

# 三维模型的空间分析

刘光伟

(昆明冶金高等专科学校,云南 昆明 650033)

**摘要:** 三维模型的空间分析是 3D GIS 的重要内容. 本文研究了三维模型的空间坐标查询、任意两点间的剖面绘制、空间距离量算、面积量算、坡度/坡向分析等功能,尤其对小型专题三维 GIS 的一次开发有重要参考意义.

**关键词:** 三维模型;空间分析

**中图分类号:** P208 **文献标识码:** A **文章编号:** 1007 - 855X(2003)05 - 0013 - 04

## The Spatial Analysis of 3D - Model

LIU Guang-wei

(Kunming Metallurgy College, Kunming 650033, China)

**Abstract:** The Spatial Analysis of 3D - Model is an important content of 3D GIS. Research is made on the functions about querying spatial coordinate of 3D - Model, drawing section plane between arbitrary two points, calculating space distance, calculating area, grade/slope - direction, and etc. It specially has important meaning for base development of mini - project of 3D GIS.

**Key words:** 3D - model; spatial analysis

## 0 引言

空间分析是研究物体的空间位置关系的一门科学,是对分析空间数据有关技术的统称,包括空间物体的数据特性、位置关系、分布情况以及空间形态,其目的是提取空间信息.随着计算机技术的不断发展,运用计算机技术模拟现实世界,对现实世界进行存储、分析、运用,已成为多个学科领域首要研究的方向和发展的目标.

自有地图以来,人们就始终在自觉或不自觉地进行着各种类型的空间分析.如在地图上量测地理要素之间的距离、方位、面积,乃至利用地图进行战术研究和战略决策等,都是利用地图进行空间分析的实例.

地理信息系统集成了多学科的最新技术,如关系数据库管理、高效图形算法、插值、区划和网络分析,为空间分析提供了强大的工具,使得过去复杂困难的高级空间分析任务变得简单易行.目前绝大多数地理信息系统软件都有空间分析功能.空间分析早已成为地理信息系统的核心内容之一,它特有的对地理信息提取、表现和传输功能,是地理信息系统区别于一般信息系统的主要功能特征.

空间分析赖以进行的基础是地理空间数据库,其运用的手段包括各种几何的逻辑运算、数理统计分析、代数运算等数学手段,最终的目的是解决人们所涉及到地理空间的实际问题,提取和传输地理空间信息,以实现辅助决策的目的.

## 1 三维空间坐标查询

### 1.1 可见点的判别与保存

显示在计算机二维屏幕上的 3D 数字模型,其屏幕上的像点  $(x, y)$  和 3D 模型的用户坐标(如直角大地坐标)  $(X, Y, Z)$  不是一一对应关系.因此,必须确定鼠标点捕捉到的 2D 屏幕坐标所对应的唯一可见的

收稿日期:2003 - 07 - 03.

第一作者简介:刘光伟(1962.3~),男,讲师;主要研究方向:GIS 理论与应用.

3D 大地坐标才能进行正确的空间分析和查询操作. 可见点的判别方法如下:

设  $I^2$  是欧氏平面上的整数集.  $R^3$  是三维欧氏空间上的实数集, 则对子集  $P$  (计算机屏幕空间)  $I^2$  和  $T$  (地面三维空间)  $R^3$ , 若  $P$  与  $T$  间存在映射关系  $f$ , 则表示为  $f: T \rightarrow P$ . 对于任意元素  $p \in P(I^2)$  及  $t \in T(R^3)$ , 若满足  $f(t = p)$  下, 有  $t_k(t_1, t_2, \dots, t_k), k \geq 2$ , 则  $p$  与模型上多个点  $(X, Y, Z)$  对应. 若有元素  $t_m, t_m = (X_m, Y_m, Z_m)$ , 使  $|t_m - E| = \min$ , 则  $t_m$  为多个点中唯一的可见点 ( $E$  为视点的位置). 在显示 3D 模型时对每一个 3D 点用此种方法进行判别可以确定屏幕 2D  $(x, y)$  点对应的 3D  $(X, Y, Z)$  信息. 根据 3D 点的信息可得到其对应的属性信息 (地形点或建筑物点), 从而保存该 2D 点对应的属性信息 ( $z$  坐标或建筑物的 ID 号).

## 1.2 三维空间坐标查询

在基于三维模型的可视化场景中, 最基本的空间查询就是空间点的三维坐标查询, 它是其他交互操作和空间分析的基础. 实现这一点实际上就是将计算机的二维屏幕坐标反解为三维空间坐标, 是透视投影的一个逆过程. 主要有两种方法: 正解变换和反解变换.

### 1.2.1 正解变换

所谓正解变换就是将三维空间点的坐标变换到计算机的二维屏幕坐标, 与鼠标所选择的目标点进行匹配, 从而获取目标点的三维坐标的方法. 将三维空间中的物体投影到二维计算机屏幕上, 可以通过投影变换矩阵来实现. 具体应用到实践中, 是利用 OpenGL 中的 `gluProject()` 函数将对象坐标映射到窗口坐标, 具体的表达如下:

```
int gluProject (Gdouble objx, Gdouble objy, Gdouble objz, const Gdouble modelMatrix[16], const Gdouble projMatrix[16], const Gint viewport[4], Gdouble *winx, Gdouble *winy, Gdouble *winz). 其中, objx, objy, objz 为对象坐标, modelMatrix 为当前的模型取景矩阵; projMatrix 为当前的投影矩阵; viewport 为当前的视口区; winx, winy, winz 为计算后的窗口坐标.
```

### 1.2.2 反解变换

所谓反解变换是指根据投影矩阵的逆矩阵将窗口坐标映射到对象坐标, 从而获得窗口目标点的三维坐标. 在 OpenGL 中主要通过 `gluUnProject()` 函数来实现, 具体表达为:

```
gluUnProject (Gdouble winx, Gdouble winy, Gdouble winz, const Gdouble modelMatrix[16], const Gdouble projMatrix[16], const Gint viewport[4], Gdouble *objx, Gdouble *objy, Gdouble *objz). 其中, winx, winy, winz 为窗口坐标, modelMatrix 为当前的模型取景矩阵; projMatrix 为当前的投影矩阵; viewport 为当前的视口区; objx, objy, objz 为计算后的对象坐标. 关键是 winz 的获取, 本文通过获取  $z$ -buffer 深度值, 建立选择机制. 主要的步骤如下:
```

利用函数 `glSelectBuffer()` 指定存放返回命令中记录的数组; 利用 `glRenderMode()` 设置 `GL_SELECT`, 进入选择方式; 利用 `glInitName()` 和 `glPushName()` 初始化; 定义选择所用的视图体; 交替发出绘图元命令和操作名栈的命令, 使每个关心的图元有合适的指定名称; 退出选择方式和处理返回的选择数据.

## 2 任意两点间剖面的生成

根据上述的空间点三维坐标的获取方法, 便可得到任意两点的三维坐标  $(X_1, Y_1, Z_1)$ 、 $(X_2, Y_2, Z_2)$ . 根据它们的平面位置  $(X_1, Y_1)$ 、 $(X_2, Y_2)$  和 TIN 模型存储的数据结构采用一定的算法可知道它们所在的 TIN 模型中的三角形号, 由起点和终点的三角形位置和它们的连线, 运用线段求交算法, 从而得到一系列的三维坐标点  $(X_i, Y_i, Z_i)$ . 根据所求交点的平面位置可以判断其是否位于三维模型上, 如果是, 则求出该连线与三维模型相交的交点. 剖面线的生成算法为:

```
StartPointXYZ[3]; // 起点的空间位置;
EndPointXYZ[3]; // 终点的空间位置;
// 创建剖面线直线方程;
CreateCutLine (&PointStruct, &PointStruct);
```

```

// 计算起点、终点所在的单元;
CalPointRow_Col ( &PointStruct ,&Row ,&Col );
// 计算它们所在的三角形指针 Tri1、Tri2;
CalTrianglePointer ();
// 计算剖面点的空间坐标;
// Tri 与剖面线相交的三角形指针;
// Edge 三角形的边指针;
Tri = Tri1;
For ( ;;)
{
    for (int I=0;I<3;I+ +)
    {
        Edge = Tri - > Edge[i];
        If (起终点的连线与 Edge 相交)
            CalIntersectionPoint ( &PointStruct ) ;// 计算交点坐标;
    }
}
if (交点在三维模型上)
    CalBuildingPoint ( &PointStruct ) ;// 计算位于三维模型上的交点坐标;
If (Tri == Edge - > LeftTriangle)
    Tri = Edge - > RightTriangle;
Else
    Tri = Edge - > LeftTriangle;
If (Tri == Tri2) // 相交的三角形为终点的三角形;
    Break;
// 计算结束;

```

### 3 空间距离查询和面积量算

用鼠标在三维模型上任意选取两个不同的点,其间的距离为该连线与三维模型的一系列交点  $(X_i, Y_i, Z_i)$  间的距离之和,利用 2 中的算法可以得到连线与模型的一系列交点的三维坐标,利用(1)式可以计算空间两点的距离:

$$S = \sum D_{i,i+1} \quad \left. \begin{array}{l} D_{i,i+1} = \sqrt{(X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2 + (Z_{i+1} - Z_i)^2} \end{array} \right\} \quad (1)$$

剖面切割后需要计算投影面积,即任意多边形在水平面上的面积.通常根据梯形法则来计算投影面积,如果一个多边形由顺序排列的  $N$  个点  $(X_i, Z_i, i = 1, 2, 3, 4, \dots, N)$  组成并且第  $N$  个点与第一个点相同,则按(2)式计算:

$$S = \frac{1}{2} \sum (X_i Y_{i+1} - X_{i+1} Y_i) \quad (2)$$

如果多边形顶点按顺时针方向排列,则计算的面积为负,反之为正.

### 4 方量的计算

建立 DEM 模型之后,通常会遇到象场地平整等土石方量的计算,实际上就是计算 DEM 体积. DEM 体积可由四棱柱和三棱柱的体积进行累加得到.四棱柱上表面可用抛物双曲面拟合,三棱柱表面可用斜平台

拟合,下表面为水平面或参考平面,按(3)式计算:

$$V = \left. \begin{aligned} & \frac{Z_1 + Z_2 + Z_3}{3} \cdot S_3 \\ & \frac{Z_1 + Z_2 + Z_3 + Z_4}{4} \cdot S_4 \end{aligned} \right\} \quad (3)$$

其中,  $S_3, S_4$  分别是三菱柱和四菱柱的底面积. 根据这个公式, 可以计算 DEM 的挖填方, 在对 DEM 进行挖或填后, 体积由原始的 DEM 体积减去新的 DEM 的体积求得, 如(4)式:

$$V = V_{\text{老DEM}} - V_{\text{新DEM}} \quad (4)$$

当  $V > 0$  时, 表示挖方,  $V < 0$  时表示填方,  $V = 0$  时表示既不挖方也不填方.

## 5 坡度 / 坡向分析

坡度 / 坡向是一种描述地形表面特征的重要变量. 坡度是指表示地表面在某点的倾斜程度的一个量, 它是矢量. 坡度矢量从数学上讲, 其模等于地表曲面函数在该点的切平面与水平面的夹角的正切, 其方向等于在该切平面上沿最大倾斜方向的某一矢量在水平面上的投影方向, 亦即坡向.

在军事地形分析中, 坡度 / 坡向是影响武器性能、部队机动、阵地设置的重要因素. 基于 DEM 进行的坡度 / 坡向计算方法较多, 其精度与计算效率各不相同, 在此列举一种方法:

如图 1 所示, 计算任意一点  $e$  的坡度 ( $T$ ), 可采用(5)式的算法:

$$T = \tan^{-1} \sqrt{t_1^2 + t_2^2} \quad (5)$$

其中:  $t_1 = (e_1 - e_3) / 2L$ ,  $t_2 = (e_4 - e_2) / 2L$  ( $e_i$  为格网点的高程,  $L$  为格网间距). 坡向 ( $A$ ) 的计算方法如(6)式:

$$A = \frac{T_2}{t_1} \quad (6)$$

$e_5$	$e_2$	$e_6$
$e_1$	$e$	$e_3$
$e_8$	$e_4$	$e_7$

图 1 DEM 格网 8 个邻域

## 6 结束语

本文研究了三维模型中的空间坐标查询、任意两点间的剖面绘制、空间距离量算、面积量算、坡度 / 坡向分析等五个方面最为基本的空间分析模型和算法, 已在实践中得到了其正确性的检验. 三维模型空间分析的内容远不止这些, 对于不同的三维专题 GIS, 应根据其专门应用领域的需求建立相应的各种空间分析模型, 以利于辅助决策功能的完善.

文中对显示在二维屏幕上的三维模型的空间三维坐标查询和任意方位任意倾角剖面的生成建模和算法, 有一定的难度和创新, 于技术方法上对三维专题 GIS 的开发者 (尤其是一次开发) 有重要的参考价值.

### 参考文献:

- [1] 郭仁忠. 空间分析(第二版)[M]. 北京: 高等教育出版社, 2001. 10.
- [2] 孙家广, 等. 计算机图形学(第三版)[M]. 北京: 清华大学出版社, 1998. 9.
- [3] 侨林, 费广正. OpenGL 程序设计[M]. 北京: 清华大学出版社, 2000. 4.
- [4] 周杨, 等. 基于 3 维地形图的空间分析算法[J]. 武汉大学测绘学院学报, 2001, (3): 40 ~ 43.