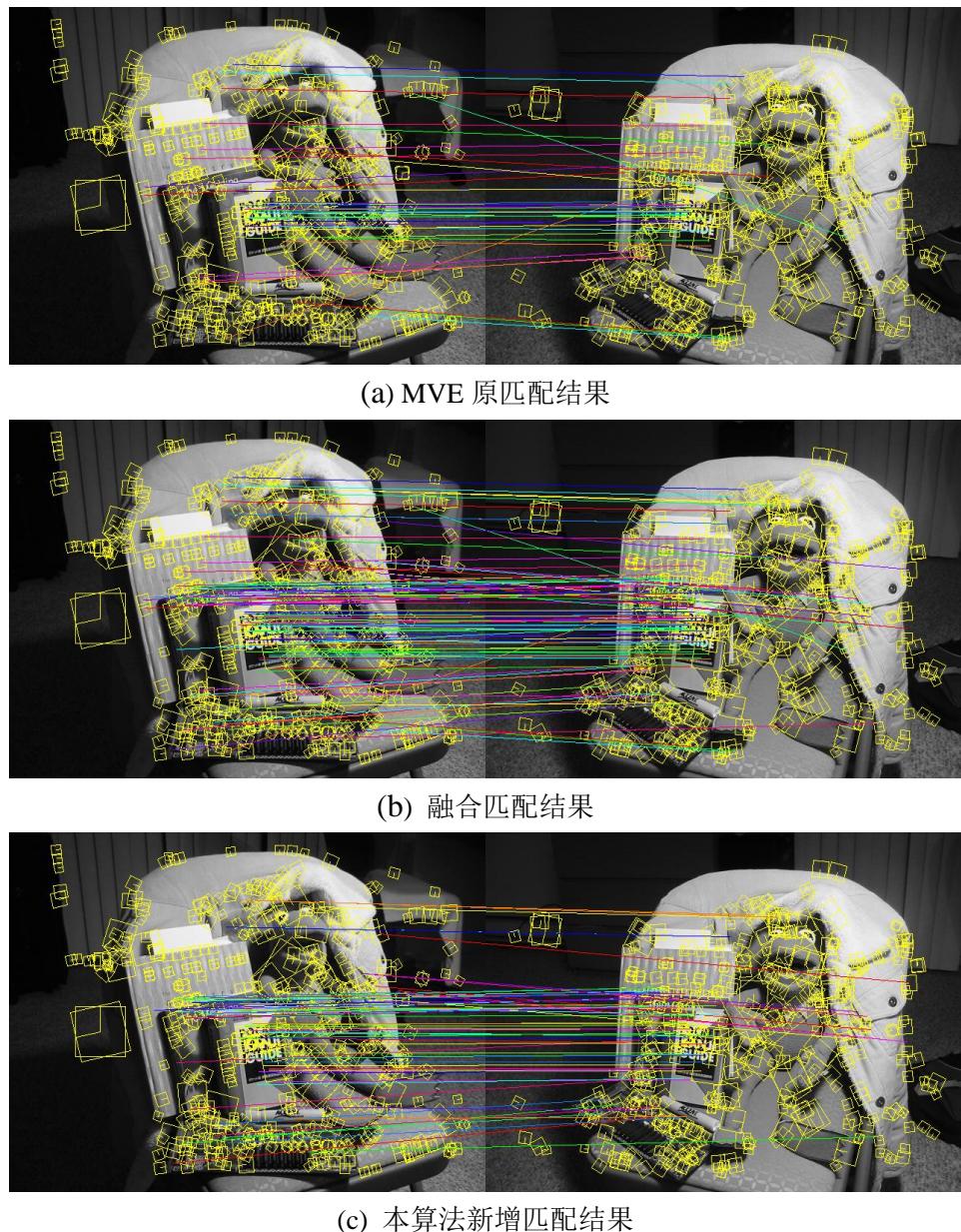


图 3-11 青蛙图片上的特征匹配算法比较

Figure 3-11. Comparison of feature matching algorithms on frog images

如图 3-11 所示，第一行图片是 MVE 算法中的特征匹配算法的结果，原算法在特征匹配阶段只通过两个视图间的特征点距离对进行判断，之后通过双向一致性验证匹配的有效性，故存在一定数目错误的匹配，第二行是本文重新匹配算法和 MVE 原匹配算法的结果，第三行是本文算法中新增加的特征匹配对。

从上述的实验，我们可以看对比原特征匹配算法的结果，本算法在原来的基础上生成了更多可靠的特征匹配结果。经过仔细观察，还可以发现，新生成的特征匹配对绝大多数都是正确的匹配。

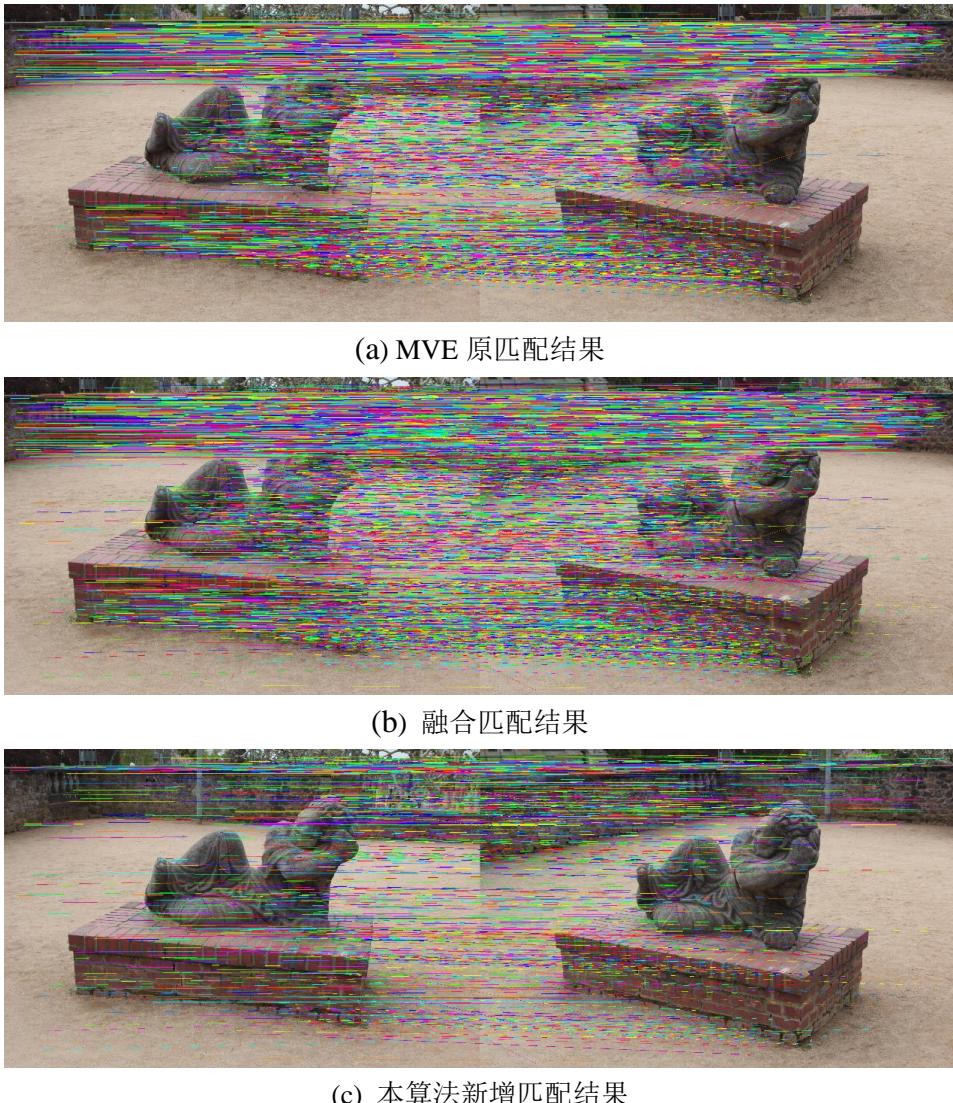


图 3-12 雕塑图像上的特征匹配算法比较

Figure 3-12. Comparison of feature matching algorithms on sculpture pictures

如图 3-12 所示，第一行图片是 MVE 算法中的特征匹配算法的结果，第二行是本文基于网格的特征匹配算法和重新匹配算法结合的结果，第三行是本文算法中新增加的特征匹配对。由于基于网格的匹配算法在验证匹配的有效性时，需要特征匹配点对附近其他匹配点对的支撑，故本算法增加了提取特征点的数目，为了公平性，原匹配算法也在相同特征点提取条件下进行。

在增加提取到特征点数目的情况下，本算法通过基于网格的特征匹配验证筛选掉了部分不符合运动规律的图像，并且通过重匹配算法，寻找到了新的符合几何一致性约束的匹配点对。

如上述对比实验的结果所示，在提取出相同特征点数目的情况下，本文的算法可以显著的增加匹配成功的特征点数目，新增加的特征匹配点对同时也是基本正确的，实现了更好的匹配效果。其具体的统计数据如下表所示。

从表 3-1 可以看到，经过基于网格统计的特征匹配算法和重匹配算法，本算法提高了特征点匹配成功的数目，并保持了匹配的质量。

表 3-1 特征匹配数目比较

Table 3-1. Comparison of number of feature matches

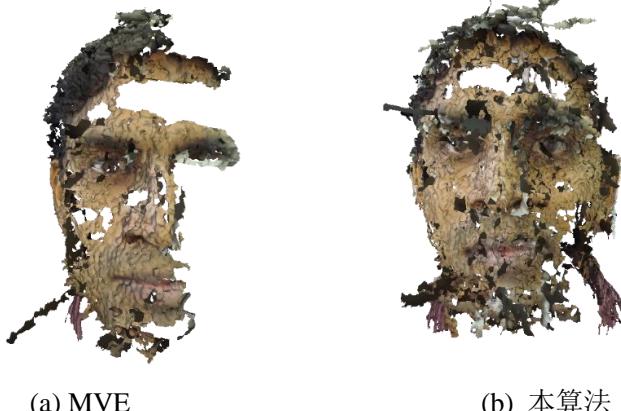
图片	图像大小	特征点数目				匹配数目		
		左图		右图		原始数目	增加数目	总数
		sift	surf	sift	surf			
青蛙	640*480	409	1196	384	1232	44	70	114
雕像	2592*1728	8520	6124	8841	5462	2800	652	3452

如下为本算法和 MVE 算法的重建结果对比，图中(a)列为 MVE 的重建结果，(b)列是本文算法的重建结果。



(a) MVE

(b) 本算法



(a) MVE

(b) 本算法

图 3-13 青蛙和人脸重建结果对比

Figure 3-13. Comparison of frog and face reconstruction results

从图 3-13 的结果可以看到，相比于 MVE 的重建结果，本文算法在重建的完整性上有了一定的提高，青蛙的右腿、躯干、图书上的孔洞都得到了弥补，在人脸数据集上，人的额头和右半边脸也得到了相应的重建。

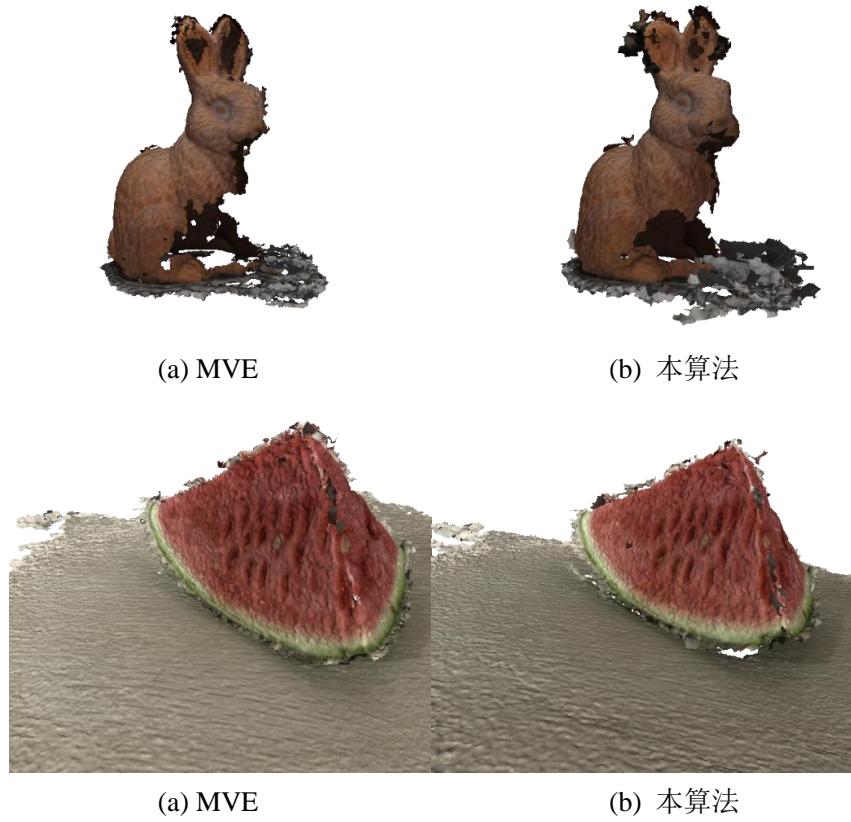


图 3-14 兔子和西瓜的重建效果对比

Figure 3-14. Comparison of rabbit and watermelon reconstruction results

如图 3-14 所示，在兔子数据集上，兔子的嘴唇和右半边脸得到重建，本算法完整性更高；而在西瓜数据集上，两者都重建出了基本一致的结果。

上述实验结果说明，经过本文的算法改进，本文算法对于三维重建的完整性有一定的提升效果。

3.6 本章小结

本文对基于增量式的重建算法进行优化，通过对其中的匹配部分进行优化，融合基于网格统计的匹配算法和重匹配算法，生成了更多可靠的匹配结果。实验表明，与原本的 MVE 重建算法相比，本文方法可以在提取相同的特征数目的情况下，生成更多可靠的匹配，并使得三维重建的结果更加鲁棒和完整。

第四章 基于机器学习的网格重拓扑和光照调整

4.1 引言

随着计算机图形学的发展，越来越多的三维模型应用于不同的领域之中，例如增强或虚拟现实、电影、游戏和历史遗迹保留等。三维网格模型目前主要通过三维扫描或多视图重建等方法获取，然而这些方法生成的模型都存在着数据量大、冗余面多或贴图杂乱难以编辑的问题，不能直接用于增强现实应用、3D 游戏等对模型质量要求高的行业。在计算机视觉中，100 多张高分辨率的图像通常能够生成高达 100 多万面的模型，需要占用大概几十兆的空间，这对网络带宽、模型渲染速度有着很大的挑战。由于建模生成的三维模型其 UV 展开比较混乱，设计人员通常难以直接确定 UV 展开图中每个分片对应于模型上的哪个位置，也就不能够对模型进行二次创作，所以我们需要能够将模型转化为可以生成规则 UV 展开图的半规则网格模型。针对上述问题，本文提出了一种基于机器学习的三维模型自动重拓扑和纹理重映射方法。

为了重新规划网格，本文的方法包含三个主要阶段。首先，使用神经网络来找到奇异点的位置。其次，本文生成半规则网格来拟合输入网格，计算和优化方向场以获得半规则网格，在优化方向场的过程中，本文通过前面的奇异点位置来对方向场进行约束。第三，从环境纹理中去除环境遮挡，在输出网格中加入无光照纹理。

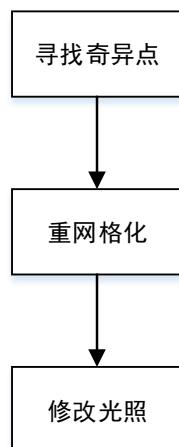


图 4-1 系统流程图

Figure 4-1. System flow chart

4.2 寻找端点

为了解决生成可编辑纹理的问题，本文引入了一种新的网格重构方法。受[19]的启发，本文使用类似的神经网络来确定 UV 映射中的奇异点，这在本文的方法中起着至关重要的作用。

奇异点是三维网格中不规整的点，某一个顶点的价是网格中和该顶点相连接的顶点的数目，一般来说，对于四边形网格，奇异点是指价为三或者五的点。在三维网格中，奇异点数目和位置都会影响生成网格质量的不同，奇异点数目多的网格，其质量较差，规则化程度也就越高。奇异点的位置一般处于网格中的特征点处，例如立方体的 8 个顶点位置，如下图所示：

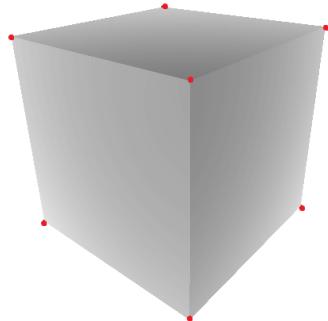
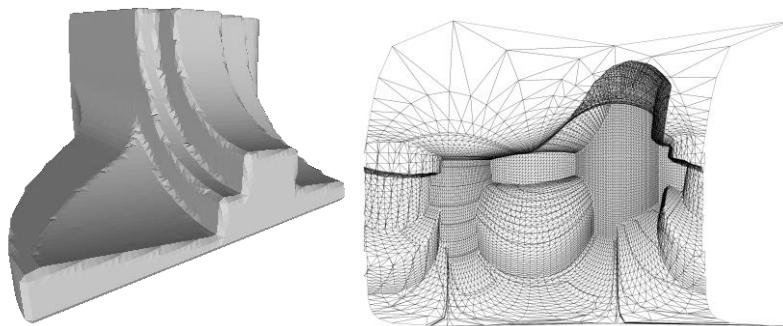


图 4-2 奇异点位置

Figure 4-2. Singular point location

这是因为所有接下来的步骤都是基于这些点的，如计算方向场，生成规则网格，以及将它们组合成一个完整的网格。



(a) 原模型文件

(b) 默认展开结果

图 4-3 对模型文件进行展开的结果

Figure 4-3. Results of expanding the model file

一般来说，三维建模生成的网格都会有太多的小块或部分面片粘连在一起，如图 4-3 所示，这两种情况都使得模型难以编辑，我们无法直观地确定每个面所对应的纹理位置，以至于设计人员无法对模型进行编辑或重新创作。例如上图所示，难以直观的观察出其展开结果和对应模型之间的关系，我们需要将模型的展

开结果更加的直观，美观。为了解决上述问题，本文设计了一种能够自动将模型分割为几个规则面片的方法。

如下图所示，为本算法期望的模型展开结果，可以从中看出，它的分片结果更加贴近原模型的特征和边。

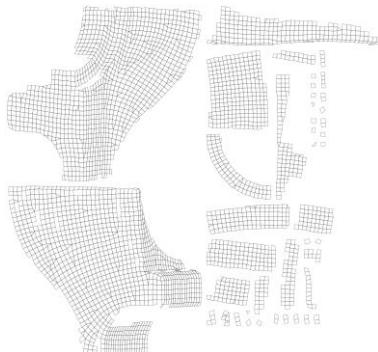


图 4-4 图分片结果
Figure 4-4. Segmentation results

本文使用了神经网络来确定 UV 映射中较为合适的奇异点位置。我们在 UV 映射中寻找合适的奇异点的位置。神经网络有三层的全连接层，每一层有 512 个隐藏单元：

$$y = W_2 \text{ELU}(W_1 \text{ELU}(W_0 x + b_0) + b_1) + b_2 \quad (4-1)$$

其中， W_0 ， W_1 和 W_2 是权重值， b_0 ， b_1 ，和 b_2 是正则项。输入值 $x \in R^6$ 是网格中每一个顶点的位置，输出项 $y \in R^3$ 是顶点的方向值。神经网络的输入是模型文件中每一个顶点的值，以及他们的边结构，输出结果是方向场。本文利用神经网络对场方向的计算结果进行模拟，通过迭代训练出该模型。

激活函数定义如下：

$$\text{ELU}(x) = \max(x, 0) + \exp(\min(x, 0)) - 1 \quad (4-2)$$

每个网格中的顶点都会被遍历一遍判断其是否为合适的奇异点位置，一个合适的奇异点位置应该位于特征点处，并且能够通过 UV 图指示出其在网格模型中对应的面片。

奇异点位置的确定可以通过判断场方向来确定。合适的奇异点位置一般对应于 UV 展开图中的几个端点处的位置，例如立方体的展开图中，8 个顶点分别都处于模型分片的边缘位置，我们通过他的位置也可以推断出该模型分片的结果是否合适。

4.3 重网格化

在这一部分中，本文优化了方向场的值，并且生成对应的半规则网格。本文

确定网格中初始方向场的方向，并通过能量函数对整体的方向场进行平滑，将其转化为全局统一的方向场，之后确定顶点位置以及顶点之间边的朝向，最后生成半规则的四边形网格。生成的四边形网格中保存有上面确定的奇异点，故不是纯四边形网格。

具体重网格化算法流程如下所示。

首先，本文根据顶点切平面的法向量来确定初始的方向场方向。方向场是用来确定和拟合网格边缘信息的一种场，它定义在切平面上的单位正交向量场，设在顶点 p 上有一个法向量 \vec{n} ，单位切向量 \vec{u} 定义为在模型表面 f 上满足同时如下条件的向量：

$$\|\vec{u}\| = 1 \quad \vec{u} \cdot \vec{n} = 0 \quad (4-3)$$

在本文中，由于本文是将三角形网格重新网格化为四边形网格，故本文采用了 4 旋转对称方向场，4 旋转对称方向场的定义如下：它是将 4 个单位切向量的集合，从单位切向量 \vec{u} 开始，本文将 \vec{u} 在切平面内，以法向量 \vec{n} 为中心，依次进行旋转，每次旋转 90 度，总共旋转三次，得到新的单位切向量 $\vec{u}_1 \quad \vec{u}_2 \quad \vec{u}_3 \quad \vec{u}_4$ ，那么 4 旋转对称方向场则是 $\vec{d} = \{\vec{u}_1, \vec{u}_2, \vec{u}_3, \vec{u}_4\}$ 。

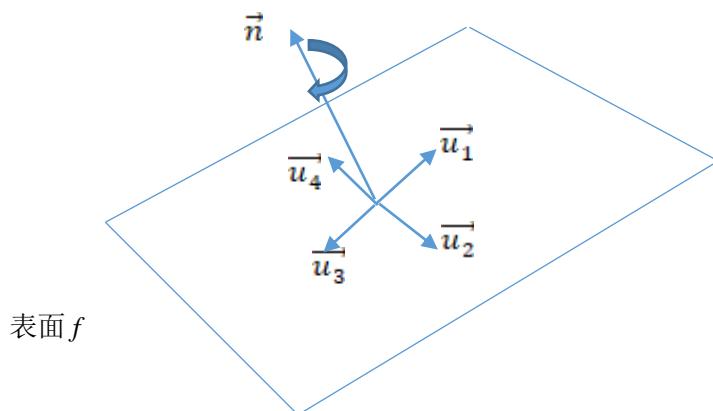


图 4-5 方向场示意图

Figure 4-5. Direction field diagram

如上图所示，顶点 p 上的最初的切向量 \vec{u}_1 通过三次绕着法向量 \vec{n} 的旋转，生成顶点 p 上的方向场 \vec{d} 。

构造完方向场信息后，本文通过下述平滑能量项来对方向场信息进行优化，通过从之前定义的奇异点位置和方向信息来生成一个满足奇异点约束的光滑方向场。

每一个顶点上的平滑的方向场方向可以用来指导生成最终网格模型上不同顶点边的方向，为了使得生成的边更加的规整，故需要将顶点上的方向场进行旋转，使得不同顶点上的方向场的方向相统一。

因为最初的方向场方向是混乱的，并且不同顶点之间的方向场有着巨大的差

异。两个不同顶点上的方向场 \vec{d}_0 和 \vec{d}_1 ，如果他们之间可以通过转化矩阵转化到另外一个平面上，那么我们称其有相同的拓扑。对于连续曲线上的任意两个顶点之间的方向场都为相同的拓扑。

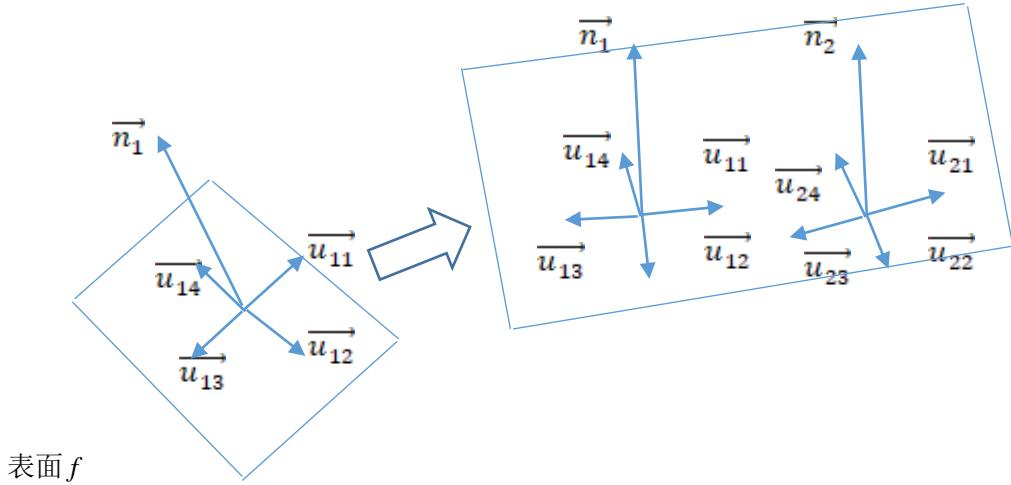


图 4-6 内部能量项转化
Figure 4-6. Internal energy item conversion

本文采用了 Ray 等人设计的平滑能量项来对方向场进行平滑，其大致思路如下^[46]：

设顶点 p_1 有法向量 \vec{n}_1 、方向场 $\vec{d}_1 = \{\vec{u}_{11}, \vec{u}_{12}, \vec{u}_{13}, \vec{u}_{14}\}$ ，顶点 p_2 有法向量 \vec{n}_2 、方向场 $\vec{d}_2 = \{\vec{u}_{21}, \vec{u}_{22}, \vec{u}_{23}, \vec{u}_{24}\}$ ，为了计算两个顶点上方向场变化的角度差，可以将顶点 p_1 上的方向场通过坐标变化，将其旋转到顶点 p_2 切平面所在的平面，然后在二维平面内计算需要旋转的角度值。

本文将旋转需要的角度值的和定义为所有顶点方向场旋转角度差的平方和，通过如下的平滑能量项 S ，优化能量项使其为最小值时，便可以得到最终网格模型的边的方向。

$$S = \sum_{P_i} \sum_{P_j} \theta(d_i, R_j d_j)^2 \quad (4-4)$$

其中，顶点 P_i 是网格中的任意顶点，顶点 P_j 是和顶点 P_i 相邻的顶点， d_i 是顶点 P_i 的方向场， R_j 是方向场 d_i 所在切平面旋转到方向场 d_j 所在切平面的旋转矩阵， θ 表示两个方向场之间的夹角，由于两个方向场都有 4 个方向，我们只需要将其方向对齐即可，故其取值范围为 0 到 90 度。

最小化能量函数的求解采用 Gauss-Seidel 迭代。在 Gauss-Seidel 迭代中，本文根据奇异点的位置和方向对场的方向进行约束，在计算顶点 P_i 处的方向场时，对 d_i 进行插值，将其更新为奇异点的方向场和原方向场的线性插值的结果，然后投影到线性空间进行归一化。

另外还有一种 Jakob 等人提出的外部能量项，和上述的能量项设计不同，在三维空间中直接计算两个方向场之间的角度^[47]。

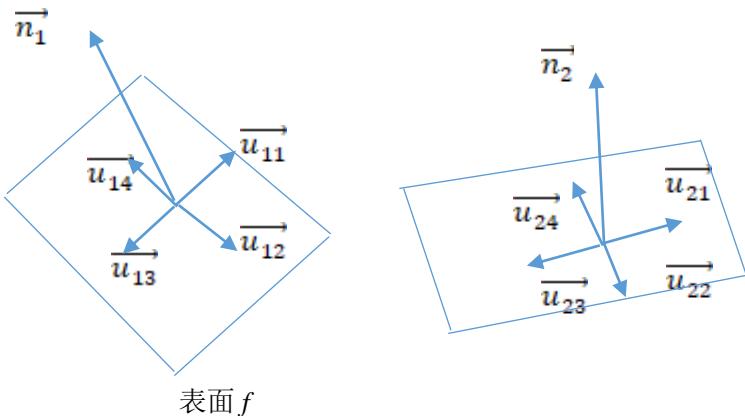


图 4-7 外部能量项

Figure 4-7. External energy item

和上述的能量项类似，外部能量项公式如下：

$$S = \sum_{P_i} \sum_{P_j} \Theta(d_i, d_j)^2 \quad (4-5)$$

其中， Θ 表示两个切平面间在三维空间中的角度差异。

在找到方向场平滑能量项的最小值后，我们需要确定半规则网格的顶点位置，和方向场的设计类似，本文为每一个顶点计算局部参数化，并定义了两种能量项来对其进行优化。

设输出网格的边长为 l ，位置场被定义为在顶点的切平面上最近的 2 维坐标点，其坐标的单位长度为 l 。本文定义顶点 p 在切平面内的所有整数表示为

$$T_p = p + l \sum_{k=0}^{s_p/2-1} t_k R_p(o, n, k) \quad (4-6)$$

其中， $p \in R^3$ 是顶点的坐标， s_p 是位置场的轴数，在本文中设定为 4， $t \in R^2$ 是在平面上各个轴的偏移量， $R_p(o, n, k)$ 是旋转后的平面，上述公式代表了所有在该平面内整数倍数的点坐标。

设网格上存在两个相邻的顶点坐标分别为 P_i 和 P_j ，如果和上述的内部方向场平滑项方法类似，也将其从 P_i 所在的切平面旋转到 P_j 所在的切平面，设 P_i 转化后的坐标为 P'_i ，那么可以将位置场上的能量项定义为两个顶点在二维平面内距离的平方和：

$$E = \sum_{P_i} \sum_{P_j} \|p_j - T_{p_i}\|^2 \quad (4-7)$$

那么 P_i 转化后的参数化顶点坐标的位置便会落在和 P_j 最近的整数坐标网格

顶点上。

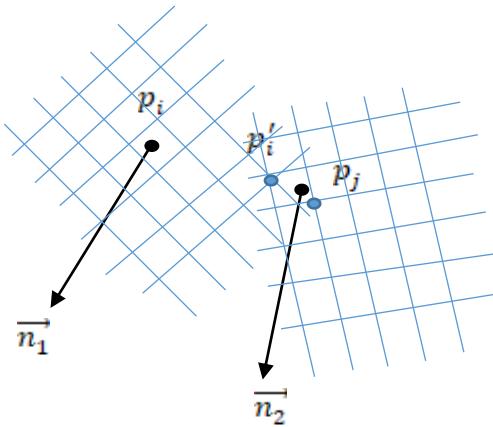


图 4-8 位置场示意图
Figure 4-8. location field diagram

如上图所示，黑色点是原位置，蓝色点是优化后位置，经过能量项的平滑，其模型顶点的优化后的位置发生改变。

类似的，如果是以外部能量项的方式来定义，那么顶点的位置通过如下的能量项来获得：

$$E = \sum_{P_i} \sum_{P_j} \|T_{p_i} - T_{p_j}\|^2 \quad (4-8)$$

其中， P_i 转化后的坐标为 P'_i ， P_j 经过转化后的坐标为 P'_j 。根据该项能量项获取到的坐标会落在离两个平面中相交最近的整数坐标网格点上。

该部分的最优化求解也是采用了 Gauss-Seidel 迭代求解法。

4.4 修改光照

在进行三维重建的过程中，由于重建算法会将建模时的光照以 RGB 信息的形式保存到贴图中，成为不可变的一部分。

而在实际应用中，需要随着场景的变换，不停地修改模型的光照信息使其更为逼真，例如在采集建模数据时存在一强光源使模型中的某一部分呈现高光，倘若在应用这一模型时环境光发生了改变，本文就无法修改模型中自带的高光。所以需要去掉模型中带有的光照信息。

本文采用了 NNAO^[51] 来实现这部分内容。NNAO 是通过神经网络来对环境光遮蔽效果进行学习的一种算法，它们的网络可以一次性计算出环境光遮蔽的近似最佳效果。下面简单介绍下 NNAO 的方法

其神经网络为一个四层的结构，其表示方式如下：

$$y = PReLU(W_n x + b_n, \alpha_n, \beta_n) \quad (4-9)$$

其中,输入参数 $x \in R^{w^2 \times 4}$ 是输入图像与 4 个 2D 纹理过滤器的点积,输出 $y \in R^4$ 是像素的遮蔽值。4 个 2D 纹理过滤器的图像如下所示:

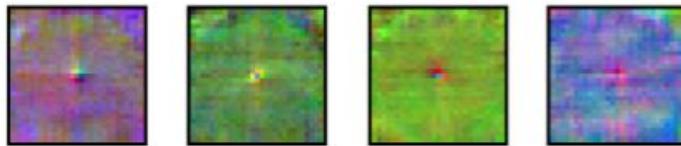


图 4-9 2D 纹理过滤器^[51]

Figure 4-9. 2D texture filter

激活函数 $PReLU()$ 是由 He 等人^[56]首次定义, 定义如下所示。

$$PReLU(x, \alpha, \beta) = \beta \max(x, 0) + \alpha \min(x, 0) \quad (4-10)$$

其网络参数如下:

$$\begin{aligned} \theta = & \{W_0 \in R^{w^2 \times 4 \times 4}, W_1 \in R^{4 \times 4}, W_2 \in R^{4 \times 4}, W_3 \in R^{4 \times 1}, \\ & b_0 \in R^4, b_1 \in R^4, b_2 \in R^4, b_3 \in R^1, \\ & \alpha_0 \in R^4, \alpha_1 \in R^4, \alpha_2 \in R^4, \alpha_3 \in R^1, \\ & \beta_0 \in R^4, \beta_1 \in R^4, \beta_2 \in R^4, \beta_3 \in R^1\} \end{aligned} \quad (4-11)$$

网络在程序中以单通道的模式渲染,首先,本文使用上述的公式计算环境光遮蔽,然后将其值反转得到对应颜色掩码如图 4-10 所示,我们以与纹理相同的方式将掩码图映射到模型上。通过上述方法,可以移除环境光对模型的影响。

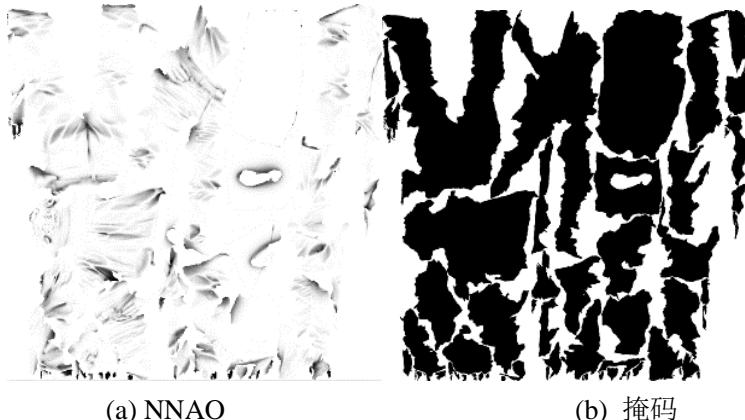


图 4-10 环境光的调整

Figure 4-10. Adjustment of ambient light

光照调整的图片如上所示,左边是 NNAO 的结果,右边是其掩码。

4.5 实验结果与分析

本文在 Instant Field-Aligned Meshes (IFM)^[47]的基础上实现了本算法。我们更新了寻找方向场的方法并且添加了调整模型环境光的功能。因此,在实验部分,本文通过观察模型来评估算法的重拓扑效果和 UV 重映射效果。本文在 2.5GHz

英特尔 i5-4200M, 8GB 内存的电脑上测试该算法。

UV 映射，一个好的 UV 映射图应当按照边和特征来分割。大多数由计算机视觉算法或激光扫描生成的模型包含过多的面，并且其 UV 映射图由混乱的面组成。

本文与 IFM 生成的网格来比较我们算法的效果。我们对四个网格模型进行处理，效果图如下所示。从左到右，从上到下分别是模型原型，原始模型的展开放效果图，IFM 的结果和本算法的结果。

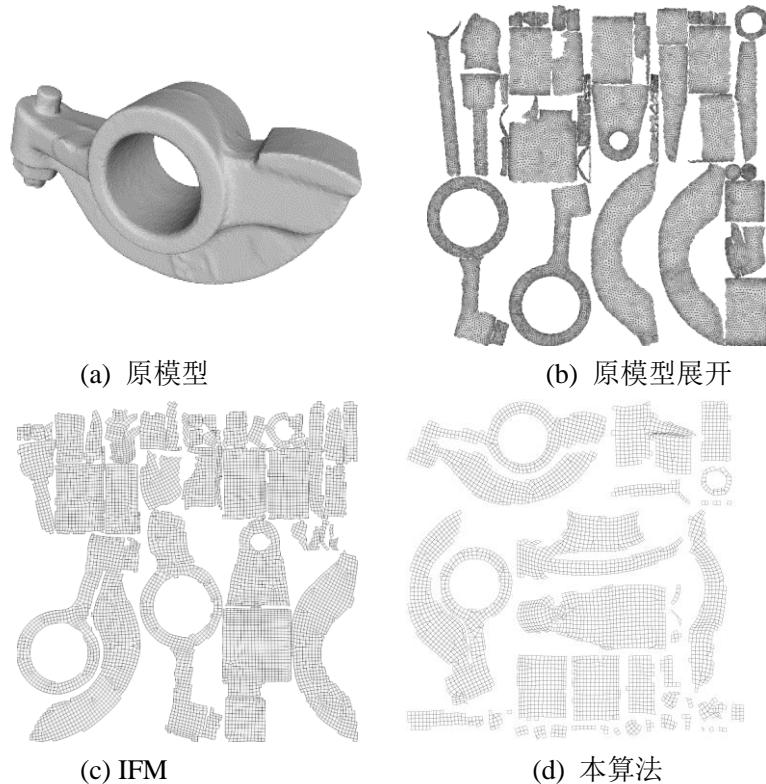
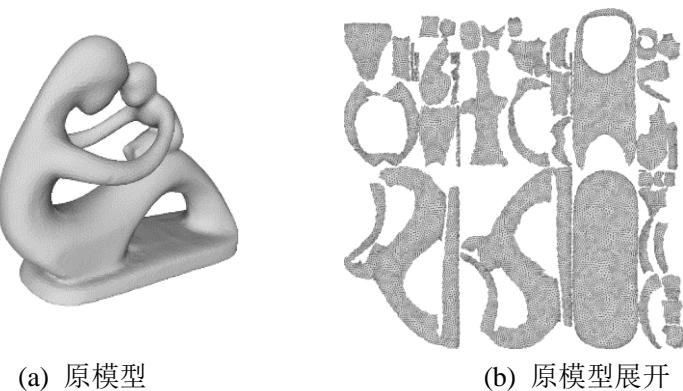


图 4-11 模型 rockerarm
Figure 4-11. Model rockerarm



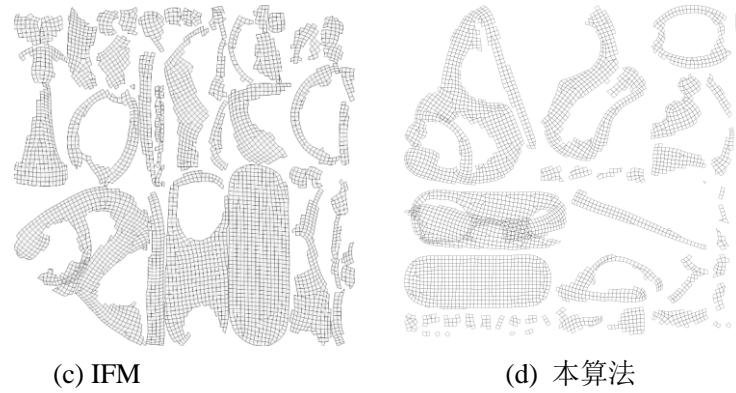


图 4-12 模型 fertility

Figure 4-12. Model fertility

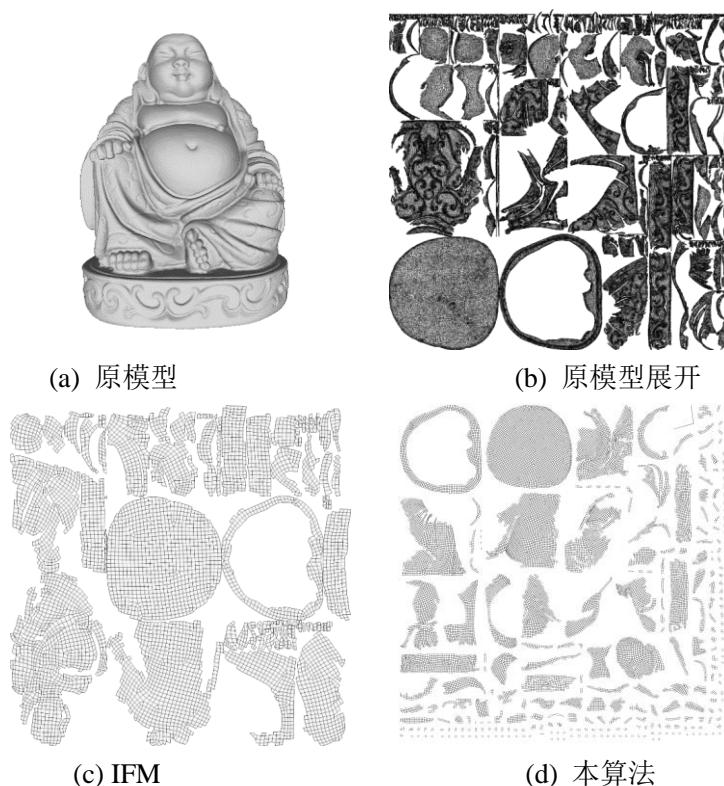
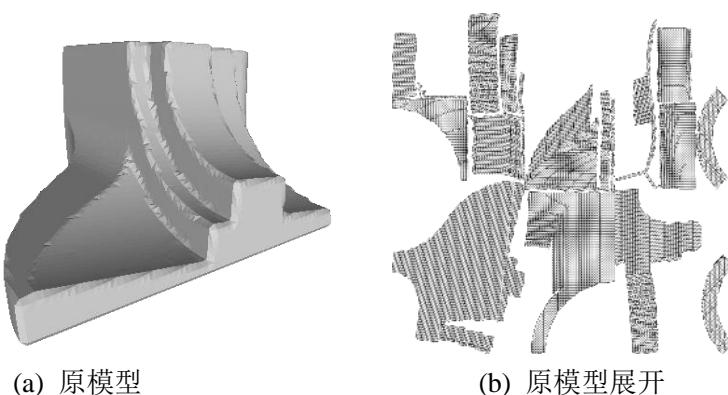


图 4-13 模型 buddha

Figure 4-13. Model buddha



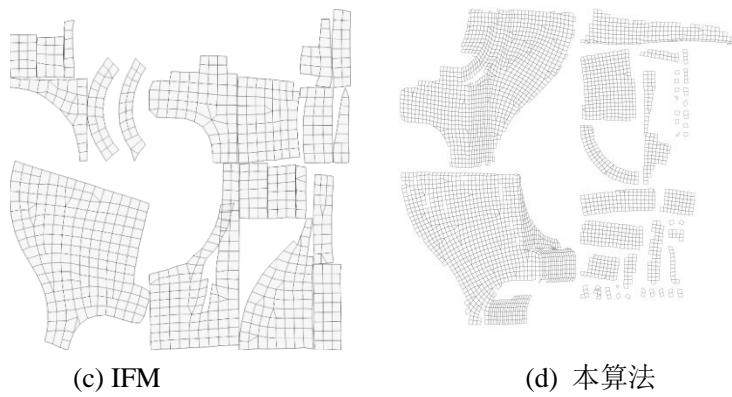


图 4-14 模型 fandisk

Figure 4-14. Model fanisk

在图中，右上角的是原始模型的展开结果，我们可以看到其面片数目较多，缺乏清晰的结构信息。左下角是用 IFM 简化后的模型，右下角是经过本算法处理后的结果。

可以明显的发现各个面片之间相互分离，并且是由半规则网格组成的，从大的面片来看，模型 rockerarm、fandisk 和 fertility 都能够相对于原本的结果能保留更多的原始模型中边的特征，对于用户来说可以更加明显的区分出其对应于原模型的哪一些区域，这允许用户对模型进行更好地编辑。

但上述仍存在一些不足之处，由于本文的算法引入了更多的奇异点数目，故在生成最后的模型时，其质量相对有些不足，其 UV 映射图中，分割的细小网格也相对应的增加了。下面对本算法的模型规整化效果进行展示。

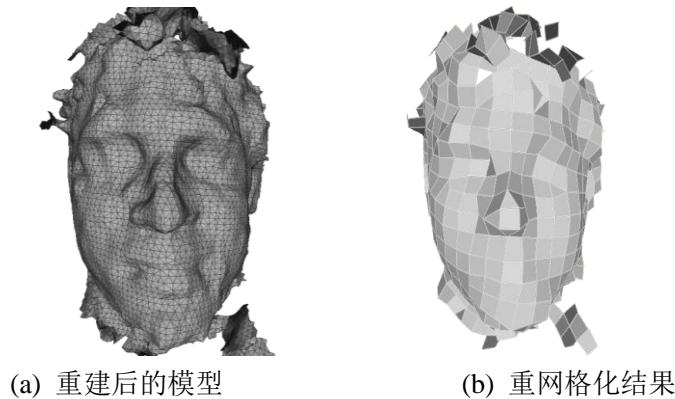
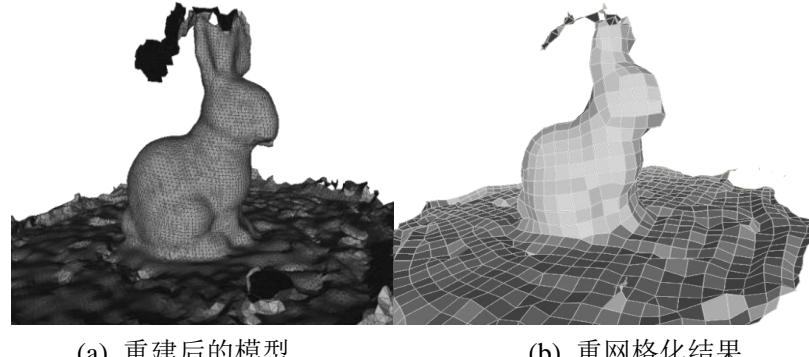


图 4-15 人头

Figure 4-15. Human head

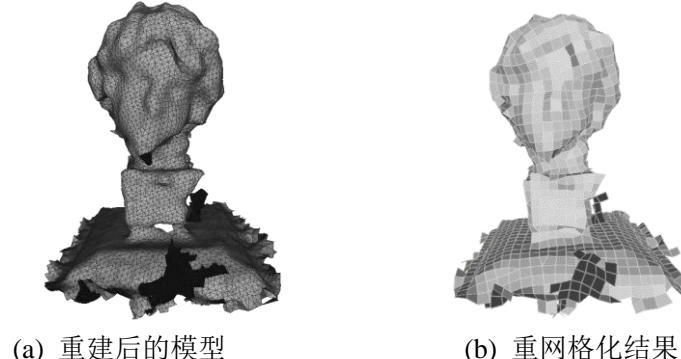


(a) 重建后的模型

(b) 重网格化结果

图 4-16 兔子

Figure 4-16. Bunny

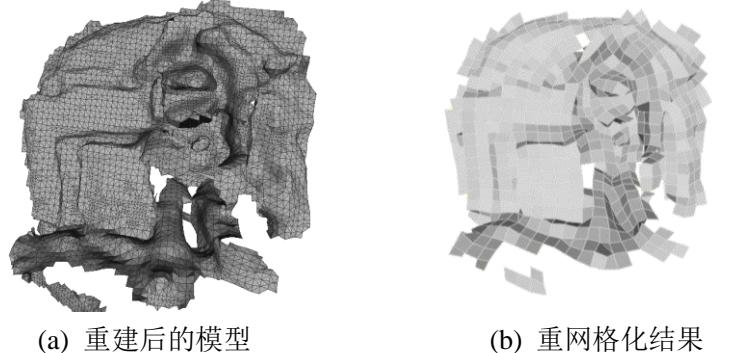


(a) 重建后的模型

(b) 重网格化结果

图 4-17 大卫

Figure 4-17. David

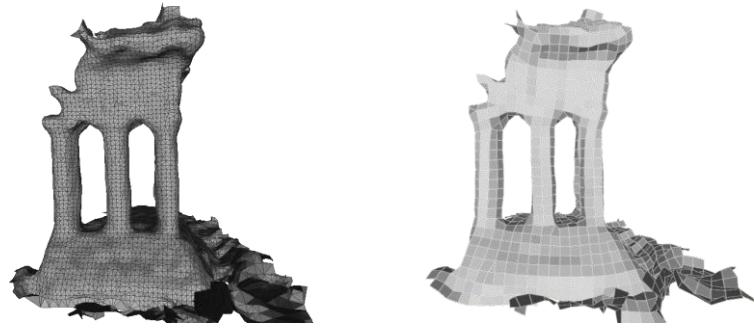


(a) 重建后的模型

(b) 重网格化结果

图 4-18 青蛙

Figure 4-18. Fog

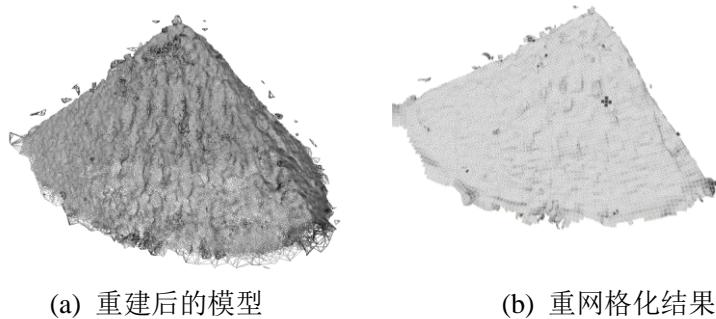


(a) 重建后的模型

(b) 重网格化结果

图 4-19 寺庙

Figure 4-19. Temple



(a) 重建后的模型
图 4-20 西瓜
Figure 4-20. Watermelon

(b) 重网格化结果

如上图所示，左边一列是第三章重建完成后的模型，模型本身的网格比较杂乱无章，面数也很多，在经过本文算法的处理之后，对模型进行重新网格化，生成的模型显著地减少了面数，并且是半规则化的。对比上述实验结果可以发现，本文算法的输出网格和原模型基本相似，在网格的几何拓扑结构上基本一致，保留了原模型的几何特点。调整环境光遮蔽的实验结果下图所示，左边是可以观察到环境光的原始的模型和纹理。当环境光照发生改变后，模型上的阴影部分难以被修改。在经过本算法的处理之后，如下图的右侧图片所示，环境光照已被移除，可以根据环境生成新的阴影。

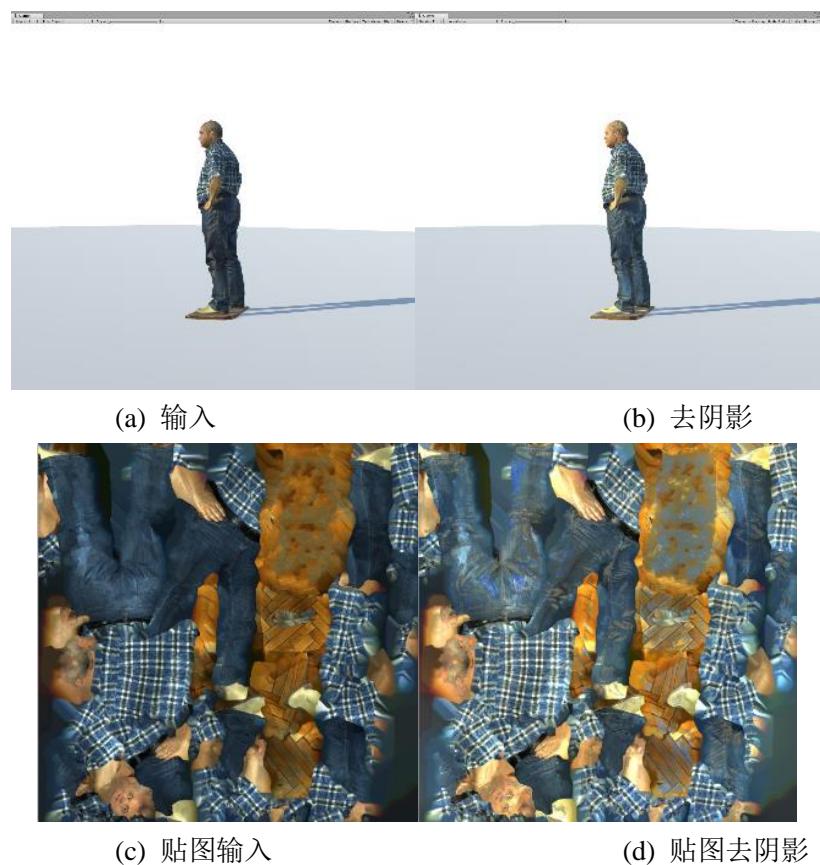


图 4-21 移除环境光的结果

Figure 4-21. Results of removing ambient light

对比左右两张图片，我们可以发现，人物模型的下巴、手部、上衣、裤子等处的阴影部分得到了减少，并且其材质本身的颜色得到了保存，这为本文算法生成的模型可以应用于不同光照场景提供了基础。图中下面两张图片是人物模型的材质贴图，左边是原材质贴图，右边是经过算法处理的材质贴图，对比其颜色和纹理，也可以验证本文算法的有效性。

4.6 本章小结

本章是对网格进行重拓扑工作，并生成了半规则化的网格，并且对模型本身的阴影进行去除。实验表明，本章的工作可以有效地生成半规则化网格，实现对三维重建结果进行重新网格化，并且去除其中的阴影，其生成网格的UV展开更加符合用户的要求。

第五章 结论与展望

5.1 结 论

随着三维模型越来越多的应用于不同的领域，三维重建系统也逐渐成为计算机视觉方向的一大研究热点。为了更好的将其应用于不同的场景中，本文主要针对三维重建系统及其应用过程中出现的一些问题，做了相关的改进工作。针对三维重建过程中的模型生成质量低的问题，本文以增量式的稀疏重建算法为研究对象，在其特征匹配阶段，融合基于网格统计的特征匹配算法和重新匹配算法，提高了模型的鲁棒性和完整性。为了使得三维重建生成的模型可以应用于不同的工业生产场景中，本文提出了基于机器学习的网格重拓扑和光照明影调整算法，本文的具体工作和研究内容如下：

(1) 研究了计算机视觉中基于多视图的三维重建算法，对现有的几类从运动恢复结构算法进行研究讨论，简要的分析了其在国内外的研究进展，并分析了各种方法的优缺点。同时，本文对重网格化技术和环境光遮蔽技术的研究现状也做了调研工作。

(2) 研究了三维重建过程中的基本算法和流程，对三维重建过程中的从图像到模型的过程进行了梳理，详细地介绍了三维重建中的理论基础，并对相关公式进行推导。同时，本文研究了重网格化技术中的理论基础。

(3) 在特征匹配中引入基于网格统计的特征匹配算法，基于运动是连续的这一特性，对相邻网格中的匹配数目进行统计，得以将错误的匹配结果过滤出去，降低了错误匹配结果对三维重建效果造成的影响，实验结果表明，它可以有效地滤除错误的匹配结果。

(4) 将重新匹配算法结合到上述的特征匹配算法中，在寻找最近匹配特征阶段，构造候选匹配特征队列，将相似程度最高的几个候选特征放入至队列中，当最匹配的特征不符合全局一致性约束和多视图几何约束时，算法验证候选匹配队列中存在的可能匹配特征点。实验结果表明了，该算法可以有效地提高特征匹配的数目，并对三维重建的结果有一定的提升。

(5) 针对模型材质难以编辑等建模后模型难以实际应用的问题，本文提出了一个三维模型自动重拓扑和 UV 重映射的方法，本文使用神经网络来自动确定奇异点位置，将模型规整化，以及移除了复杂光照环境对模型应用的影响。实验结果表明，本文方法生成的半规则网格模型能够生成可编辑的纹理材质，并且和目前最好的重网格化算法相比生成网格质量相近，在大量地减少网格面的同时，

生成的网格保留了原有的边和特征，具有较好的实际应用前景。

5.2 展望

本文实现了对原本三维重建算法效果的提升和改进，有效地生成了更加鲁棒的和完整的重建结果。同时，本文实现了三维模型自动重拓扑和 UV 重映射，生成更适合编辑的模型，同时有效地降低了网格面数，生成半规则的网格模型，为其应用于工业场景提供了可能。

尽管本文的算法对于之前算法有一定的提升效果，但仍然存在一些不足之处，后续需要进行如下工作：

(1) 在本文的实验中，我们发现，在图像角度差距较大的情况下，仍然会存在匹配失败的情况。一方面，由于较大的视线角度变化会导致物体本身会存在遮挡的情况发生，另一方面，物体本身的颜色和纹理可能因此存在变化，导致无法提取出对应的特征点，导致匹配失败的情况。针对这种问题，可以考虑之后采用不同的特征描述子或者修改特征匹配策略来进行。

(2) 在本文的实验中，我们还发现对于一些纹理较少的物体，本文的算法很难提取出足够的特征点，导致其无法生成更鲁棒的三维重建结果。针对这一问题，我们可以考虑结合深度学习技术，采用神经网络来提取特征和估计目标点的深度，实现估计出缺乏纹理物体的深度信息。

(3) 对于模型重拓扑中的模型质量问题，本算法的缺点是没有生成纯四边形网格，目前网格中存在的奇异点数目较多还可以进一步降低，可以引入面积约束来使网格顶点的分布更均匀。

参考文献

- [1] 刘旭春, 丁延辉. 三维激光扫描技术在古建筑保护中的应用[J]. 测绘工程, 2006(01): 48-49.
- [2] 王俊, 朱利. 基于图像匹配-点云融合的建筑物立面三维重建[J]. 计算机学报, 2012,35(10): 2072-2079.
- [3] 陈健堃, 彭凌霄, 仇晓兰, 丁赤飚, 吴一戎. 基于深度神经网络的 SAR 建筑目标三维重建方法[J]. 中国科学: 信息科学, 2019, 49(12): 1606-1625.
- [4] 姜慧文. 基于移动设备的实时对象三维重建[D]. 浙江大学, 2018.
- [5] 宁瑞忻, 朱尊杰, 邵碧尧, 龚冰剑, 颜成钢. 基于视觉的虚拟现实与增强现实融合技术 [J]. 科技导报, 2018, 36(09): 25-31.
- [6] 王猛, 孔繁之. 医学图像三维可视化技术及其新进展[J]. 医学影像学杂志, 2015, 25(06): 1095-1097.
- [7] 夏文玲, 顾照鹏, 杨唐胜. 实时三维重建算法的实现——基于 Kinect 与单目视觉 SLAM 的三维重建[J]. 计算机工程与应用, 2014,50(24): 199-203.
- [8] Fuhrmann S, Langguth F, Goesele M. MVE: a Multiview Reconstruction Environment[C]. //Eurographics Workshop on Graphics & Cultural Heritage. Eurographics Association, 2014.
- [9] Pollefeys M, Koch R, Gool L J V. Self-Calibration and Metric Reconstruction in Spite of Varying and Unknown Internal Camera Parameters[C]. //Proceedings of the IEEE International Conference on Computer Vision. IEEE, 1998.
- [10] Snavely N, Seitz S M, Szeliski R. Photo Tourism: Exploring Photo Collections in 3D[M]. ACM SIGGRAPH 2006 Papers. 2006: 835-846.
- [11] Wu C. Towards Linear-Time Incremental Structure from Motion[C]. //International Conference on 3D Vision. IEEE Computer Society, 2013.
- [12] Schonberger J L, Frahm J M. Structure-from-motion Revisited[C]. //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2016: 4104-4113.
- [13] Agarwal S, Furukawa Y, Snavely N, et al. Building Rome in a Day[J]. Communications of the ACM, 2011, 54(10): 105-112.
- [14] Moulon P, Monasse P, Marlet R. Adaptive Structure from Motion with a Contrario Model Estimation[C]. //Asian Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012: 257-270.
- [15] Martinec D, Pajdla T. Robust Rotation and Translation Estimation in Multiview Reconstruction[C]. //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2007: 1-8.
- [16] Enqvist O, Kahl F, Olsson C. Non-sequential Structure From Motion[C]. //Proceedings of the IEEE International Conference on Computer Vision. IEEE, 2011: 264-271.
- [17] Govindu V M. Combining Two-view Constraints for Motion Estimation[C]. //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2001: II-II.
- [18] Govindu V M. Lie-algebraic Averaging for Globally Consistent Motion Estimation[C]. //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2004: I-I.
- [19] Martinec D, Pajdla T. Robust Rotation and Translation Estimation in Multiview

- Reconstruction[C]. //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2007: 1-8.
- [20] Moulon P, Monasse P, Marlet R. Global Fusion of Relative Motions for Robust, Accurate and Scalable Structure from Motion[C]. //Proceedings of the IEEE International Conference on Computer Vision. IEEE, 2013: 3248-3255.
- [21] Jiang N, Cui Z, Tan P. A Global Linear Method for Camera Pose Registration[C]. //Proceedings of the IEEE International Conference on Computer Vision. IEEE 2013: 481-488.
- [22] Cui Z, Tan P. Global Structure-from-motion by Similarity Averaging[C]. //Proceedings of the IEEE International Conference on Computer Vision. IEEE 2015: 864-872.
- [23] Cui Z, Jiang N, Tang C, et al. Linear Global Translation Estimation with Feature Tracks[J]. ArXiv preprint arXiv:1503.01832, 2015.
- [24] Wilson K, Snavely N. Robust Global Translations with 1DSFM[C]. //European Conference on Computer Vision. Springer, Cham, 2014: 61-75.
- [25] Sweeney C, Sattler T, Hollerer T, et al. Optimizing the Viewing Graph for Structure-from-motion[C]. //Proceedings of the IEEE International Conference on Computer Vision. IEEE 2015: 801-809.
- [26] Cui H, Gao X, Shen S, et al. HSfM: Hybrid Structure-from-motion[C]. //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 1212-1221.
- [27] Zhu S, Shen T, Zhou L, et al. Parallel Structure from Motion from Local Increment to Global Averaging[J]. ArXiv preprint arXiv:1702.08601, 2017.
- [28] Farenzena M, Fusiello A, Gherardi R. Structure-and-motion Pipeline on a Hierarchical Cluster Tree[C]. //Proceedings of the IEEE International Conference on Computer Vision. IEEE 2009: 1489-1496.
- [29] Gherardi R, Farenzena M, Fusiello A. Improving the Efficiency of Hierarchical Structure-and-motion[C]. //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2010: 1594-1600.
- [30] Toldo R, Gherardi R, Farenzena M, et al. Hierarchical Structure-and-motion Recovery from Uncalibrated Images[J]. Computer Vision and Image Understanding, 2015, 140: 127-143.
- [31] Alliez P, Meyer M, Desbrun M. Interactive Geometry Remeshing[J]. ACM Transactions on Graphics, 2002, 21(3): 347-354.
- [32] Alliez P, Cohen-Steiner D, Devillers O, et al. Anisotropic Polygonal Remeshing[M]. ACM SIGGRAPH 2003 Papers. 2003: 485-493.
- [33] Marinov M, Kobbelt L. Direct Anisotropic Quad-dominant Remeshing[C]. //12th Pacific Conference on Computer Graphics and Applications. Proceedings. IEEE, 2004: 207-216.
- [34] Khodakovsky A, Litke N, Schröder P. Globally Smooth Parameterizations with Low Distortion[J]. ACM Transactions on Graphics, 2003, 22(3): 350-357.
- [35] Kälberer F, Nieser M, Polthier K. Quadcover-surface Parameterization Using Branched Coverings[C]. //Computer Graphics Forum: Blackwell Publishing Ltd, 2007, 26(3): 375-384.
- [36] Ray N, Li W C, Lévy B, et al. Periodic Global Parameterization[J]. ACM Transactions on Graphics , 2006, 25(4): 1460-1485.
- [37] Knöppel F, Crane K, Pinkall U, et al. Globally Optimal Direction Fields[J]. ACM Transactions on Graphics, 2013, 32(4): 1-10.
- [38] Aksoylu B, Khodakovsky A, Schröder P. Multilevel Solvers for Unstructured Surface Meshes[J]. SIAM Journal on Scientific Computing, 2005, 26(4): 1146-1165.
- [39] Levi Z, Zorin D. Strict Minimizers for Geometric Optimization[J]. ACM Transactions on

- Graphics, 2014, 33(6): 1-14.
- [40] Kobbelt L P, Vorsatz J, Labsik U, et al. A Shrink Wrapping Approach to Remeshing Polygonal Surfaces[C]. //Computer Graphics Forum. Blackwell Publishers Ltd, 1999, 18(3): 119-130.
- [41] Khodakovsky A, Litke N, Schröder P. Globally Smooth Parameterizations with Low Distortion[J]. ACM Transactions on Graphics, 2003, 22(3): 350-357.
- [42] Liu L, Zhang L, Xu Y, et al. A Local/Global Approach to Mesh Parameterization[C]. //Computer Graphics Forum. Blackwell Publishing Ltd, 2008, 27(5): 1495-1504.
- [43] Poranne R, Tarini M, Huber S, et al. Autocuts: Simultaneous Distortion and Cut Optimization for UV Mapping[J]. ACM Transactions on Graphics, 2017, 36(6): 1-11.
- [44] Ben-Chen M, Gotsman C, Bunin G. Conformal Flattening by Curvature Prescription and Metric Scaling[C]. //Computer Graphics Forum. Blackwell Publishing Ltd, 2008, 27(2): 449-458.
- [45] Campen M, Kobbelt L. Quad Layout Embedding via Aligned Parameterization[C]. //Computer Graphics Forum. Blackwell Publishing Ltd. 2014, 33(8): 69-81.
- [46] Ray N, Vallet B, Li W C, et al. N-symmetry Direction Field Design[J]. ACM Transactions on Graphics, 2008, 27(2): 1-13.
- [47] Jakob W, Tarini M, Panizzo D, et al. Instant Field-aligned Meshes[J]. ACM Transactions on Graphics, 2015, 34(6): 189:1-189:15.
- [48] Bommes D, Zimmer H, Kobbelt L. Mixed-integer Quadrangulation[J]. ACM Transactions on Graphics, 2009, 28(3): 1-10.
- [49] Bavoil L, Sainz M. Screen Space Ambient Occlusion[J]. NVIDIA Developer Information: Http:.Developers. Nvidia. Com, 2008, 6.
- [50] Bavoil L, Sainz M, Dimitrov R. Image-space Horizon-based Ambient Occlusion[M]. ACM SIGGRAPH 2008 talks. 2008: 1-1.
- [51] Holden D, Saito J, Komura T. Neural Network Ambient Occlusion[M]. SIGGRAPH ASIA 2016 Technical Briefs. 2016: 1-4.
- [52] 高翔. 视觉 SLAM 十四讲：从理论到实践[M]. 电子工业出版社. 2017.
- [53] Ng P C, Henikoff S. SIFT: Predicting Amino Acid Changes that Affect Protein function[J]. Nucleic Acids Research, 2003, 31(13): 3812-3814.
- [54] Alliez P , Ucelli G , Gotsman C , et al. Recent Advances in Remeshing of Surfaces[M]. Shape Analysis and Structuring. 2008.
- [55] Csurka G, Dance C R, Humenberger M. From Handcrafted to Deep Local Features[J]. ArXiv preprint arXiv:1807.10254, 2018.
- [56] He K, Zhang X, Ren S, et al. Delving Deep into Rectifiers: Surpassing Human-level Performance on Imagenet Classification[C]. // Proceedings of the IEEE International Conference on Computer Vision. IEEE, 2015: 1026-1034.

致 谢

三年的研究生生活转瞬即逝，在此想对指导过我的老师，帮助过我的同学表达诚挚的感谢！在这三年的研究生生活中，我的知识和学术水平得到了极大的提高，我的视野也得到了开阔，其中最要感谢的是我的导师张剑华副教授，是他的谆谆教导驱使我能够不断前进，教会了我如何去研究一个问题，指引了我在学习道路上的方向。

在学术论文和课题项目完成的过程中，张老师指导了我如何去发现问题，并学会如何去解决问题，感谢他在这一过程中的细心教导。在完成论文的过程中，张老师也提供了许许多多的宝贵建议，使得我的论文能够顺利的完成。

与此同时，也感谢我们计算机视觉研究所的全体同学，感谢我的师兄师姐，陈敬帛、刘儒瑜、王其超、冯缘、张少波、毛栎炳、郭建双、陈嘉伟等在学术等上给我的指导。也感谢我的同学吴佳鑫、冯宇婷、王曾媛、贵梦萍、张志飞、王豪等人，感谢他们在三年中和我相伴同窗，一起学习，一起进步。

最后，我要感谢我的父母，感谢他们这么多年来对我的支持和鼓励，感谢他们这么多年的默默付出，支持我读完这么多年书。希望他们能够身体健康，平平安安。

作者简介

1 作者简历

1995 年 07 月出生于浙江省台州市。

2013 年 09 月——2017 年 06 月，浙江工业大学计算机科学与技术学院、软件学院软件工程专业学习，攻读工学学士学位。

2017 年 09 月——2020 年 06 月，浙江工业大学计算机科学与技术学院、软件学院软件工程专业学习，攻读工学硕士学位。

2 攻读硕士学位期间发表的学术论文

- [1] Pan Z, Di M, Zhang J, Ravi S. Automatic Re-topology and UV Remapping for 3D Scanned Objects based on Neural Network[C]. //Proceedings of the 31st International Conference on Computer Animation and Social Agents. 2018: 48-52.
- [2] Zhang J, Liu J, Chen K, Pan Z, Liu R, Wang Y, Yang T, and Chen S. Map Recovery and Fusion for Collaborative AR of Multiple Mobile Devices[J]. IEEE Transactions on Industrial Informatics. May, 2020. Accepted.

3 参与的科研项目及获奖情况

- [1] 张剑华, 刘盛, 王振华, 郭东岩, 刘儒瑜, 王其超, 吴佳鑫, 潘志颖, 王曾媛, 贵梦萍. 动态视觉认知子空间构建及应用, 国家自然科学基金项目. 2019-2022.

学位论文数据集

密 级*	中图分类号*	UDC*	论文资助
公开	TP391	004	
学位授予单位名称*	学位授予单位代码	学位类型*	学位级别*
浙江工业大学	10037	学术硕士	硕士
论文题名*	基于多视图的三维重建及其模型重拓扑		
关键词*	增量式，三维重建，特征点匹配，从运动恢复结构，模型重拓扑		论文语种*
并列题名			中文
作者姓名*	潘志颖	学 号*	2111712091
培养单位名称*	培养单位代码*	培养单位地址*	邮政编码*
浙江工业大学计算机科学与技术学院	10037	杭州市潮王路 18 号	310032
学科专业*	研究方向*	学 制*	学位授予年*
软件工程	计算机视觉	3	2020
论文提交日期*	2020 年 6 月		
导师姓名*	张剑华	职 称*	副教授
评阅人	答辩委员会主席*	答辩委员会成员	
	梁荣华		
电子版论文提交格式：文本(<input type="checkbox"/>)图像(<input type="checkbox"/>)视频(<input type="checkbox"/>)音频(<input type="checkbox"/>)多媒体(<input type="checkbox"/>)其他(<input type="checkbox"/>)			
电子版论文出版(发布)者	电子版论文出版(发布)地		版权声明
论文总页数*	66		
注：共 33 项，其中带*为必填数据，为 25 项。			