第 34 卷 第 9 期 计 算 机 工 程 2008 年 5 月
Vol.34 No.9 Computer Engineering May 2008
·多媒体技术及应用· 文章编号：1000—3428(2008)09—0238—02 文献标识码：A 中图分类号：TP391.9

面向对象的城市建筑物点云数据加载和组织

路兴昌[1,2]，张艳红[1]，张爱武[2]
(1. 吉林大学地探学院，长春 130026；2. 首都师范大学三维空间信息获取与应用教育部重点实验室，北京 100037)

摘 要：采用面向对象程序设计方法对三维点云数据进行分析，研究以 Visual C++6.0 为开发平台利用图像法对激光扫描获取的城市建筑物点云数据进行组织的计算机实现方法。通过扫描关系确定点云数据中有效目标点的全局唯一索引值，将点云数据看作一幅深度图像，确定全局索引值对应的图像行列值，再利用该行列值计算每个点对应的法向量，分类和组织点云数据。实验结果表明，利用面向对象的图像处理技术可以方便、快捷地实现三维点云数据的加载和组织。

关键词：面向对象；点云数据；深度图像；组织

Object-oriented Loading and Organization on Points Cloud Data of Urban Building

LU Xing-chang[1,2], ZHANG Yan-hong[1], ZHANG Ai-wu[2]
(1. College of Geo-exploration Science and Technology, Jilin University, Changchun 130026;
2. Ministry of Education Key Laboratory of 3D Information Acquisition and Application, Capital Normal University, Beijing 100037)

【Abstract】 Three dimensional points cloud data are analyzed by using object-oriented programmer, and the computer-based realization, which is organized by points cloud data of urban building captured by laser scanning, is studied based on image process on the exploration platform of Visual C++6.0. The unique global indexes of valid target points are determined by scanning relationship between points. The image range corresponding to the global indexes is fixed on by taking points cloud data as a range image. The points cloud data are classified and organized by calculating the normals of all corresponding points by using the image ranges obtained before. The experimental results indicate that loading and organizing three dimensional points cloud data can be implemented easily and quickly by using object-oriented image technique.

【Key words】 object-oriented; points cloud data; range image; organization

1 概述

激光扫描是应用十分广泛的空间信息获取技术，它通过高速激光测量大面积高分辨率快速地获取被测对象表面每个采样点的三维坐标数据[1]，得到点的集合称为点云。利用点云数据可以建立被测对象的三维立体模型，实现真实场景的三维重建。由于点云数据包含了地物实际采样点到当前扫描视点的距离，其在二维投影平面上的点云图也被称为距离(或深度)图像[2]。每一幅深度图像代表了一个三维的点集，这些点集中的点数据和二维灰度图像中像素的组织方式类似，都是按照二维矩阵的方式来组织图像中的每个像素。与灰度图像不同的是，深度图像中每个像素代表的是一个三维点的坐标，而灰度图像中每个像素代表的是该点的颜色或灰度值[3]。

通过对被测目标对象直接采样获取的三维点云数据经过去噪和补洞等预处理后，点云数据在初始扫描时所具有的点间关系已不复存在，成为一个无任何组织的杂乱采样点集，数据文件中丢失了每个点在深度图像上的原始行列索引信息，在这种数据集上的对某指定信息的直接全局查找搜索是不可行的[4]。因此，必须首先要进行预处理后输入数据的组织工作，即建立一定的数据结构，确立预处理后的每个点在原始点集像素矩阵中的行列索引，这对于利用深度图像数据进行点云的分类和特征提取是很重要的。

在激光扫描点云数据的处理和应用中，文献[5-7]提出通过构建三角网组织点云数据，文献[4,8]直接将预处理后的点云数据通过聚类后重新组织。三角剖分对于大数据量点云数据来说，其网格化操作在时间和空间代价上都是不可接受的；直接对三维点聚类能较好地对点云数据进行分类，但在算法的理解与实现上有一定难度。本文中将基于中远景三维激光扫描系统RIEGL LMS420i[9]获取的室外建筑物表面采样数据投影到二维空间，探讨基于深度图像的点云数据在计算机中组织实现方法。

2 算法思想

在深度图像中，二维像素矩阵空间上的每一点(u, v)(u, v 分别代表该点的列值和行值)，都对应着三维空间上一个 4 维向量(x, y, z, r)。其中(x, y, z)表示实际扫描点在扫描仪坐标系下的三维坐标，r 则表示激光束返回时的强度。扫描仪坐标系下的每一个实际扫描点都有一个唯一的垂直扫描角 *Theta* 和水平扫描角 *Phi*，它们是对实际扫描点的准确特征表示，是二维像素矩阵行列值判定的依据。

本文算法实现过程描述如下：首先读取有效目标地物点云数据，得到预处理后每个有效目标地物点的垂直扫描角 *Theta*2 和水平扫描角 *Phi*2；再读取预处理前的原始点云数据，得到每个扫描采样点的索引值 ID 以及其垂直扫描角 *Theta*1、水平扫描角 *Phi*1；比较 2 次读取的垂直扫描角和水平扫描角，将垂直扫描角和水平扫描角都相等的点在原始点云中的索引

作者简介：路兴昌(1972－)，男，讲师、在职博士研究生，主研方向：三维虚拟建模；张艳红，副教授、博士；张爱武，教授、博士
收稿日期：2007-05-23 **E-mail：**luxingchang@163.com

值 ID 记录并保存下来，这样就获得了每个有效目标地物点在全部点集中的实际索引值 ID(图 1) .根据扫描的行列总数 *m_H* 及 *m_L*，计算得到每个有效目标地物点在深度图像像素矩阵中的行列索引 *i_H* 和 *i_L*(式(1)、式(2))；利用得到的深度图像像素点的行列索引计算每个点在其邻域平面的法向量，最终将各个点以点集的形式组织起来。

$$i\_H=\mathrm{ID}\%m\_H \quad (1)$$

$$i\_L=\mathrm{ID}/m\_H \quad (2)$$

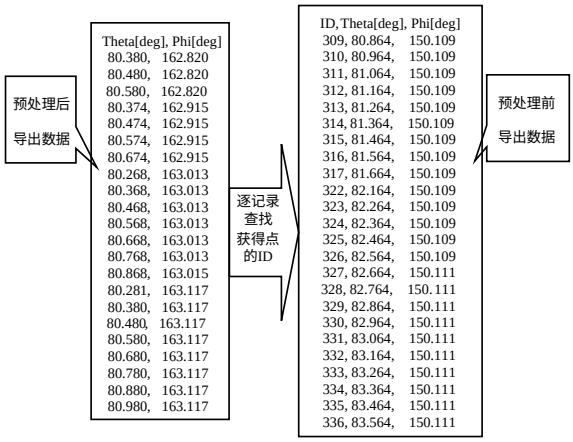

图 1　有效目标地物点全局索引值的确定

3　面向对象的编程实现

应用面向对象的程序设计方法，可以快速、方便地实现深度图像数据的加载和组织，确定有效目标点在初始全局扫描坐标系下的真实索引值，其在三维点云处理的应用集中体现在各个类的设计上。为了管理每幅深度图像的信息，首先将深度图像的信息封装成类 CRangeImage，该类充当了用户命令和实际处理之间的中间层，通过对链表中的深度图像指针的访问和操作来完成对每幅深度图像的处理。根据系统所得到数据的方式，设计了 2 个结构(structure)体来组织数据：

```
struct RangePoint
{ int iID; //点的索引
  float ptX; // x 分量
  float ptY; //y 分量
  float ptZ; //z 分量};
struct InsidePoint
{ BOOL m_bPtInside; //预处理后点的标志
  float *m_pInsidePtXYZ; //点坐标值
  int iValidPtId;   //预处理后点的索引值 };
```

在类 CRangeImage 中，分别有类型为 RangePoint 类型的指针 pRangePoint 和 InsidePoint 类型的指针 pInsidePoint。这 2 个指针都是从堆中动态分配，pRangePoint 数组的元素数目为预处理后的点数，pInsidePoint 数组单元数目是预处理前深度图像中的全部点数。为了便于对这 2 个数组的快速访问，在 RangePoint 类型中由变量 *i*ID 来记录该点在数组 pInsidePoint 中的索引，而 InsidePoint 中的元素 iValidPtId 则记录该点在数组 pRangePoint 中的索引值。

深度图像数据获取和组织的关键代码如下：

```
BOOL CRangeImage::ReadFile(CString strFileName)
{… //首先得到原始深度图像的总行数 H 和总列数 L
  pInsidePoint=new InsidePoint[H*L];
  //读取 3pf 文件得到预处理后总点数 iPtAmount
  pRangePoint=new RangePoint[iPtAmount];
  //得到预处理后文本文件 strTextFile
  //打开 strTextFile 并定位文件指针，利用文件对话框选择预处理
//前的文本文件 strOrigFile
  //打开 strOrigFile 文件，并定位文件指针
  for(int iRead=0;iRead<iPtAmount;iRead++){
  //从 strTextFile 文件中读出一条记录并分离出两个角度分别为
//theta 和 phi
   //从 strOrigFile 中读出一条记录分离出点索引和角度信息如果
//在 strOrigFile 中没有找到相等的角度就循环
   while(tempTheta!=theta 或者 tempPhi!=phi)
   { //继续读出 strOrigFile 一条记录，并分离出信息}
    pRangePoint[iRead].iID=iPtID;//保存点在全部点集中的 id
  pInsidePoint[iPtID].m_bPtInside=true;//设置点的标志
    if(pInsidePoint[iPtID].m_bPtInside)
    { //如果该点有效给该点分配坐标空间
    pInsidePoint[iPtID].m_pInsidePtXYZ=new float[3];   }
    //记录该点在有效点集中的索引值
  pInsidePoint[iPtID].iValidPtId=iRead; }
   //继续读 3pf 文件得到预处理后的点坐标
   pTotalPtSet=new C3DPointSet;
  //给预处理的点集分配一个点集对象
   for(int iBinary=0;iBinary<iPtAmount;iBinary++) {
    //读出一条 3pf 文件数据体记录，其中包括三维点坐标 x,y,z
    pRangePoint[iBinary].ptX=x;
    pRangePoint[iBinary].ptY=y;
    pRangePoint[iBinary].ptZ=z;
    //将该坐标值加入到点集对象 pTotalPtSet 中
    pTotalPtSet->InsertPtSets(x,y,z);
    int iValidId=pRangePoint[iBinary].iID;
    if(pInsidePoint[iValidId].m_bPtInside)
    { //保存该点到数组 pInsidePoint 中的相应单元处
     pInsidePoint[iValidId].m_pInsidePtXYZ[0]=x;
     pInsidePoint[iValidId].m_pInsidePtXYZ[1]=y;
     pInsidePoint[iValidId].m_pInsidePtXYZ[2]=z; } }
    //其他处理  }
```

以上代码利用对深度图像预处理后导出的文本文件和预处理前导出的文本文件获得预处理后点集中每个点的索引值信息，根据扫描仪采样点时的竖直和水平角度来获得每个点在全部点集中的ID值。确定有效目标地物点全局索引值后，利用式(1)和式(2)将获取的有效目标地物点ID值转换为二维行列索引后进行局部拟合平面法向量计算，将法向量拟合成功的点归入有效点集，法向量拟合不成功的点归入无效点集[10]，从而最终将组成深度图像的点集组织成为有效点集和无效点集 2 类。对于有效点集，设计了如下的结构体来完成对各个点的组织和标识：

```
struct singlePt
{ BOOL m_bNormal;
   float ptNormal[3];
   float ptCenter[3];
   BOOL m_bMerge;
   C3DPointSet *m_p3DPtSet;
   BOOL m_bInvalidate;
int m_iPlaneCenterIndex;
int * m_pPlaneIndex; };
```

在该结构中，成员 m_p3DPtSet 是一个容纳、组织和操作三维点集对象的指针，成员 m_bNormal 用来指示以该点为中心的局部点集进行局部小平面法向量的拟合是否成功，并将拟合成功法向量存储于成员 ptNormal 的数组中。成员

（下转第 256 页）