剖面。而且，纸质图或基于 *CAD* 的电子图件的数据维护和更新困难，很难加入新的资料。三维地质模型就是希望通过有限的资料，获取表达滑坡三维空间分布，这一过程是交互的、可控制的，采用计算机可视化技术显示，并可动态更新。

与二维图形相比，三维模型的数据量呈几何级数增长，为方便数据存贮和模型快速显示，必须考虑三维模型的数据结构。

2.2 滑坡三维地质模型数据结构

随着三维空间数据研究的深入，已经形成了各种空间数据的表达方法。根据几何特征，空间数据结构可以分为基于面表示的和基于体表示的两大类方法（*Rong Xin Li*，1995，李清泉等，1996，李青元等，2000）。基于面表示的方法是借助微小的面单元/面元素来描述对象的几何特征，基于体的表示方法则是用体信息代替面信息来描述物体内部联系。在具体描述数据结构时，可采用矢量、栅格或矢量/栅格混合的方法。三维数据结构描述形式如表 2- 1（程朋根，龚健雅，2001）。

表 2- 1 三维数据结构描述形式

几何特征	数据描述形式	
	栅　格	矢　量
基于面的表示	格网、形状模型	面片结构、边界表示、非均匀有理样条
基于体的表示	三维栅格、针状结构、八叉树	不规则四面体、结构实体几何表示

2.2.1 三维栅格数据结构

(1) 格网结构

格网结构主要用于建立数字地形模型（*DTM*）/数字高程模型（*DEM*），基于格网的方法可以很容易地产生等高线和三维格网。格网结构数据表示方法是将地形表面划为规则的网格，每个网格点对应有一个属性（高程），基本元素为点，见图 2- 1(*a*)。

(2) 形状结构

形状结构是通过对象表面点的偏导数（斜率）描述来进行的，基本元素是对象表面各单元的法向矢量，可用于对象表面的三维重建，格网点的坡度值可以用像素的灰度（明暗）变化来确定，通过坡度的变化可以求出像素之间的高程变化，最终确定 *3D* 表面，见图 2- 1(*b*)。

(3) 3*D* 栅格结构

3*D* 栅格结构是格网结构的三维扩展，将空间对象划分为立体栅格，每个栅格的元素值为 0（表示该点未被对象占有）或 1（该点被对象占有），三维栅格（体元）多采用正方单元，见图 2- 1(*c*)。

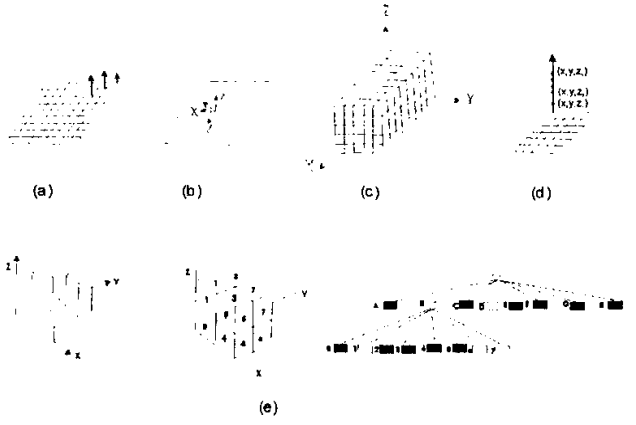


图 2- 1 栅格数据结构示意图

3D 栅格结构算法简单，能够表达对象内部的不均匀性，容易实现叠加分析、缓冲分析，但由于数据没有压缩，存贮空间浪费大，随着空间对象表达精度的提高，其数据量呈三次幂增加。

(4)针状结构

针状结构是对 *3D* 栅格结构的改进，利用行程编码技术对数据进行压缩，对每个平面位置（*X*，*Y*）在 *Z* 方向上进行行程编码，从而减少存贮空间，见图 2-1(*d*)。

(5)八叉树结构

八叉树结构是 3*D* 栅格结构的变体，是二维空间中的四叉树在三维空间的扩展，用层次式的 3*D* 空间子区域来代替大小相等、规则排列的 3*D* 栅格，见图 2-1(*e*)，这种树形结构在空间分解上具有很强的优势，它的实质是边长可对半细分的立方体填充模型。八叉树结构适用于表示体对象，特别适合于表示复杂形状的对象；内在的空间顺序使其易于显示，对布尔操作和整数特征的操作计算效率较高。八叉树结构有规则八叉树、线性八叉树等（邬伦等，2001.2，李魁星，2000.5）。

规则八叉树：直接通过八叉树结构描述空间信息，即显式地描述结点，并通过指针来实现层次（父子）关系。指针的实现可以通过建立结点索引方式实现，也可以直接利用内存指针来实现。结点的插入/删除操作可以通过直接的分配/释放内存，并修改八叉树中的指针来实现。从任意结点出发，通过指针之间的层次关系，可以遍历整个八叉树结构。规则八叉树具有直观的空间结构，其数据表示和八叉树本身完全一致，逻辑清楚、实现简单，结点增加/删除只涉及内存的分配/释放，维护简单，查询速度快，直接使用内存地址，不需比较。

线性八叉树：使用位置码来描述非空叶子结点的空间位置，每个点存贮位置码和属性值，因此，线性八叉树的插入和删除均要对该位置以后的所有节点进行移位操作。线性八叉树只保存非空叶子结点，在对象不十分复杂的情况下，可节省存贮空间，每个叶子节点结构等长，不依赖于计算机内存，可用关系型数据库来存贮和管理。