合肥工业大学

硕士学位论文

基于LiDAR点云和航空影像的城市三维重建

姓名：朱晓强

申请学位级别：硕士

专业：计算机应用技术

指导教师：刘晓平

20090401

基于 LiDAR 点云和航空影像的城市三维重建

摘　要

随着 GIS 的迅速发展和数字城市的快速建模的需要，对城市环境中各种三维信息的表达与处理变得日益迫切。近几年 LiDAR（Light Detecting And Ranging）技术的出现，为解决这一问题提供了新的途径。机载 LiDAR 获取的数据是分布于对象表面的三维点坐标，其数据集(点云)是对象的数字表面模型(DSM)。由于机载 LiDAR 能够自动地获取高精度、高密度的地球表面 3D 坐标信息，已成为生成数字地面模型(DTM)和数字高程模型(DEM)的首选工具。

针对中、低密度的机载 LiDAR 数据和航空影像的城市三维重建工作，本文主要从以下几个方面开展研究：

（1） 机载 LiDAR 技术的发展使得对大规模地表非均匀点云构网成为研究的热点，本文在 Delaunay 三角网生长法的基础上改进，提出一种新的三角网生长算法。该算法先对大规模点云进行等格网分块，然后自适应确定搜索范围。构建过程中对生成的基线分组、排序，动态删除封闭点，提高了构建三角网的速度；在整个点集范围内进行搜索，避免了插值产生的误差和模块之间的拼接。最后对大规模 LiDAR 点云数据构网，表明了该算法的有效性。

（2） 根据城市地形变化缓慢的特性，在三维 Hough 变换的基础上添加约束条件，快速生成地形的基准平面，然后根据点云 TIN 网格与基准平面的位置关系以及 TIN 三角形法线的方向提取地面点。为保持场景地形的边界，本文提出一种新的二维点云凸包生成算法，首先对点云进行分块和在多方向生成最远点而得到初始凸包，并在凸包交界的局部区域内搜索最优点来扩展初始凸包，从而提取 LiDAR 点云凸包。最后结合提取的地面点和场景的凸包生成最终的地形 TIN 模型，该方法对点集的密度要求低，提取城市地形的鲁棒性高。

（3） 从航空或卫星影像中提取道路一直是研究的热点，基于动态规划的道路提取算法是最有效的算法之一。本文基于 LiDAR 点云数据特征改进了该算法的代价函数，进而提高了基于动态规划的道路提取算法的鲁棒性。为正确地融合航拍图像和 LiDAR 点云数据，本文提出了一种航拍图像和 LiDAR 点云数据的匹配算法。最后通过试验验证了算法的正确性。

关键词：LiDAR，航空影像，三维重建，道路提取，动态规划，Hough 变换

本论文主要工作得到了国家自然科学基金国际合作交流项目“一种新的经济和环境规划分析工具，LiDAR 应用领域的扩展”（60573174）、特种显示技术教育部重点实验室开放课题基金项目“基于 LiDAR 点云数据及图像的城市建模”及炮院合作项目“数字地图与航空图像匹配系统研制”（070503D2）的共同资助。

The 3D Reconstruction of City Based on the LiDAR Point-Cloud and Aerial Images

ABSTRACT

With the rapid development of GIS and the necessity of fast modeling of digital cities, it is eager to express and deal with various 3D information in the cities. In recent years, the emergence of LiDAR provides a new way to solve the problem. The data obtained by LiDAR are 3D coordinates of points on the earth surface, which express the Digital Surface Model. LiDAR has become the preferred tool to generate Digital Terrain Model and Digital Elevation Model because it can obtain high-definition and high-density 3D coordinates of earth surface automatically.

According to the medium and low density LiDAR and aerial images, several aspects about urban 3D reconstruction are studied as follows:

(1) With the development of LiDAR, building the TIN model with the non-uniform point-clouds of the terrain surface becomes a hot research field. Based on the existing Delaunay triangulation method, a new algorithm of triangulation growth is presented in this dissertation. The algorithm divides the large-scale point clouds into uniform grids and determines the searching scope self-adaptively. During the process of building a Triangulated Irregular Network (TIN) model, the generated base-lines are grouped and the close-points are removed dynamically, which can improve the speed of reconstructing TIN in large-scale scenes dramatically. By searching the triangular vertices in the scope of the whole data set, the method can avoid errors caused by interpolation and the process of stitching between grids. The efficiency and effectiveness of the algorithm are verified by using real world data to build TIN model with large scale LiDAR point clouds.

(2) According to the traits of urban slowly-changing terrain, based on the 3D Hough transform, restricted conditions are imposed in the dissertation, which firstly builds the basic plane of the terrain, and then extract the points on the ground in term of the position relation between the point-cloud TIN and the basic plane, and of the direction of the normal of the TIN. In order to preserve the boundary points, a new algorithm of building 2D convex hull is discussed. Firstly the points are divides into grids and the initialized convex hull is gained by

seeking for the farthermost points in varied directions, then the initialized convex hull is expanded iteratively by searching the best points in local common boundary area, and finally the full convex hull from the LiDAR point-cloud can be obtained. The algorithm, which builds the final terrain TIN model by combining the points on the ground and the points of the convex hull, has low need of density of the point-cloud but has high robustness in the extraction of urban terrain.

(3) The road extraction from aerial images is always a research hotspot, and the algorithm of road extraction based on dynamic programming is one of the most efficient algorithms, which is improved in this dissertation based on LiDAR point-cloud, and the robustness is also inhanced. In order to combine the aerial image and the LiDAR point-cloud data, the algorithm of matching the aerial image with the LiDAR point-cloud data is put forward. In the end, experiments are carried out to validate the algorithm.

Keywords: LiDAR; Aerial Images; 3D Recontruction; Road Extraction; Dynamic Programing; Hough Transform

图表目录

图 1-1 机载 LiDAR 示意图 1
图 2-1 Voronoi 图和 Delaunay 三角剖分 8
图 2-2 LOP 算法 9
图 2-3 查找 Delaunay 第三点 10
图 2-4 基线右侧扩展流程 12
图 2-5 均匀条带分割法 13
图 2-6 等格网分割法 13
图 2-7 四叉树分割法 13
图 2-8 新添加的基线 13
图 2-9 点的搜索范围 14
图 2-10 排序前 14
图 2-11 排序后 15
图 2-12 边界点的处理 16
图 2-13 点云数据 17
图 2-14 LiDAR 点云二维网格 18
图 2-15 LiDAR 点云的 TIN 模型三维显示 18
图 3-1 三维 Hough 变换原理图 20
图 3-2 Hough 变换检测 LiDAR 地面参数约束 21
图 3-3 美国奥尔良城区的 LiDAR 点云 TIN 模型 21
图 3-4 提取的基准平面 22
图 3-5 提取的地面点所在的三角形 22
图 3-6 八邻域方向 23
图 3-7 内部点删除示意图 24
图 3-8 求取不同方向最远点示意图 25
图 3-9 扩展初始凸包 25
图 3-10 提取的地形网格模型 26
图 4-1 利用动态规划提取道路的迭代过程 30
图 4-2 城市 LiDAR 点云数据 31
图 4-3 高度图像 31
图 4-4 边缘图像 32
图 4-5 边缘距离图像 32
图 4-6 配准前的航拍图像 33

图 4-7 LiDAR 高度图 34
图 4-8 配准后的航拍图像 34
图 4-9 控制点位置 35
图 4-10 提取的道路 35
图 4-11 道路重建后 36
图 4-12 USAN 的三种典型形状 37
图 4-13 建筑物角点检测结果 38
图 4-14 建筑物原始模型 38
图 4-15 重建的建筑物与原始模型比较 39
图 4-16 未加纹理的建筑物重建 39
图 4-17 带纹理的建筑物重建 40

表 2-1 算法比较结果 19

独 创 性 声 明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标志和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得__合肥工业大学__或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签字：朱晓强 签字日期：2009年4月10日

学位论文版权使用授权书

本学位论文作者完全了解__合肥工业大学__有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅或借阅。本人授权__合肥工业大学__可以将学位论文的全部或部分论文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、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名：朱晓强 导师签名：

签字日期：2009年4月10日 签字日期：09年4月10日

学位论文作者毕业后去向：

工作单位： 电话：

通讯地址： 邮编：

致 谢

岁月如歌，光阴似箭，三年的研究生生活即将结束。值此论文完成之际，我要感谢长期以来对我提供热心帮助和关心的老师和同学。

首先，我要由衷地感谢我的导师刘晓平教授，三年的成长离不开他循循善诱的教导和无微不至的关怀。他严禁的治学态度、勤奋的工作作风令我敬仰，每每与刘老师相处畅谈，总是获益匪浅。和刘老师一起学习、工作的三年，是我进步最快的时期之一。刘老师不仅精心点拨我的学术研究，而且教我如何做人，这些都让我终身受益，也是我今后不断成长的源泉。

其次，我要感谢 VCC（可视化与协同计算研究室）的各位老师和同学们，他们在工作学习和日常生活中给了我诸多帮助和关怀。感谢罗月童、郑利平、路强、李琳、徐本柱、石慧老师对我的研究工作提供了很多宝贵的意见：感谢本课题组余烨、凌实、张定东、徐伟、薛晔、孙静、谢文军、钱晶晶、王晓静、闫锋、王启骏、陈韬、瞿德清、王玉培、伍国永等同学的通力合作，本文的工作与他们的辛勤劳动是分不开的，感谢他们一直以来对我的支持和鼓励。此外，感谢吴正同学对论文内容认真仔细的校对。

再次，感谢所有关心和帮助过我的老师、同学和朋友。谢谢他们一路上对我的关怀和协助。

最后要感谢我最亲爱的父母，感谢他们对我的理解和支持。

时光荏苒，三年的研究生学习生活即将结束，回想过去，思绪万千，各位老师、同学、朋友和亲人给我的关怀和帮助，使我永远难忘，这些经历会是我人生中不可多得的财富。

作者：朱晓强

2009 年 04 月 07 日

第一章 绪 论

1.1 研究背景

“数字城市”是在信息技术的飞速发展带来城市产业结构和社会发展转型的背景下提出来的，作为当前城市信息化建设的中心内容之一，“数字城市”是以计算机技术、多媒体技术和大规模存储技术为基础，以带宽为纽带，综合运用3S技术（RS遥感、GIS地理信息系统、GPS全球定位系统）、人工智能技术、虚拟仿真技术等对城市进行多分辨率、多时相、多尺度和多种类的三维描述和分析；也就是利用信息技术手段，建立城市空间的基础信息数据库，对城市所有基础设施、地理信息和社会信息进行动态检测、组织管理和应用服务的多功能、智能化的技术系统。高精度、高时效、低成本、信息共享是每一项遥感技术的发展目标，新一代激光雷达系统的出现，使得快速，高效地建立数字城市成为可能[1]。

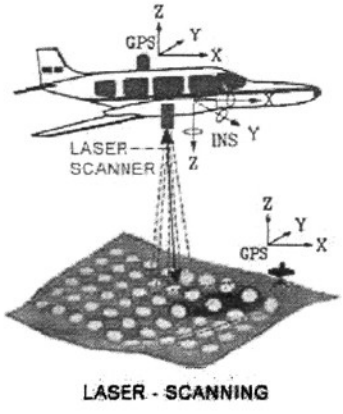


图 1-1 机载 LiDAR 示意图

LiDAR（如图1-1所示）技术是一种利用光束来探测物体和测定距离的高科技集成系统，机载GPS提供LiDAR系统的空间位置，惯性测量系统提供LiDAR激光的方向，激光系统提供激光脉冲，计算机系统提供高速、大规模数据存储空间与处理能力，代表着当前数码测绘技术的前沿。

机载LiDAR系统出现之前，地面信息的获取主要是依赖于摄影测量学。摄影测量学有着悠久的历史，从19世纪中叶至今，它从模拟摄影测量开始，经过解析摄影测量阶段，现在正向数字摄影测量阶段发展。解析摄影测量曾广泛应用生产各种地图，获取数字表面模型DSM（Digital Surface Modal）、数字地面模型DTM（Digital Terrain Models）、正射影像等。随着相关软硬件技术的发展，数字摄影测量系统DPS（Digital Photogrammetry System）的出现大大降低了成本，缩短了处理时间。特别是利用DPS进行作业，大量的处理工作，比如影像的匹配，DSM、DTM的生成，特征提取，地图的制作等基本实现自动化。然而，DSP通过立体像对匹配自动生成的DSM、DTM的精度比利用解析测图系统获得

同类产品的精度低。当相片纹理及对比度较弱，阴影的影响都会使精度迅速下降。因此人们不断研究用于生成更高精度DSM、DTM产品的许多新技术。20世纪80年代末，以机载LiDAR高技术为代表的空间对地观测技术在多等级三维空间信息的实时获取方面产生了重大突破。最近几年，随着相关技术的发展进步以及社会需求的不断扩大，机载LiDAR技术的发展日新月异。机载LiDAR系统能够快速获取精确的高分辨率的数字地面模型以及地面物体的三维坐标，进而获取地表物体的垂直结构形态，同时配合地物的视频或红外成像结果，更加增强了对地物的认识和识别能力，在摄影测量与遥感及测绘等领域具有广阔的发展前景和应用需求。机载LiDAR技术的发展为获取高时空分辨率的地球空间信息提供了一种全新的技术手段，使人们从传统的人工单点数据获取变为连续自动数据获取，提高了观测的精度和速度，使数据的获取和处理朝智能化和自动化的方向发展[2]。

1.2 国内外研究现状

近年来，国内外学者对于 LiDAR 的应用做了大量的研究。其主要研究集中在：LiDAR 数据的矫正和匹配问题、基于近距离小功率 LiDAR 测距器的目标的表面重建研究、以及基于正射影像或遥感影像的房屋建模研究等等。目前基于 LiDAR 数据的目标提取研究难点在于目标的轮廓提取比较困难。这主要和所提供的数据精度和数据质量有关。

1.2.1 TIN 构建

利用点集构建不规则三角网（Triangulated Irregular Networks，TIN）的方法目前可分为 3 类：分治-合并算法、逐点插入算法和生长算法。

（1） 分治-合并算法

Shamos 和 Hoey 提出了分治算法思想，并给出了一个生成 Voronoi 图的分治算法。Lewis 和 Robinson 将分治算法思想应用于生成 Delaunay 三角网。他们给出了一个“问题简化算法”，递归地分割点集，直至子集中只包含三个点而形成三角形，然后自下而上地逐级合并生成最终的三角网。以后 Lee 和 Schachter 又改进和完善了 Lewis 和 Robinson 的算法。

（2） 逐点插入法

Lawson 提出了用逐点插入法建立 Delaunay 三角网的算法思想。Lee 和 Schachter，Bowyer，Watson，Sloan，Macedonio 和 Pareschi，Floriani 和 Puppo，Tsai 先后进行了补充和完善。

（3） 三角网生长法

Green 和 Sibson 首次实现了一个生成 Dirichlet 多边形图的生长算法。Brassel 和 Reif 稍后也发表了类似的算法。McCullagh 和 Ross 通过把点集分块和排序改进了点搜索方法，减少了搜索时间。Maus 也给出了一个非常相似的算法。

1.2.2 地形提取

在地形滤波方面，目前研究人员通常采用类似图像处理的方法[3][4][5][6]，或者利用不规则三角网提供的点间邻接关系来设计滤波算法[7][8]。在城市环境下的 DTM 生产中，各种复杂的建筑物，如多层排列的建筑物、天井、广场、楼梯以及地下通道等，加上裸露地面上的不连续特征，使得现有滤波算法面临困难[9]。人们需要通过大量繁琐的手工编辑来获得准确的裸露地面点。据统计，在实践中，要花费 60%～80%的时间进行手工操作（包括滤波和质量控制）[9]。因此，设计稳健、高效、准确的滤波算法显得尤为重要。

而有关点云的凸壳生成，目前较为经典的平面点集凸包计算方法大体上可分为点排序与点不排序两类[10]，根据不同算法思想可分增量法[11]、格雷厄姆扫描法[12]、卷包裹法[13]、分治法[14][15][16]、实时凸包算法[17]和快速凸包算法[18]等，文献[19][20][21]对现有的各种算法进行了思路和算法复杂度的分析说明。在大多数情况下，凸包只由点集中部分点构成，其余点则存在于凸包内部。因而可以通过检测一些特殊点（如最远点[22]）来不断产生不需要研究的区域，从而通过排除非凸包顶点、减少分析的点数来提高效率，这就是快速凸包技术。

Yao 证明凸包算法的计算复杂度下限为 $O(n\log n)$。虽然有些算法在特殊情况下可以达到线性时间的复杂度，不过在最坏的情况下，计算复杂度仍不低于 $O(n\log n)$。尽管在最坏的情况下任何方法均无法突破复杂度下限，但仍有可能通过改进提出在通常情况下比传统方法更快的求取思路。文献[18]对现有的凸包生成算法进行了分析，并在文献[23]的算法基础上提出 $O(n)$ 复杂度最优算法。

1.2.3 道路提取

从航空或卫星影像中提取道路一直都是一个很具有挑战性的问题。根据道路特征提取的自动化程度，道路特征提取分为自动特征提取和半自动特征提取。自动提取方法包括基于平行线对[24]、基于二值化和知识[25]、基于窗口模型特征[26]等；半自动提取的常用算法有模板匹配[27][26]、动态规划[28]、可变模型或 Snakes 方法[29]等。到目前为止，从航空或卫星影像中全自动识别道路仍然处于不成熟的状态[30]，而且在未来的短期时间内也很难开发一套完全脱离人工干预而自动识别道路的系统。在各种半自动道路提取方法中，由于动态规划是一个解决优化问题的稳定算法，能够确保结果的最优性，成为近年来道路半自动提取的首选方法之一。动态规划法首先应用在低分辨率影像提取道路中（Sakoda 等，1993；Gruen 和 Li，1997；Merlet 和 Zerubia，1996）。Dal Poz 等[28]修改了“代价”函数，使之适用于从中高分辨率影像中提取道路。

另一方面，从机载 LiDAR（Light Detection And Ranging）数据中提取道路在近年来也有了一定的研究，Hatger 和 Brenner 在文献[32]中结合 LiDAR 数据和已经存在的数据库信息提取道路的几何属性；在 Rieger 等人的论文[33]中研究了林区范围内由 LiDAR 点云提取道路方法；为充分利用 LiDAR 数据信息，文献

[34]和文献[35]引入了强度信息辅助道路的提取。

1.2.4 建筑物提取

由 LiDAR 数据所生成的 DSM 中包含了大量关于房屋的三维信息，它们提供了许多可供推理的线索，为房屋的三维重建带来了很大的便利。但是由于地物的多样性、复杂性，从 DSM 数据重建三维房屋仍然具有许多困难[36]。

在过去的十几年内，有的研究人员将 LiDAR 数据和其他数据源结合起来，充分利用其他数据源的数据特性和 LiDAR 数据本身的特点，采用各种方法来克服以上困难完成重建工作，并取得了一些成果。一些研究人员将重点放在利用仅由 LiDAR 数据所生成的 DSM 建模，这些方法主要是利用房屋的几何特性，一般将建模分成两步：房屋的检测和重建。房屋的检测主要基于如下知识：房屋通常都比地面高，并且房屋具有一定规则的形状，在检测的过程中使用一些具体的准则将房屋和其他地物分离。

George Vosselman 和 Sander Dijkman 结合城市的地面规划设计图和从机载激光扫描所获得的点云来重建三维建筑物模型，其中屋顶的表面的高程和方向通过扩展到 3D 空间的 HOUGH 变换来较准确的获取。屋顶的轮廓线比较难以确定，因此采用城市地面规划设计图来获得精确的建筑物墙面的位置。通过将检测出来的屋顶的位置和墙面相交，就可以重建一些屋顶的边界。其他的边界通过相邻屋顶平面的相交和从点云中检测到的屋脊线来得到。采用规划设计图降低了处理的难度，基本上不需要考虑处理工作局部化的问题，因为规划图已经提供了关于独立的房屋的平面二维矢量的较精确描述，房屋的边界等都已经获得，回避了检测房屋区域的难题，但是这对数据的获取提出了更高的要求，对新建房屋也无法处理。

Hans-Gerd Maas 和 George Vosselman 使用 Delaunay 三角网从 LiDAR 数据的点云中重建房屋。他们首先采用 HOUGH 变换提取房屋各平面，然后对各平面内的点构建 Delaunay 三角网，计算平面面积，当该面积大于某一阈值时才采纳此平面为真实的房屋平面，否则剔除此平面，从而提取出房屋的真实平面。在对平面做出 Delauany 三角网后，找到该平面的外边缘轮廓线的各边，然后再调整各边的端点，使直线拟合的残差最小，由此找到房屋的外边缘轮廓线。然后对每两个平面都进行相交运算，并找出相邻平面的交线段，由平面交线和外轮廓线构成房屋屋顶结构，重建出房屋。这是一种数据驱动的建模方式，对数据质量要求较高。另外，从全局数据中进行 HOUGH 变换提取房屋各平面消耗的时间和计算量是十分可观的，当数据量较大时，成功几乎不可想象。

Morgan 和 Tempeli，研究了从 LiDAR 数据生成的 DSM 中重建房屋的方法。整个处理是从将激光扫描仪获得的不规则高程点内插成规则格网开始的。建筑物检测的主要方法是采用数学形态学的操作将房屋区域和地表面分开。为了获得建筑物的矢量描述方式，对于整块的屋顶表面通过将建筑物区域进行进一步

分割成各个小区域来获得。建筑物的每个表面的三维几何特性通过基于最小二乘的平面拟合而获得。房屋的重建是基于屋顶平面之间的连续性来完成的。屋顶重建需要提取基本图元并进行平面的交会。在检测和提取的过程中都需要用到建筑物和地形的大量的先验知识，计算量极大，过程繁琐且稳健性较差。

日本东京大学土木工程系的郭滔博士将IKONOS影像和机载激光扫描数据结合起来进行大规模的城市三维建模。他将光谱信息和高程信息结合起来进行地物的分割，通过将从IKONOS影像和DSM中所获得的多重线索融合起来进行推理，采用一种分层的方式检测出建筑物的层次、主方向、尺寸、位置等信息，重建出来的各种边缘进行编组，以组成独立的房屋，只能重建屋顶为平顶的房屋。

加拿大Galgery大学的胡勇博士提出了一个数据驱动、自动重建建筑物边界的流程。重建的边界一般是矩形、正方形和规则多边形。他首先通过一个先验的高程值来二值化DSM，从而将高于地面的部分从DSM中分割出来，获取分割后的每个备选房屋区域的边界，然后通过对边界的形状通过一个度量来分类，从而检测出可靠的建筑物边界，其次，在对建筑物边界进行确认并进行精化后，采用约束搜索、增强的HOUGH变换和连续连接技术以得到建筑物边界，通过统计方法将建筑物屋顶分成平顶和非平顶。最后，通过一种可选的精化技术重建非平顶房屋并通过垂直墙面约束纠正房屋墙面。其重建的房屋一般为独立的单层次的简单房屋，对于多层次的房屋无能为力。

在国内，从LiDAR数据中进行三维重建的研究开展的稍迟一些，但近年来已经成了研究的热点，并且取得了一定的成就。

中国科学院遥感应用研究所的尤红建等人从国内自行研制的机载激光扫描仪所获得的数据中，生成数字表面模型，并从中自动提取建筑物，取得了很多成果。主要工作是：初步完成了直接应用激光采样点的原始三维位置来提取建筑物的方法，通过DSM影像的分割、边缘提取、边缘跟踪等步骤以准确提取建筑物的边缘轮廓线。应用多边形逼近、方位角分组、确定建筑物方向、边缘线段的规格化等算法有效地对建筑物的粗略边缘线进行规格化处理，获得简单的直角规则建筑物。该方法没有考虑多层次，不规则的房屋，对屋顶的形状也几乎不予考虑。

武汉大学张小红的博士论文较详细地研究了激光测距数据的分类和规则房屋提取的方法。分类算法充分利用激光测距数据的强度信息，融合高程和强度信息进行数据分类和分割，利用多次回波激光数据进行数据分类有效地将植被提取出来，为分割相互邻接的房屋和树提供了一种有效的途径。文中提出了基于分类分割后的原始激光脚点数据提取规则房屋的方法。该方法主要针对规则房屋的重建，无需内插，房屋边界定位较为准确，但是尚未估计不规则，多层次，多直角房屋的重建，并且对于数据质量要求较高。

1.3 问题提出

传统的三维模型的重建常常使用摄影测量的手段。利用连续像对，通过影像匹配，前方交汇、内插从而得到目标的三维模型。但是，由于航拍得到的图像往往受到天气、大气折射的影响，结果往往不尽如人意。另一方面，高密度点云的获取目前仍然是一个挑战，且城市规模的高密度点云处理所需要的存储空间和计算能力是非常巨大的。

通过分析比较 LiDAR 数据和航空影像的各自特点以及相互的优缺点，本课题提出了基于稀疏 LiDAR 点云数据和航空影像的的城市建模。利用这两种数据进行三维重建的效果远比利用单一稀疏 LiDAR 数据重建的更接近客观事实。LiDAR 数据和航空影像都具有获取速度快，低成本，精度高的特点，二者各有所长，互为补充。基于 LiDAR 数据和航空影像的三维重建为数字城市、虚拟现实、三维 GIS 的原始数据提供了另一种可选的途径。

1.4 研究内容及组织结构

1.4.1 研究内容

机载激光扫描测高已成为一种先进的集成测量技术方法，具有极好的发展前景和很强的竞争力，整体经济效益明显优于传统摄影测量。国际上许多公司、研究机构投入大量人力、财力进行相关技术与系统的研究开发，并在诸多领域得到应用。鉴于机载激光扫描测高技术在许多领域的广泛应用前景，而我国在机载激光扫描技术方面的应用研究同国际发达国家相比相对落后，因而多开展该领域的研究来提高我国的技术水平是一件有着理论和实际意义的事[37]。

本文在总结国内外 LiDAR 点云数据在城市建模方面诸多研究的基础上，探索结合航空影像和 LiDAR 点云数据的关键技术，主要研究内容包括：

（1） 由于 LiDAR 成像的固有属性，生成的点云数据是疏密不均的，本文研究对海量 LiDAR 点云数据进行快速构建 Delaunay 三角网的方法。

（2） 通过对 LiDAR 点云进行二次采样生成对应的高度图，研究 LiDAR 点云数据与航空影像配准的方法。

（3） 结合航空影像和高度图，在现有基于动态规划算法的基础之上，研究半自动提取城市的主干道路的方法。

（4） 根据城市地形的起伏缓慢的特点，研究针对城市区域的地面提取较为鲁棒性的方法。

（5） 为增加城市重建的真实感，研究基于 LiDAR 点云数据的城市大型、规则建筑物的识别和重建方法。

1.4.2 组织结构

论文的主要内容安排如下：

第一章绪论部分首先介绍了机载 LiDAR 点云数据的成像原理和三维重建

的研究背景；然后针对基于 LiDAR 点云的城市三维重建相关研究进行了国内外研究现状概述；接着引入本研究的问题提出；最后简要介绍了论文工作的主要内容。

第二章主要介绍基于 LiDAR 点云的网格构建。首先介绍了 Delaunay 三角网的相关概念以及当前主流的几类 TIN 构建算法，并重点介绍了生长法构网的主要步骤和流程；然后介绍在现有生长法构网的基础上如何进行改进来适合海量非均匀分布 LiDAR 点云的快速构网，随之给出改进的若干关键技术和细节；最后对改进后的算法进行的复杂度分析和相关比较实验。

第三章主要介绍城市区域的地形提取过程。首先介绍如果利用 3D HOUGH 变换提取地形的基准平面；然后结合地面 TIN 模型的属性特征来搜索属于地面的 LiDAR 点集；最后利用 2D 凸壳生成算法保留地形的边界信息。

第四章主要介绍结合航空影像和 LiDAR 点云来提取城市大型主要人造目标。本章主要是针对道路和建筑物的提取进行介绍，首先介绍了城市主干道路的半自动提取方法；然后给出了城市大型、规则建筑物的提取和重建方法。

第五章论文的总结和展望。对本文的研究工作进行了全面的总结并提出了对今后工作的展望。

第二章　基于 LiDAR 点云的网格构建

2.1 Delaunay 三角网构建简介

2.1.1 Delaunay 概念

Voronoi 多边形的定义是，设 x 为平面上的点，则区域 $V(i)=\{x\in E_2 \mid d(x,v_i)\le d(x,v_j), j=1,2,...,N, j\neq i\}$ 称为 Voronoi 多边形。各点的 Voronoi 多边形共同组成 Voronoi 图[38]。

Delaunay 三角网的定义是，有公共边的 Voronoi 多边形称为相邻的 Voronoi 多边形。连接所有相邻的 Voronoi 多边形的生长中心 $V(i)$ 所形成的三角网称为 Delaunay 三角网，如图 2-1。

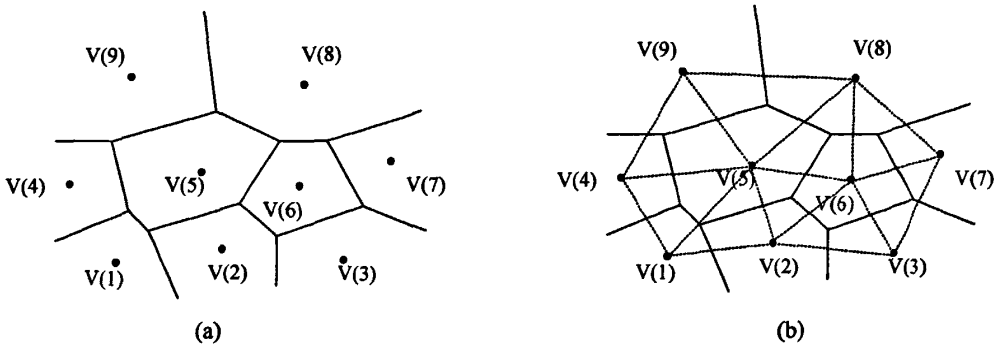


图 2-1 Voronoi 图和 Delaunay 三角剖分

Delaunay 三角网的外边界是一个凸多边形，它由节点集中的凸集形成，通常称为凸壳。Delaunay 三角网具有两个非常重要的性质：

（1）　外接圆性质。在由点集 V 所形成的 Delaunay 三角网中，其每个三角形的外接圆均不包含点集 V 中的其他任意点。

（2）　最大的最小角度性质。在由点集 V 所能形成的三角网中，Delaunay 三角网中三角形的最小角度达到最大。

由于这两个性质，Delaunay 三角网既具有极大的应用价值，又是二维平面三角网中唯一最好的[39]。

2.1.2 TIN 构建算法思想

利用点集构建不规则三角网（Triangulated Irregular Networks，TIN）的方法目前可分为 3 类：分治-合并算法、逐点插入算法和生长算法。

（1）　分治算法

Shamos 和 Hoey 提出了分治算法思想，并给出了一个生成 V-图的分治算法。Lewis 和 Robinson 将分治算法思想应用于生成 Delaunay 三角网。他们给出了一个“问题简化算法”，递归地分割点集，直至子集中只包含三个点而形成三角形，

然后自下而上地逐级合并生成最终的三角网。以后 Lee 和 Schachter 又改进和完善了 Lewis 和 Robinson 的算法。

Lee 和 Schachter 算法的基本步骤是：

1） 把点集 V 以横坐标为主，纵坐标为辅按升序排序，然后递归地执行 2）－6）步；

2） 把点集 V 分成近似相等的两个子集 V_L 和 V_R；

3） 在 V_L 和 V_R 中生成三角网；

4） 用 Lawson 提出的局部优化算法 LOP 优化所生成的三角网，使之称为 Delaunay 三角网；其 LOP 的主要思想是：如图 2-2 所示，若 $\angle ABC+\angle ADC>180^{\circ}$，则应该删除四边形 ABCD 的对角线 BD 和三角形 ABC、ADC，同时取而代之用对角线 BD 和三角形 ABD 和 BCD。

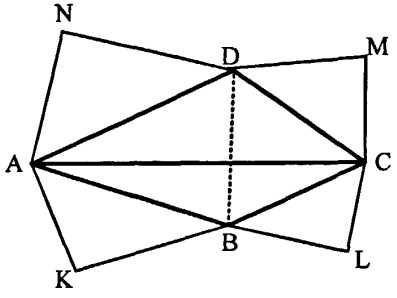


图 2-2 LOP 算法

5） 找出连接 V_L 和 V_R 中两个凸壳的底线和顶线；

6） 由底线至顶线合并 V_L 和 V_R 中两个三角网。

以上步骤显示，分治算法的基本思路是使问题简化，把点集划分到足够小，使其易于生成三角网，然后把子集中的三角网合并生成最终的三角网，用 LOP 算法保证其成为 Delaunay 三角网。不同的实现方法可有不同的点集划分法、子三角网生成法及合并法。

（2） 逐点插入法

Lawson 提出了用逐点插入法建立 Delaunay 三角网的算法思想。Lee 和 Schachter，Bowyer，Watson，Sloan，Macedonio 和 Pareschi，Floriani 和 Puppo，Tsai 先后进行了发展和完善。

逐点插入算法的基本步骤是：

1） 定义一个包含所有数据点的初始多边形；

2） 在初始多边形中建立初始三角网，然后迭代以下步骤，直至所有数据点都被处理；

3） 插入一个数据点 P，在三角网中找出包含 P 的三角形 t，把 P 与 t 的三个顶点相连，生成三个新的三角形；

4）用 LOP 算法（同分治－合并算法中的步骤 4）优化三角网。

从上述步骤可以看出，逐点插入算法的思路非常简单，先在包含所有数据点的一个多边形中建立初始三角网，然后将余下的点逐一插入，用 LOP 算法确保其成为 Delaunay 三角网。各种实现方法的差别在于其初始多边形的不同以及建立初始三角网的方法不同。

（3） 三角网生长法

Green 和 Sibson 首次实现了一个生成 Dirichlet 多边形图的生长算法。Brassel 和 Reif 稍后也发表了类似的算法。McCullagh 和 Ross 通过把点集分块和排序改进了点搜索方法，减少了搜索时间。Maus 也给出了一个非常相似的算法。

现有三角网生长算法的基本步骤是：

1）以任一点为起始点；

2）找出与起始点最近的数据点相互连接形成 Delaunay 三角形的一条边作为基线，按 D-三角网的判别法则，找出与基线构成 Delaunay 三角形的第三点。如图 2-3 所示，对于任意基线 AB，为找到满足 Delaunay 三角网性质的点 C_i，需要使 $\angle AC_iB$ 达到最大。因余弦函数在[0,360]上递减，由余弦定理，使 $\cos\angle AC_iB=(a_i^2+b_i^2-c)/2a_ib_i$ 达到最小即满足要求；

3）基线的两个端点与第三点相连，成为新的基线；

迭代 2）、3）两步直至所有基线都被处理。

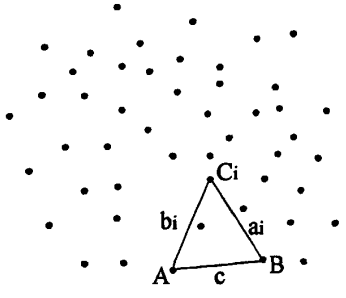


图 2-3 查找 Delaunay 第三点

上述过程表明，三角网生长算法的思路是，先找出点集中相距最短的两点连接成为一条 Delaunay 边，然后按 Delaunay 三角网的判别法则找出包含此边的 Delaunay 三角形的另一端点，依次处理所有新生成的边，直至最终完成。各种不同的实现方法多在搜寻“第三点”上做文章。

2.1.3 本文算法的选取

由于分治-合并算法时间效率高，目前使用最多，但其递归过程占用大量的内存空间，不适合大规模点云的三角化，胡金星等人在分治-合并算法的基础上对点集自适应划分，从而生成海量 DEM 数据[40]，但是需要不断将数据从内存中换进换出。逐点插入法占用内存较少，但是时间效率较差，而且在大规模点云构网后的合并过程中，有时需要插值产生新点或者很难保证边界拼接时的

Delaunay 三角形性质[41]。生长法算法简单，但时间效率也较差。在文献[42]中，Green 和 Sibson 首次实现了一个生长 Dirichlet 多边形图的生长算法。Brassel 和 Reif 稍后也发表了类似的算法[43]。McCullaugh 和 Ross 通过把点集分块和排序改进了点搜索方法，减少了搜索时间[44]。Maus 也给出了一个非常相似的算法[45]。本文对现有生长算法进行改进，自适应确定搜索范围，在 LiDAR 点云数据疏密不均的情况下也能高效率进行构网。

由于三角网生长算法的思路是，先找出点集中相距最近的两点连接成为一条 Delaunay 边，然后按 Delaunay 三角网的判别法则找出包含此边 Delaunay 三角形的另一端点，依次处理所有新生成的边，直至最终完成[46]。本文则对生长法进行改进，使其在从海量 LiDAR 点云数据中重建三维模型的过程中得到应用。

2.2 TIN 模型构建

2.2.1 算法流程

在三角网生长算法基本步骤的基础上，对其进行改进，使之能够在海量点云数据中快速构建 Delaunay 三角网，构建的主要步骤是：

Step 1：点云分块

根据网格点的密度将点云分块，并对生成的基线进行分组，设点集分 $M \times N$ 块，分别存放于 $Grid[i,j](1 \le i \le M, 1 \le j \le N)$ 中，则基线可分 M 组，分别存放于 $Base[i](1 \le i \le M)$ 中。

Step 2：构建第一条基线

从最左下角的分块中任意找一点 P_1，在以 P_1 所在分块 $Grid[i,j]$ 的临域分块中找出离 P_1 最近的点 P_2，则 P_1P_2 作为第一条基线。取 P_1、P_2 中 y 值较小的一个，不妨设为 P_1，且 $P_1 \in Grid[i,j]$，则将基线 P_1P_2 存放于基线队列 $Base[i]$ 中。

Step 3：扩展第一条基线的左侧

对于第一条基线 P_1P_2，先在其左侧搜索是否有满足 Delaunay 性质的点 Q。若存在，则 P_2Q 和 QP_1 为新的基线，并插入到相应的基线队列 $Base[k]$，其中 k 的确定方法如同插入第一条基线 P_1P_2。

Step 4：扩展基线的右侧

基线右侧的扩展过程如图 2-4 所示。

2.2.2 点云分块

三角网生长算法的核心就是求取新边构成三角网，大部分的计算都集中在从大量数据中搜寻最优的第三点，只要在搜索方法上加以改进就可以大大提高构网速度[44]。

现有的数据分块方法有：均匀条带分割（图 2-5）、等格网分割（图 2-6）、四叉树分割（图 2-7）和自适应分块等[45]。因为每一条基线第三点的查找范围

是在基线的四周进行搜索，故图 2-5 中的均匀条带分割法依然会使搜索的范围相当大。此外查找的范围不仅要在包含基线的数据块内查找，同时也要在临近数据块内查找，故数据块的划分要易于查找或计算邻接关系，因此，本文采用图 2-6 中的等网格分割法。

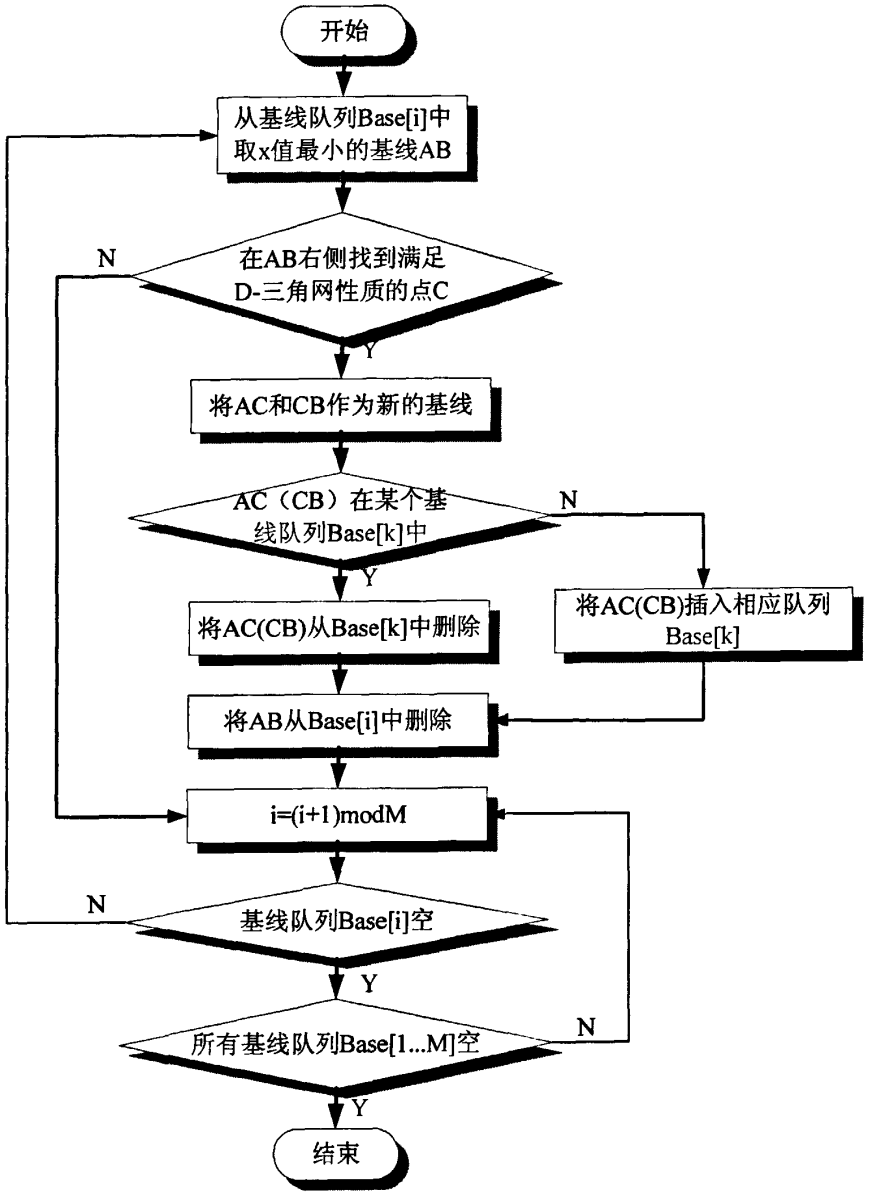


图 2-4 基线右侧扩展流程

2.2.3 确定搜索范围

以往算法是在每条基线的两侧进行扩展，本文在新添加基线的时候控制新形成的三角形在当前基线的左侧(第一条基线除外，需单独处理)，从而在每次扩展基线时只需对基线右侧进行扩展，缩小搜索范围。如图 2-8，*A*、*B*、*C*、*D*

的坐标分别 (x_1,y_1)、(x_2,y_2)、(x_3,y_3)、(x,y)，当前扩展的基线为 *AB*，*C* 是其左侧三角形的另一顶点。其右侧搜索点 *D* 的范围应取关于直线 *AB* 与点 *C* 异侧的点，由于直线 *AB* 的方程为 $F(x,y)=(y_2-y_1)(x-x_1)-(x_2-x_1)(y-y_1)=0$，当 $F(x,y)\cdot F(x_3,y_3)<0$ 时，即点 *C* 与 *D* 在直线 *AB* 的异侧，则该点可作为备选扩展点[46]，另一种方法就是按照文献[23]中的叉积先判断 C 点与 AB 的哪一侧，然后利用叉积判断 D 在另一侧。将新生成的有向基线 *AD* 和 *DB* 加入基线队列，确保三角形 *ABD* 在基线 *AD* 和 *DB* 的左侧，从而无需再对 *AD* 和 *DB* 的左侧进行扩展。

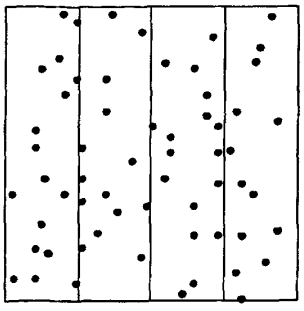


图 2-5 均匀条带分割法

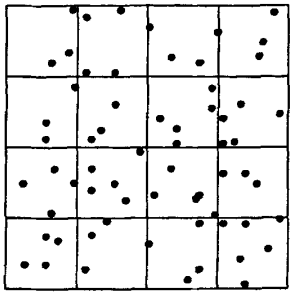


图 2-6 等格网分割法

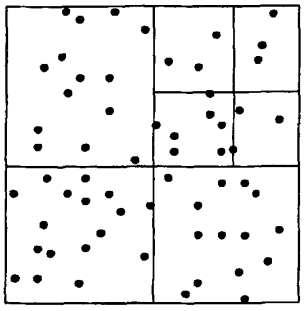


图 2-7 四叉树分割法

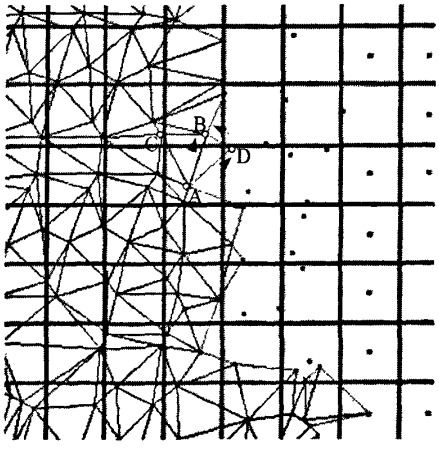


图 2-8 新添加的基线

为适应 LiDAR 点云疏密不均的情况，在构网过程中可以动态确定每条基线的扩展范围。对于待扩展的基线 *AB*（如图 2-9 所示），其扩展过程如下：

Step 1：计算 *AB* 的中点 *O*;

Step 2：在 *AB* 右侧的垂直平分线上截取 *OM=L*;

Step 3：记过 *A*、*B*、*M* 三点的圆 *Circle* 在直线 *AB* 右侧的部分所在的分块集合为 *Blocks*;

Step 4：在 *Blocks* 内搜索满足 D-三角网性质的点;

Step 5：搜索到“有效”点则结束，否则令 $L=L+\Delta L_i$，此时如果 *L* 大于预设的阈值，则停止，否则返回 *Step 2*。

该方法既能有效缩小搜索范围，同时也能保证在点集疏密不均的情况下找

到合适的候选点。

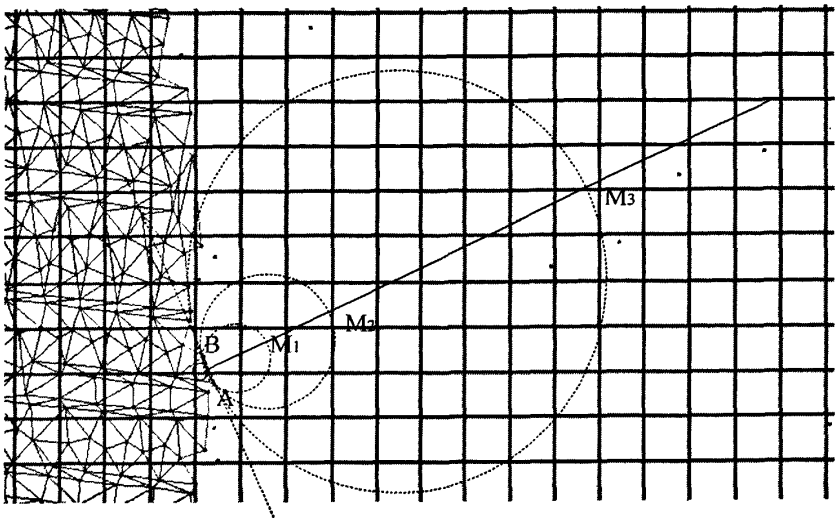


图 2-9 点的搜索范围

2.2.4 基线排序

在将生成的新基线插入到相应队列时，都要遍历所要插入的整个队列，判断该基线是否已经存在于队列当中。因此，要对基线队列的长度进行适当的控制，可以通过将基线按照端点 x 的升序排列，同时在从基线队列 $Base[i]$ 中取基线进行扩展的时候，选取队列中的第一条基线，从而避免队列中的基线数目变得很大，极大地减少内存占用和搜索时间（如图 2-10、图 2-11 所示）。

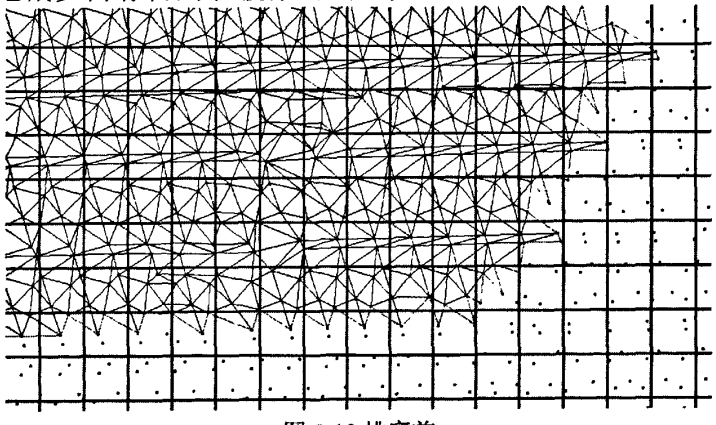


图 2-10 排序前

2.2.5 排除封闭点

封闭点[46]的定义：点 P 是离散点集 S 中一点，与 P 连接点的集合 $F(P)=\{Q_1,Q_2,...,Q_n\}$，以 P 为端点的待扩展的基线条数为 $N_Extend(P)$，如果在某一时刻有 $F(P)\neq\Phi$，且 $N_Extend(P)=0$，则称点 P 在此时是封闭的，或称点 P 是封闭点。一旦一个点成为封闭点，那么它将不可能与其它点构成满足剖分要求的三

角形。所以在扩展三角形时，应动态地将成为封闭点的点排出考虑的范围，从而减少搜索过程中的计算量[47]。

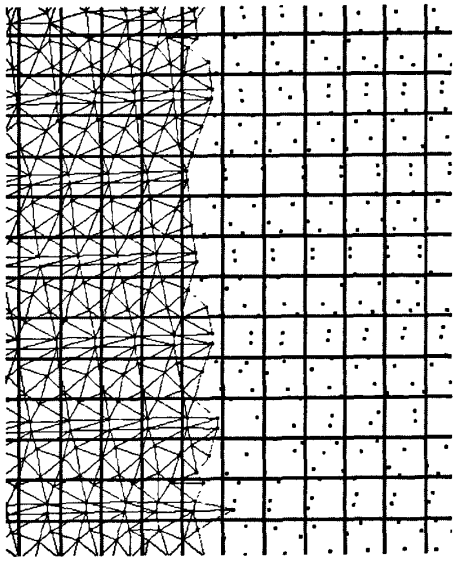


图 2-11 排序后

算法中对每一个顶点 P 都用一个变量 $N_Extend(P)$ 标记与该点相连的待扩展的基线数目，$N_Extend(P)$ 初始化为－1，添加 P 的第一条基线时 $N_Extend(P)$ 赋值 1，之后每添加一条新的基线时 $N_Extend(P)$ 加 1，每删除一条基线时 $N_Extend(P)$ 减 1，在扩展基线过程中若 $N_Extend(P)$ 为 0，则表示该点已经封闭。

另一方面，每个分块都记录当前块内的总点数、未封闭的点个数，块的数据结构如下：

```
struct_BLOCK{
int nPoint;          //总点数
int nOpenPoint;      //尚未封闭的点个数
int* pPoint;         //点序号数组
.....
};
```

若块内所有的点都已经封闭，即 nPoint=nOpenPoint，在搜索的时候直接排除该分块而不是逐个点进行判断封闭性。

2.2.6 处理边界点

在用插入法构造 Delaunay 三角形时一般是先构造点集的凸包，然后将剩余的点逐个插入到凸包中，这样可以很好地构建出点集的边界。在用三角网生成的时候一般不再构建凸包，这样在处理边界点时，会出现一些很狭长的三角形，虽然它们也满足 Delaunay 三角形的条件，但是在实际应用当中是不需要这些三角形的。所以在本文中采用在点集边界的外围添加辅助点，先构建包含辅助点

的 TIN 模型，然后从模型中删除以辅助点为顶点的三角形，通过该步骤能够很有效地避免边界的狭长三角形，而且能够很好地反映点集的边界形状。辅助点的添加方法为：若 *Grid*[*i*, *j*] 内没有点而其 8-临域方向不全有点，则在 *Grid*[*i*, *j*] 内添加一个辅助点，该辅助点的坐标为 *Grid*[*i*, *j*] 的中心位置。具体添加辅助的情况和效果如图 2-12 所示。

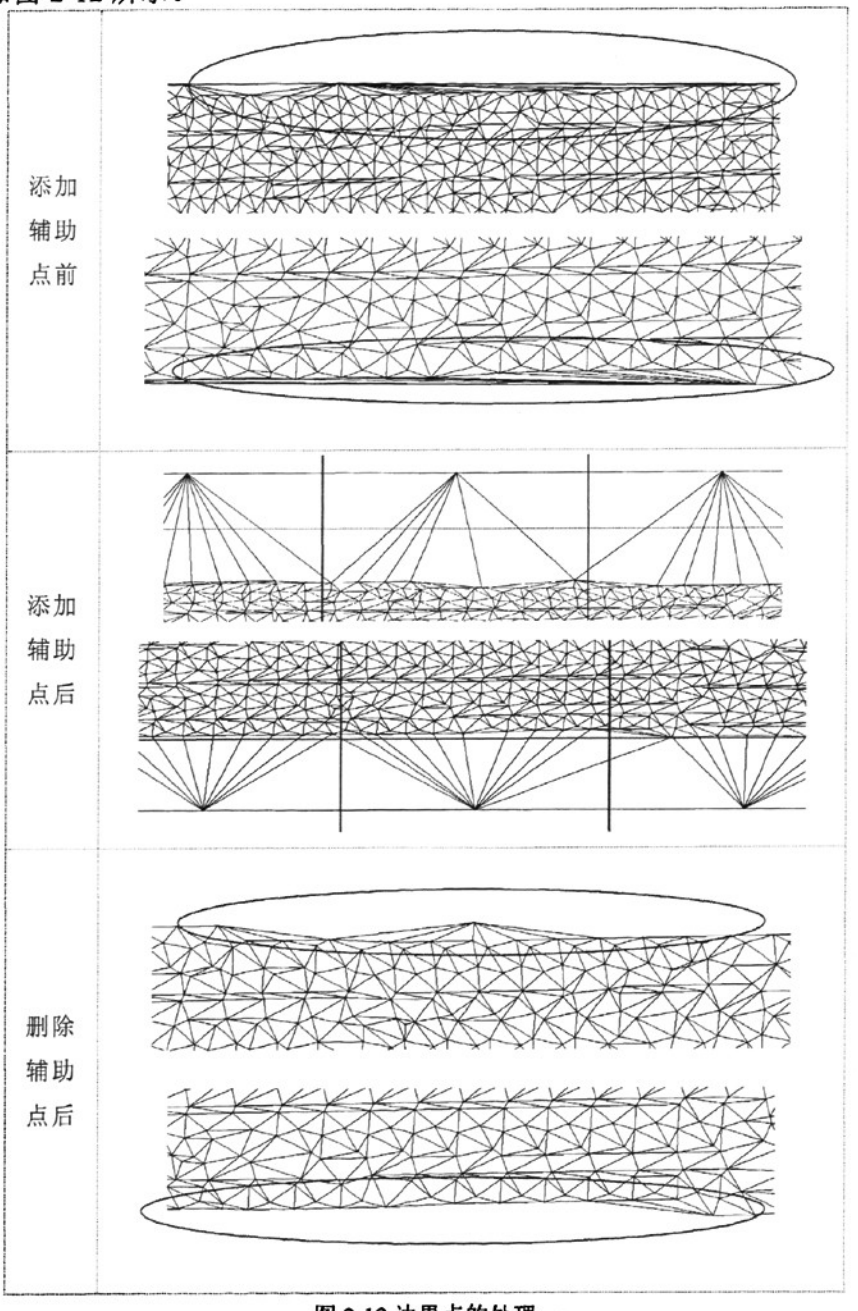


图 2-12 边界点的处理

2.2.7 算法复杂度分析

生长法构网的时间主要花费在扩展基线时候选点的搜索过程，按照生长法思想，对一定区域内的 n 个点，每查询一个“有效”点时，需要对这 n 个点逐个进行检索、计算，算法的平均复杂度为 $O(n^2)$。当点个数比较庞大时，每次查询都遍历所有的 n 个点势必会浪费很多时间。通过对点集进行合理的分块，可以利用查询附近区域内的部分点来代替查询全部 n 个点。

根据查询“有效”点的方式是否跨越分块，可以将基于分块的生长法构网分为两类。第一类是分别在每个分块内利用生长法构网，然后将每个分块构建的网格进行拼接而形成全局的网格模型[46]，第二类是在当前基线所在分块及其邻域分块内进行查询“有效”点[44][47][48]，从而避免了分块网格之间的拼接过程。对于第一类，网格的分块往往比较大，因为当分块较小时在点云稀疏的位置分块内可能就没有足够的点进行构网，且分块之间的拼接代价会很大，有时甚至需要添加过多的辅助点[41]，从而增加冗余数据并降低网格精度，假设每个分块内点的个数约为 n_1，则算法复杂度计算为 $O(n \cdot n_1)$。对于第二类的情况，为保证每次都能找到“有效”点，有两种策略，一种是分块较大，每次在较少的邻域内查询;另一种是分块较小，每次在较多的邻域内查询，不妨设每个分块内有约 n_2 个点，每次在 q_2 个邻域分块内查询，则算法的复杂度可计算为 $O(n \cdot n_2 \cdot q_2)$，虽然 n_2 或者 q_2 较小，但是 $n_2 \cdot q_2$ 结果却比较大。

本算法在第二类思想的前提下，由于采用了自适应确定查询范围的策略，可以使分块比较小，同时又能快速找到“有效”点。假设每个分块内有大约 n_3 个点，每次在 q_3 个分块内查询，则算法复杂度为 $O(n \cdot n_3 \cdot q_3)$，其中 n_3 是一个相对于 n_1 小很多的数，同时 q_3 会随着点集密度的改变自适应变化，从而使 $n_3 \cdot q_3$ 根据点云的密度的不同也自适应变化，避免了大量不必要的的查询。

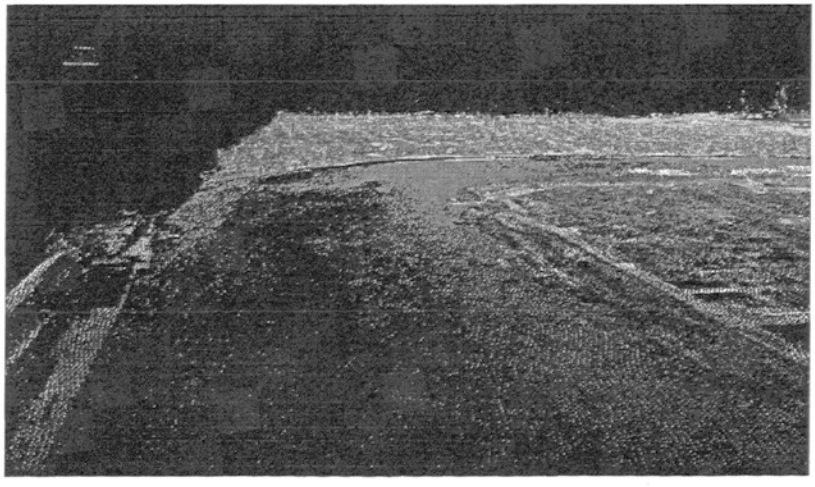


图 2-13 点云数据

另一方面本文算法在扩展每条基线时仅在基线的一侧进行查询，该策略可

以使查询次数平均减半，通过排除封闭点和封闭块也使查询次数极大地较少。经分析可知，本文算法不仅在点集密度均匀情况下快速构网，在像 LiDAR 点云数据这样密度变换剧烈的情况下，能够表现更突出的优势。

2.2.8 算法实验

本试验数据来源于美国新奥尔良市区的 LiDAR 数据。由于 LiDAR 点云具有三维坐标，因此在构建 TIN 模型时暂不考虑其高度 Z 值，利用本文算法可以生成点云（如图 2-13 所示）的二维网格（如图 2-14 所示），且自适应点集的疏密分布，在构建网格之后再将每一点作为空间点来显示。运行过程总共输入点个数 495 万，构建三角形约 867 万。图 2-15 为所构建模型的一部分。

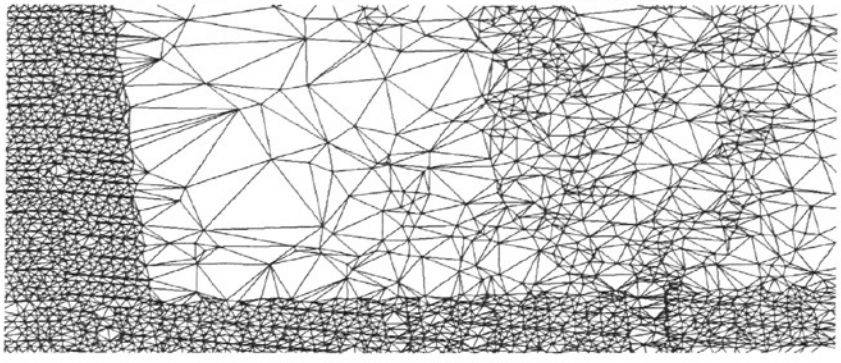


图 2-14 LiDAR 点云二维网格

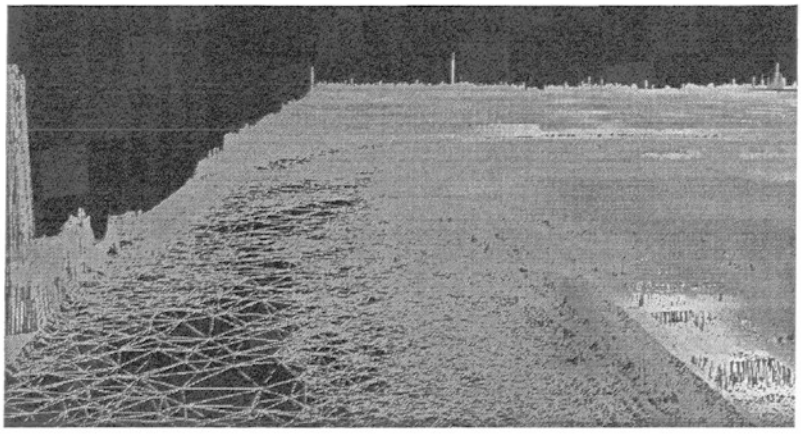


图 2-15 LiDAR 点云的 TIN 模型三维显示

从试验中我们能构看到，虽然对原始数据进行了分块，但是在后期不需要对分块进行合并，因此减少了合并分块数据所需时间;另一方面，我们在每一分块的边缘处不需要插值产生新的数据点，从而使模型中的每一个网格顶点都是机载 LiDAR 的实际采样点，能够真实地反映测量地区的实际表面形状。

为进一步说明本文算法的优势和可行性，将本文算法与改进前等格网分块的生长算法、文献[41]中的逐点插入法进行比较，比较结果如表 1-1 所示。从表中可以看出，与改进前的等格网分块算法比较，本文算法的搜索策略能够大大

加快构网速度，而且能够减少很多边界处的狭长三角形。在边界插值的逐点插入法由于要在分块的拼接处插入很多三角形，不仅浪费时间，而且增加了许多非必需三角形的数量。

表 2-1 算法比较结果

结果 / 算法	顶点个数（个）	三角形个数（个）	时间（s）
边界插值的逐点插入法	106075	203731	54.035
等格网分块的生长法	79079	160153	96.255
本文算法	79079	157511	10.126

2.3 本章小结

本章通过对三种主要的 TIN 模型的构建方法进行分析，在此基础之上针对三角形生长法进行扩展，使其能够很好的应用于大规模 LiDAR 点云的 TIN 模型构建，并且构建的结果完全使用原有的数据信息，无需进行插值和拼接过程，从而可以保证构建模型的精确性。为处理边界点，本章算法通过添加边界辅助点避免了边界狭长三角形的生成。最后将该算法应用于美国新奥尔良市区的 LiDAR 数据进行构网，验证了算法的可行性。虽然在近些年内相对于逐点插入法和分治一合并法构网，生长法构网的研究较少，但本章的改进无疑是对生长法构网研究的一个有益的尝试和补充。

第三章　城市地面提取

由于城市地形起伏缓慢，LiDAR 点云中的地面点集中于一个平面附近，因此本文首先利用三维 Hough 变换检测 LiDAR 点云中的地形基准平面的位置，然后在基准平面的附近的 TIN 中查找法线竖直的三角形，将该三角形的定点判定为地面点。为了避免滤波过程中将较多的地形边界点滤除掉，通过生成 LiDAR 点云的凸包来保持地形的边界。

本文计算凸包时运用了快速凸包算法思想，由于处理的 LiDAR 点云数量较大，因此先对点云数据进行分块，然后以块为单位对点云进行处理，从而该算法适合在海量点云数据构建凸包。

3.1　LiDAR 地面点提取

3.1.1 提取基准地面

根据城市地形变化缓慢，可以利用 3 维 Hough 变换从目标区域的 LiDAR 点云数据中提取基准地面，该提取过程利用了所有点的任何可能的地面所在平面位置，因此具有全局最优的性质，即使在点集密度比较稀疏时也能提取出地平面的近似位置，为后续的地面精确位置提取做准备。

平面 $Ax+By+Cz+D=0$ 的极坐标方程为：

$$\rho = x\sin\beta\cos\alpha + y\sin\beta\sin\alpha + z\cos\beta$$

其中 (ρ,α,β) 定义了一个从原点到面上最近的向量（如图 3-1），这个向量与该平面垂直，因此与平面一一对应。

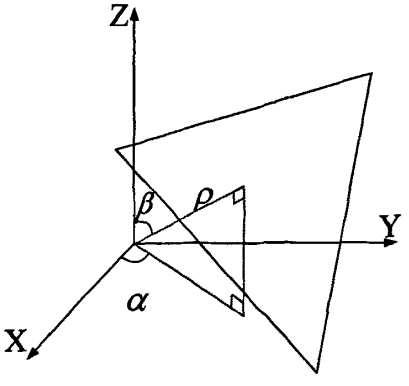


图 3-1 三维 Hough 变换原理图

考虑一个以参数 ρ、α 和 β 定义的三维空间。x，y，z 空间的任意一个平面对应了该空间的一个点。因此，平面的任意一个平面的 Hough 变换是参数空间的一个点。把上述三个参数量化以后，对于点云空间的任意一点来说，通过该点的平面仍然有很多个，即对应参数空间的很多点。对点云空间中的每一个点

都找到通过它的全部平面，并在平面参数空间对应点上计数值加 1。做完上述操作以后，实际得到了点云空间所有可能的平面所拥有的点集计数值 C，查找参数空间中的最大值 C_{max} 对应的参数为 LiDAR 点云中的地面位置。由于运用的是统计的结果，Hough 变换有很好的抗干扰能力，即使 LiDAR 点云比较稀疏，根据城市地形变化缓慢的性质也能得到地形的基准平面。此外，Hough 变换还具有明了的几何解析性，易于实现并行处理等优点。

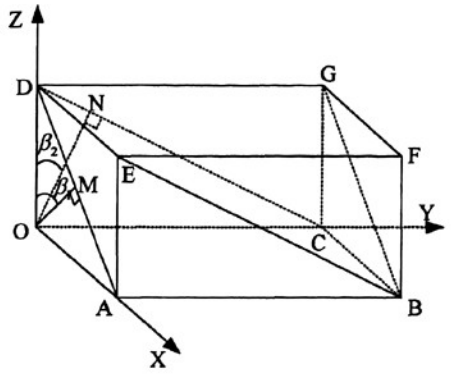


图 3-2 Hough 变换检测 LiDAR 地面参数约束

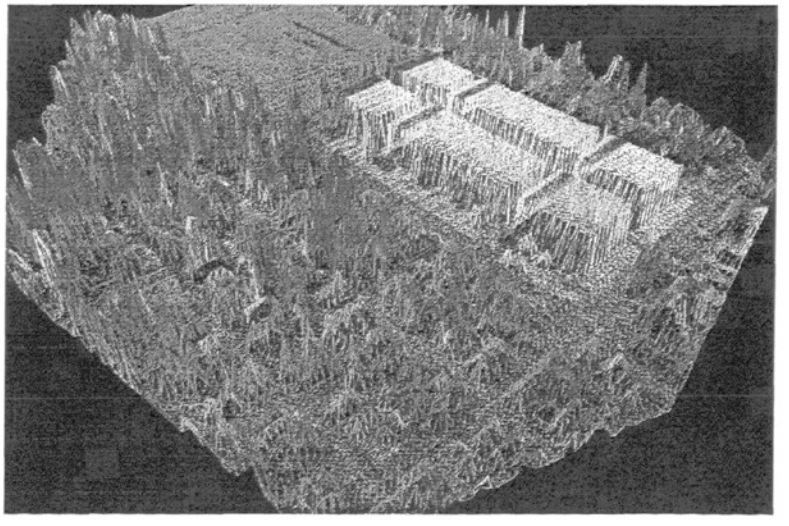


图 3-3 美国奥尔良城区的 LiDAR 点云 TIN 模型

因为三维 Hough 需要在一个三维的参数空间进行，通常运算量比较大。根据 LiDAR 点云自身的特点，可以对参数的取值范围进行限制，固定在一个较小的区间范围内，从而可以大大减少运算量。首先确定点云所在的矩形包围盒，以点云的最高点高度设为 0，将其他点与该点的高度差的绝对值为当前高度，则可以使点云的地面点位置控制在包围盒的较上位置。如图 3-2 所示，则 $\rho \in [0, \max(|OM|, |ON|)]$，$\alpha \in [0, 360]$，$\beta \in [0, \max(\beta_1, \beta_2)]$。对于图 3-3 中的城市 LiDAR 点云的基准地面提取可以得到如图 3-4 的效果，其中平面为提取的基准

地面位置。

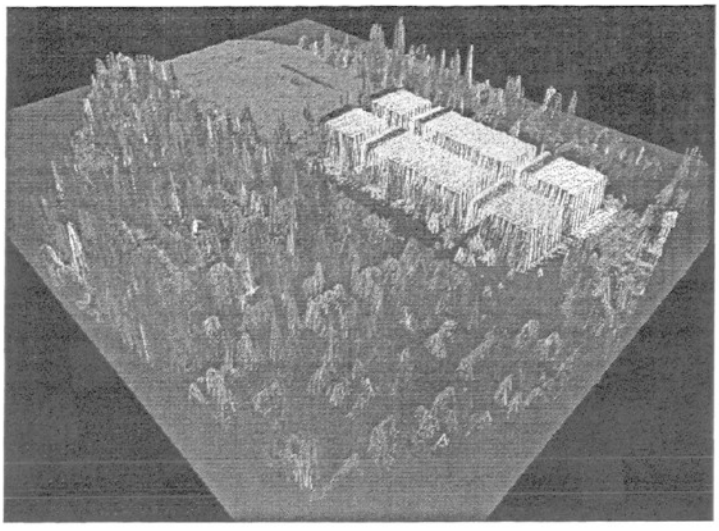


图 3-4 提取的基准平面

3.1.2地面点提取

通过三维 Hough 变换可以提取城市 LiDAR 点云中的基准地面，同时由于地面上的点集构成的 TIN 三角网格一般是比较水平的，因此我们可以将满足如下两个条件的点判定为地面点：

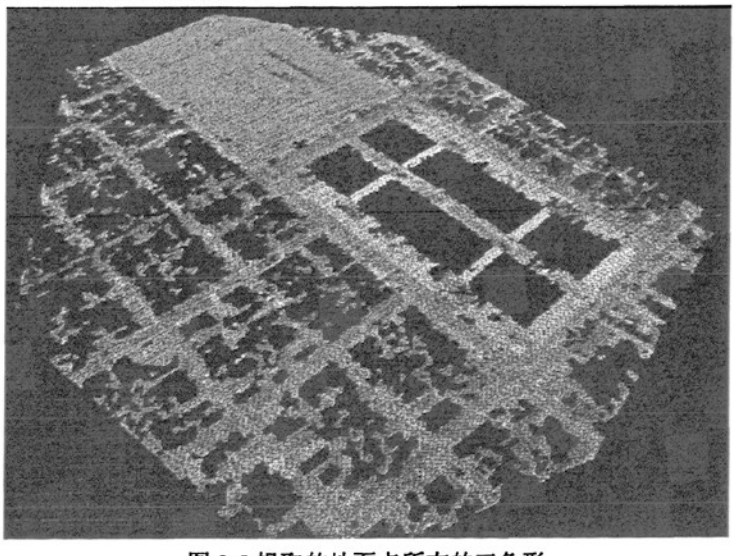


图 3-5 提取的地面点所在的三角形

（1） 根据点云的精度设定一个阈值 T，则地面点与所基准平面 d 的距离小于 T：$d<T$；

（2） 预设一个地面法线的波动范围角 θ_0，则地面点所在的三角形法线 n 与竖直方向 n_0 的夹角 $\theta<\theta_0$。

提取的效果如图 3-5 所示，其为根据图 3-4 的基准地面位置提取的地面点所在的三角形。

3.2 生成点云凸包

在对非地面点过滤的过程中，很可能会滤除掉原场景中的边界点，从而使生成的地形小于原场景。为保持边界的完整性，本文通过求取点云的凸包来构成点云的边界。因为 LiDAR 数据的获取一般都是地面物体表面坐标的海量数据，因此，为使凸包的求取高效进行，需要首先对数据进行分块。

3.2.1 点云分块

点云凸包生成算法的核心就是求取各个方向的最远点，大部分的计算都集中在从大量数据中搜寻最远的点，只要在搜索方法上加以改进就可以大大提高凸包生成速度。

现有的数据分块方法有：均匀条带分割（图 2-5）、等格网分割（图 2-6）、四叉树分割（图 2-7）和自适应分块等。由于求取凸包需要用到分块之间的邻接关系并希望边界处的分块容易查找，因此本文算法利用图 2-6 中的等格网分块法对点云数据进行预处理。

3.2.2 删除内部点

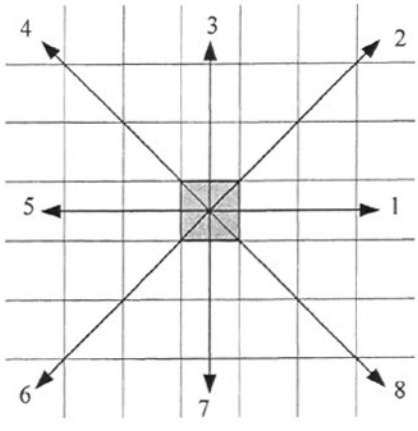


图 3-6 八邻域方向

因为点云凸包顶点都在边界分块中，因此在等格网分块的基础上，首先删除内部的顶点，从而为后面查到凸包顶点省去了大量的查询时间。首先定义分块的八邻域方向，水平向右为 1，逆时针旋转依次为 2，3，...，8，如图 3-6 所示。假设点集共分 $m \times n$ 块 $Grid[m][n]$，其中，Grid 的数据结构中包含每个方向上是否有点的变量：

```
Struct_GRID{
Int nDirection;     //标记八临域方向是否有点
……
```

}

删除内部点的步骤如下：

Step1：初始化各个分块的各个方向上均没有点，$Grid[i][j].nDirection=0(1\le i\le m,1\le j\le n)$；

下面的 *Step2* 和 *Step3* 以水平左方向 5 为例。

Step2：从左至右，查找当前行中最左端含有点的分块 $Grid[i][j]$；

Step3：将当前行中第 j 列之后的每个分块都标记左方有点，$Grid[i][r].nDirection=Grid[i][r].nDirection+1$（$j<r\le n$）。

其它方向 2 至方向 8 的处理如同 *Setp2* 和 *Step3*，当八个方向处理完毕之后即可继续处理下面的 Step4。

Step4：查询所有分块，将八个方向均有点的分块内的点删除，即若 $Grid[i][j].nDirection==8$，则删除该分块内的所有点。

通过上面的步骤可以通过一些简单的判断即可删除点云中的很多点，运算速度较快。如图 3-7 中给出了特定点集删除内部点前后的效果，其中青色部分为待删除的点集。

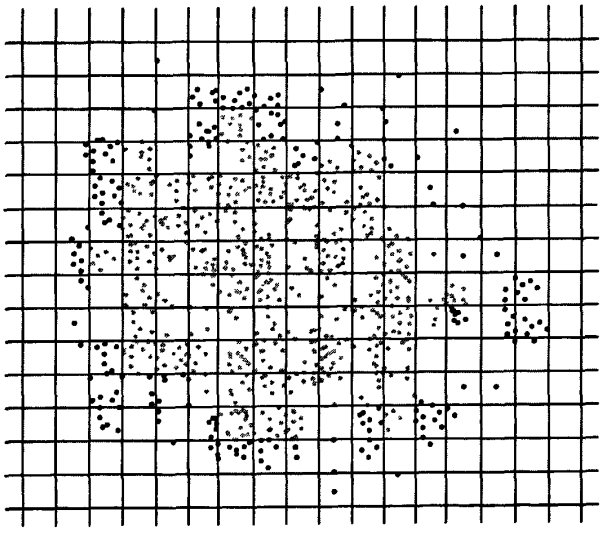


图 3-7 内部点删除示意图

3.2.3 生成初始凸包

根据凸包的性质，点集中任一方向的极值点肯定为凸包顶点。如图 3-8 所示，根据 α_i 的不同取值可以得到不同的方向直线 l_i，然后将点云向 l_i 进行投影，并记最远的投影垂线为 n_i，按逆时针方向分别取垂线 n_i 上最远的两点 *nPoint*[i][0]，nPoint[i][1]，在计算的过程中可以同时计算 α_i 方向和 $\alpha_j=\alpha_i+180°$ 方向上的最远点如图 3-8。在实际的运用中计算初始凸包一般取 $\alpha_1=0°$ 和 $\alpha_2=90°$[21]。

通过依次连接 nPoint[i][0]、nPoint[i][1]、nPoint[i+1][0]、nPoint[i+][1]...，

即可生成所需的初始凸包。连接过程中若某条垂线 n_i 上只有一点，则 nPoint[i][0]==nPoint[i][1]，同时也有可能 nPoint[i][1]==nPoint[i+1][0]，这些都作为需要删除的重复点即可。

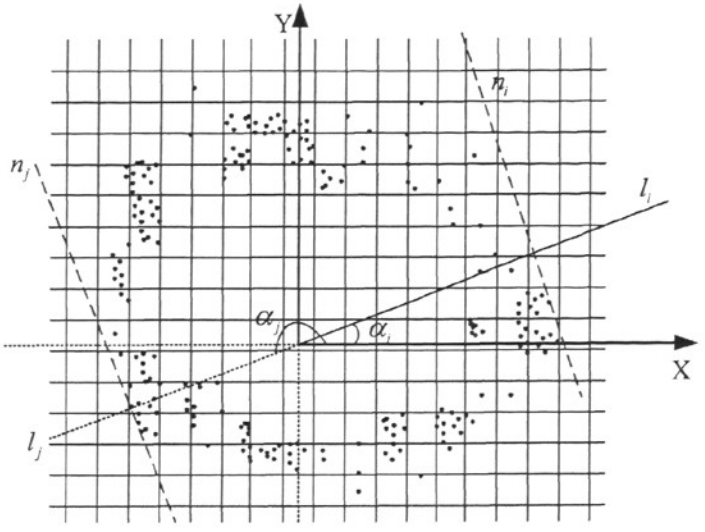


图 3-8 求取不同方向最远点示意图

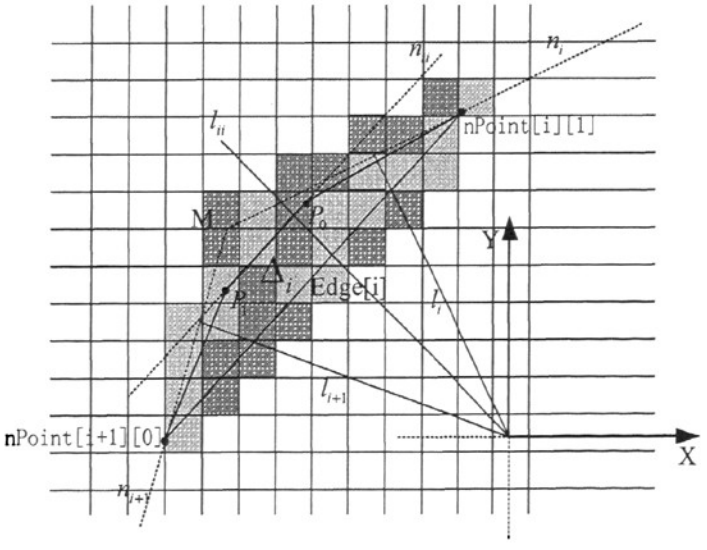


图 3-9 扩展初始凸包

3.2.4生成最终凸包

根据上节方法求取的初始凸包一般不能把所有的点集都包围起来，而是会有一些点在凸包的外面。因此，需要在初始凸包的基础上，通过加入凸包外部的点来生成所有点集的凸包。如图 3-9 所示，连接 nPoint[i][1]和 nPoint[i+1][0]

为初始凸包的一条边 Edge[i]，因为 nPoint[i][1]和 nPoint[i＋1][0]分别为方向 l_i 和 l_{i+1} 方向上的最远点，所以在边 Edge[i]之外的点都在由点 M、nPoint[i][1]和 nPoint[i＋1][0]构成的三角形 Δ_i 之内。在对 Edge[i]进行扩展的过程中只需要查询与三角形 Δ_i 相交的分块网格内的点即可，具体操作就是查找该网格内内离边 Edge[i]最远点所在直线 n_{ii} 的两端点 P_0 和 P_1，然后连接 nPoint[i][1]和 P_0、P_0 和 P_1（如果不重合）、P_1 和 nPoint[i＋1][0]成为新的凸包边，并删除 Edge[i]。重复该操作，当每条边外部不再有点时便生成最终的凸包多边形。

3.3 地形生成

要生成的地形有两部分点集构成，一部分是利用基准平面提取的地面点，另一部分是利用凸包求取算法生成的边界点。其中第二部分的点集（凸包边界点）的高度值与基准地面的距离比较远或者其所在的三角形的法线不够竖直，因此该部分点的高度需要通过附近的地面点的高度值来近似求取，求取的方法可以利用最近的地面点高度值来代替，也可以用最近的若干个点的平均高度来计算。

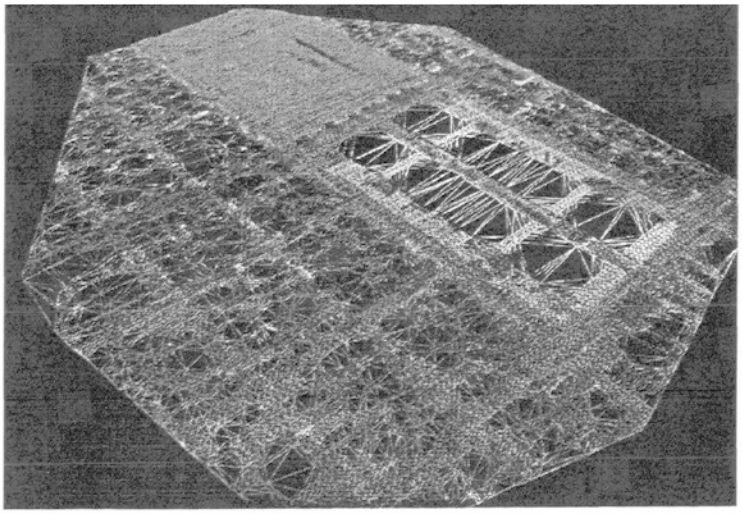


图 3-10 提取的地形网格模型

另一方面，上述的两部分点可能会有一个非空交集，因此可以利用两部分点集合的并集构建 TIN 网格模型即可得到最终的地形（如图 3-10）。

3.4 算法分析

在求取基准地面时，城市地形的特点是由于建筑物的存在使得在 LiDAR 点云中地面的点云是靠近偏下位置的，通过以 LiDAR 点云的最高点为零点位置，其他所有点的高度取为其与最高点的绝对高度差，从而地面点云在原 LiDAR 点云矩形包围盒的靠上位置。当以最小 x，y，z 为坐标原点建立坐标系时，可以使 Hough 变换的参数空间限制在一个较小的空间内，从而不仅节省了内存空间，

同时也加快了计算的速度。

本文的凸包求取算法主要是在现有的凸包求取算法在海量数据处理上的扩展，通过对 LiDAR 点云的分块，按照 8-邻域方向快速删除海量点云的内部点，该删除过程在求取初始凸包前进行，只需要利用分块的邻域关系，通过布尔变量判断块内是否含有点，运算速度比其它先计算初始凸包后判断点云与初始凸包的关系来删除内部点的算法速度要快很多。在从初始凸包进行迭代扩展的过程中，对点的搜索限制在扩展区域三角形与分块的交集内，最大限度地减少了搜索点的个数，从而加快了凸包的求取速度。

3.5 本章小结

本章首先根据城市 LiDAR 点云中地形的起伏缓慢的特性，从点云数据中利用三维 Hough 变换提取基准地面的位置，然后根据点云与基准平面的位置关系和法线方向进行提取地面点。另一方面，为保持地形的边界位置，本章通过改进后的凸包生成算法对 LiDAR 点云进行凸包计算，将提取的地面点与凸包点进行构建 TIN 网格即可得到场景范围内的地形模型。

本章算法充分利用了城市地形缓慢变化的性质，通过首先利用 Hough 变换提取基准地面，从全局的角度具有最优性，因此对于点云的密度要求不高。

第四章 城市人造目标的识别与重建

4.1 城市道路提取

由于从航拍图像中提取道路时一些经验条件难以满足，如要求道路的灰度比背景亮，道路的灰度变换不大等，而从 LiDAR 数据中提取道路对点的要求比较高，如点集密度高、需要强度信息等，因此，本文融合 LiDAR 点云和配准的航空图像两种数据源来提取道路，降低了对单一数据源的依赖性。本章首先介绍了现有的基于动态规划的道路提取方法和代价函数的意义；然后对 LiDAR 点云数据进行重新采样，并根据提取的道路特征对原有代价函数进行扩展；为精确配准航拍图像和 LiDAR 数据，本算法利用局部控制点对当前位置进行插值；最后通过试验对该算法进行验证。

4.1.1 动态规划介绍

动态规划是一种解决最优化问题的一项技术，它适合于目标估计函数中的变量不需要同时全部考虑的情况。现考虑一个相当通用的优化问题：

（1） 求代价函数 g 的最大值 M。 $g=g(x_1,x_2,...,x_n),0\le x_i\le m_i,i=1,2,...,n$。

（2） 求使代价函数达到最大值时的解 $(\overline{x_1},\overline{x_2},...,\overline{x_n})$。

如果所要求的代价函数是一组变量 $x_1,x_2,...,x_n$ 的函数，而 x_i 只取一些离散值，这样的问题只能通过穷举所有变量。然而，如果代价函数 g 由一些步骤组成，且每一步仅仅依赖于其中的部分变量，那么就可以通过多步优化过程来解决。例如，

$$g(x_1,x_2,...,x_n)=g_1(x_1,x_2,x_3)+...g_{n-2}(x_{n-2},x_{n-1},x_n) \quad (1)$$

其中在表达式 g_1 中 x_1 依赖于 x_2 和 x_3。解决的办法就是求 g_1 中有关变量 x_1 的最大值，可以通过列表的办法求解每一对 x_2 和 x_3 使 $g_1(x_1,x_2,x_3)$ 的最大值：

$$f_1(x_2,x_3)=\max_{x_1}[g_1(x_1,x_2,x_3)] \quad (2)$$

因为 $g_2,g_3,...,g_{n-2}$ 的值不再依赖于 x_1，故在这一点不需要考虑。依次类推，通过列表计算下式来消去 x_2，

$$f_2(x_3,x_4)=\max_{x_2}[f_1(x_2,x_3)+g_2(x_2,x_3,x_4)] \quad (3)$$

最后可以求出

$$f_{n-1}(x_n,x_n)=\max_{x_{n-1}}[f_{n-2}(x_{n-1},x_n)] \quad (4)$$

$$M=\max[g]=\max_{x_n}[f_{n-1}(x_n,x_n)] \quad (5)$$

从而解决原问题（1）。要解决问题（2），为求代价函数的最大值而产生的中间值 x_i 和函数值 $f_i(x_{i+1},x_{i+2})$ 必须保存在一个表中。因此，在求解的最后阶段，有 $n-1$ 个表被保存起来，通过在这些表中逆向追踪就可以解决问题（2）。

总结起来，对于一个动态规划程序可以使用下面的递归方程表达：

$$f_0(x_1,x_2)=0 \quad (6)$$

$$f_k(x_{k+1},x_{k+2})=\max_{x_k}[f_{k-1}(x_k,x_{k+1})+g_k(x_k,x_{k+1},x_{k+2})] \quad (7)$$

$$M=\max[g]=\max_{x_n}[f_{n-1}(x_n,x_n)] \quad (8)$$

$$\overline{x_n}=\arg\{\max[f_{n-1}(x_n,x_n)]\} \quad (9)$$

$$\overline{x_k}=tracing_back(\overline{x_{k+1}}) \quad (10)$$

对于动态规划算法，可以得出下面的一些结论：

（1） 本算法能够保证结果是优化的，因为变量的所有可能组合都在程序中被有效地考虑了。

（2） 动态规划算法已经被证明是一个数值稳定的算法。

（3） 该程序由迭代过程来完成。如果有n个变量，每个变量有m个可选值，算法的复杂度为$O(nm^3)$。为了降低实际执行时的复杂度，变量的个数和每个变量的候选值应该尽量减少。

该算法的一个重要特征就是可以强行添加限制条件。例如，任意两个临近的变量取值都不应该小于d，那么我们可以添加一个强制条件不等式$|x_i-x_{i+1}|>d$。如果一个候选值不能够满足这个限制条件，就直接舍弃当前的候选点，而开始下一个候选点的判断。

4.1.2 道路提取

（1） 在低分辨率航拍图像中的应用

利用动态规划从航拍图像中提取道路首先从较低分辨率的图像中[49]，将道路建立成线状的模型。道路模型中同时考虑道路的几何属性和物理属性（如道路为线状、灰度和宽度变化慢、方向变化平缓等）。由于道路可以用多边形来表达$\{p_1,p_2,...,p_n\}$，其中$p_i=(x_i,y_i)$是多边形在图像上的第i个定点。则道路模型可以用下面的代价函数和约束不等式表示：

$$E=\sum_{i=1}^{n-1}\left(\left(E_{p_1}(p_i,p_{i+1})-\beta E_{p_2}(p_i,p_{i+1})+\gamma E_{p_3}(p_i,p_{i+1})\right)\left[1+\cos(\alpha_i-\alpha_{i+1})\right]/|\Delta S_i|\right)$$

$$=\sum_{i=1}^{n-1}E_i(p_{i-1},p_i,p_{i+1}) \quad (11)$$

$$C_i=|\alpha_i-\alpha_{i+1}|<T \quad (12)$$

其中$E_{p_1}(p_i,p_{i+1})$,$E_{p_2}(p_i,p_{i+1})$，和$E_{p_3}(p_i,p_{i+1})$描述道路的各种几何和物理属性，每个表达式只依赖于连续邻接的两个点p_i和p_{i+1}；α_i表示由点p_{i-1}和p_i构成的向量方向；β和γ为正常数；$|\Delta S_i|$为p_{i-1}和p_i之间的距离；T为用户定义的由控制点构成的临近两个向量的方向改变量的上限。

分析(11)式可以知道，代价函数为一系列E_i的加权和，每一个E_i依赖于多边形P的三个连续的控制点(p_{i-1},p_i,p_{i+1})，即每个点p_i只和它的前一个点p_{i-1}、后一个点p_{i+1}直接相关。这使得可以利用动态规划来该问题，通过一系列的决策过程来达到优化的目的。根据精度要求的不同，迭代的过程可以在原有的控制点之间添加一些额外的辅助控制点。该过程大概可以从图 4-1 中看出。

（2） 在中、高分辨率航拍图像中的应用

为使(11)式也能适合从高分辨率航空影像中提取道路，文献[28]研究了中等和较高分辨率航空影像中道路中心线的提取方法，将道路的边缘信息和宽度信息融入到(11)式得到了新的代价函数如下：

$$E=\sum_{i=1}^{n-1}\left(E_{p_1}(p_i,p_{i+1})-\beta E_{p_2}(p_i,p_{i+1})+\gamma E_{p_3}(p_i,p_{i+1})+\lambda E_{p_4}(w_{i-1},w_i,w_{i+1})\right)$$
$$\left[1+\cos(\alpha_i-\alpha_{i+1})\right]/|\Delta S_i|$$
$$\approx\sum_{i=1}^{n-1}E_i(p_{i-1},p_i,p_{i+1},w_i) \quad (13)$$

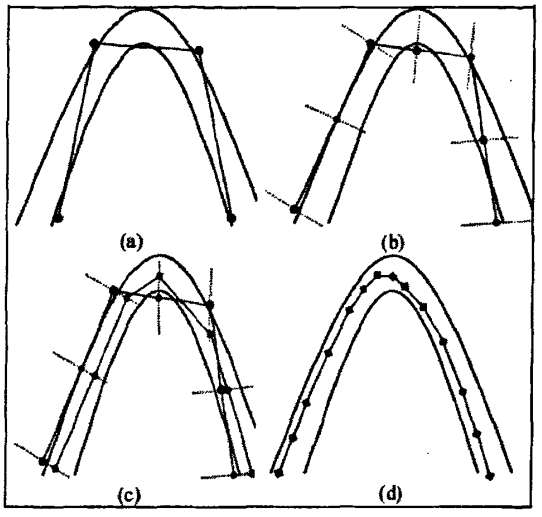


图 4-1 利用动态规划提取道路的迭代过程

4.1.3 算法改进

（1） 代价函数的改进

在(13)式中虽然融入了航空图像中道路的各种属性，但是很多条件是基于经验，当条件改变的时候，该方法很难保证道路提取的正确性，如 E_{p_1} 中假定路面的灰度高于两侧的灰度值[49]，路面的灰度值变化缓慢，这些条件在大多数情况下是成立的，但是不能绝对保证（如斑马线处灰度值变化剧烈等）。为提高算法的鲁棒性，本文中引入了另外一个数据源——LiDAR 点云，从而可以为模型(13)添加更多的路面几何信息，如路面的边缘、高度等。

（2） LiDAR 数据的重采样和特征提取

为使 LiDAR 数据能够与航空图像结合使用，并减少对数据密度的依赖性，本文对 LiDAR 点云进行重新采样。重采样的方法就是，按 LiDAR 点云的 X、Y 范围进行等网格划分，每个网格对应象素的灰度值正比于网格内的点的平均高度。由于 LiDAR 点云分布是杂乱无章、密度不均的（如图 4-2 所示），所以生成的高度图包含空洞[33]，空洞处象素用离最近的象素灰度值代替。至于网格的大小可以根据实际的点云密度来设定。该阶段生成的高度图由于灰度对应

LiDAR 点集的高度信息，为后续处理和叙述的方便，记该高度图为 $\mathrm{Im}age_1$（如图 4-3 所示），并用边缘检测方法（如 Canny 方法）生成边缘图像 $\mathrm{Im}age_2$。因为路面是平滑的，检测出的边缘不会出现在路面上，而道路与两旁的建筑物以及立交桥与两旁的地面有明显的高度差，因此在道路边缘处会检测到边缘点的存在（如图 4-4 所示）。

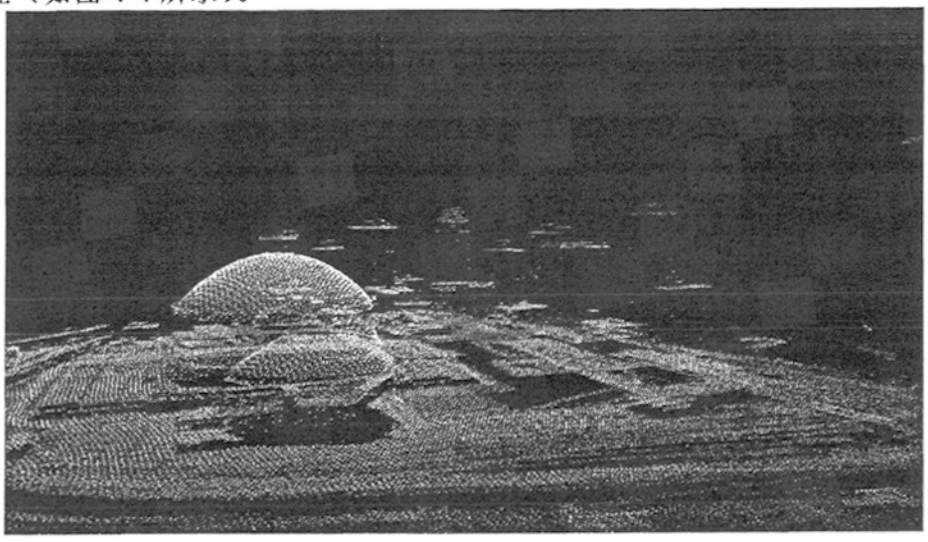


图 4-2 城市 LiDAR 点云数据

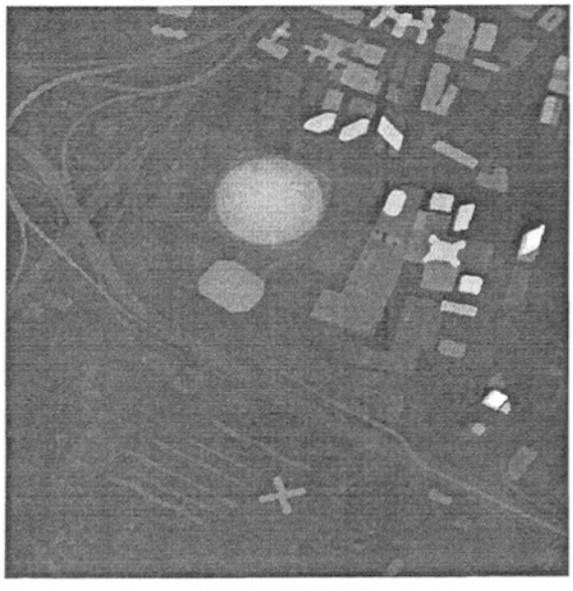


图 4-3 高度图像

为确定道路中心线的精确位置，根据边缘检测图像 $\mathrm{Im}age_2$ 生成边缘距离图像 $\mathrm{Im}age_3$，其不仅在求取道路中心线中用到，在最后求取道路宽度时也要使用该图像中的信息。$\mathrm{Im}age_3$ 生成方法为，对于 $\mathrm{Im}age_2$ 中任一非边缘点 P，求取其最近的边缘点 E_1，若 E_1 有关于 P 对称的边缘点 E_2，则将 P 和 E_1 的距离作为象素 P 点的灰度值，如图 4-5 所示。

（3） LiDAR 数据道路特征

根据图像 $\mathrm{Im}age_1$，$\mathrm{Im}age_2$和$\mathrm{Im}age_3$的生成过程和道路本身固有的属性，可以得到如下有关道路特征的描述：

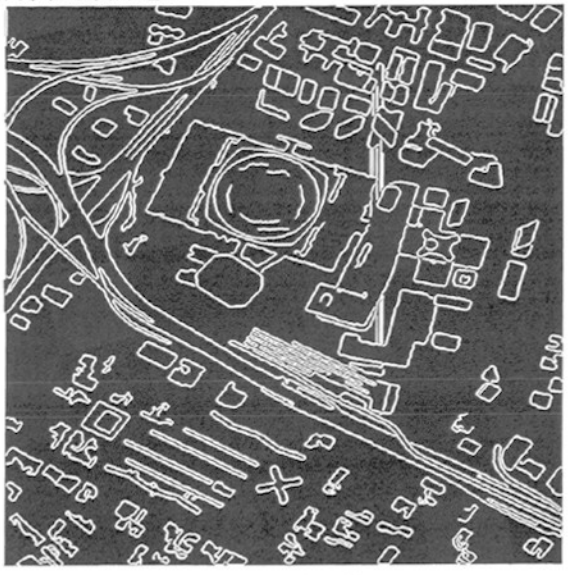


图 4-4 边缘图像

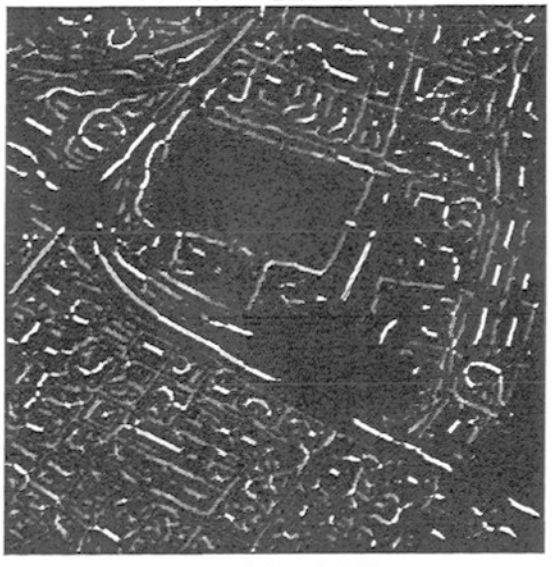


图 4-5 边缘距离图像

1） 小段范围内道路高度变化小，从而对应 $\mathrm{Im}age_1$ 中象素的灰度值变化小，则有

$$E_1 = \sum \int \Delta s_i \{G[f(s)] - G_m(\Delta s_i)\}^2 ds = \min \quad \text{.....(14)}$$

其中 $f(s)$ 为道路曲线的表达式； s 为道路曲线的弧长； Δs_i 为取定的一段道路； G 可分别表示灰度图像 $\mathrm{Im}age_1$ 和航拍图像 $\mathrm{Im}age_4$ 的灰度值；$G_m(\Delta s_i)$ 为对应图像中在道路段 Δs_i 上的平均灰度值，即 $G_m(\Delta s_i) = \int \Delta s_i G[f(s)] ds / |\Delta s_i|$

2）路面是平滑的，在边缘图像 $\mathrm{Im}age_2$ 中道路曲线一般不会横跨边缘点，也即 $\mathrm{Im}age_2$ 中的边缘点对代价函数 E 的贡献应该最小，为

$$E_2 = \sum \int \Delta s_i G[f(s)]ds = \min \quad (15)$$

其中 G 表示图像 $\mathrm{Im}age_2$ 对应图像灰度值。

3）提取的道路应离道路的中心线很近，则边缘距离图像中 $\mathrm{Im}age_3$ 中每点的灰度值对代价函数的影响为

$$E_3 = \sum \int \Delta s_i G[f(s)]ds = \max \quad (16)$$

其中 G 表示图像 $\mathrm{Im}age_3$ 对应图像灰度值。

4）道路呈现带状形态，在短距离内不会有太大的宽度变化，基于此得出道路宽度变化对代价函数的影响：

$$E_4 = \sum [\| G(p_{i+1}) - G(p_i) |] = \min \quad (17)$$

其中 G 表示图像 $\mathrm{Im}age_3$ 对应图像灰度值。

（4） 代价函数的扩充

综合航空图像和 LiDAR 数据的信息，可以从(13)-(17)式中得出最终的道路模型的代价函数：

$$E = \sum_{i=1}^{n-1} \left(E_{p_1}(p_i, p_{i+1}) - \beta E_{p_2}(p_i, p_{i+1}) + \gamma E_{p_3}(p_i, p_{i+1}) + \lambda E_{p_4}(w_{i-1}, w_i, w_{i+1}) - E_{p_5}(p_i, p_{i+1}) \right)$$
$$[1 + \cos(\alpha_i - \alpha_{i+1})] / |\Delta S_i| \quad (18)$$

其中 $E_{p_5}(p_i, p_{i+1}) = \mu_1 E_1 + \mu_2 E_2 - \mu_3 E_3 + \mu_4 E_4$，式(12)、(18)共同组成了本文道路识别的动态规划半自动提取模型。



图 4-6 配准前的航拍图像

4.1.4 数据配准

由于距离较远和影像获取环境的不稳定等因素会使获取的图像有一定的畸

变和倾斜，影响道路的提取效果和精度。因此在结合航空图像和 LiDAR 点云提取道路之前需要对两者进行配准。配准过程需要选定一些控制点作为配准依据，所提供的控制点个数和分布情况影响配准结果。配准的每一步都使用全部的控制点不仅计算量很大，而且当控制点离当前象素点较远时反而会降低配准的精度。

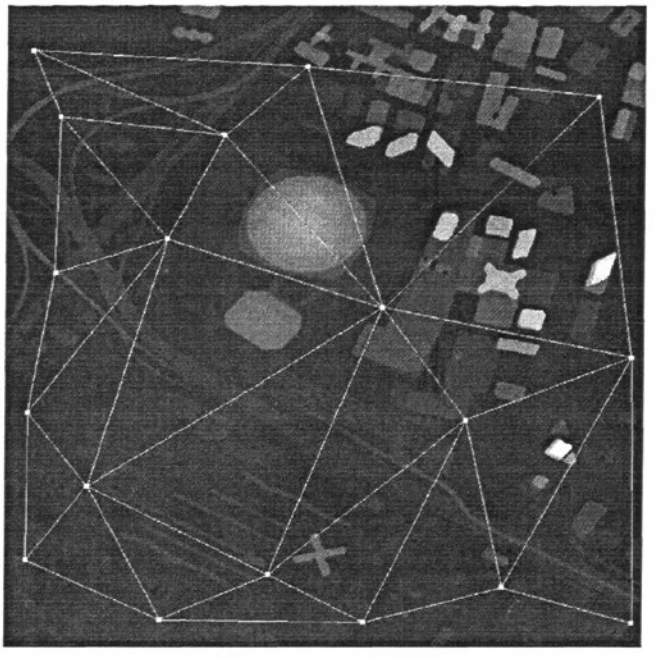


图 4-7 LiDAR 高度图

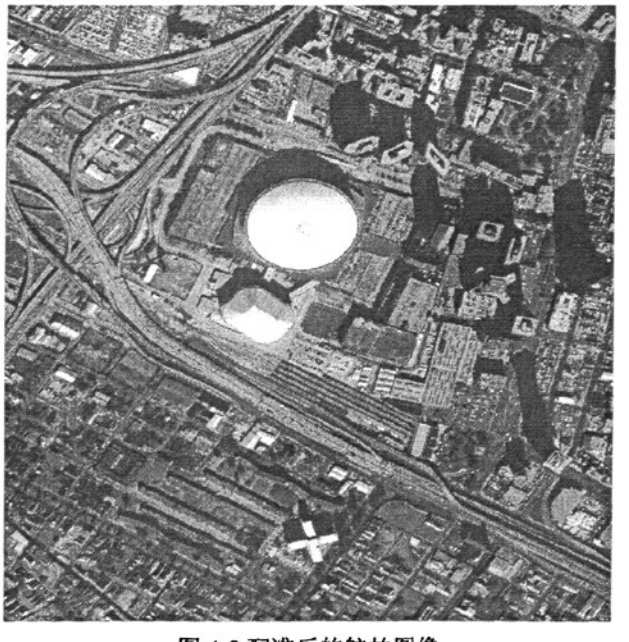


图 4-8 配准后的航拍图像

本文利用基于局部插值的思想，首先对控制点进行构造 Delaunay 三角网，

然后对每个三角形内的象素只使用该三角形的顶点进行配准[50]，既减少了计算量，也能保证配准的精确性。如图 4-6 和图 4-7 分别为配准前的航拍图像和由 LiDAR 数据生成的高度图，以及根据选取的控制点所构建的三角网，图 4-8 为与图 4-7 高度图配准的航拍图像。

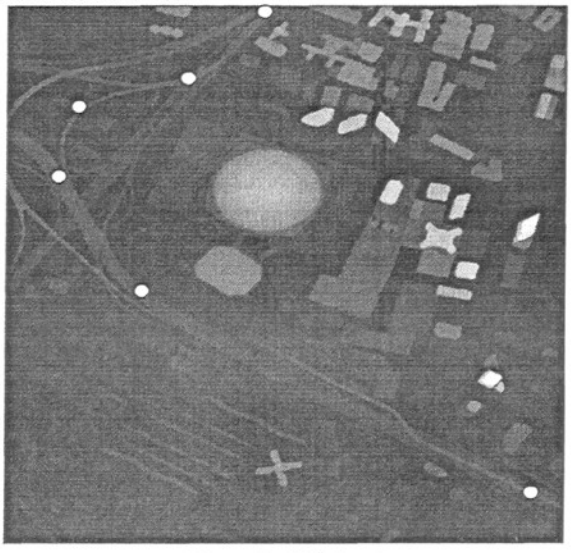


图 4-9 控制点位置

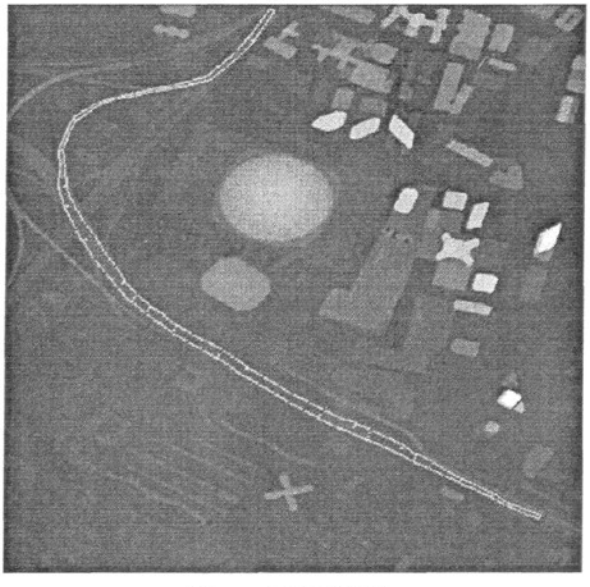


图 4-10 提取的道路

4.1.5 实验结果与分析

本文算法是半自动提取方法，首先由用户在道路附近选定一些控制点，之后的工作就由计算机来自动完成。它会自动算出道路代价函数的最大值，并根据列表保存的中间步骤信息，逆向求出使代价函数达到最大时的道路中心线位置。此外，通过道路中心线的位置和边缘距离图像 $\mathrm{Im}age_3$，算出道路中心线上

任一位置的道路宽度；根据道路中心线的位置和高度图 $Image_1$ 的生成过程计算道路的高度。

所使用的试验数据来源于在美国新奥尔良市区的 LiDAR 数据。图 4-3、图 4-4 和图 4-5 为 LiDAR 数据信息，图 4-8 为该区域的配准航拍图像，图 4-9 中为用户选取的控制点位置，图 4-10 显示所提取的道路位置和宽度信息，图 4-11 则为道路重建后的三维模型。从提取过程来看，用户提供的控制点甚至不需要在道路上，该算法仍然能够取得很好的效果。

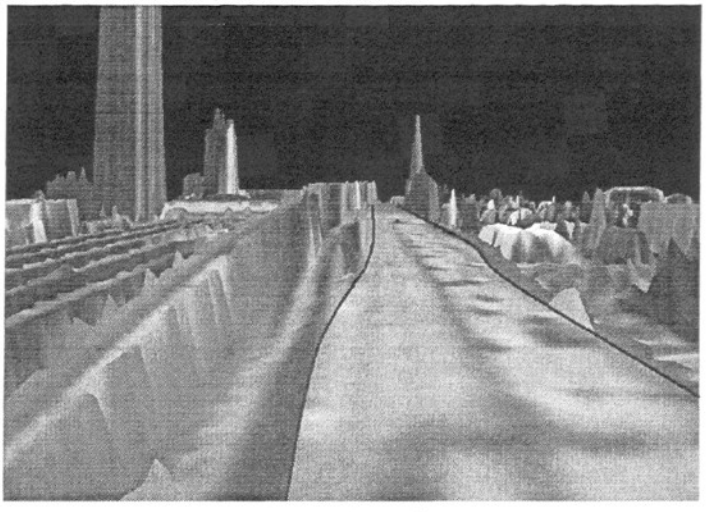


图 4-11 道路重建后

4.2 城市建筑物提取

4.2.1 区域分割

根据 LiDAR 点云高度图（如图 4-3 所示）的生成原理，目标区域内 LiDAR 三维点的高度正比于对应高度图中对应像素的灰度值。地面上的点以及靠近地面的矮小建筑物和其他物体在高度图中的像素灰度都比较偏低，而较高的建筑物对应的像素点则比较亮，因此可以通过图像处理阈值分割将高度图进行二值化处理，从而将高层建筑物区域从背景中分割出来，以便进行以后的建筑物轮廓提取和重建过程。

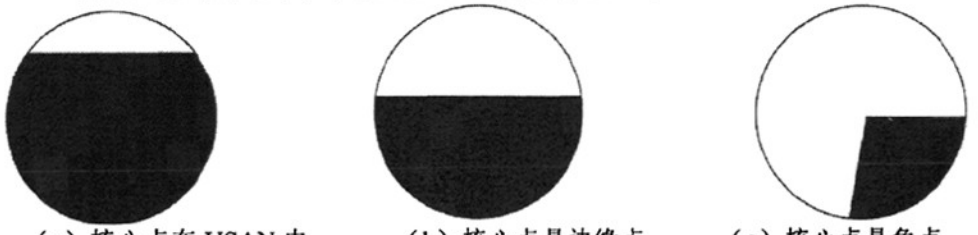
4.2.2 角点检测

目前所研究的角点检测算法较多，总体可分如下两类：基于图像边界信息和基于图像局部亮度值的方法。基于图像边界信息方法的最早的途径之一为基于边缘轮廓链码的角点检测方法。这一类方法需要对图像进行预处理，主要分为三步：首先，对图像进行预分割；然后，对预分割后得到的图像中的边界轮廓进行顺序编码，得到边缘轮廓链码；最后根据边缘轮廓链码对图像中的角点进行描述和提取。如通过链码的象素坐标寻找具有最大曲率的点；对链码进行多边形估计，寻找线段的交点或被描述成两条直线的交叉点等。

基于图像边界信息的角点检测算法需要一定的预处理，复杂度高，容易引入误差。因此近年来提出的角点检测方法大多是直接基于灰度图像的角点检测，主要分为两类：基于模板的角点检测算法和基于几何特征的角点检测算法。

基于模板的角点检测，一般首先先建立一系列具有不同角度的角点模板，然后在一定的窗口内比较待测图像与标准模板之间的类似程度，以此来检测图像中的角点。由于角点结构的复杂性，模板不可能覆盖所有方向的角点，这一类角点检测方法计算量大且比较复杂。

基于几何特征的角点检测方法主要是通过计算象素的微分几何特征来进行角点检测。其中较为典型的有 Kitchen 和 Rosenefeld 提出的基于局部梯度幅值和边界梯度方向改变率的角点检测算法。另一类基于几何特征的角点检测算法为自相关角点检测算法，这类算子不同于一般角点检测的直观定义，它们认为与临域有较大的亮度差或者其局部自相关亮度值较大的象素点为角点，具有代表性的有 Harris 算法和 Smith 与 Brady 提出的一种低层次图像处理小核值相似区的方法 SUSAN（Small Univalue Segment Assimilating Nucleus，SUSAN）算法。Harris 算法检测效果较好，但由于进行了三次高斯滤波，计算量很大。SUSAN 经过大量的实际应用，证明该算法对低层次图像处理具有较好的适用性和可靠性。本文先通过阈值分割能够将高层建筑物从高度图中分离出来，接下来利用 SUSAN 方法可以检测到每个联通区域的角点位置。



（a）核心点在 USAN 内　　（b）核心点是边缘点　　（c）核心点是角点

图 4-12 USAN 的三种典型形状

将位于圆形窗口模板中心等待检测的象素点称为核心点。假设图形中无纹理存在，称与核心点具有一样的灰度值的区域为 USAN（Univalue Segment Assimilating Nuclus）。图 4-12 给出了 USAN 的三种典型形状。从图中可以看到，当核心点位于 USAN 区域内时，USAN 区域面积最大；当核心点位于边缘时，USAN 区域相当于整个临域面积的一半；当核心点的 USAN 区域最小时，核心点是角点。利用这个原理，Smith 等人提出了最小核心值相似区域（SUSAN，Smallest Univalue Segment Assimilating）的角点检测算法。

为计算每个象素的 USAN 面积，需事先设定核心点与其圆临域中其他象素灰度值的差值门限 t，具体执行 SUSAN 时的判别函数如下：

$$c(r,r_0)=\begin{cases}1 & if \quad |I(r)-I(r_0)|\le t\\ 0 & if \quad |I(r)-I(r_0)|>t\end{cases} \qquad (19)$$

式中，$c(r,r0)$是以$r0$为核心的圆模板内的像素r 是否属于USAN的判别函数，$I(r0)$为核

心点$r0$的灰度值，$I(r)$为核心点邻域内像素r 的灰度值。则图像中任一点$r0$的USAN面积可表示如下：

$$n(r_0)=\sum_{r\in C(r_0)} c(r,r_0) \quad \text{.....(20)}$$

其中，$C(r0)$代表以$r0$为核心的圆邻域。计算完每个像素点的USAN面积后，采用一个局部非极大值抑制模板来确定最终角点。具体可由下式表述：

$$R(r_0)=\begin{cases} g-n(r_0) & if \quad n(r_0)<g \\ 0 & otherwise \end{cases} \quad \text{.....(21)}$$

式中，g 为非极大值抑制门限。

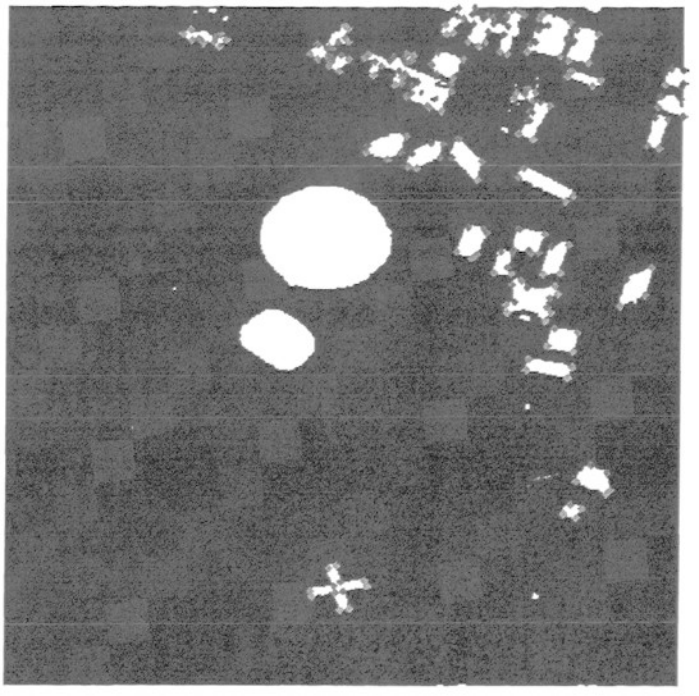


图 4-13 建筑物角点检测结果

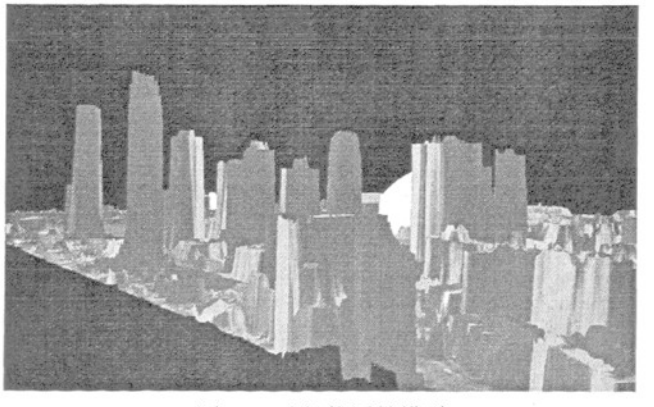


图 4-14 建筑物原始模型

通过上述分析可知，SUSAN 算法的优点在于在角点检测时不需计算梯度，不需插值而且不依赖于前期图像分割的结果，直接对像素的邻域灰度值比较即

可检测出角点，因而在图像处理中得到广泛的应用。

本文通过阈值分割能够将高层建筑物从高度图中分离出来，接下来利用者SUSAN方法可以检测到每个联通区域的角点位置（如图4-13所示）。

4.2.3 三维重建

由于高度图是根据LiDAR点云重新采样所生成的，因此，通过检测出的建筑物角点位置能够反算出每个角点在点云数据中的对应位置，同时根据角点在高度图中的位置和高度图的灰度值可以计算每个角点的高度。因重采样会产生一定的误差，楼房的角点在高度图中的灰度值可能不一致，可以通过计算同一个建筑物的多个角点位置的平均高度作为该建筑物的最终高度。如图4-14—图4-17为建筑物的重建前后比较效果。

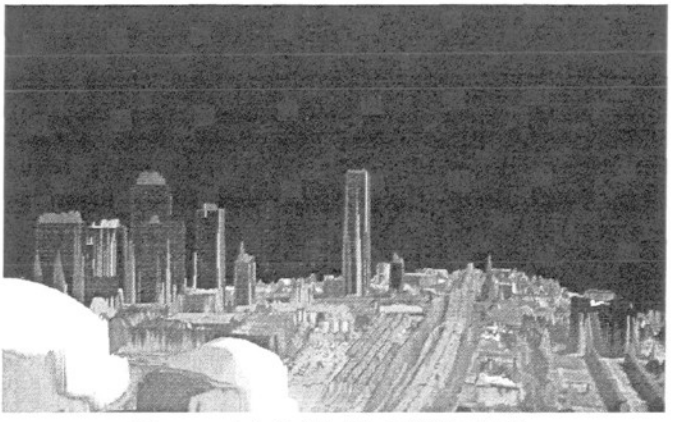


图4-15 重建的建筑物与原始模型比较

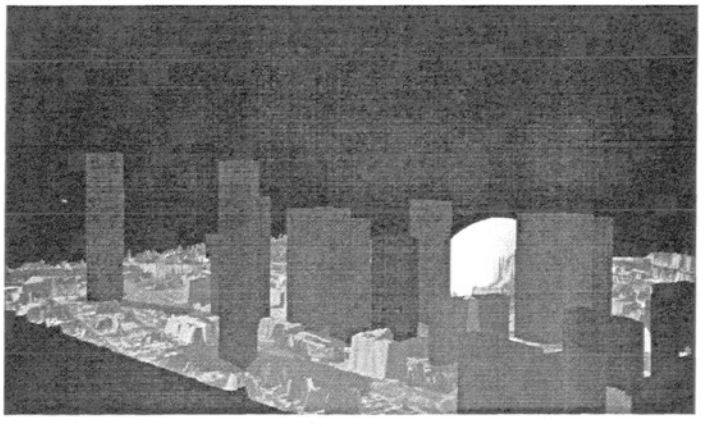


图4-16 未加纹理的建筑物重建

4.3 本章小结

本章首先总结了动态规划方法在道路提取过程中的发展，其最先应用于较低分辨率的航拍图像中，将道路抽象为一条很细的线状，为了将该方法应用于中、高分辨率的航拍图像中，研究人员引入了道路的宽度信息，从而能够提取航拍图像中的道路中心线。为了进一步提高算法的鲁棒性，本章融合航拍图像

和 LiDAR 点云两种数据源，对 LIDAR 点云数据进行重新采样，分别生成对应的高度图、边缘检测图像和边缘距离图像；结合生成的三幅图像和对应的航拍图像提取道路的特征并用于改进后的动态规划算法；最后通过试验验证了本文方法的可行性。

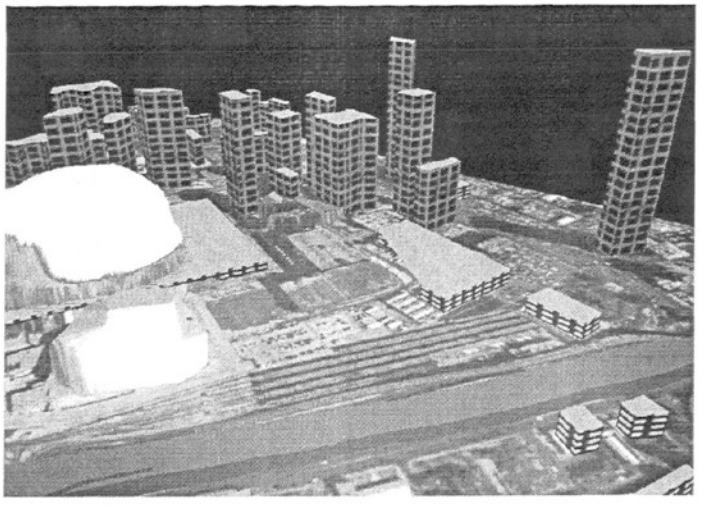


图 4-17 带纹理的建筑物重建

在建筑物提取方面，本文首先对点云生成的高度图进行阈值分割处理，并得到连通区域的边界点集合，然后利用角点检测算法得到建筑物的角点，最后根据高度图的采样原理反算建筑物角点的高度值完成建筑物的重建工作。

第五章 论文总结及展望

5.1 论文总结

本文系统总结了机载激光扫描系统的发展历史、研究现状，并展望了机载激光扫描技术的发展趋势和应用前景。在总结了前人有关基于 LiDAR 的城市重建技术之后，从当前研究和应用的需要出发，针对低密度 LiDAR 点云数据，以美国新奥尔良地区的 LiDAR 点云数据为实验数据来源，并结合航空影像对城市场景重建，研究了城市三维重建的若干关键技术。本文主要贡献可以概括如下：

（1） 对海量 LiDAR 非均匀点云数据进行自适应构网，速度快，且不会由于插值和拼接而引入误差，从而保证每一点的坐标都是实际测量获取的。同时，通过在边界处加入辅助点的方法，可以避免生成比较狭长的非必需的三角形。

（2） 文中对两种数据的配准使用局部范围的控制点进行，这样可以减少由于大气折射等因素造成的数据畸变的影响，且一个控制点的选取误差不会波及到远处的配准效果。

（3） 本文提取城市的地形点集时，使用了带限的 3D HOUGH 变换提取基准平面，因为该方法利用了大范围的点集统计特性，即使在数据密度很稀少时也具有很高的鲁棒性。

（4） 在现有动态规划算法的基础上，文中将 LiDAR 数据和航空影像进行了融合，使得半自动提取道路具有更高的鲁棒性，且可以根据 LiDAR 数据获取所提取的道路的高度信息。

5.2 论文展望

本文研究了如何利用机载 LiDAR 点云数据和航空影像来重建三维城市的一些关键技术，主要在现有算法之上进行一定的改进，使得一些算法能够更好地服务于城市三维重建工作。但是文中所研究的方法仍然有很多地方需要进一步的完善。

（1） 利用串行的方法加速海量点云的构网过程会随着数据量的增大而越来越没有吸引力，提高计算能力也是大规模点云快速构网的重要途径，将构网算法并行化已成为当前研究的热点[53]，这也是本文算法下一步的研究方向。

（2） 本文所使用的数据配准方法虽然一定程度上可以纠正由于大气折射等因素造成的图像畸变，但是其所需要的控制点太多，且不能够自动完成，因此该方面需要更多的工作要做。

（3） 在提取城市的地形时默认的一个假设是地形的起伏很小，从而使本文的算法使用具有一定的局限性；本文在提取地面之后，可能会有很多的邻接三角形可以用一个更大的多边形来近似较为合适，因为这样可以减少绘制的图元个数。

（4） 本文提取的建筑物是用规则多边形来近似的，然而实际中的建筑物往往并非如此，而且屋顶也并非一定是水平的，甚至并非是平面，本文的近似会使太多的建筑物几何特征不能很好地保存下来。

参考文献

[1] 张栋. 基于 LIDAR 数据和航空影像的城市房屋三维重建[硕士学位论文].武汉：武汉大学. 2005.

[2] 张昊. LIDAR 建筑物提取方法研究[硕士学位论文]. 天津：中国民航大学. 2007.

[3] Elmqvist M, Jungert E, Lantz F, et al. Terrain Modelling and Analysis Using Laser Scanner Data[C]. IAPRS, Annapolis, MD, 2001

[4] Roggero M, Airborne Laser Scanning: Clustering in Raw Data[C]. IAPRS, Annapolis, 2001

[5] Wack R, Wimmer A. Digital Terrain Models from Airborne Laser Scanner Data: a Grid Based Approach[C]. IAPRS, Graz, Austria, 2002

[6] Sithle G. Filtering of Laser Altimetry Data Using a Slope Adaptive Filter[C]. IAPRS, Annapolis, MD, 2001.

[7] Sohn G, Dowman I. Terrain Surface Reconstruction by the Use of Tetrahedron Model with the MDL Criterion[C]. IAPRS, Graz, Austria, 2002

[8] Axelsson P. DEM Generation from Laser Scanner Data Using Adaptive TIN Models[C]. IAPRS, Amsterdam, 2000

[9] Sithole G, Vosselman G. Experimental Comparison of Filter Algorithms for Bare-Earth Extraction from Airborne Laser Scanning Points Clouds[J]. ISPRS Journal of Photogrammetry & Remote Sensing, 2004, 59: 85-101

[10] 孔德慧，马春玲. 一种平面点集凸包与三角网综合生成的算法[J]. 计算机研究与发展, 2000, 37(7): 891-896

[11] Kallay M. The Complexity of Incremental Convex Hull Algorithm in Rd[J]. Information Processing Letters, 1984, (19): 197

[12] Graham R L. An Efficient Algorithm for Determining the Convex Hull of a Finite Point Set[J]. Information Processing Letters, 1972, (1): 132-133

[13] Javis R A. On the Identification of the Convex Hull of a Finite Set of Points in the Plane[J]. Information Processing Letters, 1973, (2): 18-21

[14] Miller R, Strut Q F. Mesh Computer Algorithms for Computational Geometry[J]. IEEE Trans Computer, 1989, 38(3): 321-340

[15] 刘新，刘任任. 一种改进的构建凸包的分治算法[J]. 计算机工程与科学, 2006, 28(8): 63-65

[16] Zhou Qihai, Huang Tao, Wu Hongyu, Li Zhongjun, Lin Xun. A New Algorithm for Finding Convex Hull with a Maximum Pitch of the Dynamical Base Line[J]. IEEE: 2008 Workshop on Knowledge Discovery and Data Mining, 2008:

630-634

[17] 王志强，洪嘉振，肖立谨. 平面点集凸包的最优实时算法[J]. 计算机学报，1998, 21(增刊): 351-356

[18] 蒋红菲. 平面点集凸包快速构建算法的研究[J]. 计算机工程与应用，2002, 20: 48-49

[19] 余翔宇，孙洪，余志雄. 改进的二维点集凸包快速求取方法[J]. 武汉理工大学学报, 2005, 27(10): 81-83, 92

[20] 郝建强. 利用正负划分性求平面点集凸包的最优算法[J]. 中国图象图形学报, 2007, 12(5): 910-916

[21] Chadnov R V, Skvortsov A V. Convex Hull Algorithm Review[J]. IEEE: Science and Technology, KORUS 2004, Proceedings, 2004: 112-115

[22] Barber C B, Dobkin D P, Huhdanpaa H. The Quickhull Algorthm for Convex Hulls[J]. ACM Transaction on Mathematical Software, 1996, 22(4):469-483

[23] 金文华，何涛，刘晓平，唐卫清，唐荣锡. 基于有序简单多边形的平面点集凸包快速求取算法[J]. 计算机学报, 1998, 21(6): 533-539

[24] Ton J., Jain A.K.,Enslin W.R.,et al.Automatic road identification and labeling in Landsat 4 TM images[J].Photogrammetric Engineering & Remote Sensing,1989,43(2): 57~276

[25] Wang F., Newkirk R.. A knowledge- based system for highway network extraction[J].IEEE Transactions on Geoscience and Remote Sensing,1988,26(5): 525~531

[26] Barzohar M., Cooper D.B.. Automatic finding of main roads in aerial images by using geometric- stochastic models and estimation[J].IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996, 18(7): 707~721

[27] 杨云，朱长青，张德. 高分辨率遥感影像上道路中心线的半自动提取[J].计算机辅助设计与图形学学报, 2007, 19(6): 781-785

[28] Pos A.,Vale G..Dynamic programming approach for semi- automated road extraction from medium and high-resolution images[J]. International Archives of the Photogrammetry,Remote Sensing and Spatial Information Sciences,2003,34(8): 87~91

[29] Armin Gruen,Haihong Li.Semi-automatic linear feature extraction by dynamic programming and LSB-snakes[J]. Photogrammetric Engineering and Romote Sensing, 1997, 63(8): 985~995

[30] Shackelford A.K., Davis C.H.. Fully automated road network extraction from high-resolution satellite multispectral imagery[C].PPProceedings of the IEEE International Geoscience and Remote Sensing Symposium,2003,Vol(1):

461~463

[31] Zhu C.,Shi W.,Pesaresi M.,*et al*.The recognition of road network from high-resolution satellite remotely sensed data using image morphological characteristics [J]. International Journal of Remote Sensing,2005,26 (24): 5493~5508

[32] Hatger C. ,Brenner C..Extraction of road geometry parameters form laser scanning and existing databases[J].Int'l Archives of the Photogrammetry,Remote Sensing and Spatial Information Sciences,Vol.XXIV,Part 3/W13,2003

[33] Rieger W.,Kerschner M.,Reiter T.,Rottensteiner F..Roads and buildings from laser scanner data within a forest enterprise[J].In：IAPRS,ISPRS Workshop,Vol. 32,La Jolla,California,1999,185~191

[34] 王涛，杨建思，廖明生. 从机载激光扫描数据中提取道路.测绘信息与工程[J]. 2006, 31(5): 10~11

[35] Simon Clode,Peter Kootsookos,Franz Rottensteiner.The automatic extraction of roads from lidar data[J]. Remote Sensing and Spatial Information Sciences ,2004, 231~236

[36] 阎平. LIDAR 数据中多层次、多直角房屋的三维重建[硕士学位论文]. 武汉：武汉大学. 2005.

[37] 张小红. 机载激光扫描测高数据滤波及地物提取[博士学位论文]. 武汉：武汉大学. 2002.

[38] 周培德. 计算几何——算法分析与设计[M].北京:清华大学出版社，南宁:广西科学技术出版社, 2000.

[39] 周杰，丁贤荣，汪德耀. 平面散点集 Delaunay 三角剖分的一种高效方法[J], 测绘信息与工程, 2003, 28(6): 21-23

[40] 胡金星，吴焕萍，潘懋，马照亭. 基于格网划分的海量 DEM 数据生成[J]. 计算机辅助设计与图形学学报, 2004, 16(1): 41-44.

[41] 武玉国，杜莹，王晓明，游雄. 大规模地形 TIN 模型的 LOD 算法设计与实现[J]. 系统仿真学报, 2005, 17(3): 665-669.

[42] Maus A. Delaunay triangulation and the convex hull of n points in expected linear time[J]. BIT, 1984(24): 151-163

[43] 武晓波，王世新，肖春生.Delaunay 三角网的生成算法研究[J]. 测绘学报, 1999, 28(1): 28-35.

[44] 何俊，戴浩，谢永强，刘宝生. 一种改进的快速 Delaunay 三角剖分算法[J]. 系统仿真学报, 2006, 18(11): 3055-3057.

[45] 栾晓岩. 一种 TIN 生成算法及其三维显示[J]. 海洋测绘, 2004, 24(5): 39-41.

[46] 徐青, 常歌, 杨力. 基于自适应分块的 TIN 三角网建立算法[J]. 中国图象图形学报, 2000, 5(6): 461-465.

[47] 赵文芳. 离散点集 Delaunay 三角网生成算法改进与软件开发[J]. 测绘工程, 2003, 12(4): 22-25.

[48] Fang TP., Pieql LA. Delaunay trianglulation using a uniform grid[J]. IEEE Computer Graphics and Application, 1993, 13(3): 36-47.

[49] Armin Gruen, Haihong Li. Road extraction from aerial and satellite images by dynamic programming[J]. JSPRS Journal of Photogrammetry and Remote Sensing, 1995, 50(4): 11~20

[50] 罗月童, 朱晓强, 刘晓平. 基于三点的航拍图像倾斜校正方法及其应用[J]. 仪器仪表学报(增刊), 2007, 28(8): 590~593

[51] 陈利军.图像角点检测和匹配算法的研究[硕士学位论文]. 西安：西安电子科技大学. 2005.

[52] 郭海霞, 解凯. 基于 USAN 的改进的角点检测算法[J]. 计算机工程, 2007, 33(22): 232~234

[53] Chen MB, Chuang TR, Wu JJ. Parallel divid-and-conquer scheme for 2D Delaunay triangulation[J]. Concurrency and Computation — Practice and Experience, 2006, 18: 1595-1612.

[54] Sangyoon Lee, Chan-lk Park, Chan-Mo Park. An improved parallel algorithm for Delauany triangulation on distribution memory parallel computers[C]. 1997 Advances in Parallel and Distributed Computing Conference (APDC '97), 1997: 131-138.

[55] Chen MB, Chuang TR, Wu JJ. A parallel divid-and-conquer scheme for Delaunay triangulation[C]. *In* 9th International Conference on Parallel and Distributed Systems (ICPADS), 2002: 571-576.

硕士研究生在读期间完成的论文

1）罗月童，朱晓强，刘晓平. 基于三点的航拍图像倾斜校正方法及其应用[J]. 仪器仪表学报（增刊），2007，28（8）：590～593
2）朱晓强，余烨，刘晓平，袁晓辉， Bill P. Buckles. 基于航拍图像和 LiDAR 点云的城市道路提取[C]. 计算机技术与应用进展·2008，2008：1～6
3）余烨，张高峰，朱晓强，凌实，刘晓平，袁晓辉，Bill P. Buckles. 基于 LiDAR 数据的辅助洪水灾害评估方法[C]. 计算机技术与应用进展·2008，2008：18～24
4）刘晓平，朱晓强，余 烨，袁晓辉，Bill P. Buckles. 基于 LiDAR 点云数据的三角网构建算法研究[J]. 软件学报（增刊），2008，19 卷：1-9
5）刘晓平，朱晓强，余烨，袁晓辉， Bill P. Buckles. 基于动态规划的城市道路提取改进算法. 工程图学学报（已录用）
6）刘晓平，余烨，张高峰，朱晓强，凌实，袁晓辉，Bill P. Buckles. 面向洪水灾害评估的城市建模与仿真. 系统仿真学报（已录用）

硕士研究生在读期间参加的科研工作

1）（炮院）合作项目（070503D2）：航空图像和数字地图匹配系统。
2）军口 863 预研项目。
3）国家自然科学基金国际合作交流项目（60573174）：一种新的经济和环境规划分析工具（LiDAR 应用领域的扩展）。
4）合肥市科技局项目：三河古镇的数字化重建和真实感漫游。
5）特种显示技术教育部重点实验室开放课题基金：基于 LIDAR 点云数据及图像的城市建模。
6）军工项目：军工某型号产品可视化子系统。

硕士研究生在读期间获奖情况

1）合肥工业大学第十一届“斛兵杯”科技作品竞赛二等奖。
2）合肥工业大学第四届研究生体育节台球联赛团体亚军。
3）全国第四届研究生数学建模三等奖。
4）全国第 19 届计算机与技术应用（CACIS•2008）学术会议优秀学生论文一等奖。
5）校优秀毕业生。