

DOI:10.13203/j.whugis20160431



文章编号:1671-8860(2018)02-0241-07

一种顾及多重约束的三维地形简化方法

李胜辉¹ 史瑞芝¹ 朱 明²

1 信息工程大学地理空间信息学院,河南 郑州,450001

2 河南工程学院材料与化学工程学院,河南 郑州,451191

摘 要:三维模型在许多领域内均有着越来越深入的应用,为保证三维模型在不同存储空间、不同图形绘制能力、不同显示分辨率的终端电子设备上显示的统一性和一致性,在很多情况下需要对三维模型进行适度的简化。对于三维地形而言,纹理颜色、地形区域边界和重要顶点是其非常重要的视觉特征,本文将这三项特征作为约束因子引入到三维地形的简化过程中,并根据这三项特征对现有的二次误差测度算法和边折叠代价的计算方法进行了改进,使其在三维地形简化过程中不会出现明显退化。本文还将改进后的算法与原来算法进行了实验对比,结果表明,改进后的简化算法不仅能够不同精度上保持三维地形视觉上的一致性,而且产生的几何误差也相对较小。

关键词:移动终端;三维模型;地形简化;纹理颜色;区域边界;顶点重要度

中图法分类号:P283; P208

文献标志码:A

近年来,三维地形建模技术在地形仿真、三维导航、虚拟现实和增强现实领域得到了越来越广泛的应用^[1]。随着多终端电子设备的不断发展,如移动手机、平板电脑、可穿戴设备等,这些移动终端也对三维地形有了新的显示需求。一般来讲,三维模型越精细越能表现出地形的细节要素,显示效果越好^[2]。但不同终端设备的存储空间、图形绘制能力、显示分辨率等存在差异,尤其是移动终端的配置往往相对较低。若要在不同终端设备上表现相同的显示效果,虽然可以通过不断提升终端设备的硬件配置来实现^[3],但仍需要利用相关简化算法对三维模型进行适度简化计算,建立不同精度的细节层次 LOD 模型(levels of detail, LOD),使之在满足对三维模型精度需求的基础上,保持显示效果的统一性和一致性^[4]。

三维模型的简化需要在减少模型多边形面数的同时,尽量保持三维模型的原始形状和拓扑结构^[5]。实现三维模型简化的算法有很多,如顶点删除^[6]、小波变换^[7]、顶点聚类^[8]、边折叠等^[9]。其中,经过改进过的边折叠算法由于具有简化效果好、简化速度快等优势,成为目前非常有效的一种简化算法。许多学者对边折叠算法进行了多项

优化和改进,比如采用局部二次误差测度来衡量边折叠的代价^[10],提高了算法运行效率和简化效果,将纹理颜色考虑在内,有效保持简化模型的视觉特征等等^[11]。

对于三维地形而言,除了纹理颜色之外,地形的区域边界、重要顶点特征等均为三维地形非常重要的视觉特征^[12],所以在简化过程中,这些视觉特征也应当被考虑在内,而目前的地形简化算法却较少充分考虑这方面的问题。本文对边折叠算法的误差测度算法和折叠代价的计算方法进行了改进,综合考虑了三维地形的区域边界、纹理颜色和顶点重要度等视觉要素,并将三者作为特征约束因子应用到三维地形的简化过程当中,尽量减少视觉特征在三维地形简化过程中受到的影响。此外,本文还将改进后的算法与未考虑特征约束的边折叠算法进行了实验对比,以进一步定量评估改进算法的简化效果。

1 边折叠简化算法

边折叠简化算法是一种适用于三角形网格的简化算法,如图 1 所示。通过对边 (v_i, v_j) 进行折

收稿日期:2017-05-11

项目资助:国家自然科学基金(61301231)。

第一作者:李胜辉,博士生,主要从事三维地图交互与显示和地图制图与出版等领域的理论与方法研究。lishenghui2006@126.com

通讯作者:史瑞芝,博士,教授。ruizhishi@sina.com

叠操作,产生新的顶点 v_n ,并用 v_n 代替 v_i 和 v_j ,连接原有顶点,从而实现三角形面数的删减。在边折叠简化算法的计算过程当中,关键在于折叠边的选择和新顶点位置的确定。选择折叠边时要保证简化后的模型与原始模型的误差尽可能的小,一般是采用计算折叠代价的方式评估删除折叠边对整体三维模型所造成的影响,折叠代价越大,折叠后对模型造成的影响越大;反之,折叠后对模型的影响越小。根据折叠代价的大小可以对待折叠边进行排序,优先删除折叠代价较小的边,然后再重新计算新模型各边的折叠代价并排序,如此循环,直至模型简化至符合要求为止。

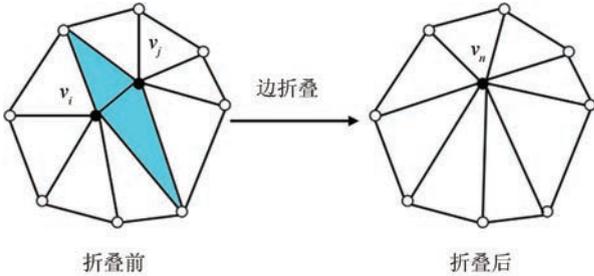


图1 边折叠

Fig.1 Edge Collapse

边折叠代价可以通过计算顶点与其所有相邻三角形的几何距离进行度量^[13],在欧式三维空间内,任何一个三维平面均可用 $\mathbf{n}^T \mathbf{v} + d = 0$ 来表示,其中 $\mathbf{n} = [a, b, c]^T$,为平面的法向量,则顶点 $\mathbf{v} = (x, y, z)$ 到该平面距离的平方 $D^2(\mathbf{v})$ 可表示为:

$$D^2(\mathbf{v}) = (\mathbf{n}^T \mathbf{v} + d)^2 = (\mathbf{n}^T \mathbf{v} + d)(\mathbf{n}^T \mathbf{v} + d) = \mathbf{v}^T (\mathbf{n} \mathbf{n}^T) \mathbf{v} + 2d \mathbf{n}^T \mathbf{v} + d^2 \quad (1)$$

从式中可以看出, $D^2(\mathbf{v})$ 明显是一个二次的多项式,所以该算法又称为二次误差测度算法,同时式(1)也可用式(2)代替表示:

$$Q(\mathbf{v}) = \mathbf{v}^T \mathbf{A} \mathbf{v} + 2\mathbf{b}^T \mathbf{v} + c \quad (2)$$

其中, $\mathbf{Q} = (\mathbf{A}, \mathbf{b}, c) = (\mathbf{n} \mathbf{n}^T, d \mathbf{n}, d^2)$ 。

一般情况下,顶点 \mathbf{v} 会与多个三角形面相连,顶点 \mathbf{v} 到这些三角形面的距离平方均可采用式(2)的方法进行计算,然后将这些距离的平方相加,这样就可以得到三维模型中每个顶点的二次误差测度。当边 $(v_i, v_j) \rightarrow v_n$ 进行折叠的时候,边 (v_i, v_j) 的二次误差测度采用 $Q = Q_i + Q_j$ 计算即可,边 (v_i, v_j) 的折叠代价则可以用新顶点 v_n 处产生的误差来计算,即 $Q(v_n) = Q_i(v_n) + Q_j(v_n)$ 。通过计算每条边的折叠代价可以建立一个升序序列,优先删除序列中折叠代价较小的边,从而使每次简化都能保证简化模型与原始模

型的几何差异尽可能的小。

在计算边折叠代价之前,折叠后新顶点的坐标位置也需要进行确定。一般有两种新顶点位置的确定方法,一种是子集位置确定,另一种是最优位置确定。前者采用 v_i 和 v_j 中折叠代价较小的顶点来作为新的顶点,这种方法运算简单,速度较快,但简化后的模型与原始模型的几何差异会比较明显。所以,本文选择最优位置确定方法来计算新顶点的坐标位置。最优位置确定方法理论上也比较简单,通过对边 (v_i, v_j) 的二次误差测度 Q 在 x, y, z 方向上求偏导,即 $\partial Q / \partial x = \partial Q / \partial y = \partial Q / \partial z = 0$,求解这个三元一次方程组,即可得到新顶点的坐标位置 (x_n, y_n, z_n) 。由于 Q 在点 (x_n, y_n, z_n) 处是极小值,所以折叠代价也最小,因此通过这种方法计算得到的新顶点能够较大程度上减少简化模型与原始模型之间的几何差异。

2 特征约束因子

三维地形除了几何特征之外,还会有许多其他属性,比如区域边界、颜色纹理、顶点重要度等,这些特征属性在模型简化过程中也是需要特别处理和保留的。在前文的模型简化过程中,仅以顶点与相邻平面的距离来计算折叠代价,进而决定边的折叠顺序,一些重要的属性特征并没有得到特别处理和保留,所以在模型简化过程中很容易造成这些重要特征的退化。因此,本文结合三维地形的特点,介绍3种特征约束因子,并将其结合到折叠代价的计算过程当中,用以保留三维地形的一些重要特征。

2.1 区域边界约束

三维地形的边界是非常重要的特征,在三维地形的简化处理过程中需要进行特别保留。如图2所示,通过对三维地形模型中的所有边进行标记,将边分别标记为内边、一般边和边界边,以区分不同边的类型。内边是指一条边的两个端点都不在边界上,一般边指一条边的两个端点有一个在边界上,边界边则两个端点均在边界上,任何一条边 (v_i, v_j) 都属于这三者之一^[14]。通过调整3种类型边在计算过程中的权重,可以使边界边的折叠代价最大,内边的折叠代价最小,从而延迟对边界边的简化,使其得到较好的保留。由于三维地形的立体特征,若要保持地形整体结构在简化过程中的一致性,不仅需要保留区域边界,还需要保证地形的整体结构在简化过程中保持一致,为了解决这个问题,本文引入了顶点重要度这项指

标,具体在 § 2.3 进行阐述。

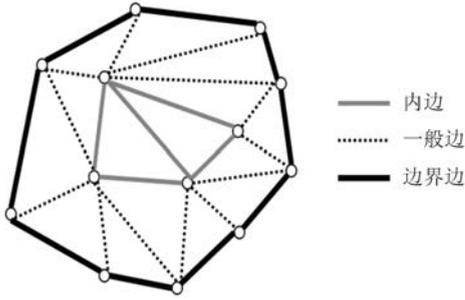


图 2 3 种类型的边

Fig.2 Three Types of Edge

2.2 纹理颜色约束

除了几何特征之外,三维地形模型为了增强地形场景的真实感,会将纹理影像贴附在模型的表面,从而使三维模型在视觉上与真实地形更加接近^[15]。在三维地形的简化过程中,如果不把纹理颜色信息考虑在内,仅仅根据几何特征进行计算,会使简化后的模型与原始模型的纹理颜色存在较大差异,影响三维地形的视觉一致性。

对于具有纹理的三维地形模型,每个顶点除了空间坐标 $V_{\text{position}} = (x, y, z)$ 之外,还需要增加纹理贴图的坐标 $V_{\text{texture}} = (u, v)$ 。这样对于三维地形模型中的每个顶点而言,原来是用 3 个坐标值进行表示,增加纹理之后,就需要采用 5 个值进行表示,即顶点 $V_{\text{vertex}} = (V_{\text{position}}, V_{\text{texture}}) = (x, y, z, u, v)$,也可以看做将每个顶点的表示从原来的三维扩展到了五维,前 3 个值表示代表空间坐标,后两个值代表纹理坐标。

2.3 顶点重要度约束

顶点重要度是衡量三维模型表面凹凸特性的一项指标,一般来讲,顶点越凸起,则该顶点越重要,在模型简化过程中越应当进行保留。反映在三维地形上,这些重要顶点主要分布在山顶、山脊、陡坡等区域,这些区域由于角度变化比较明显,与其他区域相比,对于保持地形的整体结构和轮廓更为重要。目前计算顶点重要性的方法有很多,比如高斯曲率、体积近似估计、平均法向量等等^[16]。相比较而言,计算平均法向量的方法对于三维模型重要顶点的保持具有较好的效果^[17]。

如图 3 所示,平均法向量是计算顶点 V_0 所有相邻平面法向量的平均值来衡量顶点的重要性。式(3)为平均法向量的计算公式, m 为相邻平面的个数。如果 V_0 相邻平面的法向量与 V_0 方向向量的方向越接近,顶点 V_0 所处的位置就越平坦,该顶点越不重要,可以被优先删除。反之,如果相

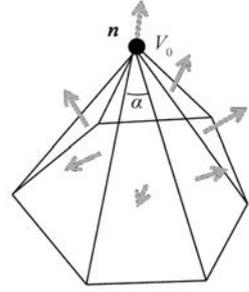


图 3 平均法向量

Fig.3 Average Normal Vector

邻三角形面法向量与顶点 V_0 的方向差别越大,顶点 V_0 所处的位置就越凸起,该顶点就越重要,此时对 V_0 的删除也应当做延后处理。

$$N = \frac{\sum_{i=1}^m n_i}{m} \tag{3}$$

此外,实验表明,如果将顶点 V_0 的方向向量与相邻三角形面的夹角作为权值,加入到平均法向量的计算过程中,会对三维模型顶点特征的保持具有较好的运算效果^[18]。图 3 中的 α 为顶点 V_0 与相邻三角形的夹角,式(4)为改进后的平均法向量计算公式:

$$N = \frac{\sum_{i=1}^m \alpha_i n_i}{\sum_{i=1}^m \alpha_i} \tag{4}$$

3 顾及多重约束的三维地形的简化算法

现有的二次误差测度算法是通过计算原始模型与简化模型之间的几何误差来进行简化的,并没有涉及到其他属性特征的计算。对于三维地形而言,如果将区域边界、纹理颜色、顶点重要性等约束因子考虑在的话,每个顶点不仅要有 3 个空间坐标值描述其空间位置,还应该有的属性值来描述这些约束因子。

在各待折叠边进行数据初始化的时候,通过对各个边进行标记,分别标记为内边、一般边和边界边,能够完成区域边界特征的描述。纹理坐标可以用 $V_{\text{texture}} = (u, v)$ 进行表示,顶点特征约束主要是通过法向量进行计算,所以各个顶点的法向量可以用 $V_{\text{normal}} = (a, b, c)$ 进行表示。如果用八维向量描述模型中的每个顶点,则对于每个顶点 V ,前 3 个值描述空间坐标,其他值用来描述顶点的纹理坐标和法向量,如式(5)所示。

$$\begin{aligned} V &= (V_{\text{position}}, V_{\text{texture}}, V_{\text{normal}}) \\ &= (x, y, z, u, v, a, b, c) \end{aligned} \quad (5)$$

三维地形中的每个顶点都可以用式(5)中的8个值描述其空间坐标、纹理坐标和法向量,然后根据这8个值应用现有的折叠代价计算框架进行计算。不过,目前的二次误差测度计算公式实际上是根据每个顶点的空间坐标值计算距离,没有涉及到属性值的误差计算,所以需要对该计算方法进行相应的改进。

对于三维地形中任意一个三角形面 $T=(p, q, r)$, 其中 p, q, r 为三角形面的3个顶点, 每个顶点都有8个值, 即顶点 $p=(p_x, p_y, p_z, p_u, p_v, p_a, p_b, p_c)$ 。所以相当于用3个八维的点来定义了一个二维的平面, 此时的二次误差测度应该是计算顶点到这个3个八维点所定义二维平面的距离。如图4所示, 令 e_1, e_2 为三角形面 T 平面上两个正交的单位向量, p 为原点, 则顶点 v 到平面 T 的距离就等于顶点 v 到正交单位向量 e_1 和 e_2 所定义平面的距离。

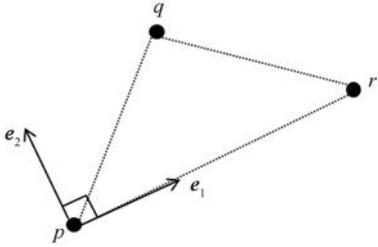


图4 正交单位向量 e_1 和 e_2

Fig.4 Orthonormal Vectors e_1 and e_2

由于 e_1, e_2 为三角形面 $T=(p, q, r)$ 上两个正交的单位向量, 所以 e_1, e_2 可以用顶点 p, q, r 进行表示, 如式(6)、式(7)所示。由于三维模型中的每个顶点都有8个值进行描述, 所以可以看做把顶点放在一个原点为 p 的八维空间内, 每个点都有八个互相正交的单位向量进行描述, 即 $e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8$, 这8个向量均为单位向量且互相正交。

$$e_1 = \frac{q-p}{\|q-p\|} \quad (6)$$

$$e_2 = \frac{r-p - (e_1 \cdot (r-p)) e_1}{\|r-p - (e_1 \cdot (r-p)) e_1\|} \quad (7)$$

根据目前简化算法原理, 需要计算顶点到相邻三角形面距离的平方。同理, 对于八维空间中的任意一点 v , 需要计算该点到相邻三角形面 $T=(p, q, r)$ 距离的平方 $D^2(v)$ 。令 $u=p-v$, p 为坐标原点, 则 u 长度的平方可表示:

$$\|u\|^2 = (u \cdot e_1)^2 + (u \cdot e_2)^2 + \dots + (u \cdot e_8)^2 \quad (8)$$

式(8)可变形为:

$$\begin{aligned} \|u\|^2 - (u \cdot e_1)^2 - (u \cdot e_2)^2 \\ = (u \cdot e_3)^2 + (u \cdot e_4)^2 + \dots + (u \cdot e_8)^2 \end{aligned} \quad (9)$$

由于 p 为坐标原点, 从式(9)可以明显看出, 式(9)的等号右边为顶点 v 到单位向量 e_1, e_2 所定义平面距离的平方, 换言之, 式(9)等号右边即为顶点 v 到三角形面 $T=(p, q, r)$ 距离的平方。所以对于三维模型中的任一顶点 v , 它到其相邻三角形面距离的平方可以用式(10)表示:

$$\begin{aligned} D^2(v) &= \|u\|^2 - (u \cdot e_1)^2 - (u \cdot e_2)^2 \\ &= u^T u - (u^T \cdot e_1)(e_1^T \cdot u) - (u^T \cdot e_2)(e_2^T \cdot u) \end{aligned} \quad (10)$$

将 $u=p-v$ 代入式(10), 可得:

$$\begin{aligned} D^2(v) &= v^T v - 2p^T v + p \cdot p \\ &\quad - v^T (e_1 e_1^T) v + 2(p \cdot e_1) e_1^T v - (p \cdot e_1)^2 \\ &\quad - v^T (e_2 e_2^T) v + 2(p \cdot e_2) e_2^T v - (p \cdot e_2)^2 \end{aligned} \quad (11)$$

式(11)可表示为二次误差测度公式的形式, 即 $Q(v) = v^T A v + 2b^T v + c$, 其中 A, b, c 分别如式(12)所示。经过改进后的二次误差度量公式, 将纹理、顶点重要性约束因子考虑在内, 使折叠代价的计算不仅仅依据几何误差, 还加入纹理和法向量的误差计算:

$$\begin{cases} A = 1 - e_1 \cdot e_1^T - e_2 \cdot e_2^T \\ b = (p \cdot e_1) e_1 + (p \cdot e_2) e_2 - p \\ c = p \cdot p - (p \cdot e_1)^2 - (p \cdot e_2)^2 \end{cases} \quad (12)$$

4 实验验证

4.1 实验步骤

改进后的算法综合考虑了区域边界、颜色纹理、顶点重要性3项特征, 其运算过程主要包括以下6个步骤。

1) 三维地形数据初始化, 根据每条边两个端点的情况, 将每条边标记为内边, 一般边和边界边;

2) 利用改进后的二次误差测度算法计算每条边的折叠代价;

3) 根据每条边的类型, 增加边界边的权值, 然后计算加权后的折叠代价;

4) 根据折叠代价大小进行排序, 建立一个升序序列, 并存放于堆中;

5) 选取折叠代价最小的边进行折叠, 并利用

1)~4)步计算与折叠边端点相关所有边的折叠代价,然后更新折叠代价的排序;

6)重复执行 1)~5)步,直至简化至符合要求为止。

为了定量评估改进算法对三维地形一致性的保持效果,本文采用三维模型误差计算方法对简化模型和原始模型的误差进行估计^[19]。三维地形的误差估计主要依据两个指标,一是最大误差,二是平均误差^[20]。最大误差的算法如式(13)和式(14)所示,式(13)中的 $d(v, v')$ 为三维空间内两点间的欧式距离, $e(v, S)$ 为简化后模型中的一点到原始模型 S 中所有点距离的最小值,式(14)中的 $E(S_1, S_2)$ 表示简化模型 S_1 与原始模型 S_2 的最大误差,在实际计算过程中 $E(S_1, S_2)$ 有时与 $E(S_2, S_1)$ 并不相等,本实验取两者较大值作为三维地形简化前后的最大误差,平均误差的计

算如式(15)所示:

$$e(v, S) = \min_{v' \in S} d(v, v') \quad (13)$$

$$E(S_1, S_2) = \max_{v \in S_1} e(v, S_2) \quad (14)$$

$$E_m(S_1, S_2) = \frac{1}{|S_1|} \int_{S_1} e(v, S_2) ds \quad (15)$$

4.2 实验环境及实验结果

本文算法所运行的硬件环境为 i5-2310, 3.00 GHz 处理器,显卡为 NVIDIA Geforce GT420,内存为 3 G 的计算机,系统环境为 Windows 7 Professional,软件开发工具为 Microsoft Visual Studio 2010 和 OpenGL。简化前的三维地形模型共有 30 868 个面,15 718 个顶点和 566 个边界边。图 5 为根据几何距离进行边折叠简化的原来算法的简化效果,图 6 为考虑边界特征、纹理颜色、顶点重要性等多重特征约束的改进后的三维地形简化效果。

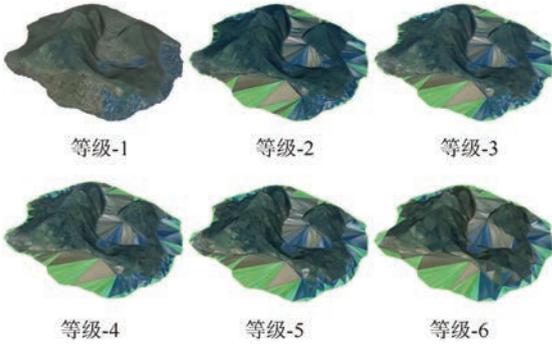


图 5 原来算法简化效果

Fig.5 Simplification Result by the Original Algorithm

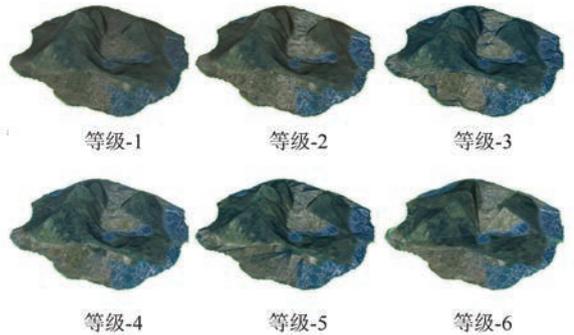


图 6 改进算法简化效果

Fig.6 Simplification Result by the Improved Algorithm

从图 5 和图 6 可以看出,在将三维地形简化至 20 000 面以下时,仅根据几何距离进行简化的三维地形就出现了纹理颜色错乱,部分边界特征退化的现象,而顾及多重特征约束的简化算法对三维地形的边界特征、纹理颜色和重要顶点均有较好的保持效果,在不同的精度上基本可以保持与原始三维模型在视觉上的一致性。表 1 为简化后的三维地形各项数据和误差,从渲染耗时来看,改进算法与原来算法的时间消耗差别不大,但改进算法能够更好保持三维地形在简化过程中视觉

感受上的一致性。同时,简化等级越高的地形渲染耗时越少,说明地形简化算法对提高三维地形的绘制效率具有非常重要的意义。另外,由于改进算法对边界边进行了加权处理,所以边界边在简化过程中没有受到损失。简化后的三维地形误差可以采用 Hausdorff 距离度量,从表 1 最大误差和平均误差的对比结果可以看出,与未考虑特征约束的边折叠算法相比,对于同一简化水平的三维地形,改进后的三维地形简化算法所产生的最大误差和平均误差均相对较小。

表 1 简化数据对比

Tab.1 Contrast of the Simplification Data

简化等级	三角形面数/个		顶点数/个		边界边数/个		渲染耗时/ms		最大误差/m		平均误差/m	
	原来算法	改进算法	原来算法	改进算法	原来算法	改进算法	原来算法	改进算法	原来算法	改进算法	原来算法	改进算法
等级-1	24 693	24 692	12 625	12 630	555	566	673	672	0.096	0.051	0.021	0.013
等级-2	12 339	12 338	6 295	6 453	251	566	426	431	0.913	0.617	0.210	0.091
等级-3	6 171	6 170	3 178	3 369	187	566	264	273	3.248	1.783	1.012	0.483
等级-4	3 085	3 084	1 604	1 826	124	566	191	199	5.328	3.611	2.093	1.007
等级-5	1 541	1 540	812	1 054	83	566	112	117	7.715	4.322	3.132	1.909
等级-6	769	768	413	668	55	566	77	83	9.127	6.315	5.012	2.368

5 结 语

对于三维地形简化而言,地形的边界特征、纹理颜色和重要顶点是其非常重要的属性特征,本文将这些特征作为约束因子引入到三维地形的简化过程中,并根据这些约束因子的特点对边折叠算法中二次误差测度和折叠代价的计算方法进行了改进,实验结果表明:经过改进后的三维地形简化算法对地形的边界特征、纹理颜色和重要顶点具有较好的保持效果,而且简化产生的几何误差也相对较小。

不过,由于顾及多重约束的简化方法是对角度变化比较明显的区域进行了特别的处理和保留,这些区域的保留效果也比较好。但对于角度变化相对平缓的区域,如部分鞍部、缓坡等区域会进行较大程度的舍弃。因此,如何在简化过程中平衡这些问题仍然需要进一步的实验和研究。

参 考 文 献

- [1] Wang Baoshan. Study on the Coal Mine Virtual Reality System Three Dimensional Data Model And Virtual Technology and Arithmetic[D]. Zhengzhou: Information Engineering University, 2006(王宝山, 煤矿虚拟现实系统三维数据模型和可视化技术与算法研究[D]. 郑州: 信息工程大学, 2006)
- [2] Wang Guangxia, Zhu Changqing. LOD Modeling and Accuracy Analysis of Terrain Information[J]. *Acta Geodaetica et Cartographica Sinica*, 2005, 34(3):228-232(王光霞, 朱长青. 地形信息的 LOD 建模及精度分析[J]. 测绘学报, 2005, 34(3):228-232)
- [3] Wang Hongwei. The Study and Realization of Global Terrain Rendering[D]. Zhengzhou: Information Engineering University, 2007(王洪伟, 全球三维地形显示研究与实现[D]. 郑州: 信息工程大学, 2007)
- [4] Liu Xiao, Liu Zhen, Mei Xiangdong. Rendering Optimization Based on Real-time LOD Simplification[J]. *Journal of Hunan University of Science & Technology (Natural Science Edition)*, 2015, 30(4):92-96(刘晓, 刘镇, 梅向东. 基于实时 LOD 简化绘制的渲染优化方法[J]. 湖南科技大学学报(自然科学版), 2015, 30(4):92-96)
- [5] He Zhengwei. Spatio-Temporal Data Modeling and Visualization for Dynamic Phenomenon[D]. Wuhan: Wuhan University, 2009(何正伟, 动态现象的时空数据建模与多维可视化[D]. 武汉: 武汉大学, 2009)
- [6] Luo Kun, Huang Kuidong, Lian Mingming. New Method of Triangular Mesh Simplification Based on Vertex Culling[J]. *Microelectronics & Computer*, 2009, 26(5):142-148(罗鸱, 黄魁东, 连明明. 基于顶点删除的三角网格模型简化新方法[J]. 微电子与计算机, 2009, 26(5):142-148)
- [7] Xu Xiaoli. Real-Time Rendering of Dynamic Terrain By Wavelet Transform[D]. Qingdao: China University of Petroleum, 2008(徐晓莉, 基于小波变换的动态地形的实时绘制[D]. 青岛: 中国石油大学, 2008)
- [8] Chen J, Li M, Li J. An Improved Texture-related Vertex Clustering Algorithm for Model Simplification[J]. *Computers & Geosciences*, 2015, 83:37-45
- [9] Yin Zhihui, Li Chaokui, Wu Baiyan, et al. Simplification Method for LOD Model with the Visual Features Preserved[J]. *Geomatics and Information Science of Wuhan University*, 2014, 39(7):772-776(殷智慧, 李朝奎, 吴柏燕, 等. 保持视觉特征的 LOD 模型简化算法研究[J]. 武汉大学学报·信息科学版, 2014, 39(7):772-776)
- [10] Dang J W, Liu Y W, Wang Y P, et al. Quadradric Error Metric Mesh Simplification Algorithm Based on Discrete Curvature[J]. *Journal of Computer Applications*, 2011, 31(4):1 010-1 012
- [11] Lu Wei, Zeng Dinghao, Pan Jingui. Mesh Simplification for 3D Models with Feature-Preserving[J]. *Journal of Software*, 2009, 20(3):713-723(卢威, 曾定浩, 潘金贵. 支持外观属性保持的三维网格模型简化[J]. 软件学报, 2009, 20(3):713-723)
- [12] Teng Shujuan. LOD Model Simplification Method Based on Preserved Vision Feature[J]. *Computer Engineering and Applications*, 2010, 46(33):164-168(腾书娟. 一种保持视觉特征的 LOD 模型简化算法[J]. 计算机工程与应用, 2010, 46(33):164-168)
- [13] Han K C, Kim H S, Lee K H. An Improved Mesh Simplification Method Using Additional Attributes with Optimal Positioning[J]. *International Journal of Advanced Manufacturing Technology*, 2010, 50(1-4):235-252
- [14] Hussain M, Okada Y, Niijima K. Feature-Preserving and Memory-Efficient Simplification of Polygonal Meshes[J]. *International Journal of Modeling & Simulation*, 2005, 25(4):1 703-1 712
- [15] Chen Xiaozheng, Li Zhenxiao. A Territory Texture Mapping Algorithm Based on Area Ratio[J]. *Computer Simulation*, 2007, 24(8):174-177(陈晓铮, 李震霄. 一种基于表面积比例的地形纹理贴图算法

- [J]. 计算机仿真, 2007, 24(8):174-177)
- [16] Hoppe H, Deroose T, Duchamp T, et al. Surface Reconstruction from Unorganized Points[J]. *Acm Siggraph Computer Graphics*, 1999, 26(2):71-78
- [17] Liu Xiaoli, Liu Zeyi, Gao Pengdong, et al. Edge Collapse Simplification Based on Sharp Degree[J]. *Journal of Software*, 2005, 16(5):669-675 (刘晓利, 刘则毅, 高鹏懂, 等. 基于尖特征度的边折叠简化算法[J]. 软件学报, 2005, 16(5):669-675)
- [18] Wang Fang, Cong Wenjing, Zhu Haitao. Research of Simplification Algorithm for Modeling Based on Normal Important Degree of Triangular facets[J]. *Computer & Digital Engineering*, 2011, 39(7):6-8 (王芳, 丛文静, 祝海涛. 基于顶点法向量重要度的模型简化算法研究[J]. 计算机与数字工程, 2011, 39(7):6-8)
- [19] Loop C, Schaefer S, Ni T. Approximating Subdivision Surfaces with Gregory Patches for Hardware Tessellation[J]. *Acm Transactions on Graphics*, 2009, 28(5):1-9
- [20] Franc M, Skala V. Mesh Simplification with Respect to a Model Appearance [C]. Proceedings Spring Conference on Computer Graphics, Slovakia, Bratislava, 2006

Simplification Approach for 3D Terrain with Multi-constraints Consideration

LI Shenghui¹ SHI Ruizhi¹ ZHU Ming²

1 Institute of Geographic Space Information, Information Engineering University, Zhengzhou 450001, China

2 Department of Material and Chemical Engineering, Henan University of Engineering, Zhengzhou 451191, China

Abstract: Many application areas require complex 3D models. In order to ensure the uniformity and consistency of 3D models across different electronic devices but in many cases, 3D models must be simplified. In 3D terrains, textures, colors, regions, boundaries, and vertices are very important visual features. In order to preserve the visual characteristics of the 3D terrain, these visual features are introduced as constraint factors in the simplification process of 3D terrains. Considering the texture color, region boundary and vertex importance, a new simplified scheme is proposed based on the quadric error metric to preserve these features during the simplification process. The improved simplification scheme is compared with traditional algorithm in an experiment. The results show that the improved simplification scheme can decrease geometric error while preserving visual features.

Key words: mobile terminal; 3D model; terrain simplification; texture color; region boundary; vertex importance

First author: LI Shenghui, PhD candidate, specializes in the theories and methods of 3D map display, cartography and map publication. E-mail: lishenghui2006@126.com

Corresponding author: SHI Ruizhi, PhD, professor. E-mail: ruizhishi@sina.com

Foundation support: The National Natural Science Foundation of China, No. 61301231.