

学号 200631610274

密级

# 武汉大学本科毕业论文

## 基于单视影像的三维重建技术研究

院（系）名 称：测绘学院

专 业 名 称 ： 测绘工程

学 生 姓 名 ： 冯 晨

指 导 教 师 ： 邓 非      副教授

二〇一〇年六月

# **BACHELOR'S DEGREE THESIS OF WUHAN UNIVERSITY**

## **Research On Single View Reconstruction**

College : School of Geodesy and Geomatics

Subject : Geomatics

Name : Chen Feng

Directed by : Fei Deng, Associate Professor

June , 2010

# 郑 重 声 明

本人呈交的学位论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本学位论文的知识产权归属于培养单位。

本人签名： 冯晨

日期： 2010-06-09

# 摘 要

本论文主要研究了通过人工环境的单张影像重建场景模型的相关算法。

首先，本文指出了相机模拟在本研究中的重要意义——验证算法以及数值实验。为此研究了相机模型在摄影测量、计算机视觉以及计算机图形学中的不同表现形式，并且推导了模型参数的转换公式。以此为基础实现了相机模拟程序，并且将其应用于真实照片与虚拟场景的联合显示。

其次，本文研究了相机自标定。通过对本研究目的的分析指出，对于单像建模而言，在所有标定算法中，灭点标定方法是最为适用的。在推导完灭点标定的理论公式后，给出了一个极为直观的几何解释。为了验证并分析灭点标定算法，结合相机模拟程序，提出了一套对该算法进行模拟数值分析的算法。该算法显示单像灭点标定稳定性不高。接着介绍了一种新近提出的稳健估计算法 J-linkage，对其在灭点标定中的应用进行了实验，并指出，由于灭点标定本身的病态性质，即使使用稳健估计，其标定结果仍然属于弱标定。

最后，本文分别研究了两种单像建模算法。在将单像建模抽象化描述为含约束的最优化问题之后，首先详细研究了各种几何约束及其数学表达形式。受此启发提出一种序贯式重建方法，并且在实验后指出其缺点。为克服此缺点，本研究分别修改了 Sturm 和 Grossmann 的算法，提出了整体式重建算法，并再次结合相机模拟程序进行数值模拟实验，给出了一个很有意义的结论：像点量测误差虽然对于灭点标定的影响很大，但是对于整体式单像建模的重建结果的影响却相对较小，故经过弱标定的图像即可用于单像重建。实验表明，改进后整体式算法的解算精度要高于原始算法。

**关键词：**单像建模；灭点标定；稳健估计；J-linkage；相机模拟

# ABSTRACT

This paper explores single view reconstruction of man-made environments.

In the first place, this paper points out the significance of camera simulation in this research--for algorithms tests and numerical experiments. After an introduction of camera models' different appearances in Photogrammetry, Computer Vision and Computer Graphics, a series of transform equations of the models' parameters between those different appearances are given. On the basis of these, a camera simulation software is implemented and applied in the display of photos in virtual scenes.

After this, the camera self calibration algorithms are studied. Considering the purpose of this research, this paper claims that calibration from vanishing points is the most appropriate calibration approach. A geometric interpretation of the algorithm is given after introducing its basic idea and equations for solving the camera intrinsic. Along with camera simulation, this paper proposed a series of algorithms to numerically simulate the above calibration approach and analyze its robustness, which shows that calibrating the camera from a single view is not robust. Then a newly proposed robust estimation algorithm called J-linkage is introduced and experiments of its application in vanishing point detection shows that because of the illness of vanishing point estimation, even if the robust estimation technique is applied, it is still a weak calibration.

Then we come to the core of this paper which focuses on single view reconstruction (SVR). Since firstly the problem is abstracted as an optimization problem with constraints, several geometric constraints and their respective math representations are carefully examined, and a sequential SVR algorithm is heuristically proposed, tested and its natural short comings are pointed out. An improved algorithms is proposed and tested numerically, leading to a very meaningful conclusion: the influence of errors in image points to the vanishing point calibration is much bigger than that to the improved SVR algorithm, meaning weak calibrated images could be used for SVR. Experiments also proved that the improved algorithm's is better than the original one.

**Key words:** Single View Reconstruction; Vanishing Point Calibration; Camera Simulation; Robust Estimation; J-linkage

# 目 录

第 1 章	绪论 .....	1
1.1	研究目的 .....	1
1.2	研究现状 .....	2
1.3	论文结构 .....	6
第 2 章	相机模拟 .....	8
2.1	概述 .....	8
2.2	相机模型 .....	9
2.2.1	摄影测量 .....	9
2.2.2	计算机视觉 .....	10
2.2.3	计算机图形学 .....	11
2.3	模型参数转换 .....	13
2.3.1	摄影测量与计算机视觉 .....	14
2.3.2	计算机视觉与计算机图形学 .....	15
2.4	模拟相机实现 .....	16
2.5	本章小结 .....	16
第 3 章	相机自标定 .....	17
3.1	概述 .....	17
3.2	灭点标定 .....	18
3.2.1	理论推导 .....	19
3.2.2	几何解释 .....	20
3.2.3	实验结果 .....	21
3.3	灭点检测 .....	25
3.3.1	多实例模型稳健估计——J-linkage .....	27
3.3.2	J-linkage 在灭点检测中的应用 .....	31
3.3.3	实验结果 .....	32
3.4	本章小结 .....	33
第 4 章	单像建模 .....	34
4.1	概述 .....	34
4.2	序贯式重建 .....	35

4.2.1	几何约束 .....	36
4.2.2	实验结果 .....	38
4.3	整体式重建 .....	40
4.3.1	基于 Sturm 算法 .....	40
4.3.2	基于 Grossmann 算法 .....	43
4.3.3	实验结果 .....	45
4.4	本章小结 .....	49
第 5 章	总结与展望 .....	50
参考文献	.....	52
本科期间发表的论文和科研情况	.....	57
致谢	.....	59

# 第1章 绪论

从二维影像获得三维模型，这是摄影测量学和计算机视觉的基本任务；从三维模型产生二维影像，这是计算机图形学的基本任务。三者结合，学科交叉，产生了基于影像的建模（Image Based Modeling, IBM）这样一个前沿的研究领域，它使得用户可以利用输入的影像，在不同于原影像视角的新视角下对场景进行观察。本文所要探讨的基于单视影像的三维重建技术，就是该领域中的一个热门话题。

## 1.1 研究目的

在众多应用领域（如古建筑与古文物的三维重建、数字城市、生物医学和工业生产）中，用户经常需要原始场景的相关三维信息，使得人们可以通过影像来量测三维物体。同样，为了获得室内或室外建筑的虚拟三维模型，也需要这些信息。众多不同类型的用户，如城市设计规划师、建筑师、房地产公司、电影制作商、电子/电脑游戏制作商、三维地理信息系统数据提供商等等，都会对这种方式建立虚拟模型的方式感兴趣。

计算机视觉和摄影测量学的一个基本任务，就是从影像获取物方的三维信息。根据其基本理论，拍摄照片是一个将物方的三维信息映射到照片二维平面的过程，由于这是一个多对一的映射，因此必然存在着信息的损失，即深度信息的丢失。为了通过照片恢复出物体的三维坐标，传统的摄影测量学和计算机视觉研究表明，需要通过至少两幅或更多幅影像才能达到实现。

然而经常会出现这样的情况：用户只有人工场景的单张影像，比如希望研究城市历史的学者所提供的某栋早已不存在的大楼的一张老照片，或者是文艺复兴时期以后某位画家绘制的一副古建筑的油画。此时多像建模方法就无法发挥作用了。

所幸的是，实践经验告诉我们，很多情况下，即使人眼只观察一张照片，或者只用一只眼睛观察周围的环境，我们还是能够迅速的在大脑中恢复出物体的形状（图 1.1）。这就启发我们，通过单张影像进行三维重建是可能的。

因此，为了能够方便的从用户提供的单张相片，或者是满足中心投影的图画，重建出场景的三维模型，以作为各种虚拟现实、增强现实应用的数据来源，有必要研究单视影像的三维重建技术。





(a) 武汉大学信息学部，主教学楼      (b) 密歇根大学安娜堡分校，Rackham 大楼

图 1.1 仅观察单张影像亦可迅速在大脑中恢复物体（尤其是建筑物）的形状。

## 1.2 研究现状

纵观基于影像进行建模的各种方法，可以分为以下几类：基于多视影像、基于主动视觉和基于单视影像。

基于多张影像建模的方法主要有<sup>[1]</sup>：

立体视觉。其基本原理是从两个（或多个）视点观察同一目标，以获取在不同视角下的感知影像，通过计算同名像点坐标的位置偏差（即视差），然后利用三角测量的原理来获取景物的三维信息，这一过程与人类双目视觉的立体感知过程是类似的。立体视觉是一种被动获取三维信息的手段。如何自动实现在不同视角图像上寻找同名点（影像匹配）是立体视觉的核心问题所在。

运动图像序列法。根据由图像中像点的运动场和光流场求解目标的三维信息。当目标与摄影机发生相对运动时，图像中的像点坐标也发生了变化，每个点的变化包含了目标和摄影机相对运动的信息，所有像点坐标的变化矢量构成运动场；同时像点的亮度也发生变化，所有像点亮度的变化矢量构成光流场。运动场和光流场包含了目标点之间的空间关系及其摄影机相对于目标之间的运动关系。

基于主动视觉的建模方法主要为结构光方法<sup>[1]</sup>。由已知空间数学模型的投影光线、投影平面、投影曲面及投影光束和记录光线的影像组成。相对于立体视觉而言，结构光测量系统仍然由两个视觉通过三角测量的原理获取目标的三维信息，但是其中一个视觉是生成结构光，另外一个视觉是记录结构光。根据生成的结构光在目标上形成的纹理，可以分为点、线和面结构光。结构光测量属于主动视觉的手段，结构光测量特别适合于缺乏纹理或无纹理的目标。由于已知一个视觉的方向，故只需要在其影像上提取结构光纹理的成像就可以获取目标表面的信息。

研究人员就结构光的生成设备，生成结构光的类型，如何对结构光影像进行处理从而实现目标的三维重建进行了广泛而深入的研究，并在隧道工程、工业器件、飞机及船体等大型工业目标的应用中取得了良好的效果。

以上两类是较为传统的三维重建方法，然而结合心理学、神经生理学的研究人们发现，人脑可以提取所观测影像中的光照、阴影、纹理和几何结构等信息，从而帮助理解场景。从这几种信息出发，则产生了几类不同的利用单张影像恢复场景三维信息的方法<sup>[2]</sup>，对应于传统的多像三维重建（Multiple View Reconstruction），这种方法被称之为单视影像三维重建（SVR，Single View Reconstruction）：

阴影恢复形状法（Shape from shadow）。利用单幅图像中目标表面的明暗变化来恢复表面各点的相对高度或表面法向量等参数值。对于实际图像，目标表面的亮度可能受到多种因素的影响，光源参数、物体表面材料的性质和形状、摄影机相对于目标的位置关系。

纹理恢复形状法（Shape from texture）。借助目标表面的纹理进而恢复表面的形状。在获取图像的透视投影的过程中，目标原始的纹理结构会发生变化，这种变化随纹理所在表面的朝向而变，因而图像带有目标的方向信息。

光度立体学方法（Shape from shading）。利用影像中目标的亮度恢复其三维模型的方法，是在不改变摄影机与目标相对位置的情况下，利用不同的光照条件得到多幅图像。由于每幅图像中各点的亮度是光源向量、目标表面法向量及其表面反射系数的函数，故可以利用多幅影像之间的亮度关系获取目标的三维信息。

基于几何结构信息的方法。该类方法是目前单视影像三维方法中被研究得最深入，重建结果的视觉效果最好的。下文我们均以单像建模作为这种方法的简称。

这种完全基于几何结构信息的单像建模方法的核心思想在于这样一个事实，当观察图 1.1 这种结构化场景的单张影像时，我们一般都会对所观察物体的形状有大致的把握，但是却不能精确确定场景的度量信息，例如某个夹角或是建筑物长宽的比例。然而通过几何理论，我们就可以将二维观测量与三维点联系起来，再利用代数工具将问题进行定量化重述，辅以参数估计，最终就可以从单张影像计算出原始场景的三维信息。这里所谓的“结构化场景”指的是满足一定规则几何性质（例如共面，平行，垂直）的三维点集所构成的场景。

最早开始利用几何结构信息进行场景三维重建的是来自于计算机图形学领域的 Youichi Horry 等人于 1997 年提出的“Tour Into the Picture (TIP)”技术<sup>[3]</sup>。算法

针对于包含单个灭点的图像，检测到灭点后，利用灭点将场景分割为所谓的蛛网模型（Spidery Mesh），结合前景-背景检测算法，将检测得到的前景物体至于蛛网中的相应位置，从而得到一个近似的三维重建结果（算法流程见图 1.2）。与此方法类似的还有研究基于影像建模（IBM，Image-Based Modeling）的 E. Guillou 等人的单像近似三维建模方法<sup>[4]</sup>。但是这类算法由于其对场景做了过强的假设（单灭点透视，立方体建模），从而限制了其应用。

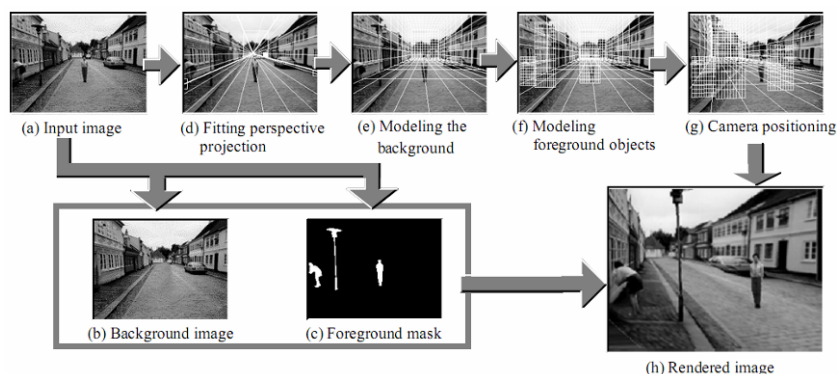


图 1.2 TIP 算法流程<sup>[3]</sup>

与此同时，英国牛津大学机器人实验室视觉几何研究组<sup>1</sup>的 Antonio Criminisi 开始了一系列的单像量测<sup>[5-7]</sup>的相关研究，并于 2000 年正式提出所谓单像量测（SVM，Single View Metrology）的概念<sup>[8-9]</sup>。所谓单像量测，意指通过单张影像上的几何信息（尤其是灭点、灭线等）进行测量。以此为基础，该实验室的 David Liebowitz 进行了一系列单、多像三维重建方面的研究<sup>[10-12]</sup>，特别是相机自标定理论。他们的研究引入了射影几何的相关理论，并利用代数射影几何进行相关的误差分析，为后来的单像建模奠定了较为坚实的数学理论基础。

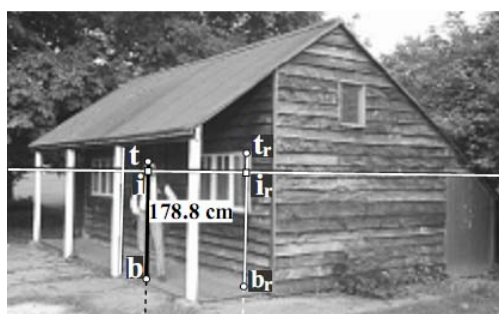


图 1.3 通过单张影像中的参考直线 $t_r b_r$ 的长度测得人的身高为 178.8cm，真实升高为 180cm<sup>[8]</sup>。

<sup>1</sup> <http://www.robots.ox.ac.uk/~vgg/>

同一时期研究单像建模的还有法国国家信息与自动化研究所 (INRIA) 的 Peter F. Sturm<sup>1</sup>。他所提出的交互式重建分段平面物体的单像建模方法<sup>[13]</sup>能够在对相片进行灭点标定之后, 利用场景几何约束 (平行, 垂直, 共面) 进行整体解算, 这使得其方法相对而言最为简便, 而重建所得的视觉效果也令人满意。并且他还进一步研究了利用单张全景影像进行三维重建的方法<sup>[14]</sup>。其后, 他的学生 Marta Wilczkowiak 又提出了利用平行四面体进行单像相机标定和三维重建的方法<sup>[15]</sup>。Sturm 的方法后来被加以改进, 利用含非线性约束的优化方法对重建后的场景进行优化, 以进一步提高视觉效果<sup>[16]</sup>。

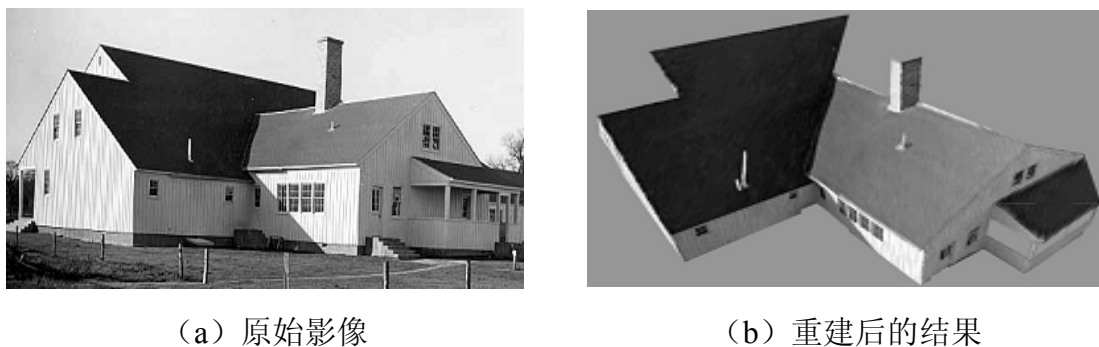


图 1.4 Peter F. Sturm 的交互式分段平面物体的单像建模方法<sup>[13]</sup>

葡萄牙 Lisbon Tech. University 的 Etienne G. Grossmann<sup>2</sup>专门针对单像建模进行研究<sup>[17-21]</sup>。除了利用上述平行、共面、垂直等条件, 他研究了新的几何约束, 如有关某平面的对称性。而且, 他将满足这些约束的点进行了参数化, 从而可以放入最大似然估计、最小二乘估计等框架中进行求解。

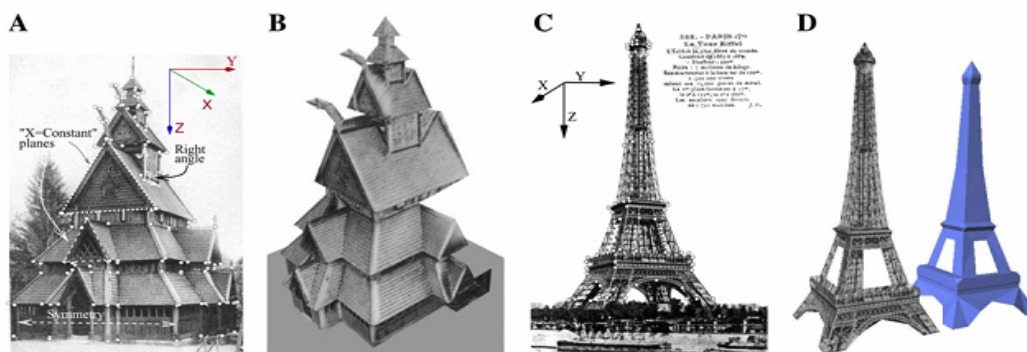


图 1.5 E. G. Grossmann 的单像建模方法<sup>[20]</sup>

此外, 与 P. E. Debevec 的多像建模<sup>[22]</sup>中所用方法类似, David Jelineck 等人从

<sup>1</sup> <http://perception.inrialpes.fr/people/Sturm/>

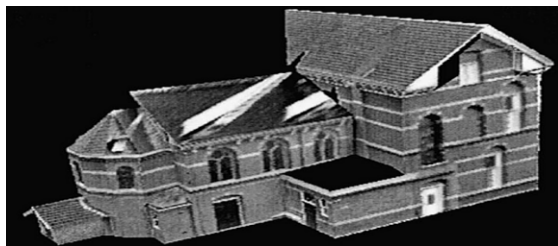
<sup>2</sup> <http://users.isr.ist.utl.pt/~etienne/>

参数化的模型入手进行单像重建<sup>[23]</sup>，提供了一类不同的解决方案。

不仅仅是计算机视觉和计算机图形学的研究者，摄影测量学者也进行了单像建模的研究。荷兰 Delft 大学的 Frank van den Heuvel 从基于线的摄影测量（Line Photogrammetry）出发，研究建筑物摄影测量。他所提出的单像建模方法区别于上述各种基于像点测量的方法，其用户测量的是影像直线段而非像点<sup>[24]</sup>。



(a) 原始影像+用户量测的线段



(b) 重建后的结果

图 1.6 Heuvel 的基于线段测量的单像建模方法<sup>[24]</sup>

以上方法的共同特点是，被重建的场景要求满足分段平面的条件，需要用户交互（并非全自动算法）。为了重建任意自由曲面，有计算机图形学界的研究者提出，利用用户指定表面向量进行单像重建<sup>[25]</sup>；为了达到全自动重建，卡耐基梅隆大学的 Hoiem 等学者从机器学习的角度出发，提出了一系列诸如基于贝叶斯方法、基于马尔可夫随机场的统计推断单像重建方法<sup>[26]</sup>。然而，这些方法的重建效果均差强人意，目前尚无法达到前述基于纯几何信息重建方法的效果。

### 1.3 论文结构

本论文所要达到的研究结果是，在用户给定的单张影像上进行一定的交互式量测，利用场景的先验几何约束，重建出场景的三维模型，以供虚拟现实应用。

全文包含以下几个假设作为前提：

1. 影像不存在径向畸变；
2. 影像是结构化的分段平面场景的影像；
3. 在影像上可以找到三组相互垂直的直线方向；

第一个假设可以通过影像纠正满足。而根据大量的日常生活经验可知，后两个假设在人工场景中，尤其是建筑场景，也是很容易满足的。

全文主要内容分为以下三个章节：

第二章相机模拟，介绍相机模型在摄影测量、计算机视觉和计算机图形学中的不同表现形式，并推导其相互转化的关系，指出相机模拟在本研究中的重要意义；

第三章相机自标定，推导灭点标定的理论公式，并给出其直观的几何解释；介绍一种新的多实例模型稳健估计算法 **J-linkage**，以及其在灭点检测中的应用，给出相关的实验结果和分析；

第四章单像建模，研究物方几何约束及其数学表达，提出序贯式重建算法；根据实验指出其缺陷，结合现有研究成果修改该算法，并给出实验结果和相关分析。



## 第2章 相机模拟

如果学习只在于模仿，那么我们就不会有科学，也不会有技术。

——高尔基

### 2.1 概述

所谓相机模拟 (Camera Simulation)，是指利用一定的相机模型，给定相应的相机参数以及已知的物方对象，在计算机中模拟照相，获得像方对象的整个过程。其中，“相机模型”指的是一个数学函数，将物方对象映射成像方对象；“对象”指的是各种三维或者二维的几何元素，例如点、线与面。

在本研究中，相机模拟具有重要意义。首先，由于已知模拟相机的各种参数、物方和像方对象，我们可以利用这些准确的信息来检验算法的正确性。例如，利用物方和像方对象可以检验第二章中的灭点标定算法，而其后的基于点和基于线的单像重建算法则可以利用像方对象和相机参数进行检验。其次，通过给这些准确信息（比如像方对象）加入人为的不同程度的随机误差（噪声），则可以测试算法的稳定性。实际上，正是由于相机模拟具有这些重要作用，不仅仅是单像重建，在计算机视觉和摄影测量的各类研究中，相机模拟得到了大量应用<sup>[27-28]</sup>。

一般来说，存在两类相机模拟的方法：数值模拟法和可视化模拟法。数值模拟法是指直接根据相机数学模型和物方对象，通过数值计算得到像方对象。这种方法实现简单，但是由于整个过程没有涉及任何图形图像操作，使得模拟不够直观。尤其是设置相机位置以及姿态的时候，很容易导致所得像方对象不在相片像幅之内，抑或物方对象在相片背面，从而得不到像方对象。而可视化模拟法则利用计算机图形学技术，通过图形图像编程（如 OpenGL/OpenSceneGraph 或者 Direct3D），利用其内部投影函数与针孔相机模型的等效关系，不但可以同数值模拟法一样得到像方对象，还可以利用图形用户界面响应用户交互事件，可视化地动态设置相机位置与姿态，并且获得模拟照片。其整个过程实时、动态、可视化地模拟了真实世界中拍摄照片的过程，十分便于后续研究，因此本研究中就采用 OpenSceneGraph 图形编程的方法实现了可视化相机模拟。

本章分以下三节阐述相机模拟过程：第 2.2 节详细讨论了摄影测量、计算机视

觉以及计算机图形学中的相机模型；第 2.3 节基于三种相机模型，推导了其相机参数相互转换的公式；第 2.4 节则根据前两节中的理论模型，说明了如何在 OpenSceneGraph 中实现一个相机模拟系统。

## 2.2 相机模型

如概述中所言，相机模型实际上是对实际成像过程的一种数学抽象，是用数学语言描述相机如何将物方对象映射到像方对象，即函数  $f: \mathbb{R}^3 \rightarrow \mathbb{R}^2$ ，其中  $\mathbb{R}^3$  和  $\mathbb{R}^2$  分别表示三维和二维欧氏空间。本研究中主要使用普通相机，一般用针孔成像模型来描述其成像物理过程，即物方点与相机中心连线与像平面的交点为像点。

由于不同学者采用的数学工具不同，对于坐标系的定义也不一样，这导致相机模型  $f$  的具体表现形式不尽相同，因此相应参数的表现形式亦不相同。以下分别从摄影测量、计算机视觉和计算机图形学三大流派阐述其各自使用的相机数学模型。

### 2.2.1 摄影测量

摄影测量学派是最早进行相机模型研究的，其世界观基于欧式几何，即其建立模型时，物方与像方对象都是在欧式空间中进行描述的。

由图 2.1 (a)，我们定义如下几个坐标系：

- $(X, Y, Z)$  地面摄影测量坐标系
- $(u, v, w)$  像空间辅助坐标系
- $(x, y, -f)$  像空间坐标系
- $(x, y)$  像平面坐标系

值得注意的是，摄影测量中的像平面坐标系是平面右手系（即从  $x$  轴正向逆时针旋转 90 度为  $y$  轴正向），而数字影像的图像坐标系为平面左手系。

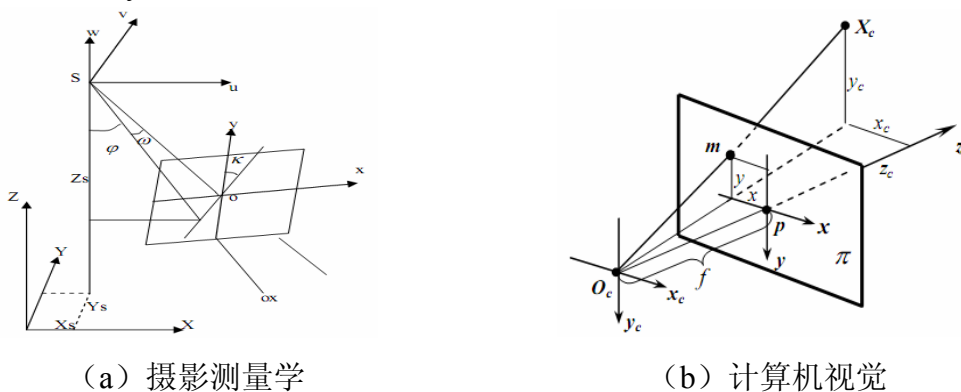


图 2.1 摄影测量和计算机视觉的相机模型坐标系



一个物方点 $(X, Y, Z)$ 通过如下方式被映射到像方点 $(x, y)$ :

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \rightarrow \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} X - X_s \\ Y - Y_s \\ Z - Z_s \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ -f \end{pmatrix} = R_{ph} \begin{pmatrix} u \\ v \\ w \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \end{pmatrix},$$

其中 $f$ 表示相机主距,  $(X_s, Y_s, Z_s)$ 表示相机中心在地面摄影测量坐标系中的坐标(也称为外方位线元素); 而根据图 2.1 (a) 中三个角度 $\kappa, \omega, \varphi$ (也称为外方位角元素)所计算得到的旋转矩阵表示如下, 其各个元素的计算公式可在文献<sup>[29]</sup>中找到:

$$R_{ph} = \begin{pmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{pmatrix} \quad (2.1)$$

综合上述变换关系, 并且考虑到像主点的偏移 $(x_0, y_0)$ , 我们得到摄影测量中最重要的概念, 实际上也就是摄影测量的相机模型, 即共线方程式<sup>[30]</sup>:

$$\left. \begin{aligned} x - x_0 &= -f \frac{a_1(X - X_s) + b_1(Y - Y_s) + c_1(Z - Z_s)}{a_3(X - X_s) + b_3(Y - Y_s) + c_3(Z - Z_s)} \\ y - y_0 &= -f \frac{a_2(X - X_s) + b_2(Y - Y_s) + c_2(Z - Z_s)}{a_3(X - X_s) + b_3(Y - Y_s) + c_3(Z - Z_s)} \end{aligned} \right\}. \quad (2.2)$$

### 2.2.2 计算机视觉

与摄影测量学派不同, 计算机视觉学者们将自己的视觉世界建立在射影几何的基础上, 即利用齐次坐标(Homogeneous Coordinate)来表达射影几何对象。如一个二维射影空间 $\mathbb{P}^2$ 中的点 $\mathbf{x} = (x_1, x_2, x_3)^T$ , 当 $x_3 \neq 0$ 时表示二维欧式空间 $\mathbb{R}^2$ 中的点 $(x_1/x_3, x_2/x_3)^T$ , 否则表示 $\mathbb{P}^2$ 中的理想点(亦即无穷远点), 三维空间可依此类推<sup>[31]</sup>。

由图 2.1 (b), 有如下坐标系:

$$\begin{aligned} (X, Y, Z) & \quad \text{世界坐标系} \\ (x_c, y_c, z_c) & \quad \text{摄像机坐标系} \\ (x, y) & \quad \text{像平面坐标系} \end{aligned}$$

值得注意的是, 与第 2.2.1 节中所述不同, 计算机视觉中的像平面坐标系与数字影像图像坐标系一致, 均为平面左手系(即从  $x$  轴正向顺时针旋转 90 度为  $y$  轴正向)。

一个物方点 $(X, Y, Z, 1)$ 通过如下方式被映射成像方点 $(x, y, 1)$ :

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = R_{cv}(I, -C_{cv}) \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \rightarrow s \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = K \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix},$$

其中,  $R_{cv}$ 表示旋转矩阵(可以采用欧拉角或四元数<sup>[32]</sup>等多种方法计算),  $I$ 表示单位阵,  $C_{cv}$ 表示相机中心在三维欧式空间中的坐标,  $s$ 表示非零比例因子,  $K$ 表示相

机内参数构成的矩阵：

$$K = \begin{bmatrix} \alpha_x & \gamma & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix},$$

其中 $\alpha_x, \alpha_y$ 表示相机主距， $\gamma$ 表示像平面 x 轴和 y 轴的不垂直因子。

同样的，将上述过程进行综合，我们可以得到计算机视觉中的相机模型：

$$\mathbf{x} = K R_{cv}(I, -C_{cv})\mathbf{X} = P\mathbf{X}, \quad (2.3)$$

而计算机视觉中著名的相机矩阵即为公式(2.3)中的

$$P = K R_{cv}(I, -C_{cv}) = K(R_{cv}, T) \quad (2.4)$$

其中平移向量 $T = -R_{cv}C_{cv}$ ，也经常作为相机中心的另一种表达方式。

### 2.2.3 计算机图形学

虽然计算机图形学与计算机视觉一样也采用齐次坐标来表达空间对象，但是与计算机视觉和摄影测量都不同的是，计算机图形学研究的是如何将已有的三维信息表达为二维信息，这就导致其相机模型具有不同于前两者的特殊性。以下我们以 OpenGL（Open Graphics Library）为例来阐述其相机模型建立的过程。

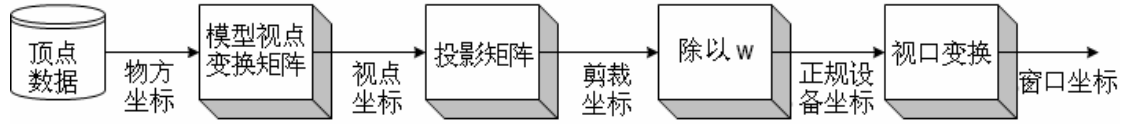


图 2.2 OpenGL 顶点变换过程（译自文献<sup>[33]</sup>）

图 2.2 描述了 OpenGL 将三维顶点变换到二维点的过程，我们将其分述如下：

#### 1. 模型视点变换（ModelView Transform）

模型视点变换用于将物方坐标系中的坐标变换到视点坐标系（即相机坐标系）内。一般来说计算机图形学里存在模型变换和视点变换两个概念。模型变换用于将物方局部坐标系（object space）下的坐标变换至世界坐标系（world space）下，而视点变换则将点从世界坐标系变换至视点坐标系（eye space，亦称相机坐标系或观察坐标系）。由于这两者在实质上没有区别，因此在 OpenGL 中两者被合并成了同一个变换，即模型视点变换（GL\_MODELVIEW）：

$$\begin{pmatrix} x_{eye} \\ y_{eye} \\ z_{eye} \\ w_{eye} \end{pmatrix} = M_{ModelView} \cdot \begin{pmatrix} x_{obj} \\ y_{obj} \\ z_{obj} \\ w_{obj} \end{pmatrix} = M_{View} \cdot M_{Model} \cdot \begin{pmatrix} x_{obj} \\ y_{obj} \\ z_{obj} \\ w_{obj} \end{pmatrix}. \quad (2.5)$$

值得一提的是，视点矩阵 $M_{View}$ 的逆矩阵即为相机的位置与姿态矩阵，这一点

可以从空间变换的相对性来理解。

## 2. 投影变换 (Projection Transform)

计算机图形学中通过将用户定义的一个相机坐标系下的视景体线性变换到剪裁坐标系 (clip space) 中的单位正方体来表达投影关系。视景体若为普通立方体, 则投影关系为正射投影; 若为棱台形状 (Frustum), 则投影关系为透视投影<sup>[27]</sup>:

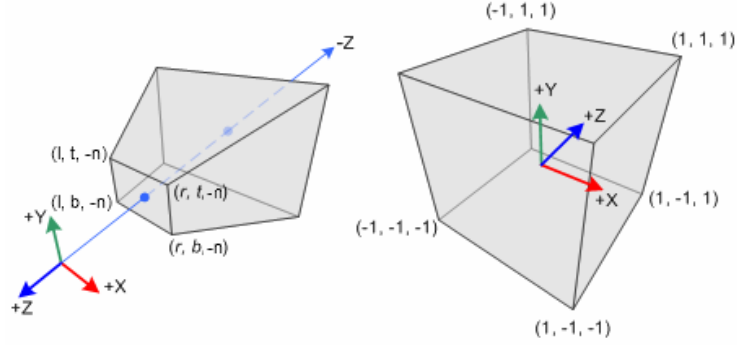
$$\begin{pmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} \end{pmatrix} = M_{Projection} \cdot \begin{pmatrix} x_{eye} \\ y_{eye} \\ z_{eye} \\ w_{eye} \end{pmatrix}. \quad (2.6)$$


图 2.3 左: 相机坐标系中的透视投影视景体;

右: 投影变换后位于剪裁坐标系中的视景体

这里我们主要讨论透视投影, 则公式(2.6)中

$$M_{Projection} = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix}, \quad (2.7)$$

其中,  $l, r, b, t, n > 0, f > 0$  六个参数中的前四个表示视景体近平面的位置和尺寸, 后两个表示近平面和远平面距视点的距离, 见示意图 2.3。根据公式(2.7), 近平面上的四个点  $(l, t, -n)$ 、 $(r, t, -n)$ 、 $(l, b, -n)$  和  $(r, b, -n)$  分别被变换到剪裁坐标系中单位立方体上的四个点  $(-1, 1, -1)$ 、 $(1, 1, -1)$ 、 $(1, -1, -1)$  和  $(-1, -1, -1)$ , 而远平面上的四个点也得到类似的变换。注意, 与相机坐标系不同, 剪裁坐标系是左手系。

## 3. 透视除法 (Perspective Division)

通过透视除法我们得到正规设备坐标 (NDC, Normalized Device Coordinate):

$$\begin{pmatrix} x_{ndc} \\ y_{ndc} \\ z_{ndc} \end{pmatrix} = \begin{pmatrix} x_{clip}/w_{clip} \\ y_{clip}/w_{clip} \\ z_{clip}/w_{clip} \end{pmatrix}. \quad (2.8)$$

故 NDC 就是剪裁坐标的非齐次表达形式, 其中的  $z_{ndc}$  表达的是像素点的深度信息。

#### 4. 视口变换 (Viewport Transform)

所谓视口，是指四个参数 $x_0, y_0, w, h$ ，它们指定了图像相对于当前 OpenGL 窗口被平移和拉伸的程度，变换后得到窗口坐标 (Window Coordinate)：

$$\begin{pmatrix} x_w \\ y_w \end{pmatrix} = \begin{pmatrix} \frac{w}{2}x_{ndc} + (x_0 + \frac{w}{2}) \\ \frac{h}{2}y_{ndc} + (y_0 + \frac{h}{2}) \end{pmatrix}, \quad (2.9)$$

正规设备坐标中的  $z$  分量用于记录深度信息以便提供给 OpenGL 的深度缓存 (Z-Buffer)，与相机模型无关，这里就不给出其变换形式了。窗口坐标系是以窗口左下角为原点，水平向左为  $x$  轴正向，垂直屏幕向外为  $z$  轴正向的右手坐标系。

由于计算机图形学在投影过程中需要保留顶点的深度信息已确保进行后续渲染的正确性（比如各种消隐算法），因此在综合上述变换以获得计算机图形学的相机模型时，我们需要忽略有关深度信息的部分，从而得到如下相机模型：

$$\mathbf{x} = M'_{Window} M'_{Projection} M'_{ModelView} \mathbf{X}, \quad (2.10)$$

$$M'_{Window} = \begin{pmatrix} \frac{w}{2} & 0 & x_0 + \frac{w}{2} \\ 0 & \frac{h}{2} & y_0 + \frac{h}{2} \\ 0 & 0 & 1 \end{pmatrix}, \quad (2.11)$$

$$M'_{Projection} = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} \\ 0 & 0 & -1 \end{pmatrix}, \quad (2.12)$$

$$M'_{ModelView} = R_{gl}(I, -C_{gl}), \quad (2.13)$$

类似第 2.2.2 节，公式(2.13)中 $R_{gl}$ 表示旋转矩阵， $C_{gl}$ 表示相机中心， $I$ 表示单位阵。

### 2.3 模型参数转换

在理解了这三大流派的相机模型后，有必要详细讨论其模型参数的相互转换，这对于之后的相机模拟具有重要意义。

总结第 2.2 节中三种相机模型，我们发现其相机参数均可分为两大类：内参数和外参数。所谓外参数，是描述如何将物方对象变换至相机坐标系的各种参数（即摄影测量中的外方位元素，计算机视觉中的 $R_{cv}$ 和 $C_{cv}$ ，计算机图形学中的模型视点变换矩阵）。而所谓内参数，是描述如何将相机坐标系下的三维点投影成像平面二维点的各种参数（即摄影测量中的内方位元素，计算机视觉中的  $K$ ，计算机图形学中的投影矩阵和视口矩阵）。以下我们将分别推导摄影测量和计算机视觉，以及计算机视觉和计算机图形学之间的相机模型内外参数转换公式。

### 2.3.1 摄影测量与计算机视觉

为了同时考虑计算机视觉和摄影测量的坐标系建立过程，我们对两方的相机模型建立过程进行综合。首先应当认识到这一点，当照片拍摄完成之后，其光学中心，相片在空间中的位置和姿态以及物方对象都已经在空间中确定了，这与我们使用哪种相机模型来描述这个过程无关。

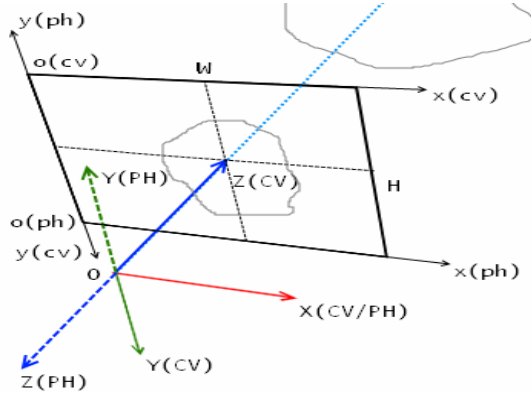


图 2.4 相机坐标系以及像平面坐标系（CV 表示计算机视觉，PH 表示摄影测量）

考虑图 2.4，给定光学中心  $O$ ，以及宽高分别为  $W$  和  $H$  的照片（注意这里的照片是正片，即物方对象与照片在光学中心的同侧）。首先看两方如何建立坐标系。

联系第 2.2.1 节，摄影测量的相机坐标系：以连接像主点和光学中心的轴为相机坐标系的  $Z$  轴，从像主点指向光学中心为  $Z$  轴正方向；以过光学中心  $O$  并且平行于照片水平方向的轴为  $X$  轴，水平向左为  $X$  轴正方向； $XYZ$  轴构成右手坐标系。

而联系第 2.2.2 节，与摄影测量相反，计算机视觉的相机坐标系是以从光学中心指向像主点为  $Z$  轴正方向的右手坐标系。因此两者的相机坐标系恰好存在一个绕  $X$  轴旋转  $180^\circ$  的关系。

不同于一般摄影测量学以像片框标连线中点为像平面原点<sup>[29]</sup>，我们这里采用像片左下角点作为摄影测量像平面原点。而计算机视觉中的像平面坐标系一般即为数字图像坐标系，因此以图像左上角点为原点（见图 2.4）。

根据上述分析，我们可以得到以下结论：

1. 由于一般摄影测量使用相机为量测相机，其像平面  $x$  与  $y$  轴的不垂直因子很小可以视作为零，那么若其主距为  $f$ （以像素为单位），主点为  $(x_0^{ph}, y_0^{ph})$ ，像片高为  $H$ ，则计算机视觉内参数矩阵相应即为

$$K = \begin{pmatrix} f & 0 & x_0^{ph} \\ 0 & f & H - y_0^{ph} \\ 0 & 0 & 1 \end{pmatrix}; \quad (2.14)$$

2. 摄影测量的外方位线元素 $(X_s, Y_s, Z_s)$ 和计算机视觉的相机中心 $C_{cv}$ 完全一致，即

$$C_{cv} = (X_s \ Y_s \ Z_s)^T; \quad (2.15)$$

3. 根据摄影测量的外方位角元素以及文献<sup>[29]</sup>中的公式计算出了九个旋转矩阵元素，则计算机视觉中的旋转矩阵为

$$R_{cv} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} R_{ph} = \begin{pmatrix} a_1 & b_1 & c_1 \\ -a_2 & -b_2 & -c_2 \\ -a_3 & -b_3 & -c_3 \end{pmatrix}. \quad (2.16)$$

### 2.3.2 计算机视觉与计算机图形学

由图 2.3 和图 2.4 我们可以发现，计算机图形学（由 OpenGL 实现）中的相机坐标系与摄影测量完全一致，因此其外方位元素转换完全类似：

$$C_{gl} = C_{cv}, \quad (2.17)$$

$$R_{gl} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} R_{cv}. \quad (2.18)$$

以下我们主要推导两者内参数的相互转换，不失一般性，我们忽略计算机视觉中的像平面 x 轴和 y 轴的不垂直因子 $\gamma$ ，即默认其为零。

首先根据第 2.2.3 节最后的公式(2.11)和(2.12)，我们将两个矩阵合并得到

$$\begin{aligned} M_{WindowProjection} &= M'_{Window} \cdot M'_{Projection} \\ &= \begin{pmatrix} \frac{w}{2} & 0 & x_0 + \frac{w}{2} \\ 0 & \frac{h}{2} & y_0 + \frac{h}{2} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} \\ 0 & 0 & -1 \end{pmatrix} \\ &= \begin{pmatrix} \frac{nw}{r-l} & 0 & \frac{w(r+l)}{2(r-l)} - x_0 - \frac{w}{2} \\ 0 & \frac{nh}{t-b} & \frac{h(t+b)}{2(t-b)} - y_0 - \frac{h}{2} \\ 0 & 0 & -1 \end{pmatrix} \end{aligned} \quad (2.19)$$

将相机坐标系下的坐标与上式相乘，展开得到

$$\left. \begin{aligned} x &= -\frac{nw}{r-l} \frac{X_{eye}}{Z_{eye}} - \frac{w(r+l)}{2(r-l)} + x_0 + \frac{w}{2} \\ y &= -\frac{nh}{t-b} \frac{Y_{eye}}{Z_{eye}} - \frac{h(t+b)}{2(t-b)} + y_0 + \frac{h}{2} \end{aligned} \right\}, \quad (2.20)$$

类比第 2.2.1 节中的公式(2.2)，即摄影测量共线方程，结合第 2.3.1 节中的公式(2.14)，考虑到 OpenGL 中的视口大小 $w, h$ 即为照片尺寸，我们得到如下计算机视觉和计算机图形学中内参数转换公式

$$K = \begin{bmatrix} \frac{nw}{r-l} & 0 & -\frac{w(r+l)}{2(r-l)} + x_0 + \frac{w}{2} \\ 0 & \frac{nh}{t-b} & \frac{h(t+b)}{2(t-b)} - y_0 + \frac{h}{2} \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.21)$$

## 2.4 模拟相机实现

由以上分析，笔者在 OpenSceneGraph (OSG) 里实现了相机模拟和相机浏览工具<sup>1</sup>。前者用于模拟相机，可以输出相片、该相片对应的相机参数和物方点在该相片上对应的像点；后者读取模型文件、相片和相应的相机参数文件，重现照相时的情景。后者亦可将真实照片添加到虚拟场景中，提高其真实感。用户只需要根据虚拟场景中的三维点与照片上像点的对应关系，利用相机标定工具计算出相机的内外参数（内参数  $K$  矩阵，旋转矩阵  $R$  以及相机中心  $C$ ），生成相片浏览器所需的相机参数文件，就可以实现上述功能（见图 2.5）。

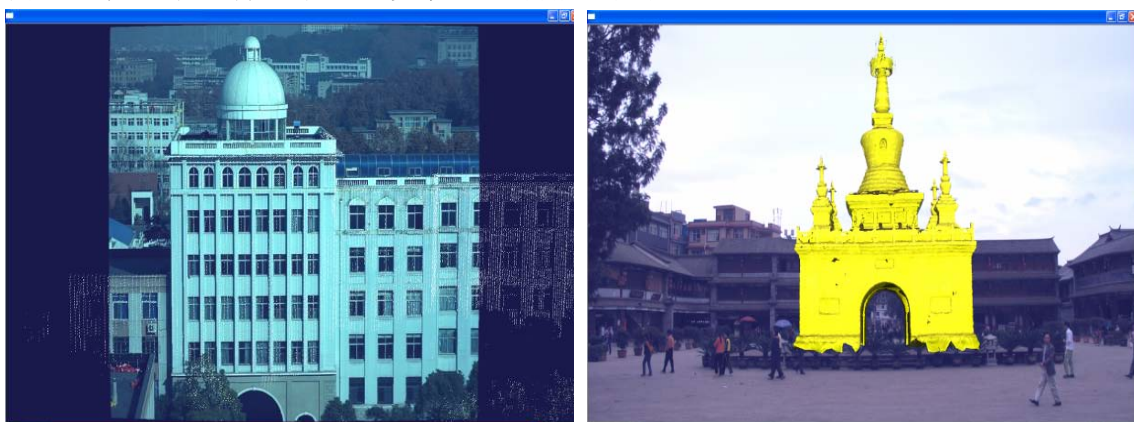


图 2.5 将真实照片与虚拟场景联合显示

## 2.5 本章小结

本章我们首先阐述了相机模拟在整个摄影测量学以及计算机视觉研究中的重要性，然后分别从摄影测量学，计算机视觉以及计算机图形学的角度阐述了各自相机模型的建立过程，并且深入研究了三种模型参数的相互转换公式，最后根据这些公式在 OpenSceneGraph 中实现了相机模拟工具以及相机浏览工具。

值得注意的是，在本章内我们并未考虑相机畸变模型，所有推导都忽略了各种畸变模型（包括径向畸变、偏心畸变等等）。这是因为，畸变模型可以很方便的根据准确的像点以及像主点进行数值模拟，没有必要在 OpenSceneGraph 中以图像的方式体现出来。

---

<sup>1</sup> 由于 OSG 内部的点坐标是使用行向量表达，即其矩阵对点的变换表达为  $V' = V \cdot M$ ，故按照前两节公式对应设置矩阵时也应该首先进行转置。

## 第3章 相机自标定

Photographers deal in things which are continually vanishing and when they have vanished there is no contrivance on earth which can make them come back again.

——Henri Cartier Bresson

### 3.1 概述

所谓相机标定 (Camera Calibration, 亦有文献称为相机定标, 摄影测量学中一般称为相机检校), 是指使用各种测量手段确定相机的内部参数 (像主点、焦距、各种畸变参数等等, 亦即第 2.2.2 节中式(2.3)里的标定矩阵  $K$ ) 的过程。

相机标定在计算机视觉、摄影测量中有着重要的意义: 通过标定畸变参数, 可以将畸变影像上量测所得像点纠正到理想位置, 从而符合理想的小孔成像模型; 而标定像主点、焦距之后, 则可以恢复出像点在空间中的相对位置关系, 这使得我们在之后的三维重建步骤中可以获得欧氏 (Euclidean) 而非仿射 (Affine) 意义下的重建结果<sup>[34]</sup>。

将众多相机标定算法进行总结, 我们可以将其分为两大类: 传统标定方法与相机自标定方法<sup>[35]</sup>。传统方法中比较著名的有摄影测量中的光束法<sup>[30]</sup>, 计算机视觉中的 Tsai 两步法<sup>[36-37]</sup>, 各种直接线性变换 (DLT, Direct Linear Transformation) 方法<sup>[31, 38]</sup>, 以及张正友等人提出的平面标定方法<sup>[39]</sup>等等。其共同特点是, 需要提供已知的物方度量信息, 诸如三维点的空间坐标、三维直线的空间坐标或者是已知的空间角度等等。这要求该方法必须配合经过精确量测的标定参照物 (calibration rig) 一起使用, 例如摄影测量中使用的高精度标定控制场、张正友标定方法中使用的棋盘标定物。

与传统方法不同, 相机自标定 (Self-Calibration) 不要求使用标定参照物。所谓自标定, 意思是标定过程中不需要对所拍摄的场景进行度量, 而仅仅利用多幅影像间的对应关系信息 (如利用基础矩阵的标定)、影像自身的运动信息 (如基于主动视觉的方法)、或者是先验的场景结构信息 (灭点标定)。

自标定在本研究中有着重要的意义。因为本研究所希望达到的目标是, 任意给定一张中心投影生成的影像, 即可对其进行三维重建。由于无法事先对生成该影



像的相机进行标定，因此传统的标定方法在本研究中都不适用；又由于与本研究针对的是单张影像的三维重建，因此自标定方法中基于基础矩阵的方法和基于主动视觉的方法也都不适用。可见，对于本文所讨论的单视影像三维重建而言，灭点标定是唯一可行的标定方法。

本章主要讨论相机自标定中的灭点标定方法，并且介绍了一种新的稳健估计及其在灭点标定中的应用：第 3.2 节介绍了利用灭点标定的理论基础，并且给出了一个直观的几何解释；第 3.3 节讨论了如何检测一幅图像中的灭点，介绍了多实例模型稳健估计方法 J-linkage，以及如何将其应用在灭点检测中。

## 3.2 灭点标定

灭点 (Vanishing Point, 也译作消隐点)，是三维空间中一组平行直线经过中心投影后在投影平面上相交所得的点。在人工环境中，往往存在大量的平行线，例如门窗和建筑物的轮廓线。因此在对人工环境所拍摄的照片中也就存在着对应的灭点，例如火车的两根铁轨、房屋的垂线（见图 3.1），分别都在像平面上相交于一点。

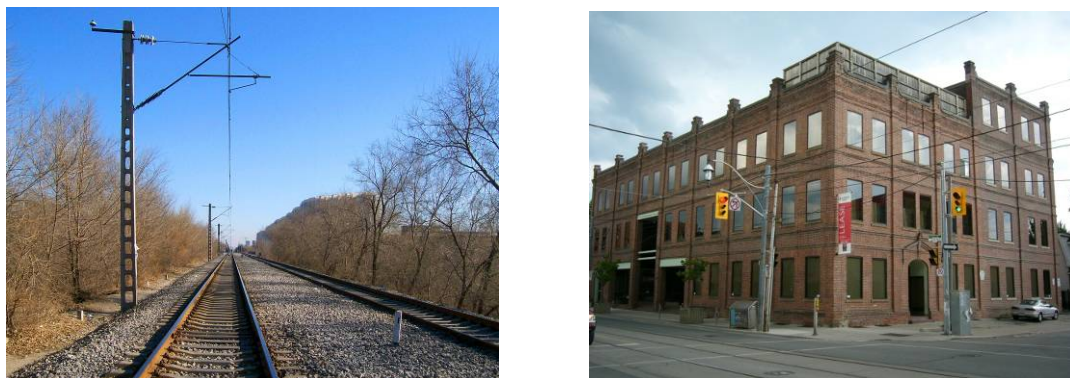


图 3.1 左：平行铁轨在相片上交于灭点；右：建筑物垂直轮廓线交于灭点<sup>1[40-41]</sup>

实际上我们可以认识到，一个灭点即对应空间中的一组平行直线组成的集合，从而对应一个空间方向，而不同的空间方向所形成的灭点也不同。根据 B.Caprile 等人的研究<sup>[42]</sup>，只要在像平面内确定了三个灭点，且这三个灭点所对应的三个空间方向两两垂直，那么就可以根据这三个灭点的位置确定相机的三个重要内参数：像主点 $(x_0, y_0)$ 和相机主距 $f$ 。当然这里存在几个假设：

1. 相机的成像单元为正方形，即纵横比 (Aspect Ratio) 为 1，亦即第 2.2.2

---

<sup>1</sup> 图片引用自 York Urban 数据库，由 Denis 等人提供。

节中的主距 $\alpha_x, \alpha_y$ 相等;

2. 第 2.2.2 节中像平面 x 轴和 y 轴的不垂直因子 $\gamma = 0$ .

以上假设等价于认定式(2.3)中的标定矩阵具有如下形式:

$$K = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

根据 J. M. Coughlan 和 A. L. Yuille 在研究<sup>[43-45]</sup>中提出的所谓“曼哈顿世界假设 (Manhattan World Assumption)”——“大多数室内或室外城市场景都是基于笛卡尔坐标系统的, 我们称其为曼哈顿网格。很自然的, 这个网格给观察者提供了一个参考框架”——我们可以认为, 在人工场景 (尤其是各种建筑场景) 的相片中, 一般很容易找到三组相互垂直的直线。而上述的另外两个假设, 现有的数码相机也很容易满足, 或者近似满足, 且其所带来的误差对于重建后的虚拟场景的视觉效果并无显著影响。

因此, 以下我们将基于上述前提和假设, 给出灭点标定的理论推导<sup>[46]</sup>, 以及一个几何上的直观解释, 并且给出灭点标定的实验结果及相关讨论。

### 3.2.1 理论推导

不失一般性的, 我们可以令场景中三个相互垂直的方向分别为 X、Y 和 Z 方向, 那么根据公式(2.3)我们可以得到

$$\begin{bmatrix} \lambda_1 u_1 & \lambda_2 u_2 & \lambda_3 u_3 \\ \lambda_1 v_1 & \lambda_2 v_2 & \lambda_3 v_3 \\ \lambda_1 & \lambda_2 & \lambda_3 \end{bmatrix} = \mathbf{P} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad (3.2)$$

其中的 $\lambda_i$ 表示未知的非零比例因子,  $(u_i, v_i), i = 1, 2, 3$ 分别表示 X, Y, Z 方向在像平面上对应的灭点的坐标。代入公式(2.4)和式(3.1)得到

$$\begin{bmatrix} \lambda_1 u_1 & \lambda_2 u_2 & \lambda_3 u_3 \\ \lambda_1 v_1 & \lambda_2 v_2 & \lambda_3 v_3 \\ \lambda_1 & \lambda_2 & \lambda_3 \end{bmatrix} = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{R}, \quad (3.3)$$

从而推出

$$\mathbf{R} = \begin{bmatrix} \lambda_1(u_1 - x_0)/f & \lambda_2(u_2 - x_0)/f & \lambda_3(u_3 - x_0)/f \\ \lambda_1(v_1 - y_0)/f & \lambda_2(v_2 - y_0)/f & \lambda_3(v_3 - y_0)/f \\ \lambda_1 & \lambda_2 & \lambda_3 \end{bmatrix}. \quad (3.4)$$

由于旋转矩阵  $\mathbf{R}$  具有正交性, 我们可由上式前两列正交的条件推出

$$\lambda_1 \lambda_2 ((u_1 - x_0)(u_2 - x_0)/f^2 + (v_1 - y_0)(v_2 - y_0)/f^2 + 1) = 0, \quad (3.5)$$

又由于 $\lambda_i \neq 0 (i = 1, 2, 3)$ , 故上式等价于

$$(u_1 - x_0)(u_2 - x_0) + (v_1 - y_0)(v_2 - y_0) + f^2 = 0, \quad (3.6)$$

类似的, 由第二列和第三列、第三列和第一列的正交条件我们得到

$$(u_2 - x_0)(u_3 - x_0) + (v_2 - y_0)(v_3 - y_0) + f^2 = 0, \quad (3.7)$$

$$(u_1 - x_0)(u_3 - x_0) + (v_1 - y_0)(v_3 - y_0) + f^2 = 0. \quad (3.8)$$

由式(3.6)减去式(3.7)得到

$$(u_1 - u_3)(u_2 - x_0) + (v_1 - v_3)(v_2 - y_0) = 0, \quad (3.9)$$

类似的, 由式(3.7)减去式(3.8)、式(3.8)减去式(3.6)得到

$$(u_2 - u_3)(u_1 - x_0) + (v_2 - v_3)(v_1 - y_0) = 0, \quad (3.10)$$

$$(u_1 - u_2)(u_3 - x_0) + (v_1 - v_2)(v_3 - y_0) = 0. \quad (3.11)$$

联立式(3.9)和式(3.10), 写成矩阵形式有

$$\begin{aligned} \mathbf{Ax} &= \mathbf{b}, \\ \mathbf{A} &= \begin{bmatrix} u_1 - u_3 & v_1 - v_3 \\ u_2 - u_3 & v_2 - v_3 \end{bmatrix}, \\ \mathbf{x} &= \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}, \\ \mathbf{b} &= \begin{bmatrix} (u_1 - u_3)u_2 + (v_1 - v_3)v_2 \\ (u_2 - u_3)u_1 + (v_2 - v_3)v_1 \end{bmatrix}, \end{aligned} \quad (3.12)$$

从而可以直接解得像主点

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}. \quad (3.13)$$

值得注意的一点是, 因为式(3.11)可以由式(3.9)和式(3.10)推出, 所以它并不构成一个独立的方程, 不需要参与构成  $\mathbf{A}$  矩阵, 所得的线性方程组即可有唯一解, 所以这和文献<sup>[47]</sup>中第 2.5.1 节中所说将以上三式一起构成线性方程组并且使用所谓“最小二乘法求解”实质上是等价的。

解得像主点后, 可以直接根据式(3.6)求解相机主距

$$f = \sqrt{-(u_1 - x_0)(u_2 - x_0) - (v_1 - y_0)(v_2 - y_0)}. \quad (3.14)$$

式(3.13)和式(3.14)即为利用三个对应方向相互垂直的灭点进行相机标定的计算公式。

### 3.2.2 几何解释

根据式(3.9)我们可以看出其几何意义为: 灭点  $X$  和灭点  $Z$  的连线与灭点  $Y$  和像主点的连线相互垂直; 类似的式(3.10)和式(3.11)的几何意义分别为: 灭点  $Y$  和

灭点 Z 的连线与灭点 X 和像主点的连线相互垂直，灭点 X 和灭点 Y 的连线与灭点 Z 和像主点的连线相互垂直。这表明，像主点应该是灭点 X、Y 和 Z 所构成三角形的垂心。实际上，这一点通过立体几何可以得到一个非常直观的解释。

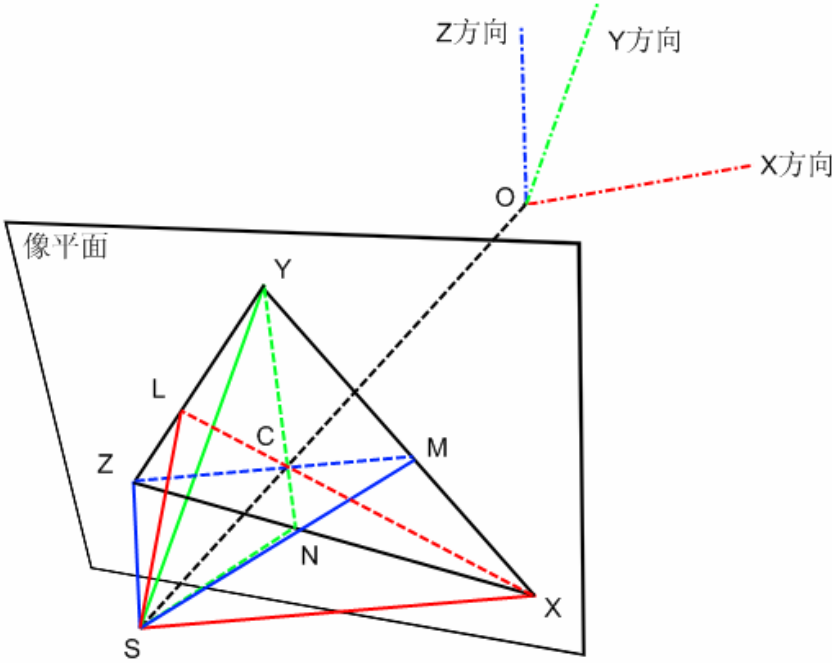


图 3.2 灭点标定的几何解释

将灭点 X、Y 与 Z 分别与相机光学中心 S 相连，根据灭点定义可知，SX，SY 和 SZ 两两垂直，又有像主点 C 与 S 的连线和像平面垂直，可见点 S、X、Y 和 Z 构成了一个直角四面体<sup>1</sup>。由立体几何容易证明，直角四面体的斜面上的高 SC 与斜面的交点 C 即为斜面三角形 XYZ 的垂心。

### 3.2.3 实验结果

利用第 2.4 节中所实现的相机模拟程序，笔者模拟了一幅图像（见图 3.3）用于进行灭点标定的实验。表 3.1 给出了该相片对应的模拟相机参数。表 3.2 给出了单位立方体的 8 个像点的真实坐标（图像坐标系，即以图像左上角点为原点）。

---

<sup>1</sup> 所谓直角四面体,是指由同一点出发的, 两两互相垂直的三条棱所构成的四面体。其中两两垂直的三条棱称为直角棱, 两两垂直的三个面称为直角面, 另一个面称为斜面。

表 3.1 模拟相片的相机参数（对应图 3.3，其中内参数和像幅单位为像素）

内参数	$\mathbf{K} = \begin{bmatrix} 7.009616e+002 & 0 & 2.650000e+002 \\ 0 & 7.009616e+002 & 2.350000e+002 \\ 0 & 0 & 1 \end{bmatrix}$
旋转矩阵	$\mathbf{R} = \begin{bmatrix} 8.621022e-001 & -4.109904e-001 & 2.964230e-001 \\ 7.096253e-002 & -4.812859e-001 & -8.736866e-001 \\ 5.017411e-001 & 7.742420e-001 & -3.857527e-001 \end{bmatrix}$
相机中心	$\mathbf{C} = [-6.795583e-001 \quad -1.582206e+000 \quad 1.545251e+000]^T$
投影矩阵	$\mathbf{P} = \begin{bmatrix} 7.372619e+002 & -8.291436e+001 & 1.055567e+002 & 2.067133e+002 \\ 1.676512e+002 & -1.554160e+002 & -7.030726e+002 & 9.544522e+002 \\ 5.017411e-001 & 7.742420e-001 & -3.857527e-001 & 2.162058e+000 \end{bmatrix}$
像幅	$W = 500, H = 500$

表 3.2 像点真实位置（单位：像素）

点号	X 坐标	Y 坐标
0	9.560952e+001	4.414555e+002
1	3.543718e+002	4.212418e+002
2	2.504510e+002	2.811739e+002
3	4.216155e+001	2.721235e+002
4	1.757975e+002	1.415183e+002
5	4.607158e+002	1.839431e+002
6	3.166862e+002	8.636628e+001
7	8.992409e+001	3.762470e+001

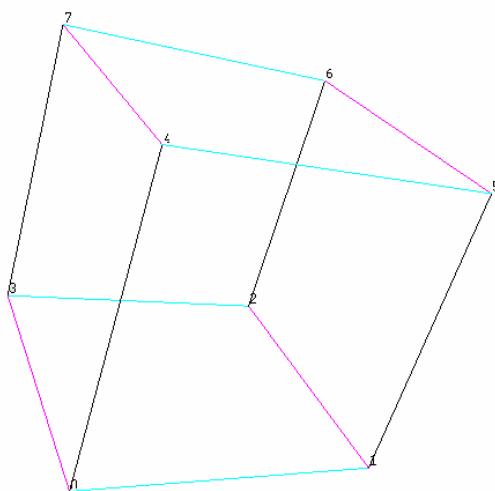


图 3.3 单位立方体的模拟相片

在没有量测误差的情况下，笔者在 Matlab 中实现上述灭点标定算法，解得相机内参数为

$$\mathbf{K} = \begin{bmatrix} 700.961582262149 & 0 & 265.000270785556 \\ 0 & 700.961582262149 & 235.000144133067 \\ 0 & 0 & 1 \end{bmatrix},$$

上述结果说明了灭点标定算法的正确性。

接下来考虑对灭点标定算法进行误差分析。由于灭点是三维射影空间中无穷远点在像平面上所成的像，而无穷远点在三维欧氏空间中并不存在，不便直接对其行误差分析。因此，不同于张祖勋等人<sup>[48]</sup>的解析分析方法，这里笔者采用数值模拟实验的方法来分析灭点标定的计算精度。

---

### 算法 3.1 给定像点量测标准差的误差分析

---

#### 输入

$N$ : 模拟量测次数

$\sigma$ : 像点量测标准差

$f^{true}$ : 相机主距真值

$(x_0^{true}, y_0^{true})$ : 相机主点真值

$P$ : 像点真值的数组

$L_i$ : 直线的数组，其每一个元素为(S, E)，分别表示直线起点和终点在 $P$ 中的下标， $i = 1, 2, 3$ 分别代表 X、Y 和 Z 方向的直线数组

#### 输出

$F$ : 灭点标定所得相机主距真误差数组

$X$ : 灭点标定所得相机主点 X 坐标真误差数组

$Y$ : 灭点标定所得相机主点 Y 坐标真误差数组

#### 算法

对  $n \leftarrow 1$  到  $N$ :

    给  $P$  中每个像点的坐标分别加上一个随机误差  $(x, y)$ ,  $x \sim N(0, \sigma^2)$ ,  $y \sim N(0, \sigma^2)$

    根据式(3.13)和式(3.14)，利用  $P$ ,  $L_i$  进行标定，得到主距  $f$  和主点  $(x_0, y_0)$

$F_n \leftarrow f - f^{true}$

$X_n \leftarrow x_0 - x_0^{true}$ ,  $Y_n \leftarrow y_0 - y_0^{true}$

---

利用算法 3.1 以及表 3.1 和表 3.2 的数据，我们进行了一次  $N=50$ ,  $\sigma = 0.5$  (像点最大量测误差约为两个像素) 的实验，所得数据绘制成曲线图 (图 3.4)。从图

中我们可以发现，像点量测的误差对于灭点标定的结果有较大的影响，尤其是主距的标准差达到了 6.03 个像素，最大真误差的绝对值超过了 15 个像素。

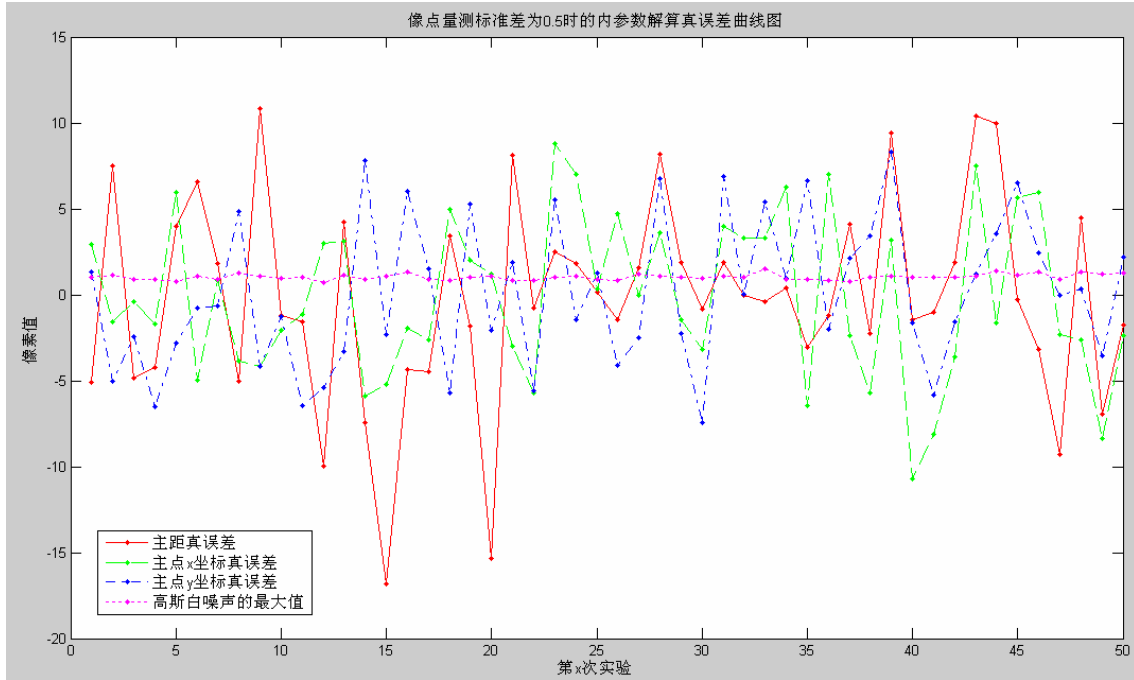


图 3.4 五十次模拟实验的内参数解算真误差曲线图，像点量测标准差为 0.5 像素，主距标准差为 6.0332，主点 X 坐标标准差为 5.0111，主点 Y 坐标标准差为 3.8418。

为了进一步分析像点量测误差和灭点标定解算误差的关系，笔者在算法 3.1 的基础上设计了另一个算法 3.2 以进行模拟实验。根据  $N = 100$ ， $\sigma_{min} = 0.1$ ， $\sigma_{max} = 2$ ， $\delta = 0.1$ ，以及表 3.1 和表 3.2 的数据，算法 3.2 获得了灭点标定解算误差虽量测误差变化曲线图（图 3.5）。由该图可以看到，解算和量测标准差近似呈线性关系，且在量测标准差为 2 个像素时，解算标准差被放大到 20 个像素左右，这和谢文寒<sup>[49]</sup>论文中给出的结论“（单像灭点标定）只能进行弱标定”是一致的。

---

### 算法 3.2 分析灭点标定解算误差随像点量测误差的变化

---

#### 输入

$\sigma_{min}, \sigma_{max}$ : 像点量测标准差的最小值和最大值

$\delta$ : 像点量测标准差的变化步长

以及算法 3.1 要求的所有其他输入

#### 输出

$F'$ : 每次灭点标定所得相机主距真误差的标准差数组

$X'$ : 每次灭点标定所得相机主点 X 坐标真误差的标准差数组

---

$Y'$ : 每次灭点标定所得相机主点  $Y$  坐标真误差的标准差数组

### 算法

$n = 1, \sigma = \sigma_{min}$

若  $\sigma \leq \sigma_{max}$ , 则

调用算法 3.1 进行标定, 得到数组  $F, X, Y$

$F'_n \leftarrow std(F), X'_n \leftarrow std(X), Y'_n = std(Y)$ ,  $std(M)$  表示求数组  $M$  的标准差

$n \leftarrow n + 1, \sigma \leftarrow \sigma + \delta$

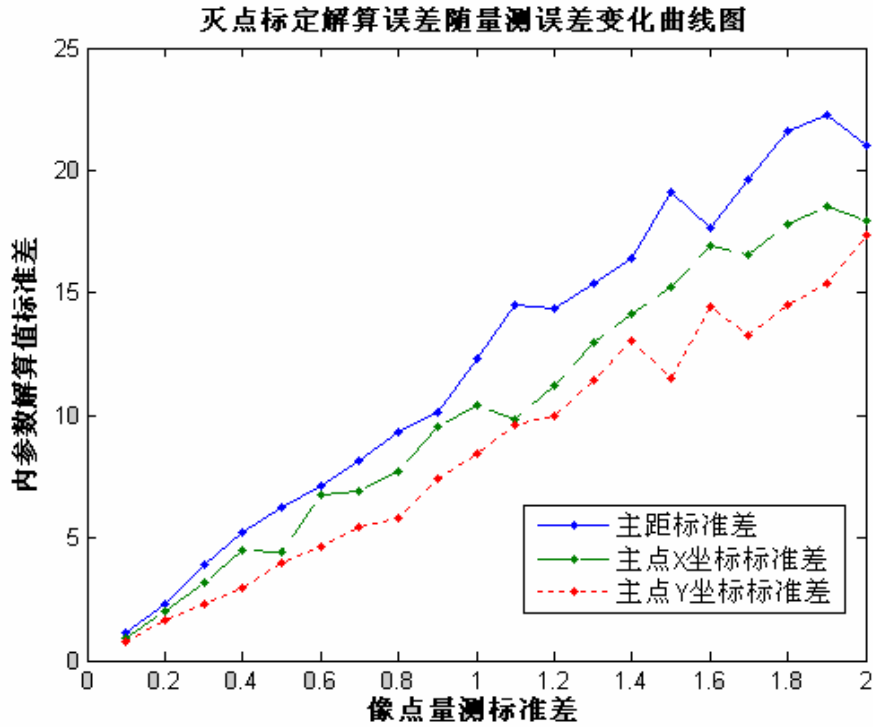


图 3.5 灭点标定解算误差随量测误差变化曲线图。在每个确定的像点量测标准差下, 重复模拟一百次解算过程, 统计内参数解算真误差的标准差, 绘制而成。

对于灭点标定的误差理论分析, 可以进一步参考谢文寒博士论文<sup>[49]</sup>。虽然灭点标定的精度不高, 但考虑到本文的目的是为了向虚拟现实等应用提供三维模型, 对于标定精度的要求没有一般摄影测量那么高, 且根据实验, 使用灭点标定对于三维重建结果的视觉效果影响不大, 因此是可以接受的。

### 3.3 灭点检测

从前一节我们可以看到, 灭点标定中的一个重要步骤就是检测灭点。前一节中我们使用的方法可以看成是手动地在影像上选择直线。然而为了提高灭点标定的



自动化程度，计算机视觉和摄影测量的学者提出了许多不同的算法来达到自动检测灭点的目的。

考虑到灭点有可能位于像平面的无穷远处，Barnard 等人<sup>[50]</sup>提出在所谓“高斯球（Gaussian sphere）”——即球心位于相机光学中心，半径为 1 的单位球——的球面上做类似 Hough 变换的方式来检测灭点：每一条检测出来的直线对应于高斯球上的一个大圆，灭点对应于众多大圆的交点，通过对高斯球的表面进行量化并记录落入其中的交点的个数，峰值点处即为灭点。然而这种方法和一般的 Hough 变换有着一样的缺点，即检测精度受高斯球表面量化程度的限制，较耗费内存空间，而且受噪声影响较大。

在计算机视觉界，一种被称之为 RANSAC（Random Sample Consensus，随机采样一致性）<sup>[51]</sup>的稳健估计方法被广泛地用于对包含大量噪声的数据进行模型参数估计，比如利用 RANSAC 解基础矩阵。其基本思想与通常我们使用的最小二乘估计不同，不是使用尽可能多的数据点去获得模型的估计，而是使用尽可能少的可行数据去尽量扩大一致性数据集，最后再使用最大的一致性数据集内的数据点来进行最优估计。这种思想使得 RANSAC 方法能够很好的剔除数据中的外点（outliers，亦即粗差），从而获得稳健的估计。

然而，由于 RANSAC 方法是针对数据集中只存在模型的单个实例这种情况来设计的，因此，对于多实例模型的稳健估计问题（例如从数据集中估计  $N$  条直线），RANSAC 方法就失效了。这是因为在这种问题中，同时存在着外点和伪外点（pseudo-outliers），后者的定义<sup>[52]</sup>为“是当前实例的外点，却是其他实例内点的点”。

不同于 RANSAC 研究所有数据在某个模型假设上的残差的分布，残差直方图分析<sup>[53]</sup>提供了一个新的思路：分析某个单独数据点在不同模型假设上的残差的分布，并指出该残差的分布模式反映了模型实例的情况。每个数据点的这种残差分布的峰值对应着真实的模型，因为通过随机采样生成的模型假设通常应该聚集在真实模型的周围，这也是随机 Hough 变换能够应用的一个基础。这个特点也就决定了残差直方图分析方法可以像随机 Hough 变换一样，自动发现模型实例的数量，而不是像 multiRANSAC<sup>[54]</sup>那样需要预先指定模型数量。然而实验表明<sup>[55]</sup>残差直方图分析方法在寻找残差分布模式的时候存在很多问题导致实验结果并不理想。

受 RANSAC 和随机直方图分析的启发，Toldo 和 Fusiello 最近提出了一种新的多实例模型稳健估计方法——J-linkage<sup>[55]</sup>，这种方法被众多实验证明优于

multiRANSAC, 随机 Hough 变换 (RHT) [56], 残差直方图分析等多实例模型估计方法, 而灭点检测又是典型的多实例模型估计问题, 因此我们接下来首先介绍 J-linkage 方法, 然后再说明它是如何应用于灭点检测的, 最后给出相关实验结果及分析。

### 3.3.1 多实例模型稳健估计——J-linkage

J-linkage 算法的基本思想是: 将数据点放入其相似概念空间 (conceptual space) 中进行分析, 属于同一模型实例的数据点会聚集在相似概念空间中。

可见, J-linkage 中最重要的概念就是相似概念空间。它不同于随机 Hough 变换中使用的参数空间 (parameter space) 和残差直方图分析中的残差空间 (residual space) ——前者由于需要对连续的参数空间进行量化, 直接导致随机 Hough 变换中的各种缺点; 而后者又因为对残差模式估计的困难而影响了残差直方图分析的实际效果。相似概念空间其实是源自模式识别中的相似概念表示方式: 给定一个对象  $x$  和  $C$  个类别, 则该对象的相似概念表示为其对每个类别的后验概率的向量  $[P(x | class_1), P(x | class_2), \dots, P(x | class_C)]$ 。而在 J-linkage 中每个数据点的相似概念则表示为数据点对于每个模型假设的倾向情况的向量 (这和残差直方图分析中的单个数据点对于各模型假设的残差的分布是类似的)。

从 J-linkage 的基本思想可知, 该算法框架包含两大步骤: 首先是有数据点生成概念空间, 然后是在该空间中进行聚类。接下来我们分别介绍这两个步骤。

#### 1. 生成相似概念空间

首先应当确定空间的维度  $M$ 。由上述相似概念的解释可知, 相似概念空间的维度即为模型假设的个数。在这里, 所谓模型假设, 就是根据从数据点集合中随机抽取的  $n$  个数据点所计算的到的模型,  $n$  是确定一个模型所需的最小数据点个数, 这  $n$  个点即构成所谓的最小采样集 (MSS, minimal sample set)。 $M$  的大小是与数据点集合中外点的百分比相关的, 它必须足够大才能够使算法以某个确定的概率  $\rho$  获得某确定数量  $K$  的不包含外点的最小采样集, 其计算公式为

$$\rho = 1 - \sum_{k=0}^{K-1} \binom{M}{k} p^k (1-p)^{M-k}, \quad (3.15)$$

其中  $p$  表示从数据点集合中抽样得到一个完全由内点 (inliers) 组成的最小采样集的概率, 这个概率与抽样方法、MSS 的大小有关。

然后是生成  $M$  个模型假设。J-linkage 在这里采用了一种优先选择邻近空间中的数据点的方式来生成 MSS。也就是说，当一个点  $x_i$  已经被选择之后，那么接下来点  $x_j$  被选中的概率是

$$P(\mathbf{x}_j | \mathbf{x}_i) = \begin{cases} \frac{1}{Z} \exp - \frac{\|\mathbf{x}_j - \mathbf{x}_i\|^2}{\sigma^2} & \text{if } x_j \neq x_i, \\ 0 & \text{if } x_j = x_i \end{cases} \quad (3.16)$$

每个 MSS 生成之后，都可以利用其内的数据点计算出一个模型实例的假设。

最后是数据点在相似概念空间中的定位，即将数据点用其相似概念表示。若模型用函数  $F(p, \theta) = 0$  表示，数据点为  $p_i$ ，模型实例用其参数  $\theta_j$  表示，则模型实例  $j$  的内点  $i$  应该满足

$$|F(p_i, \theta_j)| \leq \varepsilon \quad (3.17)$$

其中  $\varepsilon$  即为 RANSAC 算法中的一致性阈值。那么所有满足式(3.17)的数据点被称为模型实例  $j$  的一致集 (CS, consensus set)。通过一致集可以构造用于描述数据点对于每个模型倾向情况的倾向集合 (PS, preference set)，也就是其相似概念。维度为  $M$  的相似概念空间中，单个数据点的倾向集合可以用一个  $1 \times M$  的 0-1 向量表示：

$$\begin{aligned} \mathbf{PS}_i &= [ps_{i,1} \ ps_{i,2} \ \cdots \ ps_{i,M}]_{1 \times M} \\ ps_{i,j} &= \begin{cases} 1 & |F(p_i, \theta_j)| \leq \varepsilon \\ 0 & \text{otherwise} \end{cases}, \end{aligned} \quad (3.18)$$

我们称这个向量  $\mathbf{PS}_i$  为数据点  $i$  的倾向向量。若数据点集合的大小为  $N$ ，则所有数据点的倾向向量组成的  $1 \times M$  矩阵称之为倾向矩阵

$$\mathbf{PM} = \begin{bmatrix} \mathbf{PS}_1 \\ \mathbf{PS}_2 \\ \vdots \\ \mathbf{PS}_N \end{bmatrix}_{N \times M}, \quad (3.19)$$

这个倾向矩阵就是数据点集合的相似概念空间的表现形式。

## 2. 相似概念的聚类

相似概念空间生成之后，即可对其倾向集合中的  $N$  个  $M$  维倾向向量进行聚类。J-linkage 采用的聚类方式是一种自底向上的凝聚聚类方式 (agglomerative clustering)：从所有包含单个实例的类别开始，聚类算法每次扫描都会将两个具有最小距离的类别合并。这一类聚类方式被称之为 linkage 算法，而不同的 linkage 算法使用的距离测度也不相同。J-linkage 算法所采用的是一种被称之为 Jaccard 距离的测度，这也是 J-linkage 算法名字的由来。

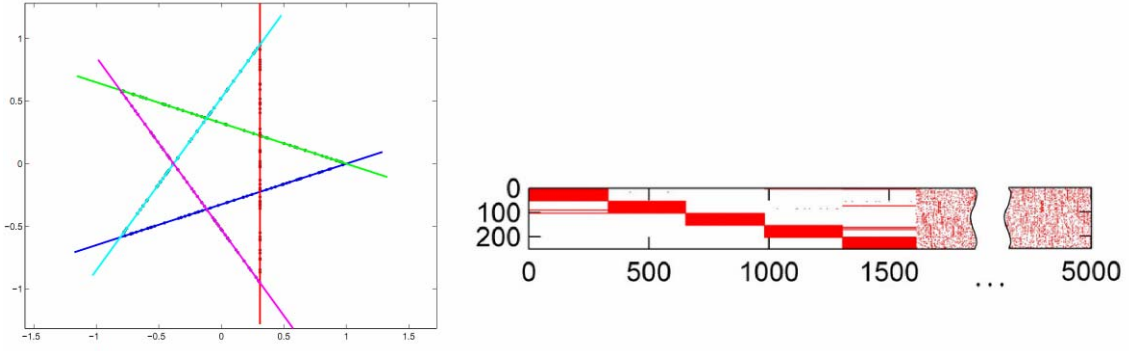


图 3.6 左：包含 250 个数据点，分别落在五角星的 5 条线段上；右：倾向矩阵，行为数据点的倾向向量，列为模型实例假设，矩阵在行方向由所聚类别进行了排序，在列方向上由所聚类别的大小进行了排序<sup>[55]</sup>。

**定义 3.1** Jaccard 距离。给定两个集合  $A$  和  $B$ ，其 Jaccard 距离为

$$d_J(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}, \quad (3.20)$$

其中符号  $|A|$  表示集合  $A$  的元素个数。

从定义 3.1 可知，Jaccard 距离表示的是两个集合的重叠程度——0 表示  $A$  和  $B$  是同一个集合，1 表示  $A$  和  $B$  是不相交集合。由于倾向向量表示了数据点的倾向集合，因此有必要使用向量语言重述公式(3.20)。为此我们给出定义 3.2 和定义 3.3，利用它们可以得到倾向向量的 Jaccard 距离公式(3.23)，并给出完整的 J-linkage 聚类算法（算法 3.3）。

**定义 3.2** 倾向向量的与运算。给定两个  $n$  维倾向向量  $A$  和  $B$ ，其与运算  $\otimes$  生成一个新的  $n$  维向量

$$\begin{aligned} \mathbf{C} &= \mathbf{A} \otimes \mathbf{B} \\ C_i &= \begin{cases} 1 & A_i = B_i = 1 \\ 0 & \text{otherwise} \end{cases}, i = 1, 2, \dots, n, \end{aligned} \quad (3.21)$$

它对应了集合的交运算。

**定义 3.3** 倾向向量的或运算。给定两个  $n$  维倾向向量  $A$  和  $B$ ，其或运算  $\oplus$  生成一个新的  $n$  维向量

$$\begin{aligned} \mathbf{C} &= \mathbf{A} \oplus \mathbf{B} \\ C_i &= \begin{cases} 1 & A_i + B_i \geq 1 \\ 0 & \text{otherwise} \end{cases}, i = 1, 2, \dots, n, \end{aligned} \quad (3.22)$$

它对应了集合的并运算。

定义 3.4 倾向向量的 Jaccard 距离。给定两个  $n$  维倾向向量  $A$  和  $B$ ，其 Jaccard 距离公式为

$$d_J(A, B) = \frac{\sum(A \oplus B) - \sum(A \otimes B)}{\sum(A \oplus B)}, \quad (3.23)$$

其中  $\sum C$  表示对倾向向量  $C$  的所有元素求和。

---

### 算法 3.3 J-linkage 聚类算法<sup>[55]</sup>

---

输入

$PM_{N \times M}$ : 倾向矩阵，包含  $N$  个数据点的  $M$  维倾向向量  $PV_i$

输出

$N$  个数据点的聚类

算法

1. 初始化，每个倾向向量自成一类
  2. 在所有倾向向量中，选择两个倾向向量  $PV_i, PV_j$ ，满足  $d = d_J(PV_i, PV_j)$  最小
  3. 若  $d < 1$ ，从  $PM$  中删去  $PV_i, PV_j$ ，添加  $PV_i \oplus PV_j$ ，转到步骤 2
  4. 否则结束算法，聚类完成。
- 

在得到所有数据点的聚类之后，则可以利用每一个聚类中的数据点进行最小二乘等最优估计，不同的聚类对应着不同的模型实例，从而完成多实例模型的稳健估计。

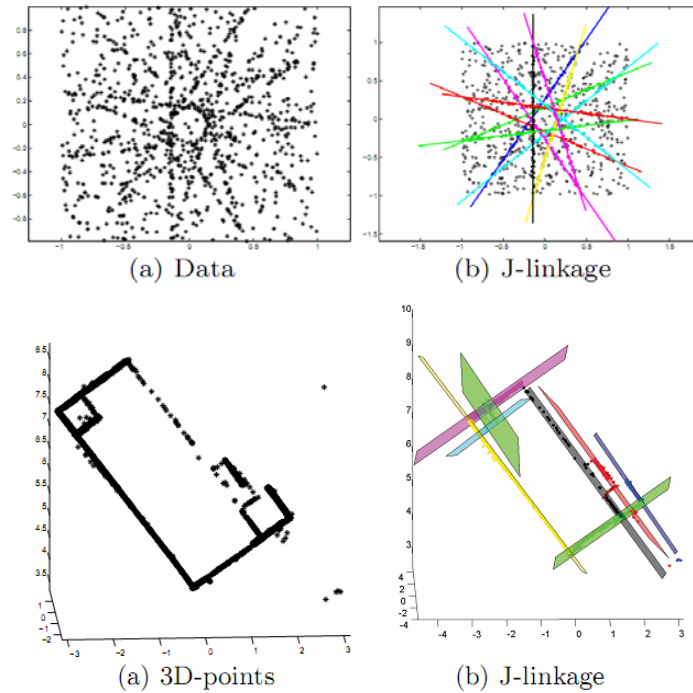


图 3.7 J-linkage 进行多实例模型估计的几个例子<sup>[55]</sup>

### 3.3.2 J-linkage 在灭点检测中的应用

考虑到点和直线的对偶性,就很好理解 J-linkage 算法在灭点检测中是如何应用的了。J. P. Tardif 就利用 J-linkage 给出了一种非迭代式的灭点检测算法<sup>[57]</sup>。

---

#### 算法 3.4 J-linkage 式灭点检测算法

---

输入

$\varepsilon_{1 \dots N}$ : 自动检测得到的  $N$  条边

$\phi$ : 一致性阈值 (恒定为 2 个像素)

$M$ : 灭点假设的个数, 相似概念空间的维度

输出

$\mathcal{V}$ : 灭点集合

算法

生成倾向矩阵  $PM$ , 对  $m \leftarrow 1$  到  $M$ :

    随机从  $\varepsilon$  中选择两条边组成边集  $S$ ,

    由  $S$  计算灭点,  $\mathbf{v}_m \leftarrow V(S)$ ,

    对  $n \leftarrow 1$  到  $N$ :  $PM_{n,m} \leftarrow D(\varepsilon_n, \mathbf{v}_m) \leq \phi$

利用 J-linkage 算法和  $PM$  对边进行聚类

对于每个聚类, 利用各种最优估计, 分别计算出其灭点, 并得到灭点集合  $\mathcal{V}$

---

其中函数  $D(\varepsilon_n, \mathbf{v}_m)$  类似于式(3.17)中的  $F$ , 即为各直线对于灭点的一致性度量函数。根据 D. Liebowitz 等人<sup>[10]</sup>提出的灭点估计方法, 有下式

$$[\mathbf{v}, \mathbf{l}_1, \dots, \mathbf{l}_N] = \arg \min_{\mathbf{v}, \mathbf{l}_1, \dots, \mathbf{l}_N} \sum_{i=1}^N d^2(\mathbf{l}_i, \mathbf{e}_{i,1}) + d^2(\mathbf{l}_i, \mathbf{e}_{i,2}), \quad (3.24)$$

且满足  $\mathbf{l}_i^T \mathbf{v} = 0, i = 1, \dots, N$ , 其中  $\mathbf{e}_{i,1}, \mathbf{e}_{i,2}$  分别表示检测到的第  $i$  条线段 (通过 Canny 边缘提取算子以及边缘分割算法获得) 的两个端点,  $d^2(\mathbf{l}, \mathbf{e})$  表示点到直线距离的平方。该方法即估计这样一个灭点, 使得过这个灭点的  $N$  条直线到其对应的  $N$  条线段的两个端点的距离的平方和最小。

但是考虑到式(3.24)的优化需要迭代求解 (LM, Levenberg-Marquardt), 而函数  $D(\varepsilon_n, \mathbf{v}_m)$  在算法 3.4 中会被大量调用 ( $N \times M$  次), 因此不适合直接使用式(3.24)作为灭点一致性度量函数, 为此 J. P. Tardif 提出了一种近似的闭合求解方案<sup>[57]</sup>

$$D(\varepsilon_j, \mathbf{v}) = d(\hat{\mathbf{l}}, \mathbf{e}_{j,1}) = d(\hat{\mathbf{l}}, \mathbf{e}_{j,2}), \hat{\mathbf{l}} = \frac{\mathbf{e}_{j,1} + \mathbf{e}_{j,2}}{2} \times \mathbf{v}, \quad (3.25)$$

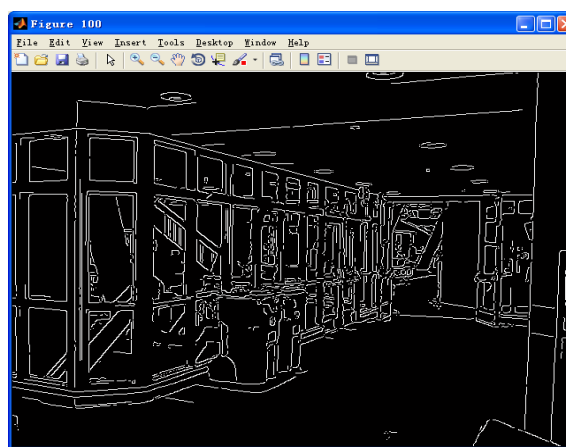
即给定一个灭点以及一条线段时，该线段对于这个灭点的一致性度量即为灭点和线段中点连线到线段任意一端点的距离。由于可以证明，灭点和线段中点的连线是过灭点的所有直线中能够最小化线段端到该直线距离的最大值的直线，故式(3.25)的含义就是这个最小化的最大距离。

### 3.3.3 实验结果

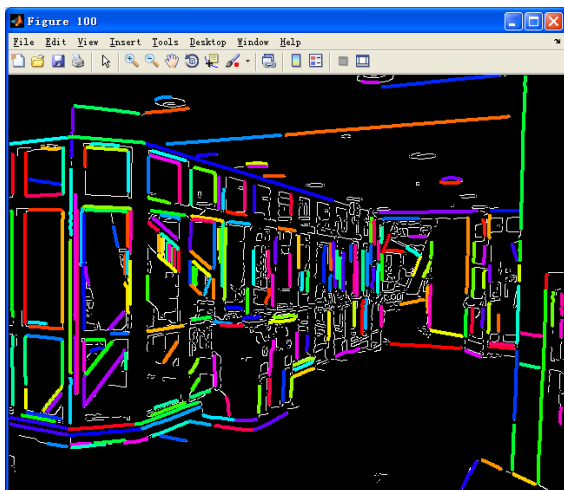
根据 J. P. Tardif 文献<sup>[57]</sup>中所提供的网址<sup>1</sup>，笔者下载并且编译了 Matlab 混合 C++ 程序 VPdetection，并且利用 York Urban 数据库<sup>[40-41]</sup>中的人工环境进行了实验。



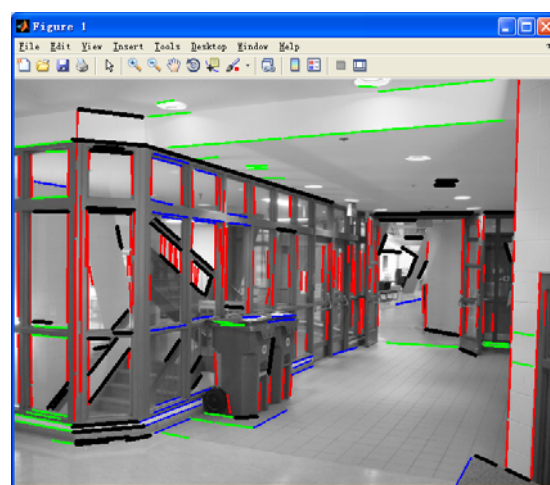
(a) 原始影像



(b) Canny 算子检测所得边缘



(c) 边缘分割所得的直线段



(d) 使用 J-linkage 对线段进行聚类

图 3.8 VPdetection 处理 York 数据库中 P1020816.jpg 图像的结果。(d) 中红绿蓝三色线段分别代表程序根据已有内参数检测出的三个相互垂直的主方向线段。

<sup>1</sup> <http://www-etud.iro.umontreal.ca/~tardif/fichiers/VPdetection.tar.gz>

对 York Urban 数据库中的 P1020816.jpg 图像进行自动灭点检测及标定实验，VPdetection 的输出结果如下：

```
[main] Ground truth Focal length is 674.917975
[FACADE_getEdgelets2] Found 245 lines
[FACADE_getEdgelets2] normalization and subpixel accuracy by line fitting
[FACADE_getVP_JLinkage] Not calibrated| SOLVE_VP: 2
[genHyp] done (500 trial, 500 sol)
[mexJLinkage] Done in 235 iteration
[FACADE_getVP_JLinkage] nb Class 10
[FACADE_selfCalib] focal length are 673.444158
```

可见算法检测得到 245 条线段，聚集成 10 类（即检测得到 10 个灭点），最终通过所聚集线段最多的前三个灭点标定所得相机主距结果为 673.444158 像素，与真值 674.917975 相差较小（注，这里默认相机主点为图像中心）。其线段聚类结果见图 3.8。

值得指出的是，通过大量实验可以发现，由于灭点标定问题本身的病态性，利用这种算法自动检测灭点后，进行灭点标定存在一定程度的不稳定性。

### 3.4 本章小结

本章首先介绍了灭点标定的相关理论推导，给出了利用三个空间对应方向相互垂直的灭点进行相机标定的理论公式。为了更加直观的介绍灭点标定，笔者给出了一个几何解释。之后，通过手动在影像上选取直线组的方式，在 Matlab 中编程实现了灭点标定的算法。且结合第 2 章 的相机模拟，进行了模拟数值分析实验，得出灭点标定受像点量测误差影像较大，标定精度不高，只能进行弱标定的结论。这一结论和相关文献的理论分析是一致的<sup>[49]</sup>。

然后，本章着重介绍了一种新的稳健估计方法——J-linkage。这种算法是针对数据点集合中存在同一模型的多个实例的情况而设计的，其基本思想是将数据点放入其相似概念空间中进行分析，而属于同一模型实例的数据点会聚集在相似概念空间中。而检测一幅图像中的灭点的任务实际上可以被化归为这种多实例模型的稳健估计问题，因此进一步介绍了一种利用 J-linkage 框架进行灭点检测的算法，并且给出了相关的实验结果。



## 第4章 单像建模

Don't get involved in partial problems, but always take flight to where there is a free view over the whole single great problem, even if this view is still not a clear one.

——Ludwig Wittgenstein

### 4.1 概述

从绪论的研究现状介绍（第 1.2 节）我们可以发现，利用几何结构信息进行三维重建方法可以大致分为两类，一类是基于模型的方法，另一类则是基于约束的方法<sup>[58]</sup>。前者所重建的结果是大量几何基元（如平行四面体、棱柱、圆柱等等）所组装形成的，而后者所处理的主要是场景的大量正规几何性质（如平行、正交等等）。

在基于模型的方法<sup>[22-23]</sup>中，场景是定义于一个 CAD 体系中的。例如图 3.1 中的建筑物，就可以分解为几个不同尺寸和不同位置长方体。通过影像量测数据来拟合这种 CAD 中的参数化模型，其尺寸、位置以及朝向就可以被确定下来。而“场景必须能被分解成几何形状基元”作为这种方法的假设前提，成为了限制该方法应用的主要因素。而在基于约束的方法中，这种限制就不存在了<sup>[20]</sup>。只要有足够的正规几何性质使得重建是唯一的，那么任何形状都可以适用。

如果令  $\mathbf{x} = \{x_i \mid x_i \in \mathbb{R}^2, i = 1, 2, \dots, N\}$  表示观察到的像点的二维坐标的集合， $\mathbf{X} = \{X_i \mid X_i \in \mathbb{R}^3, i = 1, 2, \dots, N\}$  表示对应待重建物方点的三维坐标的集合， $\mathbf{K}$  表示相机内参数矩阵，函数  $f, g$  分别表示相机模型和几何约束，那么基于约束的三维重建方法可以表示为

$$(\mathbf{X}, \mathbf{K}) = \arg \min_{\mathbf{X}, \mathbf{K}} \|\mathbf{x} - f(\mathbf{X}, \mathbf{K})\|^2, \quad (4.1)$$

且满足

$$g(\mathbf{x}, \mathbf{X}, \mathbf{K}) = 0. \quad (4.2)$$

注意，考虑到本文单像建模的目的是为虚拟现实等应用提供模型，并不需要模型的绝对坐标，故这里的函数  $f, g$  不同于第 2.2.2 节中的一般相机模型，与相机在世界坐标系中的平移量（ $\mathbf{C}$ ）、旋转量（ $\mathbf{R}$ ）无关。

现有的基于约束的单视影像三维重建的方法一般都是将各种正规几何性质表

达成测量值的线性约束，即(4.2)中的 $g$ 为 $\mathbf{x}$ 的线性函数。而通过分析可以发现，利用平面的法线方向可以很方便的将未知物方点三维坐标构成这样一个线性系统。因此，在进行重建之前，有必要先估计出场景中待重建的平面的法线方向。而这正好可以通过第三章中相机灭点标定所得到的相机内参数和各个灭点来完成。

本章首先将研究约束函数 $g$ 的表现形式，并依此得出一种序贯式（物方点的三维坐标是依次计算出来的）的单像建模算法，并给出该算法的实验结果以及分析；在指出该种方法的缺陷之后，结合现有研究成果，介绍并修改一种整体式重建算法，给出实验结果及分析。

## 4.2 序贯式重建

根据摄影测量和计算机视觉的基本理论，当一张相片经过标定，确定了其内参数之后，此时得到恢复的是光束之间的相对关系。即每量测相片上的一个点，等价于知道了空间中一条相对于相机主光轴确定的光束。而物方点即位于这条光束之上，唯一需要确定的就是点在这条光束上的位置。换句话说，就是此时的物方点由未标定前的三个自由度降为了只有一个自由度。

若 $\mathbf{p} \in \mathbb{R}^3$ 为某像点 $\mathbf{x} \in \mathbb{R}^2$ 所代表光束的单位方向向量，则物方点三维坐标 $\mathbf{X}$ 为

$$\mathbf{X} = \lambda \mathbf{p} \quad (4.3)$$

$$s\mathbf{p} = \mathbf{K}^{-1}[\mathbf{x}, 1]^T \quad (4.4)$$

其中 $\|\mathbf{p}\| = 1$ ， $s$ 表示任意尺度因子，而 $\lambda \in \mathbb{R}$ 表示物方点到相机光学中心的距离，满足 $\lambda > 0$ ，我们称之为物方点的深度，也就是待确定的那个自由度。可见单像建模的任务就是要确定像点集合 $\mathbf{x} = \{\mathbf{x}_i \mid \mathbf{x}_i \in \mathbb{R}^2, i = 1, 2, \dots, N\}$ 中各点所对应的深度的集合 $\{\lambda_i \mid \lambda_i \in \mathbb{R}, i = 1, 2, \dots, N\}$ ，或表示成 $\mathbf{\Lambda} = [\lambda_1, \lambda_2, \dots, \lambda_N]$ 。

若按照传统的摄影测量求解方式来考虑的话，如果不通过多视影像之间的同名点对应关系，这个深度集合是没有办法求解得到的。然而根据单像建模的研究可知，利用几何约束条件可以唯一或者相对的确定该集合。所谓相对的确定，意指解得 $\mathbf{\Lambda}' = s\mathbf{\Lambda}$ ， $s$ 表示任意尺度因子，类似于摄影测量中相对定向的步骤。如果有关于场景某几个物方点间的确定距离，或已知某一个或几个像点的深度，该尺度因子也可以得到确定，从而获得确定解，这也类似于摄影测量中绝对定向的步骤。

由上述可见，几何约束条件在单像建模中具有极其重要的意义，因此接下来我们就来研究物方点之间的各种几何约束的数学表达，以及由此启发得出的一种序贯式单像建模算法。

### 4.2.1 几何约束

#### A. 已知物方点约束

在所有约束中，这种约束是最为简单的。当我们已知一个物方点的空间坐标为  $(X, Y, Z)$  时（由于不考虑相机的平移以及旋转，世界坐标系即相机坐标系），根据式(4.3)，该约束的数学表达式为

$$\lambda = \sqrt{X^2 + Y^2 + Z^2}, \quad (4.5)$$

也就是说，此时深度即为物点到相机光学中心（即为原点）的距离。

#### B. 平行四边形约束

在人工场景中，最常见的正规几何约束就是平行约束，比如建筑的门窗等部分一般都满足对边平行且相等的条件。而由初等几何知识容易证明，空间中的四个三维点  $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4$  若能按顺序构成一个平行四边形，等价于满足以下条件

$$\mathbf{P}_1 + \mathbf{P}_3 = \mathbf{P}_2 + \mathbf{P}_4, \quad (4.6)$$

即对角线的中点必定重合。

利用这个性质，可以得到以下三个有关平行四边形约束的数学表达式。当已知空间平行四边形的某一个点的空间坐标时，通过剩余三个点的像点坐标可构成一个  $3 \times 3$  的线性方程组，从而求得相应三个点的深度。不失一般性，我们假设已知点为  $\mathbf{P}_1 = (X_1, Y_1, Z_1)^T$ ，剩余三个像点通过式(4.4)可计算得到其所对应光束的单位方向向量  $\mathbf{p}_i = (x_i, y_i, z_i)^T, i = 2, 3, 4$ ，则此时平行四边形约束表达为

$$\begin{bmatrix} -x_2 & x_3 & -x_4 \\ -y_2 & y_3 & -y_4 \\ -z_2 & z_3 & -z_4 \end{bmatrix} \begin{bmatrix} \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} -X_1 \\ -Y_1 \\ -Z_1 \end{bmatrix}, \quad (4.7)$$

类似的，对于已知点的个数为两个、三个时，平行四边形约束表达分别为

$$\begin{bmatrix} x_3 & -x_4 \\ y_3 & -y_4 \\ z_3 & -z_4 \end{bmatrix} \begin{bmatrix} \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} -X_1 + X_2 \\ -Y_1 + Y_2 \\ -Z_1 + Z_2 \end{bmatrix}, \quad (4.8)$$

$$\begin{bmatrix} -x_4 \\ -y_4 \\ -z_4 \end{bmatrix} \begin{bmatrix} \lambda_4 \end{bmatrix} = \begin{bmatrix} -X_1 + X_2 - X_3 \\ -Y_1 + Y_2 - Y_3 \\ -Z_1 + Z_2 - Z_3 \end{bmatrix}. \quad (4.9)$$

#### C. 共面约束

给定空间三个不共线的点，即可确定一个平面，而此时若量测了一个像点，且其物方点恰好位于此平面上，则相当于利用一条光线和此平面相交，即可确定该物方点的深度，这就是所谓的共面约束。

由基本的几何知识可知，若四个点  $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4$  共面，且任意三点不共线，则

必有以下几何条件

$$\mathbf{P}_4 - \mathbf{P}_1 = \alpha(\mathbf{P}_2 - \mathbf{P}_1) + \beta(\mathbf{P}_3 - \mathbf{P}_1), \quad (4.10)$$

其中 $\alpha, \beta$ 不同时为零。

由此性质，若已知的三个物方点为 $\mathbf{P}_i = (X_i, Y_i, Z_i)^T, i = 1, 2, 3$ ，则第四个点的深度可由以下线性方程组求得

$$\begin{bmatrix} x_4 & X_2 - X_1 & X_3 - X_1 \\ y_4 & Y_2 - Y_1 & Y_3 - Y_1 \\ z_4 & Z_2 - Z_1 & Z_3 - Z_1 \end{bmatrix} \begin{bmatrix} \lambda_4 \\ \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix}, \quad (4.11)$$

其中符号含义与式(4.7)相同。

#### D. 共线约束

给定空间两个点，即可确定一条直线，而此时若量测了一个像点，且其物方点恰好位于此直线上，则相当于利用一条光线和此直线相交，即可确定该物方点的深度，这就是所谓的共线约束。

由基本的几何知识可知，若有三个点 $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ 共线，且任意两点不重合，则必有以下几何条件

$$\mathbf{P}_3 - \mathbf{P}_1 = \gamma(\mathbf{P}_2 - \mathbf{P}_1), \quad (4.12)$$

其中 $\gamma \neq 0$ 。

由此性质，若已知的两个物方点为 $\mathbf{P}_i = (X_i, Y_i, Z_i)^T, i = 1, 2$ ，则第三个点的深度可由以下线性方程组求得

$$\begin{bmatrix} x_3 & X_2 - X_1 \\ y_3 & Y_2 - Y_1 \\ z_3 & Z_2 - Z_1 \end{bmatrix} \begin{bmatrix} \lambda_3 \\ \gamma \end{bmatrix} = \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix}, \quad (4.13)$$

其中符号含义与式(4.7)相同。

#### E. 垂直约束

给定空间中两个点，即可确定一条直线 $l$ 。而此时若量测了一个像点，且其物方点与前两个点中的某一个点的连线恰好垂直于 $l$ ，亦可确定该物方点的深度，即所谓的垂直约束。

由基本几何知识可知，若有三点 $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ 所成两条直线垂直，则必有以下几何条件

$$(\mathbf{P}_3 - \mathbf{P}_2)(\mathbf{P}_1 - \mathbf{P}_2) = 0, \quad (4.14)$$

从而可得其数学表达式为

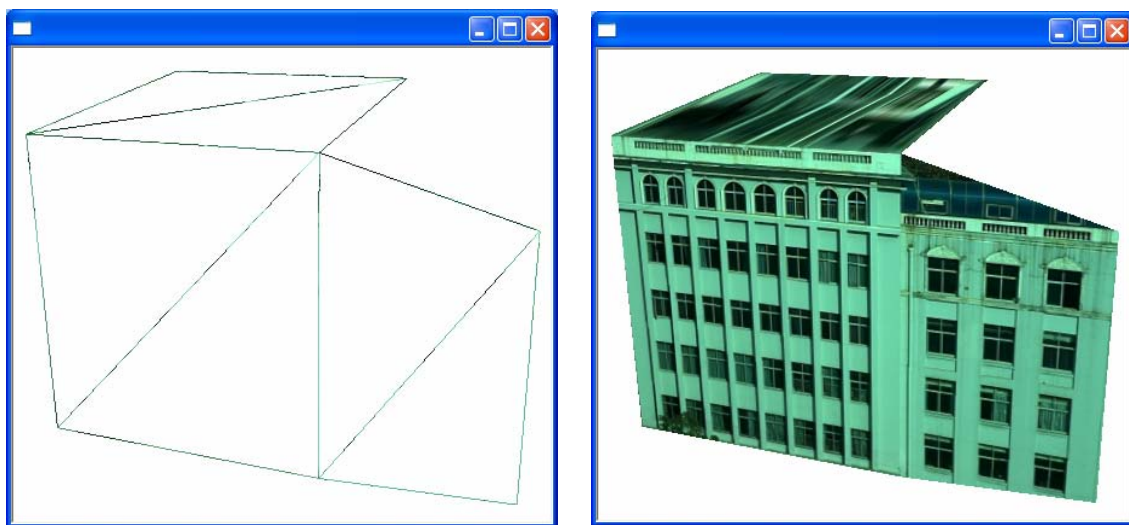
$$\lambda_3 = \frac{AX_2 + BY_2 + CZ_2}{Ax_3 + By_3 + Cz_3}, \quad (4.15)$$

其中  $A = X_1 - X_2, B = Y_1 - Y_2, C = Z_1 - Z_2$ , 符号含义与式(4.7)相同。

分析上述各种几何约束可以看出,除了“已知物方点约束”外,其他所有约束都依赖于某个或某几个已知点(可以是事先给出的,也可以是之前计算得到的)。这就启发我们得到一种序贯式的重建算法,即通过按照一定的顺序应用这些约束条件,将物方点依次重建出来。这种序贯式算法的特点在于,若已经重建了  $i$  个点  $P_1, P_2, \dots, P_i$ , 则第  $i+1$  个点必定可以由前  $i$  个点的三维坐标通过式(4.7)、(4.8)、(4.9)、(4.11)、(4.13)或(4.15)中的某一约束表达式求解得到。

## 4.2.2 实验结果

根据序贯式单像建模算法,笔者使用 C++结合计算机视觉开源库 OpenCV 1.2 (Open Computer Vision Library)和三维计算机图形学开源库 OpenSceneGraph 2.8,开发了一个序贯式单像建模的解算核心程序。



(a) 点线式框架

(b) 将原影像作为点线框架的纹理贴图

图 4.1 图 1.1 中武汉大学信息学部主教学楼的序贯式单像重建结果,依次使用式(4.5)、(4.7)、(4.8)以及(4.11)解算所得

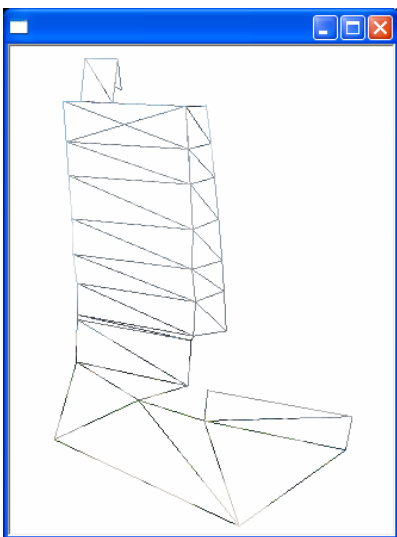
该控制台程序读入一个重建文件 (\*.cst), 其中分别包含影像文件内参数、量测所得像点坐标、第一点深度(即已知物方点约束)、有序约束列表(用于序贯式重建)以及三维构网的拓扑结构等六块信息,并最终生成物方点的三维坐标序列。



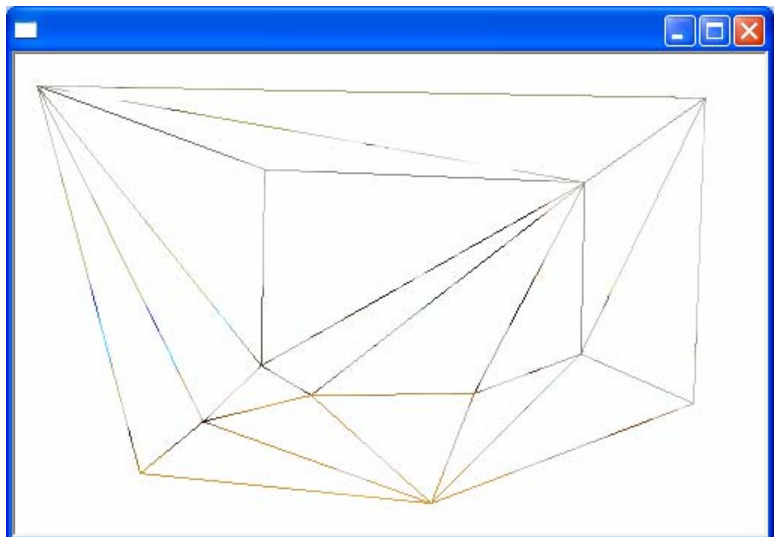
(a) 原始图像-室外场景



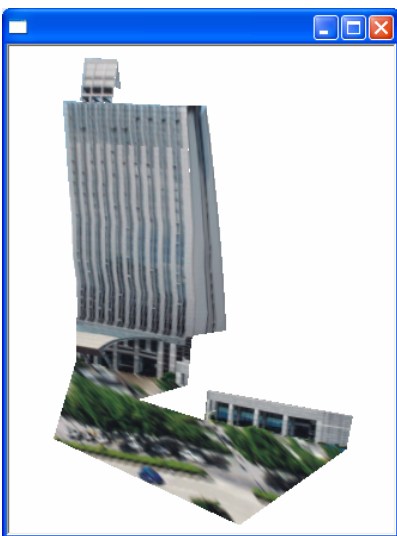
(b) 原始图像-室内场景



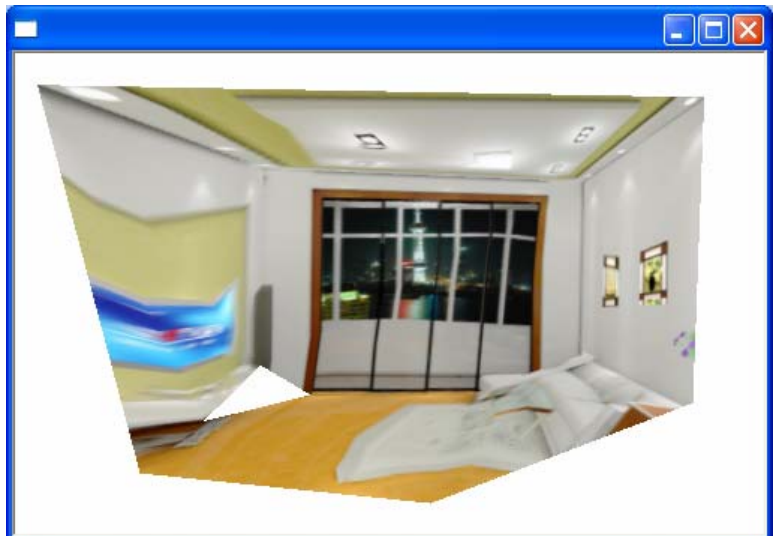
(c) 室外重建结果（框架）



(d) 室内重建结果（框架）



(e) 室外重建结果（附纹理）



(f) 室内重建结果（附纹理）

图 4.2 室内、室外场景的序贯式单像建模实例

通过以上几个实验可以看到，序贯法重建结果的视觉效果勉强能令人满意。

重新思考序贯法可以发现，其存在的一个最大的问题就在于“序贯”二字。由于像点的量测不可避免的存在误差，则序贯式解算所带来的问题就是误差的累积。另外，序贯式算法是一种恰定式而非超定式（over-determined）的算法，即提供的约束条件恰好足以求解出所有点的深度，因此无法进行平差计算。这些都会影响解算结果的视觉效果。

例如图 4.1 中的大楼正面，采用的约束是平行四边形约束，而实际上大楼正面是长方形，不仅仅是平行四边形约束，还应该存在垂直约束。故由于误差影响，实际解算出来的正面只能满足平行约束，而没有垂直的性质。

解决这个问题的方法无外乎提供更多约束，使之变成超定式系统。为此有两种修改思路。最直接的思路是类似文献<sup>[16]</sup>中所使用的方法，即首先通过原序贯式算法求解物方点的初值，然后提供更多的几何约束，通过含有约束条件的非线性优化来改进初始值。然而非线性优化存在各种问题（比如收敛性问题），这使得我们思考能否使用一种线性优化方法来求解。

### 4.3 整体式重建

实际上 Sturm 等人的方法<sup>[13]</sup>以及 Grossmann 的方法<sup>[17-21]</sup>就是整体的超定式线性解决方案。下面我们分别阐述基于 Sturm 和 Grossmann 算法，并与序贯式算法中的平行四边形约束相结合的改进后的整体式单像建模算法。

#### 4.3.1 基于 Sturm 算法

Sturm 等人的算法（以下简称 Sturm 算法）核心思想在于以下两点：

其一，利用灭点确定场景主方向。由于第 4.2.1 节中很多约束都可以表述为未知向量和已知向量的关系。如共面约束式(4.10)，可表述为未知向量和面法线垂直

$$(\mathbf{P}_4 - \mathbf{P}_1) \cdot \mathbf{n} = 0, \mathbf{n} = (\mathbf{P}_2 - \mathbf{P}_1) \times (\mathbf{P}_3 - \mathbf{P}_1) \quad (4.16)$$

同样，垂直约束式(4.14)也可以表述为未知向量和已知向量垂直

$$(\mathbf{P}_3 - \mathbf{P}_2) \cdot \mathbf{n} = 0, \mathbf{n} = \mathbf{P}_1 - \mathbf{P}_2 \quad (4.17)$$

而共线约束也可类似表达为

$$(\mathbf{P}_3 - \mathbf{P}_2) \times \mathbf{n} = 0, \mathbf{n} = \mathbf{P}_1 - \mathbf{P}_2 \quad (4.18)$$

由此可以考虑先利用灭点检测所得到的各个灭点来计算场景中的已知的方向（称之为主方向，dominant directions） $\mathbf{n}_1, \mathbf{n}_2, \dots, \mathbf{n}_M$ 。然后利用这些主方向，将原来的

某些非线性约束表达为线性形式。而主方向（即平面法向量）的计算可以有两种方式，若已知位于该平面的两个灭点的齐次坐标  $\mathbf{v}_i, \mathbf{v}_j \in \mathbb{R}^3$ ，则该平面法向量为<sup>[31]</sup>

$$s\mathbf{n} = \mathbf{K}^T \mathbf{l}, \mathbf{l} = \mathbf{v}_i \times \mathbf{v}_j \quad (4.19)$$

若已知该平面和某个灭点所代表的空间方向垂直，则该平面法向量为<sup>[31]</sup>

$$s\mathbf{n} = \mathbf{K}^T \mathbf{l}, \mathbf{l} = \mathbf{K}^{-T} \mathbf{K}^{-1} \mathbf{v} \quad (4.20)$$

其二，平面拟合。既然已经有了各个平面的法向量，则各种垂直约束实际上已经蕴含在这些法向量中了，接下来需要做的就是通过共面的点进行平面拟合，最小化各点到其各自所属平面的距离的平方和即可。

由式(4.3)，物方点已经通过其深度进行了参数化，类似的，已知法线的平面亦可进行参数化<sup>[13]</sup>

$$\Pi = \begin{bmatrix} \mathbf{n} \\ d \end{bmatrix} \quad (4.21)$$

其中  $\|\mathbf{n}\| = 1$ ，则  $d$  表示相机中心（原点）到该平面的距离。

若用户量测了  $N$  个像点  $[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ ，由式(4.4)得其单位方向向量  $[\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N]$ 。由式(4.19)或(4.20)计算得  $M$  个面的单位法向量  $[\mathbf{n}_1, \mathbf{n}_2, \dots, \mathbf{n}_M]$ ，且这  $M$  个面通过一些点连接起来，即对于任意两个面  $\Pi_i, \Pi_j$ ，至少存在一个点  $X_k$  同时位于其上， $\mathbf{X}_k \in \Pi_i \cap \Pi_j, \forall i, j = 1, 2, \dots, M, i \neq j, k = 1, 2, \dots, M$ ，则此时式(4.1)和(4.2)可以近似表达为

$$[\lambda_1, \lambda_2, \dots, \lambda_N \mid d_1, d_2, \dots, d_M] = \arg \min_{[\lambda|\mathbf{d}]} \sum_{i=1}^N \sum_{j=1}^M (d_j + (\mathbf{n}_j^T \mathbf{p}_i) \lambda_i)^2 C_{i,j} \quad (4.22)$$

$$C_{i,j} = \begin{cases} 1 & \mathbf{X}_i \in \Pi_j \\ 0 & \mathbf{X}_i \notin \Pi_j \end{cases}, \forall i = 1, 2, \dots, N, j = 1, 2, \dots, M \quad (4.23)$$

令  $h(\mathbf{d}, \lambda)$  表示式(4.22)中的目标函数

$$h(\mathbf{d}, \lambda) = \sum_{i=1}^N \sum_{j=1}^M \left( d_j^2 + 2(\mathbf{n}_j^T \mathbf{p}_i) \lambda_i d_j + (\mathbf{n}_j^T \mathbf{p}_i)^2 \lambda_i^2 \right) C_{i,j}$$

由多元函数极值理论可知，上式取极小值的条件为其偏导数为零，则有

$$\begin{aligned} \frac{\sigma h}{\sigma d_j} &= 2 \left[ \sum_{i=1}^N C_{i,j} d_j + \sum_{i=1}^N C_{i,j} (\mathbf{n}_j^T \mathbf{p}_i) \lambda_i \right] = 0, \forall j = 1, 2, \dots, M \\ \frac{\sigma h}{\sigma \lambda_i} &= 2 \left[ \sum_{j=1}^M C_{i,j} (\mathbf{n}_j^T \mathbf{p}_i) d_j + \sum_{j=1}^M C_{i,j} (\mathbf{n}_j^T \mathbf{p}_i)^2 \lambda_i \right] = 0, \forall i = 1, 2, \dots, N \end{aligned} \quad (4.24)$$

若将以上两式写成矩阵形式，则可以得到一个有关  $N+M$  个未知数的齐次线性



方程组，它有着非常优美的对称形式

$$\left( \begin{array}{cccc|cccc} A_1 & & & & D_{11} & D_{12} & \cdots & D_{1N} \\ & A_2 & & & D_{21} & D_{22} & \cdots & D_{2N} \\ & & \ddots & & \vdots & \vdots & & \vdots \\ & & & A_M & D_{M1} & D_{M2} & \cdots & D_{MN} \\ \hline D_{11} & D_{21} & \cdots & D_{M1} & B_1 & & & \\ D_{12} & D_{22} & \cdots & D_{M2} & & B_2 & & \\ \vdots & \vdots & & \vdots & & & \ddots & \\ D_{1N} & D_{2N} & \cdots & D_{MN} & & & & B_N \end{array} \right) \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_M \\ \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (4.25)$$

简写后即为

$$\begin{pmatrix} \mathbf{A} & \mathbf{D} \\ \mathbf{D}^T & \mathbf{B} \end{pmatrix} \begin{pmatrix} \mathbf{d} \\ \lambda \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} \quad (4.26)$$

其中

$$\begin{aligned} A_j &= \sum_{i=1}^N C_{i,j} \\ B_i &= \sum_{j=1}^M C_{i,j} (\mathbf{n}_j^T \mathbf{p}_i)^2 \\ D_{i,j} &= C_{i,j} (\mathbf{n}_j^T \mathbf{p}_i) \end{aligned} \quad (4.27)$$

可以使用奇异值分解（SVD，Singular Value Decomposition）算法求解该线性系统。

由前述可知，上述解法中已经包含了序贯法的大多数约束（共面，垂直），却未能包含平行四边形约束和已知物方点约束，因此将这两个约束加入，得到 Sturm 算法的修改版本，我们称之为整体式重建算法。

根据平行四边形约束式(4.6)，可得其条件方程为

$$\begin{bmatrix} x_1 & -x_2 & x_3 & -x_4 \\ y_1 & -y_2 & y_3 & -y_4 \\ z_1 & -z_2 & z_3 & -z_4 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (4.28)$$

若用户指定了  $K$  个平行四边形，根据上式可得一个新的齐次线性方程组

$$\mathbf{E} \lambda = \mathbf{0} \quad (4.29)$$

$3K \times NN \times 1 \quad N \times 1$

将该方程组与已知物方点约束式(4.5)一同并入式(4.26)可得

$$\begin{pmatrix} \mathbf{A} & \mathbf{D} \\ \mathbf{D}^T & \mathbf{B} \\ \mathbf{0} & \mathbf{E} \\ \mathbf{0} & \mathbf{F} \end{pmatrix} \begin{pmatrix} \mathbf{d} \\ \lambda \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \lambda_0 \end{pmatrix} \quad (4.30)$$

$M \times M \quad M \times N$   
 $N \times M \quad N \times N$   
 $3K \times M \quad 3K \times N$   
 $1 \times M \quad 1 \times N$

其中

$$\mathbf{F} = \begin{bmatrix} \mathbf{0} & 1 & \mathbf{0} \\ 1 \times (i-1) & & 1 \times (N-i) \end{bmatrix}$$

表示第  $i$  个像点的深度已知为  $\lambda_0$ 。将式(4.30)简写成

$$\mathbf{M}\mathbf{x} = \mathbf{q}$$

则可得其最小二乘意义下的解为

$$\mathbf{x} = (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \mathbf{q}$$

从而完成该单像建模算法的解算。

### 4.3.2 基于 Grossmann 算法

Grossmann 的算法（以下简称 Grossmann 算法）与 Sturm 算法的相同之处在于，都需要在进行重建之前，首先计算出主方向以用于构成线性系统。然而其与 Sturm 算法的最根本区别在于，它将所求物方点的坐标直接视为未知参数，而非利用像点的投影约束对其进行参数化。并且该算法将各种约束分为硬约束和软约束，从而保证解算结果必定满足一定的视觉特性。

若令像点  $[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ ,  $\mathbf{x}_i \in \mathbb{R}^2$  对应的待求物方点坐标为  $[\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_N]$ ,  $\mathbf{X}_i \in \mathbb{R}^3$  则 Grossmann 算法按照如下方法进行解算。

首先，最为直观的约束就是像点对于物方点的投影约束，即式(4.1)。Grossmann 算法中非常巧妙的利用了向量叉乘来表达了这一约束（而传统的摄影测量中，其对应的共线方程则是一个非线性的表达方式）

$$\begin{bmatrix} \mathbf{x}_m \\ 1 \end{bmatrix} \times (\mathbf{K}\mathbf{X}_m) = \varepsilon_m \approx 0 \quad (4.31)$$

上式所表达的几何意义即为，物方点与相机中心的连线应该与对应像点和相机中心的连线共线。由于像点的量测存在误差，因此这个约束是一个软约束，即应当进行最小二乘以优化等式左端的误差项  $\varepsilon_m$ 。将向量  $[\mathbf{x}_m, 1]^T$  表达为反对称阵形式

$$\begin{bmatrix} \mathbf{x}_m \\ 1 \end{bmatrix} \times = \begin{bmatrix} 0 & -1 & x_m^2 \\ 1 & 0 & -x_m^1 \\ -x_m^2 & x_m^1 & 0 \end{bmatrix} \quad (4.32)$$

则式(4.31)可以重写为

$$\left( \begin{bmatrix} 0 & -1 & x_m^2 \\ 1 & 0 & -x_m^1 \\ -x_m^2 & x_m^1 & 0 \end{bmatrix} \mathbf{K} \right) \mathbf{X}_m = \mathbf{S}_m \mathbf{X}_m = \varepsilon_m \approx 0 \quad (4.33)$$

从而可得所有像点投影约束构成的线性方程组

$$\begin{bmatrix} S_1 & & & \\ & S_2 & & \\ & & \ddots & \\ & & & S_N \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{bmatrix} = AX = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_N \end{bmatrix} \approx 0 \quad (4.34)$$

除了投影约束，就是物方点的几何约束了，即式(4.2)。Grossmann 算法考虑了共面，对称两种约束。

若两物方点  $X_n, X_m$  共属于同一个法线为  $\mathbf{n}$  的平面，则其约束为

$$\mathbf{n}^T(X_n - X_m) = 0 \quad (4.35)$$

若两物方点  $X_n, X_m$  关于第三个物方点  $X_1$  在某个方向  $\mathbf{n}$  上对称（如图 4.3），则

$$\mathbf{n}^T(X_1 - X_n) = -\mathbf{n}^T(X_1 - X_m) \quad (4.36)$$

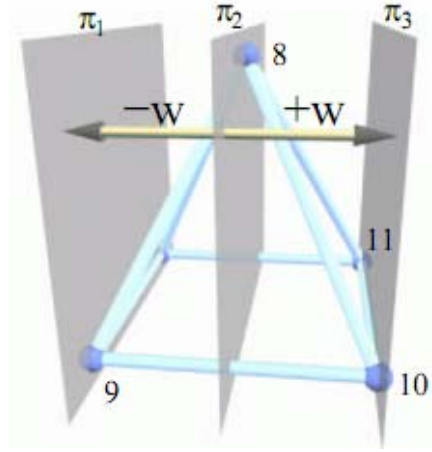


图 4.3 点 9 和点 10 关于点 8 在平面  $\pi_2$  的法线方向上对称<sup>[18]</sup>

将所有几何约束组成线性方程组，即的硬约束

$$BX = 0 \quad (4.37)$$

为了保证所求得的物方点必定满足上式，Grossmann 算法采用了以下解法：

1. 求解式(4.37)中矩阵  $B_{M \times (3N)}$  ( $N$  个物方点,  $M$  个约束) 的零空间 (nullspace) 的所有正交基所构成的矩阵  $U_{(3N) \times Q}$  (其每一列为一个正交基);
2. 与式(4.34)中的系数矩阵构成齐次线性方程组如下, 并利用 SVD 分解求解其非平凡解  $V^*$

$$(AU)V = 0 \quad (4.38)$$

3. 可得物方点坐标向量, 其中  $\mu$  为任意尺度因子

$$X = \mu UV^* \quad (4.39)$$

类似式(4.29), 我们可以将平行四边形约束加入式(4.37), 即的改进后的 Grossmann 算法。改进后的算法由于可以利用平行四边形约束, 从而使得我们可以不必预先求出场景中每一个需要求解的面的法线, 而直接利用平行四边形约束解

算得到部分特殊位置的点。

### 4.3.3 实验结果

依据以上介绍的整体式单像建模算法，笔者在 Matlab 中实现了其解算核心程序，并且对图 3.3 单位立方体的模拟相片进行了重建实验，图 4.4 显示了重建结果的两个不同视角，表 4.1 给出了重建所得的物方点三维坐标（指定第一点深度为 10）。从给出的重建结果可以看出，其效果十分理想。

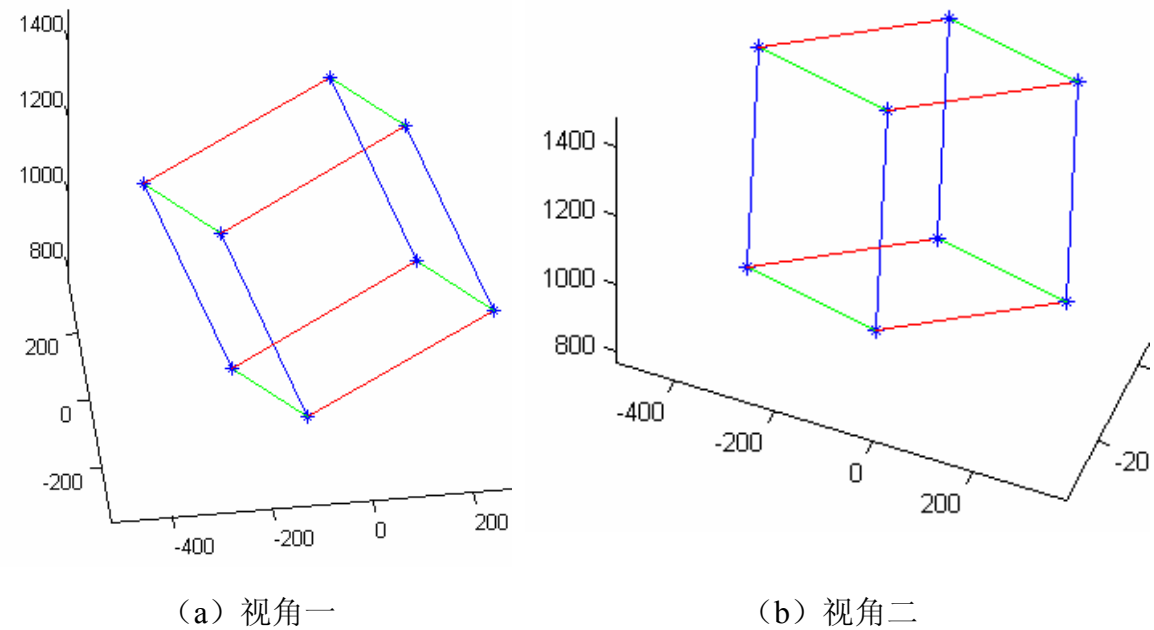
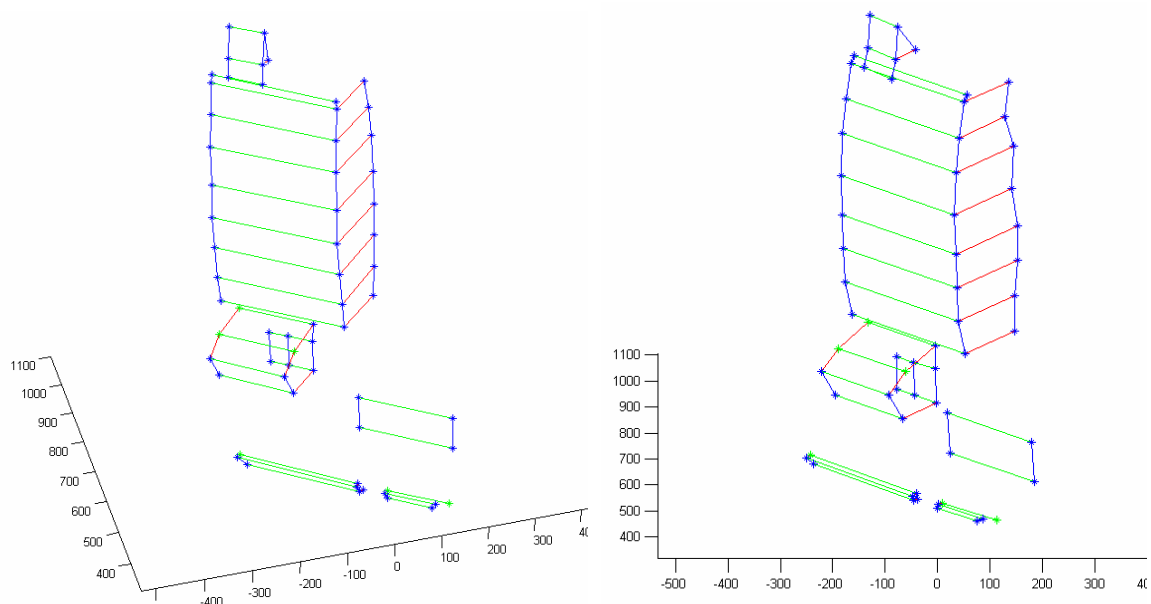


图 4.4 对图 3.3 单位立方体的模拟相片的重建结果，重建所使用的像点不含量测误差。

表 4.1 图 4.4 重建所得物方点三维坐标

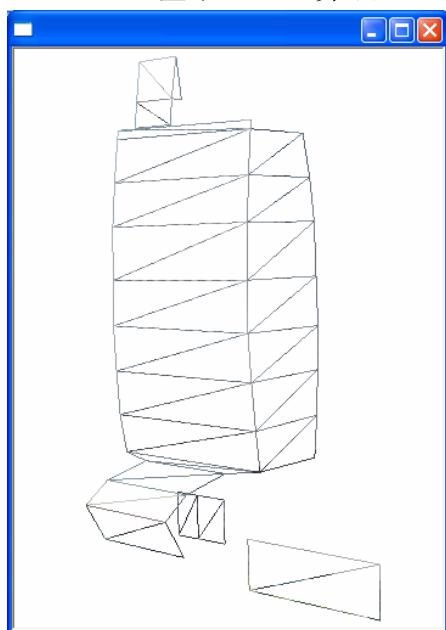
点号	X 坐标	Y 坐标	Z 坐标
0	-2.25821380869430	2.75233643711650	9.34479076978960
1	1.46794140285431	3.05904844625876	11.5134040723852
2	-0.308432657314151	0.978845911257058	14.8598130228550
3	-4.03458733575216	0.672133733820366	12.6911997729749
4	-0.977021123674627	-1.02388900696406	7.67750000054263
5	2.74913394768833	-0.717176883501363	9.84611333375984
6	0.972760545216229	-2.79737949784738	13.1925222454239
7	-2.75339466425196	-3.10409155769740	11.0239089737183

对比序贯式重建所得的结果，笔者利用整体式算法对图 4.2 中 (a) 图所示建筑物进行了重建实验，所得结果令人满意（如图 4.5）。



(a) 基于 Sturm 算法

(b) 基于 Grossmann 算法



(c) a 中数据点线式框架



(d) a 中数据贴纹理后结果

图 4.5 整体式算法对图 4.2 中 (a) 图所示建筑物的重建结果

对比上图 (a) 和 (b) 的几个细节可以发现，基于 Grossmann 算法的重建结果在部分细节上的视觉效果不如基于 Sturm 算法的结果。究其原因在于，用户指定的硬约束条件可能过强，而 Grossmann 算法又保证硬约束必须满足，在像点量测存在误差的情况下，导致重建结果反而不如 Sturm 算法。

另外，笔者还对其他真实数据进行了实验，使用的图像是 York Urban 数据库

中的图像（图 3.1 中右图）。其重建结果如图 4.6（指定第一点深度为 1000）。

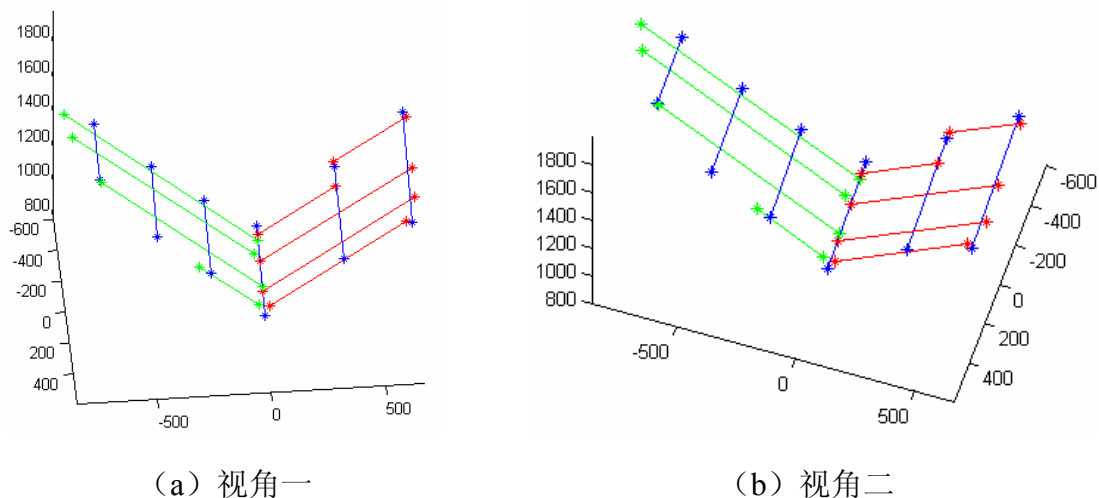


图 4.6 York Urban 数据库图像 P1030001.jpg 重建结果

类似第 3.2.3 节中的数值模拟误差分析，笔者也利用这张模拟影像对整体式重建算法对于像点量测误差的稳定性进行了数值分析。

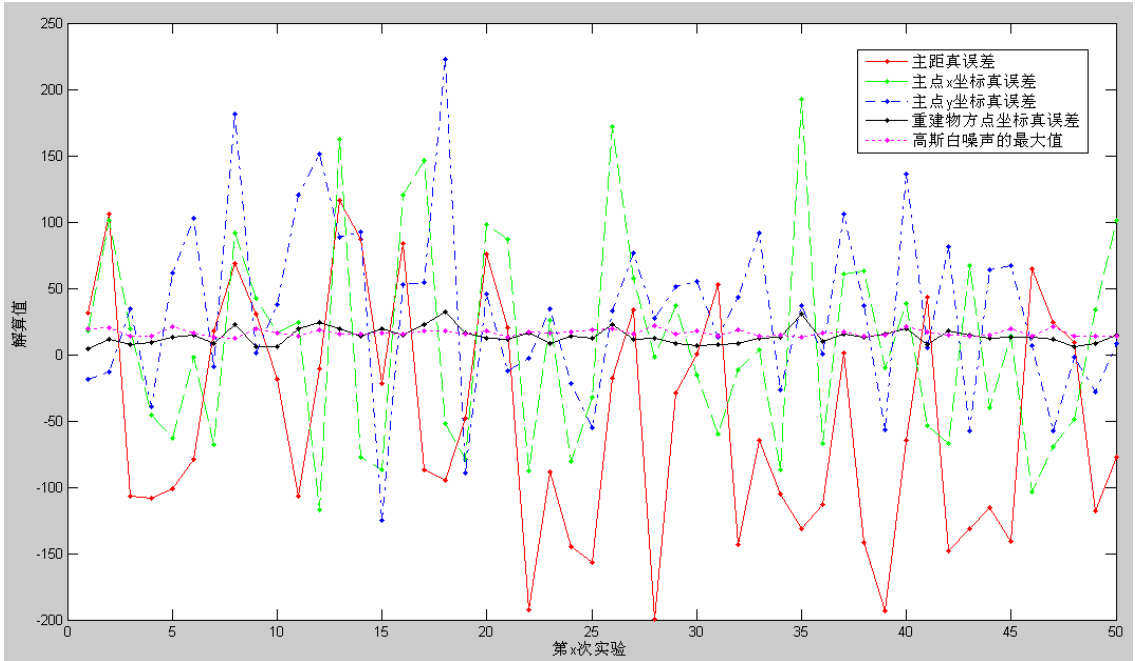


图 4.7 像点量测标准为 8 个像素，重复试验 50 次，整体式单像建模误差分析图

用于生成以上两幅误差分析图的数值模拟算法类似于算法 3.1 和算法 3.2，其中的重建物方点真误差的含义是，将不含误差的像点进行整体式重建所得到的结果（见表 4.1）认为是物方点的真值，与含有误差的像点重建所得结果比较，对应点距离之和即为此处的重建物方点真误差。

观察两幅误差分析图我们可以发现一个有意义的结论，即像点量测误差虽然对于灭点标定的影响很大（量测误差为 8 个像素时，主距的真误差可以达到 200 个

像素之大), 但是对于整体式单像建模的重建结果的影响却相对较小。这对于我们的重建是很有帮助的, 因为根据第 3 章 的结论, 灭点标定只能进行弱标定。而我们的模拟误差分析表明: 经过弱标定的图像即可用于单像重建。

同时, 笔者还对比了原始的 Sturm 算法与添加平行四边形约束后的整体式重建算法, 得到两者在同样误差条件下, 求解物方点坐标的真误差之差值的曲线图 (图 4.9)。由此图可以明显看出, 添加平行四边形约束后解算精度得到明显提高。

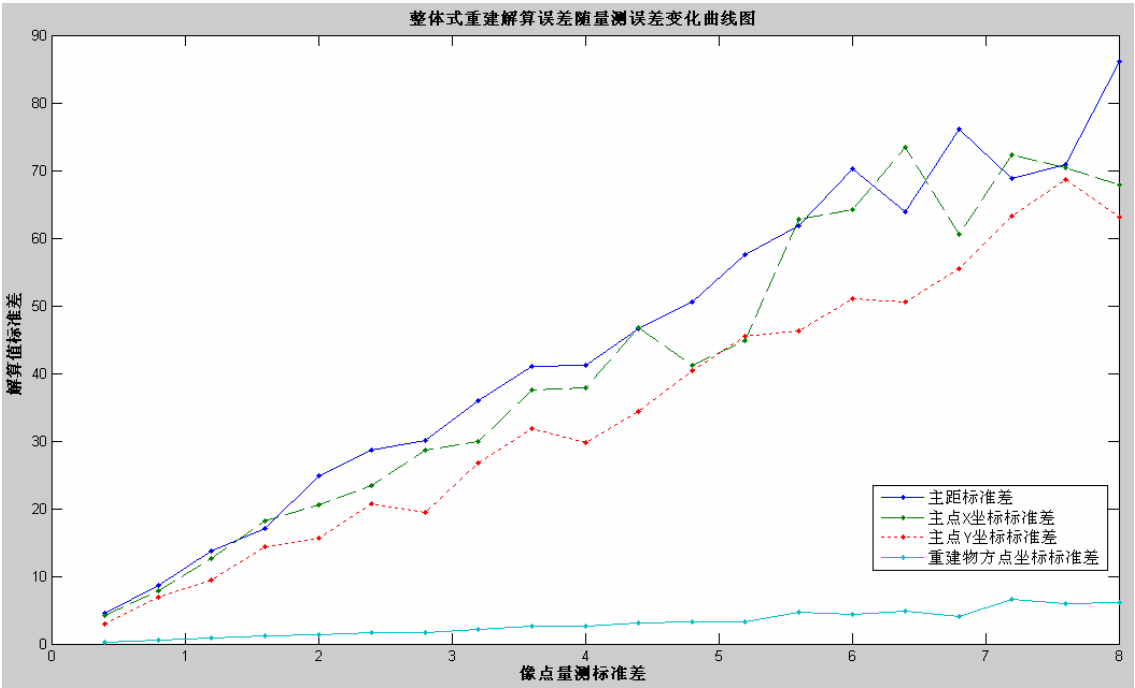


图 4.8 量测标准差从 0.4 变至 8 像素, 每次重复试验 100 次, 结果标准差分析图

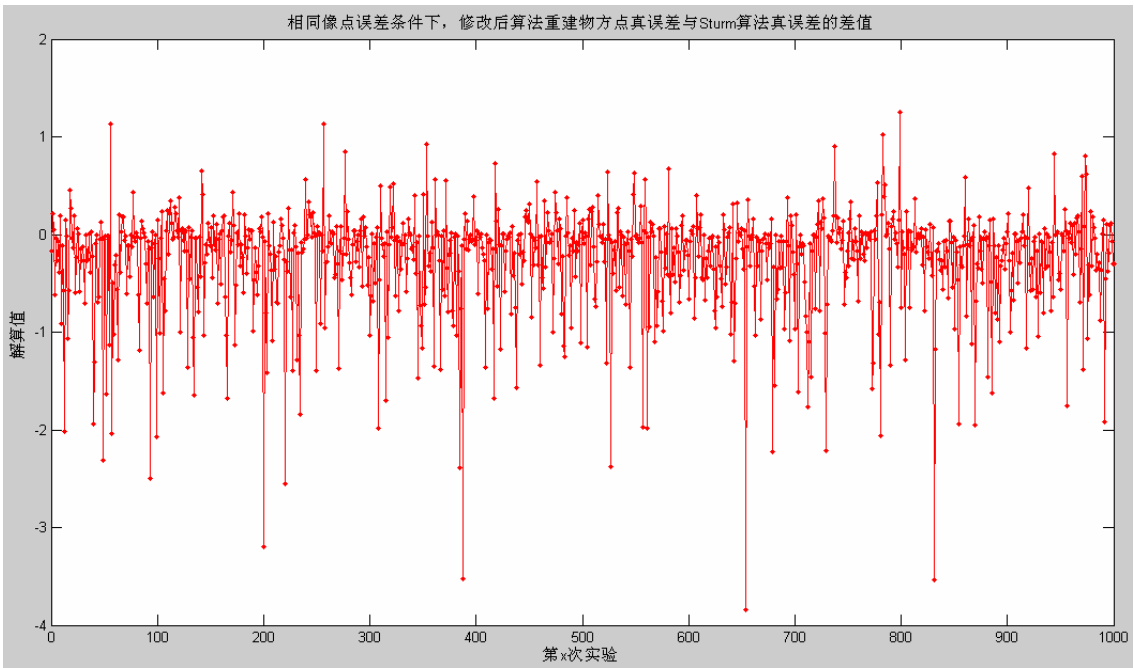


图 4.9 像点误差相同, 修改后算法重建物方点真误差与 Sturm 算法真误差的差值

## 4.4 本章小结

本章首先详细研究了作为基于约束的单像建模方法核心的各种几何约束条件，给出了其相应的数学表达式，并且受此启发，提出了一种直接的序贯式单像重建方法。在进行了相关实验后指出，序贯式单像重建的重要缺点在于是一种恰定式的系统，而非可以进行最小二乘估计的超定式系统，故而由于像点量测所引入的误差会不断累积，从而影响重建结果的视觉效果。

为了解决序贯式重建算法所存在的问题，本章接下来分别介绍了 Sturm 和 Grossmann 提出的线性最小二乘重建算法，指出两者均其包含了序贯式算法中的大部分约束条件，并且将序贯式算法中的其他约束条件纳入其算法框架内，修改其算法为整体式单像建模算法。

在之后的整体式单像重建算法的实验中，利用与第 3.2.3 节中类似的数值误差分析算法，分析了整体式单像重建算法对于像点量测误差的稳定性。实验结果表明，像点量测误差对于灭点标定的影响较大，但是对于整体式单像重建算法的影响相对很小，因此得出结论：经过弱标定的图像即可用于单像重建。对模拟影像和真实影像的实验以及和原始 Sturm 算法的对比表明，整体式单像重建算法是比较稳健，可以用于生成虚拟现实等应用中所需的三维模型数据。



## 第5章 总结与展望

本文首先从摄影测量的交叉学科——计算机视觉以及计算机图形学入手，研究了相机模型的不同表现形式，并且给推导出了相关的模型参数转换公式。在此基础上实现了相机模拟程序，并且指出，相机模拟程序在本研究中具有重要意义：它能够按实验者的要求直观的模拟相机的拍摄过程，并且获得相机内外参数以及严格符合相机模型的像点，这对于之后的相机自标定算法以及单像重建算法的验证与相关误差数值分析是十分必要的。

然后，本文讨论了相机自标定。通过对于本研究目的的分析指出，对于单像建模而言，灭点标定这种自标定方法是最为适用的。在完成了对相互垂直的三个灭点标定相机的理论推导之后，给出了一个极为直观的几何解释，说明了相机主点为何必定为这三个灭点所构成三角形的垂心。结合相机模拟程序，对于该灭点标定算法进行了验证，并且设计了一套数值模拟算法来检验其稳定性。模拟算法表明：灭点标定的解算标准差和像点量测误差的标准差近似呈线性关系，且在量测标准差为 2 个像素时，解算标准差被放大到 20 个像素左右，这和已有博士论文中给出的结论“（单像灭点标定）只能进行弱标定”是一致的。之后详细介绍了一种新的多实例模型稳健估计算法——J-linkage，简介并实验了其在灭点自动检测中的应用，最后指出：通过大量实验可以发现，由于灭点标定问题本身的病态性，利用这种算法自动检测灭点后，进行灭点标定存在一定程度的不稳定性。

最后是本文的核心环节——单像建模。在将单像建模抽象化描述为含约束的最优化问题之后，首先详细研究了各种几何约束及其数学表达形式。在此基础上提出一种启发式的序贯式重建方法，并且在实验后指出其重要缺点在于是一种恰定式的系统，而非可以进行最小二乘估计的超定式系统，故而由像点量测所引入的误差会不断累积，从而影响重建结果的视觉效果。为克服此缺点，结合之前研究的一种平行四边形约束，分别修改了 Sturm 和 Grossmann 的算法。再次结合相机模拟程序进行的数值模拟程序给出了一个很有意义的结论：像点量测误差虽然对于灭点标定的影响很大，但是对于整体式单像建模的重建结果的影响却相对较小，故经过弱标定的图像即可用于单像重建。实验还表明，添加了平行四边形约束的整体式算法的解算精度要高于原始算法。

根据已有的研究可以发现，适合进行单像建模的约束条件（即可以通过人的先验知识从图像中获得的）已经得到了大量的研究（平行、共面、垂直、对称等等），将来需要进一步研究的重点应该放在以下几个方面：

1. 灭点标定的自动化；
2. 场景几何约束的自动化或半自动化检测；
3. 多种单像建模思路的融合（基于纯几何约束，纹理，光照等等）。

这其中，特别是第二点，如果能够得以实现，则可以解决基于几何约束进行单、多像重建方法长期以来为人诟病的缺点——过多的用户操作（labor-intensive），而又不丢失基于几何约束的方法的最大优点——相对而言重建精度最高，视觉效果最好。

从绪论中可以看到，人脑完成单像建模这个任务所依赖的是其提取得到的所观测影像中的光照、阴影、纹理和几何结构等信息。然而现有的研究分别都是单独利用每一类信息进行重建，尚未出现综合利用多种信息进行单像重建的方法。这启示我们可以考虑融合多种单像建模思路。例如，采用光照、纹理等信息分别判断影像各区间所对应物方表面的法线方向，结合影像区间的边缘作为整体法单像重建的输入信息，这样的思路，很有可能改进现有的基于机器学习的自动化单像重建，应该作为下一步研究的重点。

## 参考文献

- [1] 侯文广. 基于普通数码相机实现三维重建的应用研究 [D]. 博士论文; 武汉大学, 2006.
- [2] 章毓晋. 图像处理和分析 [M]. 北京: 清华大学出版社. 1999.
- [3] YOUICHI H, KEN-ICHI A, KIYOSHI A. Tour into the picture: using a spidery mesh interface to make animation from a single image [M]. Proceedings of the 24th annual conference on Computer graphics and interactive techniques. ACM Press/Addison-Wesley Publishing Co. 1997.
- [4] GUILLOU E, MENEVEAUX D, MAISEL E, BOUATOUCH K. Using vanishing points for camera calibration and coarse 3D reconstruction from a single image [J]. The Visual Computer, 2000, 16(7): 396-410.
- [5] CRIMINISI A, REID I, ZISSERMAN A. A plane measuring device; proceedings of the 8th British Machine Vision Conference, Colchester, 1997 [C].
- [6] CRIMINISI A, ZISSERMAN A, VAN GOOL L J, BRAMBLE S K, et al. A new approach to obtain height measurements from video; proceedings of the SPIE, 1999 [C].
- [7] CRIMINISI A, REID I, ZISSERMAN A. Duality, Rigidity, Stratified Metrology and Planar Parallax; proceedings of the ECCV, 1998 [C].
- [8] CRIMINISI A, REID I, ZISSERMAN A. Single view metrology [J]. International Journal of Computer Vision, 2000, 40(2): 123-48.
- [9] CRIMINISI A. Single-view metrology: Algorithms and applications [M]. DAGM Symposium. 2002: 224.
- [10] LIEBOWITZ D, ZISSERMAN A. Combining scene and auto-calibration constraints; proceedings of the 7th International Conference on Computer Vision, Kerkyra, Greece 1999 [C].
- [11] LIEBOWITZ D, ZISSERMAN A. Metric rectification for perspective images of planes; proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 1998 [C].
- [12] LIEBOWITZ D, CRIMINISI A, ZISSERMAN A. Creating architectural models from images; proceedings of the EuroGraphics, 1999 [C].

- [13] STURM P, MAYBANK S J. A method for interactive 3d reconstruction of piecewise planar objects from single images; proceedings of the British Machine Vision Conference, Nottingham, England, 1999 [C].
- [14] STURM P. A method for 3D reconstruction of piecewise planar objects from single panoramic images; proceedings of the IEEE Workshop on Omnidirectional Vision, Hilton Head Island, USA, 2000 [C].
- [15] WILCZKOWIAK M, BOYER E, STURM P. Camera calibration and 3D reconstruction from single images using parallelepipeds; proceedings of the 8th IEEE International Conference on Computer Vision, Vancouver, Canada, 2001 [C].
- [16] LOURAKIS M I A, ARGIROU A A. Enforcing scene constraints in single view reconstruction; proceedings of the EuroGraphics, 2007 [C].
- [17] GROSSMANN E, ORTIN D, SANTOS-VICTOR J. Single and multi-view reconstruction of structured scenes; proceedings of the ACCV, 2002 [C].
- [18] GROSSMANN E, VICTOR J S. Using geometric cues for 3D reconstruction from a single view [R]: CVOnline, 2002.
- [19] GROSSMANN E, SANTOS-VICTOR J. Maximum likelihood 3D reconstruction from one or more images under geometric constraints; proceedings of the BMVC, 2002 [C].
- [20] GROSSMANN E, SANTOS-VICTOR J. Least-squares 3D reconstruction from one or more views and geometric clues [J]. Computer vision and image understanding, 2005, 99(2): 151-74.
- [21] GROSSMANN E. Maximum Likelihood 3D Reconstruction From One or More Uncalibrated Views Under Geometric Constraints [D]. PhD Thesis; Universidade Técnica de Lisboa, 2002.
- [22] DEBEVEC P E. Modeling and rendering from photographs: a hybrid geometry-and image-based approach [D]. PhD Thesis; Department of Computer Science, University of California at Berkeley, 1996.
- [23] TAYLOR C J, JELINEK D. Reconstruction of linearly parameterized models from single images with a camera of unknown focal length [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999, 23(767-74).
- [24] VAN DEN HEUVEL F A. 3D reconstruction from a single image using geometric constraints [J]. ISPRS Journal of Photogrammetry and Remote Sensing, 1998, 53(6): 354-68.

- [25] ZHANG L, DUGAS-PHOION G, SAMSON J S. Single-view modeling of free-form scenes [J]. The Journal of Visualization and Computer Animation, 2002, 13(4): 225-35.
- [26] HOIEM D, EFROS A A, HEBERT M. Automatic photo pop-up [J]. ACM Transactions on Graphics (TOG), 2005, 24(3): 584.
- [27] 张祖勋, 苏国中, 郑顺义, 张剑清. OpenGL 成像机理及其与摄影测量方位元素的相关分析 [J]. 武汉大学学报·信息科学版, 2004, 29(7): 570-4.
- [28] 苏国中, 郑顺义, 张剑清, 张祖勋. OpenGL 模拟摄影测量方法研究 [J]. 中国图象图形学报, 2006, 11(004): 540-4.
- [29] 王佩军, 徐亚明. 摄影测量学 [M]. 第一版. 武汉: 武汉大学出版社, 2005.
- [30] 冯文灏. 近景摄影测量——物体外形与运动状态的摄影法测定 [M]. 第一版. 武汉: 武汉大学出版社, 2002.
- [31] HARTLEY R, ZISSERMAN A. Multiple View Geometry in computer vision [M]. 2nd Edition. Cambridge: Cambridge University Press, 2003.
- [32] HORN B K P. Closed-form solution of absolute orientation using unit quaternions [J]. Journal of the Optical Society of America, 1987, 4(4): 629-42.
- [33] AHN S H. OpenGL Transformation [ES/OL]. 2008.  
[http://www.songho.ca/opengl/gl\\_transform.html](http://www.songho.ca/opengl/gl_transform.html)
- [34] FAUGERAS O. Stratification of three-dimensional vision: projective, affine, and metric representations [J]. Journal of the Optical Society of America A, 1995, 12(3): 465-84.
- [35] 丘茂林, 马颂德, 李毅. 计算机视觉中摄像机定标综述 [J]. 自动化学报, 2000, 26(1): 45-7.
- [36] TSAI R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses [J]. IEEE Journal of robotics and Automation, 1987, 3(4): 323-44.
- [37] TSAI R Y. An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision, 1986 [C]. IEEE Computer Society Press.
- [38] ABDEL-AZIZ Y I, KARARA H M. Direct linear transformation from comparator coordinates into object space coordinates in close-range

photogrammetry; proceedings of the Symposium on Close-Range Photogrammetry, 1971 [C].

[39] ZHANG Z. Flexible camera calibration by viewing a plane from unknown orientations; proceedings of the ICCV, 1999 [C].

[40] DENIS P, ELDER J, ESTRADA F. Efficient edge-based methods for estimating manhattan frames in urban imagery; proceedings of the European Conference on Computer Vision, 2008 [C].

[41] DENIS P. Efficient edge-based methods for estimating manhattan frames in urban imagery [D]. M.Sc. Thesis; York University, Canada, 2008.

[42] CAPRILE B, TORRE V. Using vanishing points for camera calibration [J]. International Journal of Computer Vision, 1990, 4(2): 127-39.

[43] COUGHLAN J M, YUILLE A L. Manhattan World: compass direction from a single image by Bayesian Inference; proceedings of the ICCV, 1999 [C].

[44] COUGHLAN J M, YUILLE A L. Manhattan world: Orientation and outlier detection by Bayesian Inference [J]. Neural Computation, 2003, 15(5): 1063-88.

[45] COUGHLAN J M, YUILLE A L. The Manhattan world assumption: Regularities in scene statistics which enable Bayesian inference [J]. ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS, 2001, 845-51.

[46] CIPOLLA R, DRUMMOND T, ROBERTSON D. Camera calibration from vanishing points in images of architectural scenes; proceedings of the British Machine Vision Conference, 1999 [C].

[47] 高满屯, 曲仕茹, 李西琴. 计算机视觉研究中的投影理论和方法 [M]. 西安: 西北工业大学出版社. 1998.

[48] 张祖勋, 吴军, 张剑清. 建筑场景三维重建中影像方位元素的获取方法 [J]. 武汉大学学报: 信息科学版, 2003, 28(003): 265-71.

[49] 谢文寒. 基于多像灭点进行相机标定的方法研究 [D]. 博士论文; 武汉大学, 2004.

[50] BARNARD S T. Interpreting perspective images [J]. Artificial Intelligence, 1983, 21(4): 435-62.

[51] FISCHLER M A, BOLLES R C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography [J]. Communications of the ACM, 1981, 24(6): 381-95.

- [52] STEWART C V. Bias in robust estimation caused by discontinuities and multiple structures [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(8): 818–33.
- [53] ZHANG W, KÖSECK J. Nonparametric estimation of multiple structures with outliers [J]. Dynamical Vision, 60–74.
- [54] ZULIANI M, KENNEY C S, MANJUNATH B S. The multiRANSAC algorithm and its application to detect planar homographies; proceedings of the IEEE International Conference on Image Processing, 2005 [C].
- [55] TOLDO R, FUSIELLO A. Robust multiple structures estimation with J-Linkage; proceedings of the ECCV, 2008 [C].
- [56] XU L, OJA E, KULTANEN P. A new curve detection method: randomized Hough transform (RHT) [J]. Pattern Recognition Letters, 1990, 11(5): 331–8.
- [57] TARDIF J P. Non-Iterative Approach for Fast and Accurate Vanishing Point Detection; proceedings of the ICCV, 2009 [C].
- [58] ROBERTSON D, CIPOLLA R. An interactive system for constraint-based modeling; proceedings of the British Machine Vision Conference, 2000 [C].

## 本科期间发表的论文和科研情况

### 发表论文

1. 邹进贵, **冯晨**, 控制网最小独立闭合环搜索算法研究。地理空间信息, 2008, 第 6 卷, 第 6 期, 97-99 页.
2. 邓非, 徐国杰, **冯晨**, 管海燕, 机载 LiDAR 点云数据与航空影像结合的建筑物重建. 测绘信息与工程(2009, 已接收).
3. Deng Fei, Chu YanLai, Zhang Peng, **Feng Chen**, Liang JingYong, The Visualization Simulation of Remote-Sensing Satellite System, The 2009 International Workshop on Virtual Changing Globe for Visualization and Analysis.
4. **冯晨**, 杨成, 谢玲, 基于切点检测的圆心定位算法, 2008, 中国大学生数学建模竞赛.
5. WAN Jun-kun, SHEN Wen-bin, YANG Qian, **FENG Chen**, Investigations of determining the gravity frequency shift by Doppler reduction technique. Geomatics and Information Science of Wuhan University (submitted).
6. WAN Jun-kun, SHEN Wen-bin, YANG Qian, **FENG Chen**, Experimental investigations of the geopotential difference between two stations based on the GPS signals. Surveying and Mapping Science, 2009, supplement: 23~25.
7. YANG Qian, SHEN Wen-bin, WAN Jun-kun, **FENG Chen**, WANG Jinling. Techniques for extracting the gravity frequency shift from GPS signals to determine the orthometric height. IGNSS Symposium 2009.Queensland.(accepted).

### 科研项目

1. 2009 至今, “单视影像三维重建”, 主要完成者, 导师: 邓非副教授
2. 2009, “交互式点云数据处理软件”, 完成者之一, 导师: 邓非副教授
3. 2008 至今, “利用 GPS 信号确定海拔高差”, 完成者之一, 导师: 申文斌教授
4. 2008, “数字武汉三维地理信息平台”, 主要软件研发人员, 导师: 邓非副教授
5. 2008, “海量三维数据管理工具”, 项目负责人, 导师: 邓非副教授

### 获奖情况

1. 2009, 夏坚白测绘优秀青年学子奖全国二等奖
2. 2009, “陈永龄院士优秀学生科技创新奖学金” 二等奖



3. 2008, 中国大学生数学建模竞赛湖北赛区一等奖
4. 2008, “百度杯” 华中北区大学生 ACM 程序设计竞赛一等奖
5. 2008, “高德杯” 科技论文写作大赛二等奖

## 致谢

四年时光，弹指一挥，转瞬即逝。回首沉思，种种迷惘、彷徨、焦虑、痛苦，已然逝者如斯，沉淀于心的是良师益友所带来的清晰、果断、冷静和幸福。

常言道，“师父领进门，修行在个人”。然而领你进门的人往往决定了你走进的是一扇什么样的门，进门后很长一段时间内你修行的种种也往往会自觉或不自觉的模仿自这个人。邓非老师对我而言就是这样一个人。从大一下学期在邓老师的指导下研究并完成大学生科研项目“海量三维数据管理工具”开始，到后来最终产生本文的单像建模项目，我从邓老师身上学到了很多。他不仅引导我进入了摄影测量、计算机视觉和计算机图形学的交叉研究领域，也教会了我如何管理时间、开展研究，言传身教中更让我学到了研究之外的种种知识。邓老师，谢谢您！

“仰之弥高，钻之弥坚”，这是我在申文斌老师指导下进行全国大学生科研“利用 GPS 信号测定海拔高差”的过程中对他的强烈感受。他不仅有着很高的学术成就，且平易近人。尤记在茶馆讨论项目的那个下午，他与我们分享做研究的心得，查阅文献、学习数学工具的方法。“授人以鱼，不如授人以渔”，申老师，谢谢您！

我要感谢茅盛、李刚、李海亮和董苏扬等几位师兄。我很怀念 08 年夏天的工作室和华莱士炸鸡，301 机房中关于计算机视觉技术的讨论，以及经管院内演讲台上的青春与激昂，他们渊博的知识、对技术的把握和对生活的激情，使我获益匪浅。而在我申请出国深造的一路上，没有他们的支持，我很难走下来。

我要感谢李端顺、卢亮、陈林、李俊峰、楚燕来、沈智娟、覃宇、王煦、杨司飞、万俊堃和杨茜等同学。他们对数学的直觉、对挫折的淡定、对技术的追求、对理想的坚持、对人的真诚、对社会的适应和对理论的钻研，深深地影响着我。

我要感谢周涵超和康崴铃。初中相识、高中同班、到大学同校，我们胜似兄弟。

最后，我要感谢我的父亲冯美君先生和母亲邓祝容女士。他们给了我足够的自由空间来做感兴趣的事，使我发挥所长；授予我丰富的社会经验，使我能处理好学习、生活和工作中的种种问题；指出并包容我的种种缺点，使我有充分的时间接受并改正。还要感谢我的女友李莎莎，在我为实验焦躁的时候，给我鼓励和支持，陪伴着我完成了本文。

北美研究生入学考试写作部分中有一题要求论述“我们主要通过与社会群体的认同而来界定我们自身”，我想，以上就是我的回答。