

# 基于 ArcGIS Engine 的三维地形可视化与分析系统设计

## Design of the 3D Landform Visualizing and Analyzing System Based on ArcGIS Engine

郭丽萍 (东南大学地理信息工程系 南京 210096)  
(南京军区 73603 部队 210049)

**摘要:** 在介绍组件式 GIS 技术、ArcGIS Engine 组件及 GIS 地形分析技术的基础上,以 ArcGIS Engine 为开发平台,通过系统设计、具体功能设计与开发的详细说明及代码展示,介绍了 DEM 生成、显示、浏览、地形分析的二次开发过程的关键技术,为 3D 开发提供参考。

**关键词:** 组件式 GIS 技术 ArcGIS Engine 数字高程模型(DEM) 地形分析

### 1 引言

本文主要探讨基于组件技术的 ArcGIS Engine 进行三维地形可视化和分析的开发,并介绍基于该技术的一些通用三维分析功能模块的开发实例。

组件式 GIS 是指基于组件对象平台,以一组具有某种标准通信接口,允许跨语言应用的组件提供的 GIS。GIS 组件之间以及 GIS 组件与其他组件之间,可以通过标准的通信接口实现交互,嵌入通用的开发环境(如 VB、VC、.Net 等)实现 GIS 功能及开发 GIS 应用系统<sup>[1]</sup>。采用组件式结构的地理信息系统,与传统的开发方式相比,可以降低开发难度,提高开发效率,增强系统的灵活性和开放性。

ArcGIS Engine 是美国 ERSI 公司推出的由一组核心 ArcObject 包组成并用于构建定制应用的完整 GIS 组件库,可用于建立自定义独立地理信息系统(GIS)应用程序,支持多种应用程序接口(Application Program Interfaces, APIs),拥有许多高级 GIS 功能,而且构建在工业标准基础之上<sup>[2]</sup>。

### 2 数字地形分析开发简介

数字地形分析是以数字高程数据为基础来推导二维或三维地形特征以及确定流域、集水区、水流路径等

等<sup>[3]</sup>。数字地形分析离不开 DEM 这个载体,其主要任务有两个方面,一是地形数据的基本量算,二是数据的地形特征分析。地形数据的基本量算包括如何确定点的高程、两点之间的距离和方位以及给定区域的面积、表面积和体积的计算;数据的地形特征分析则是地形特征识别及地理对象的相关关系分析。从地形分析的复杂性角度,可以将地形分析分为两大部分:一部分是基本地形因子(包括坡度、坡向、粗糙度等)的计算。另一部分是复杂的地形分析,包括可视性分析、地形特征提取、水系特征分析、道路分析等。

ArcGIS Engine 数字地形分析组件主要分布在 GeoAnalyst 类库中。GeoAnalyst 类库包含支持核心空间分析功能的对象。这些功能用在 SpatialAnalyst 和 3Danalyst 两个类库中。3Danalyst 类库包含了操作三维场景的对象,其中 Scene 对象是 3Danalyst 类库中主要对象之一<sup>[4]</sup>。ISurface 是 GeoAnalyst 类库中用于数字地形分析的主要接口。

### 3 系统总体框架设计

本系统在 Visual C#.Net 平台下,基于 ArcGIS Engine 组件库进行开发设计。空间数据采用数据库方式存储,属性、图形集中存储在 ArcSDE + SQLServer 中。

利用通用关系型数据 (SQLServe) 和空间数据管理模块 (ArcSDE), 将地理空间数据和属性数据统一无缝地集成在关系数据库中, 提供对空间数据的高效操作。

系统的总体设计见图 1。

是地形表面简单的数字表示, 描述地球表面形态多种信息空间分布的有序数值阵列<sup>[5]</sup>。数字高程模型是进行地形可视化分析的重要基础数据。

在 ArcGIS Engine 中提供了两种数据来生成地形的三维表面模型。一种是规则格网数据 (Raster), 它是基于单元的数字地形地面最一般的存放格式, 是连续地形表面模型的栅格化表示。它用一个矩阵格网来存放, 矩阵的每一格网元素存放单元的高程<sup>[6]</sup>; 另一种是不规则三角网数据 (TIN), 它将按地形特征采集的点按一定规则连接成覆盖整个区域且互不重叠的许多三角形, 是三维空间的分段线性模型。Raster 表面数据的缺点是不能准确表示地

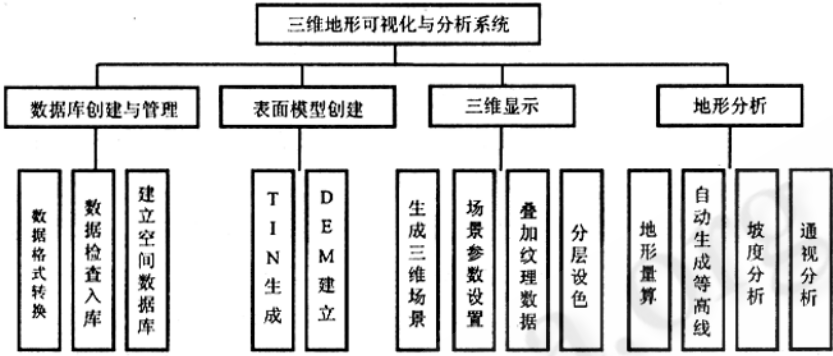
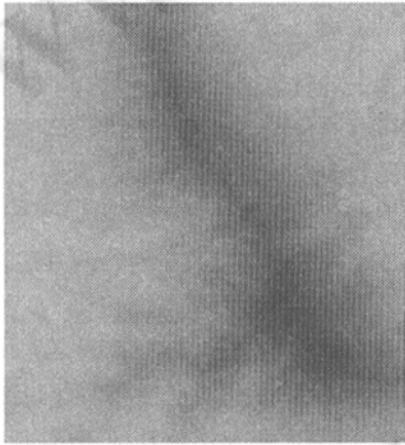


图 1 系统总体设计框架



(a) 规则格网数据生成的地形表面



(b) 不规则三角网数据生成的地形表面

图 2

### 4 系统功能开发

本系统的功能模块主要包括三维表面 (DEM) 的创建模块、三维显示模块、三维地形分析模块。下面就对这三个模块分别进行介绍。

#### 4.1 三维表面 (DEM) 的创建

要建立三维表面首先要了解数字高程模型。数字高程模型是利用一个任意坐标场中大量选择的已知 (X,Y,Z) 坐标点对连续地面的一个统计表示, 即 DEM

形的结构与细部, 并且数据量过大, 通常要进行压缩存储; 而 TIN 表面能较精确描述复杂地形, 但数据存贮方式比 Raster 复杂<sup>[7]</sup>。

从用户需求的角度上讲, 通常需要从矢量等高线数据直接内插生成相应的 DEM 数据。在 ArcEngine 组件库中的接口函数中提供了直接由线类型的等高线数据生成 TIN 的方法, 但没有提供直接由等高线数据生成 Raster 的方法, 所以只有通过间接的方法来实现: 一种方法是先将等高线数据生成 TIN, 再将 TIN 转为 Ras-

ter 数据;另一种方法是将等高线数据转为点类型数据,再利用点内插生成 Raster 数据。

以下是使用反距离加权 (IDW) 插值法得到 Raster 数据的关键语句,内插生成的 Raster 数据的图形见图 2(a)。

```
// * * * * * 通过远端数据源得到一个 Feature-
Class * * * * *
//根据实际配置情况通过接口 IPropertySet 设置 ArcS-
DE 连接参数
IWorkspace pWorkspace = pWorkspaceFactory. Open
(pPropertyset, 0);
IFeatureWorkspace pFeatureWorkspace = ( IFeature-
Workspace) pWorkspace;
IFeatureClass pFeatureClass = pFeatureWorkspace.
OpenFeatureClass( strDatabaseName);
// * * * * * 产生新的 Raster 内插操作 * * * * *
*
IInterpolationOp pInterpolationOp = new RasterInterpo-
lationOpClass();
// 通过 IFeatureClassDescriptor 获得选中的高程信息
字段 fieldName
pFeatureClassDescriptor. Create ( pFeatureClass, null,
fieldName);
// 通过 IRasterAnalysisEnvironment 设置输出网格大
小,通过 IRasterRadius 设置搜索范围 pRadius, 使用
FeatureDescriptor 作为 IinterpolationOp 的输入对象进
行内插
IGeoDataset pFCDescriptorGeo = ( IGeoDataset) pFea-
tureClassDescriptor;
IGeoDataset pOutputRaster = pInterpolationOp. IDW
(pFCDescriptorGeo, power, pRadius, ref bar);
// 通过 IRasterBandCollection 保存内插结果 Raster
IDataset pRasterDataset = pRasterBandC. SaveAs ( Ras-
tername, pRasterWorkspace, " GRID" );
```

在生成 TIN 的方法中,等高线和离散点都被看成是一种数据类型 (feature class) 来处理,因此两者生成 TIN 的过程是相同的。其实现过程主要是:在得到数据源 pFeatureClass 后,通过接口 ISpatialReference 得到 pFeatureClass 的 SpatialReference, 赋给 IEnvelope 接口对象;利用 ITINEdit 接口的 InitNew 方法创建 TIN;使用

ITINEdit 接口的 AddFromFeatureClass 方法添加高程信息到 TIN;最后保存 TIN。生成 TIN 的效果图见图 2(b)。从显示效果来看, TIN 很明显要比 Raster 的显示精度高。

## 4.2 三维显示

通过上述方法,得到了 DEM 地形表面。接下来要生成三维场景,实现三维显示。在该模块中主要设计了设置场景参数、叠加纹理数据、分层设色的功能。

通过设置三维场景的基本参数可以改变三维场景的基本属性。三维场景的基本参数设置包括垂直拉伸系数设置、背景颜色设置、范围设置、光照设置等。用户可以通过图 3 所示系统对话框对基本参数进行设定,并可以在屏幕上直接进行交互指定观察点和目标点的位置,以从不同角度和高度更直观、更逼真的观察地形。

叠加纹理数据包括叠加影像数据、图像数据和矢量数据。叠加影像数据和图像数据通常要考虑影像、图像与 DEM 表面之间的对应关系,即所谓的纹理映射。建立纹理映射要利用类库 GraphicsLayer3DClass 对象组件进行具体功能开发。具有了映射关系的纹理数据就可以通过设置基高,贴在地形表面,使生成的三维地形图既有立体感又有真实感。

分层设色法是将地貌按高度划分为若干带,每一带设置成不同的颜色以表示地貌的方法。其关键是设计一个最有表现力的“色层高度表”。通过分析表示区域的特征确定等高距、划分高程带和确定相应的颜色。以色相、饱和度或明度变换与地貌高度变化相适应的原则,以高度表为设色基础,反映地貌类型、高程的垂直分布与水平分布特征。

在 ArcGIS Engine 中,对于 TIN 和 Raster 两种不同的数据结构,需要采用不同的接口和方法实现分层设色功能。下面以 Raster 为例简单介绍一下实现步骤: 1) 从三维场景获得 IRasterLayer 接口对象。2) 通过 IRasterClassifyColorRampRenderer 接口创建 RasterClassifyColorRampRenderer 对象,同时获得它的 IRasterRender 接口对象。3) 根据高程带数和起始色及终止色创建颜色列表。并赋给 IEnumColors 接口对象。4) 将颜色赋给 SimpleFillSymbol 对象,再把 SimpleFillSymbol 赋给 RasterClassifyColorRampRenderer 对象。5) 把 RasterClassifyColorRampRenderer 对象赋给 IRasterLayer 接

口对象。6) 刷新, 完成。

### 4.3 三维地形分析

三维地形分析功能主要是由 ISurface 接口和 ISurfaceOp 实现, ISurface 接口提供了很多函数, 如夸大高程倍数的 Zfactor 函数、生成等高线的 Contour 函数、计算坡向的 GetAspectDegrees 函数、计算坡度的 GetSlopeDegrees 函数、返回通过查询点的等高线的 GetContour 函数、得到查询点高程的 GetElevation 函数、计算表面面积的 GetSurfaceArea 函数和计算体积的 GetVolume 函数等。ISurfaceOp 接口提供了控制表面(主要指 Raster 表面)操作的方法和函数, 常用的如生成等高线的 Contour 函数、计算坡度值的 Slope 函数、用于实现表面点集通视分析的 Visibility 函数。开发人员可以直接使用这些函数进行二次开发。下面是系统中一些空间分析功能的应用。

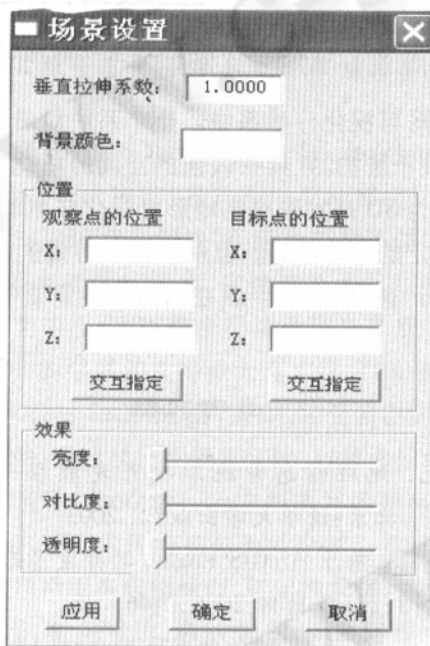


图3 三维场景基本参数设定对话框

#### 4.3.1 地形量算

坡度用来表示地表面的倾斜情况, 地面上任一点坡度定义为该点的法线与过该点的铅垂线(高程 Z 方向)的夹角。坡度在 Raster 表面和 TIN 表面中的计算是不同的。在 TIN 中, 它等于点所在三角形中高程最大变化率; 在 Raster 表面中, 它等于每个像素和相邻八个像素的高程最大变化率。坡向是指整个坡表面的朝向。坡向是以正北方向为  $0^\circ$ , 顺时针方向  $0^\circ \sim 360^\circ$  来

进行度量的。如果研究的表面是水平面, 则表示这个表面没有坡向, 设坡向值为  $-1$ 。

地形量算模块实现了三维场景任意点高程、坐标、坡度、坡向信息的计算, 基于 DEM 表面的距离计算以及某一高程基准面以上或以下面积和体积的计算。

#### 4.3.2 自动生成等高线

在 GIS 中经常需要通过 DEM 来生成该地区的等高线数据。从 DEM 内插等高线主要包括两个步骤: 首先从 DEM 跟踪等高线点, 其次是进一步插补加密等高线点以形成光滑曲线(即等高线的拟合或光滑)。根据 DEM 的数据结构, 搜索与跟踪等高线的方法又有基于规则网格的矢量法和栅格法以及基于三角网的矢量法之分。

在 ArcGIS Engine 中, 对于 TIN 和 Raster 两种不同的数据源, 需要采用不同的数据接口和方法来生成等高线。因此, 在系统中采用了重载 CreateContour 函数的方式。下面分别给出了 TIN 和 Raster 自动生成等高线的关键词句。

```
// * * * * 从 TIN 数据源自动生成等高线 * * * * *
// pLayer 是输入的 TIN 要素层对象, Interval、BaseValue
// 分别是等高线间隔和基等高线值
// pFeatureClass 是输出的 FeatureClass 对象, Contour
// 是输出等高线的高程字段名
ITinSurface pTinSurface = (ITinSurface) ((ITinLayer)
pLayer). Dataset;
pTinSurface. Contour ( BaseValue, Interval, pFeature-
Class, " Contour" , 0);
// * * * * 从 Raster 数据源自动生成等高线 * * * * *
*
// pRasterDataset 是加载的 Raster 数据集, Interval、
// BaseValue 同上
IGeoDataset pInputDataset = (IGeoDataset) pRaster-
Dataset;
ISurfaceOp pSurfaceOp = new RasterSurfaceOpClass ( );
object BASE = BaseValue;
IGeoDataset pOutput = pSurfaceOp. Contour ( pInput-
Dataset, Interval, ref BASE);
```

#### 4.3.3 坡度分析

坡度分析模块计算了整副 DEM 表面的坡度图, 以坡度图的分级显示方式直观的表示了坡度的大小。由

于 TIN 和 Raster 数据源不同,它们需要采用不同的数据接口和方法来生成坡度图。因此,在系统中同样采用了重载 CreateSlope 函数的方式。对于 Raster 数据源,在实现过程中主要利用了接口 ISurfaceOp 中的 Slope 方法。而对于 TIN 数据源,主要利用了接口 ITIN-Surface 中的方法 QueryPixelBlock 方法。坡度分析效果图见图 4。图中坡度越小越黑,坡度越大越白。

#### 4.3.4 通视分析

通视分析是指在三维空间中观察者是否发现目标的一种分析。它可分三种,第一种是两点之间的通视情况分析,即点对点的通视;第二种是已知视点,分析视点的视野问题,即点对线的通视;第三种是在某观察点上可视区域的分析,即点对区域的通视。点对点通视分析的核心是调用 ISurface 接口的 GetLineOfSight 方法。在 ArcGIS Engine 中没有直接提供对 TIN 的可视区域通视分析方法,因此对 TIN 的通视分析需要先将 TIN 转为相应的 Raster 数据。对于 Raster 的可视区域通视分析的主要步骤有:1) 使用 ISurfaceOp 创建一个 RasterSurfaceOp 对象,同时得到 RasterSurfaceOp 对象的 IRasterAnalysisEnvironment 接口对象;2) 利用 IRasterAnalysisEnvironment 接口设置输出通视分析图的网格大小;3) 利用 ISurfaceOp 的 Visibility 方法,生成通视分析图;4) 保存坡度图。可视区域分析结果见图 5,白色区域即为观察点的可见范围。



图 4 由 TIN(图 2 b)生成的坡度图

## 5 结束语

近 20 年来,真实感三维可视化技术,一直是计算机 3D 开发的前沿领域,发展迅速。而三维地形可视化

是 3D 开发的基础,因此,对地形三维可视化的研究更是三维研究中的重点。由于独立开发难度太大,本系统以 ESRI 公司的 ArcGIS Engine 组件作为开发工具,结合面向对象的可视化开发语言,高效、方便地开发出了

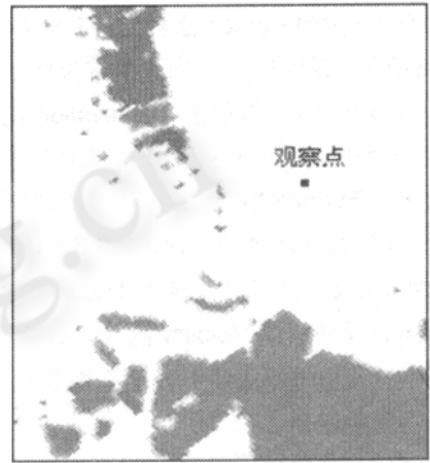


图 5 由 Raster(图 2 a)生成的通视图

三维地形可视化分析系统。整个系统可靠性好、易于移植、便于维护,并具有较强的空间分析功能,可以在实践上为 3D 开发者提高地形可视化技术和应用能力提供一定的经验基础。但系统对大容量的 DEM 数据,刷新显示速度较慢,今后在这方面还需要作进一步的研究。

## 参考文献

- 1 刘光, 地理信息系统二次开发教程——组件篇 [M], 北京:清华大学出版社, 2003.
- 2 谢小蕙, 基于 ArcGIS Engine 的开发原理和方法的探讨[J], 城市勘测, 2006, 2:46~48.
- 3 毕华兴, 基于 DEM 的数字地形分析[J], 北京林业大学学报, 2005, 27(2):49~53.
- 4 ESRI Corp. ArcGIS Engine Developer Guide
- 5 李志林, 数字高程模型 [M], 武汉:武汉大学出版社, 2000.
- 6 祝烈煌、周洞汝, 数字流域的三维可视化及基本要素提取[J], 计算机应用, 2000, 20(增刊):27~29.
- 7 韩鹏, 地理信息系统开发——ArcObjects 方法 [M], 武汉:武汉大学出版社, 2005.