文章编号:1671-6833(2006)02-0084-04

基于二次误差测度的递进网格色彩模型生成算法

谭同德,郝齐辉,陈争艳,赵红领,李润知

(郑州大学信息工程学院,河南 郑州 450001)

摘 要:现有的递进网格生成算法不仅效率低,而且大多只能完成几何特征的简化,没有考虑网格模型的其它表面属性.针对这些问题,提出一种新算法.该算法把基于点到一组平面距离平方和的二次误差测度,从三维几何空间推广到包含属性信息的多维空间.这种二次误差测度表示了简化网格与初始网格的几何特征和属性信息的匹配程度,利用二次误差值最小原理,指导网格简化操作的进行.实验结果表明,该算法不仅效率高,而且可以保证简化模型同初始模型在几何特征和颜色信息上具有高相似度.

关键词:二次误差测度;递进网格;边折叠;细节层次

中图分类号:TP 391 **文献标识码**:A

0 引言

在计算机图形学中,经常采用多边形网格(通常是三角形网格)描述物体.由表面重构技术生成的多边形网格中的多边形数目非常多.在很多情况下,我们并不需要对物体的细节刻画得很详细的模型.例如,当观察者离物体较远或物体在运动时,我们只看到物体的大概轮廓,用三角形数目较少的粗糙网格表示即可;只有当观察者离物体较近或物体静止时,我们才需要一个比较精细的网格模型.

传统的网格的表示方法有几个缺点:首先,数据量庞大,不易于存储、传输和绘制,尤其在交互式可视化和虚拟现实中,要求图形生成具有"实时性",过于庞大的网格显然是不适合的;其次,这种表示法不支持递进传输和有选择精化;再次,用这种表示法不易于建立光滑的多细节层次的近似模型.

为了解决上述问题,H. Hoppe 提出了基于能量优化的 PM 生成方法[1].该算法的基本思想是:以边折叠作为网格简化的基本操作,最初采用全局能量最小化方法来计算折叠所生成的新点坐标,并记录下该折叠操作,最后形成一组网格简化信息的序列.然后通过一系列顶点分裂操作就可以动态地恢复成原始网格.这种方法的主要缺点是:由于要建立和求解复杂的全局能量优化方程,因而计算复杂性高,很难达到实时.此外,目前国内外许多科研人员对递进网格模型进行了研究,提出了一些相关算法[2-6].但这些算法大多只支持具有几何信息的网格模型,而对于包含颜色、纹理等其它信息的网格模型,尚没有较好的算法.

基于上述讨论,本文作者在深人研究了前人的工作以后,提出并实现了一种递进网格模型的生成算法.该算法仍使用基于边折叠的网格简化方法,但在选择折叠操作所生成的新点坐标时,不是用能量优化方法,而是使用扩展二次误差测度方法[7].同大多数以往算法相比,本文的算法不仅更快速,而且可生成带有颜色信息的递进网格模型.

1 算法描述

1.1 递进网格的生成

假设原网格为 M_n,每次从 M_n 中找出一条边进行折叠(edge collapse),经过 n 次折叠后,得到一个相对粗糙的简化网格 M_0,其中 m_0 为网格 M_0 的顶点数目.边折叠的逆过程是点分裂(vertex split),如图1所示.在 M_0 的基础上逐步的把点分裂的信息添加上去,就能恢复原网格 M_n.

$$M_n \xrightarrow{\text{edgecol}_{n-1}} M_{n-1} \xrightarrow{\text{edgecol}_{n-2}} \cdots M_1 \xrightarrow{\text{edgecol}_n} M_0$$

$$M_0 \xrightarrow{\text{vsplit}_0} M_1 \xrightarrow{\text{vsplit}_1} \cdots M_{n-1} \xrightarrow{\text{vsplit}_{n-1}} M_n$$

收稿日期:2006-01-22;修订日期:2005-03-15

基金项目:河南省科技攻关项目(0324210045)

作者简介:谭同德(1951-),男,上海人,郑州大学教授,博士,主要从事计算机辅助设计、图形图像处理以及虚拟现实.

由于边折叠操作是一个可逆过程,我们可以把它的对偶操作—顶点分裂表示成五元组 *vsplit* $(s, l, r, cv_s, cv_{m0+i})$,其中 cv_s 和 cv_{m0+i} 分别是 M_i 的后续网格 M_{i+1} 的 s 点和新顶点 m_0+i 的信息值.这样,$(M_0, \{\mathrm{vsplit0}, \mathrm{vsplit1}, \cdots, \mathrm{vsplit}_{n-1}\})$ 就构成了递进网格.

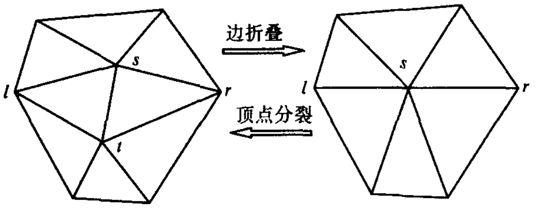


图 1　边折叠与顶点分裂操作

Fig.1　The operation of edge collapse and vertex split

1.2　边折叠操作

边折叠操作需要解决两个关键问题:一是如何选择网格中的一条边进行折叠;二是如何确定边折叠后新顶点的位置.解决上述两个问题的原则是:折叠的边及其折叠后新顶点位置的选取应使化简后的网格与原网格的误差尽可能小.

笔者采用扩展二次误差测度作为本文算法的误差度量标准,来解决以上问题.扩展二次误差测度建立在原始二次误差测度的基础之上,也是采用点到一组平面的距离平方和作为误差测度.它继承了原始二次误差测度具有较高的计算速度,较小的内存消耗,得到的简化网格具有较高质量的优点.同时,它又把二次误差测度从三维欧氏空间推广到多维欧氏空间,可以使任意多维属性的处理在一个统一的框架下进行.

1.2.1　*扩展二次误差测度的公式推导*

扩展二次误差测度把顶点信息的表示由一个三维的几何向量,扩展为包含几何和属性信息的多维向量;把基于点到一组平面距离平方和的二次误差测度从三维欧氏空间推广到多维欧氏空间.下面以能够处理具有颜色属性的扩展二次误差测度为例,加以说明公式推导过程:

由于顶点的空间位置坐标值的范围和颜色分量值的范围不同,为了能够使这两种属性对简化误差的评估起到等价的作用,在网格数据初始化时,首先要将网格模型的几何空间坐标和颜色分量值归一化到(0,1)之间.另外,算法中处理的是顶点的颜色,而实际图形文件中的数据多是三角面片的颜色.实现时对多颜色的顶点,将该顶点在各个面的颜色取平均值后,再赋给该顶点,实践证明这种方法在处理模型色彩均匀变化的部分时效果尤其不错.

现在所用的顶点 $\boldsymbol{v} = [x\ y\ z\ r\ g\ b]^{\mathrm{T}}$ 是一个六元组,是多元空间中的一个点,对于多元空间中的点到其平面的距离,做如下考虑:在多元空间 $\mathbf{R}^n$(在色彩模型的情况下,$n=6$)中有一个三角形 $\boldsymbol{T} = (\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{r})$,设 $\boldsymbol{p} = [p_x\ p_y\ p_z\ p_r\ p_g\ p_b]^{\mathrm{T}}$,$\boldsymbol{q} = [q_x\ q_y\ q_z\ q_r\ q_g\ q_b]^{\mathrm{T}}$,$\boldsymbol{r} = [r_x\ r_y\ r_z\ r_r\ r_g\ r_b]^{\mathrm{T}}$.这样,三角形 $\boldsymbol{T} = (\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{r})$ 所在的平面,可以用两个相互垂直的向量和一个点来表示,如图 2 所示.

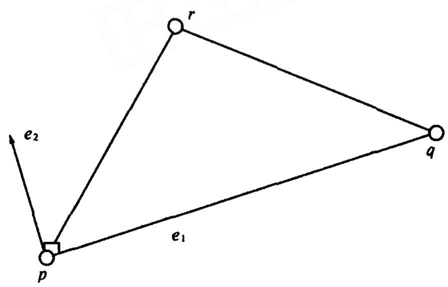


图 2　三角形 T 的表现形式

Fig.2　A manifestation of the triangle T

$$\boldsymbol{e}_1 = (\boldsymbol{q}-\boldsymbol{p}) / \|\boldsymbol{q}-\boldsymbol{p}\|;$$

$$\boldsymbol{e}_2 = ((\boldsymbol{r}-\boldsymbol{p}) - \boldsymbol{e}_1\cdot(\boldsymbol{r}-\boldsymbol{p})\boldsymbol{e}_1) / \|(\boldsymbol{r}-\boldsymbol{p}) - \boldsymbol{e}_1\cdot(\boldsymbol{r}-\boldsymbol{p})\boldsymbol{e}_1\|.$$

理论上,在多元空间 $\mathbf{R}^n$ 中,可以分别求出 $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3, \cdots, \boldsymbol{e}_n$,它们是 n 元空间中相互垂直的单位矢量.

现在求 $\mathbf{R}^n$ 空间中任意一点 $\boldsymbol{v}$ 到平面 $\boldsymbol{T}$ 的距离.设 $\boldsymbol{u} = \boldsymbol{p} - \boldsymbol{v}$,则有

$$\|\boldsymbol{u}\|^2 = \boldsymbol{u}^{\mathrm{T}}\boldsymbol{u} = (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_1)^2 + (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_2)^2 + (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_3)^2 + \cdots + (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_n)^2.$$

上式所表示的实际上是勾股定理的扩展形式.

把上式重写成如下形式:

$$\|\boldsymbol{u}\|^2 - (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_1)^2 - (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_2)^2 = (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_3)^2 + \cdots + (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_n)^2.$$

注意到式子右边便是 $\boldsymbol{v}$ 到平面 $\boldsymbol{T}$ 的距离的平方.于是有

$$D^2 = \|\boldsymbol{u}\|^2 - (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_1)^2 - (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_2)^2 = \boldsymbol{u}^{\mathrm{T}}\boldsymbol{u} - (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_1)(\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{u}) - (\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e}_2)(\boldsymbol{e}_2^{\mathrm{T}}\boldsymbol{u}).$$

可以定义三元组 $\boldsymbol{Q} = (a, b, c)$ 来表示 D^2,$\boldsymbol{Q}$ 称作二次误差测度或二次误差矩阵,$\boldsymbol{Q}(\boldsymbol{v})$ 称作二次误差值.

$$D^2 = \boldsymbol{Q}(\boldsymbol{v}) = \boldsymbol{v}^{\mathrm{T}}\boldsymbol{a}\boldsymbol{v} + 2\boldsymbol{b}^{\mathrm{T}}\boldsymbol{v} + c.$$

求出对应的 $\boldsymbol{a}, \boldsymbol{b}, c$,以便求"边折叠"后新顶点的位置矢量,则 $\boldsymbol{a}, \boldsymbol{b}, c$ 的值如下:

$$\boldsymbol{a} = \boldsymbol{I} - \boldsymbol{e}_1\boldsymbol{e}_1^{\mathrm{T}} - \boldsymbol{e}_2\boldsymbol{e}_2^{\mathrm{T}};$$

$$b=(\boldsymbol{p}\cdot\boldsymbol{e}_1)\boldsymbol{e}_1+(\boldsymbol{p}\cdot\boldsymbol{e}_2)\boldsymbol{e}_2-\boldsymbol{p};$$

$$c=\boldsymbol{p}\cdot\boldsymbol{p}-(\boldsymbol{p}\cdot\boldsymbol{e}_1)^2-(\boldsymbol{p}\cdot\boldsymbol{e}_2)^2.$$

在这里,$\boldsymbol{a}$ 是一个 $n\times n$ 的矩阵,$\boldsymbol{b}$ 是一个 n 维向量,c 是一常数.

将顶点的各个相邻面的误差矩阵相加,就得到该顶点的误差矩阵.

1.2.2 折叠边及折叠后新顶点位置的选取

网格中每条边的误差矩阵为边的两个端点的误差矩阵之和:$\boldsymbol{Q}=\boldsymbol{Q}_1+\boldsymbol{Q}_2$,每次应选择当前误差值 $\boldsymbol{Q}(\boldsymbol{v})$ 最小的边进行折叠.

要使折叠后的误差值最小,则须对误差代价 $\boldsymbol{Q}(\boldsymbol{v})=\boldsymbol{v}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{v}$ 求偏导,令偏导值为0,解出最佳折叠后新顶点位置向量 $\boldsymbol{v}=-\boldsymbol{a}^{-1}\boldsymbol{b}$,相应的最小值为 $\boldsymbol{Q}(\boldsymbol{v})=-\boldsymbol{b}^{\mathrm{T}}\boldsymbol{a}^{-1}\boldsymbol{b}+c$.有时 $\boldsymbol{a}$ 无法求逆矩阵,也可在两个端点中选择一个代价较小的作为新顶点.这样的好处是程序更简单,速度更快,内存占用更小,但简化的效果上要差一点.

1.3 算法梗概

根据上面的讨论,本文提出的递进网格的生成方法可用下面的算法描述:

Step1 读入初始网格 $M=M_n$.

Step2 对 M 中的所有边,计算折叠误差,并根据误差的大小进行排序.

Step3 取折叠误差最小的边进行折叠操作,记录折叠信息,更新与之相关的所有信息.

Step4 重复 Step3,直至当前的简化网格满足要求.

Step5 输出简化网格 M_0 和边折叠记录序列(即递进网格).

2 算法实现的细节及实验结果

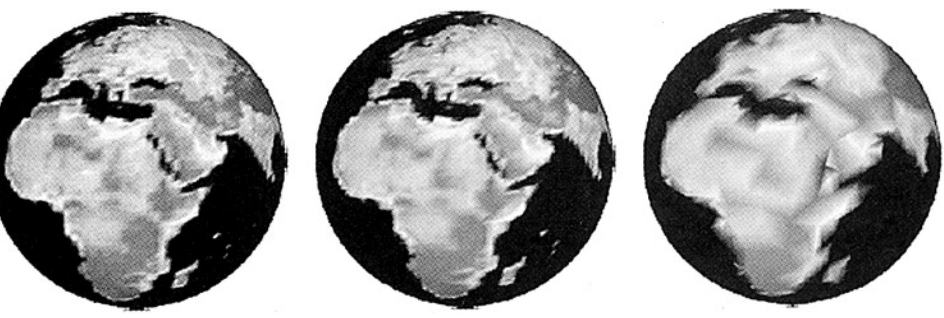
2.1 边界和颜色间断的保持

网格模型的间断,如褶皱、开放的边界和不同色彩区域间的边界等,常常是非常重要的视觉特征.因此,保持这些特征对于生成具有较高质量的简化模型非常关键.

前面提到的基本算法已经能够处理表面形状的间断(如褶皱).对于模型的边界和不同颜色区域的分界线(称为特征边),我们采取下面的方法处理:在预处理过程中标志出模型的边界边和特征边.为包含这些边的面做垂面,计算垂面的二次误差测度,然后将这些二次误差测度附加到边界顶点或特征边顶点的二次误差测度中.这样可以保证边界和分界线在简化过程中得到有效的保持.

2.2 实验结果

本文所介绍的算法已在P4 2.4G,256M内存,NVIDIA Geforce2 MX 400 64M显卡联想品牌机上用VC++6.0和DirectX8.0实现,系统为Microsoft Windows XP.图3是一组实例,执行结果验证了算法的有效性.从图中可以看出,不仅网格模型的几何信息得到了较好的保留,其颜色信息也得到了较好的保留.



(a) 73 728个面片 (b) 20 000个面片 (c) 3 000个面片

图3 具有色彩属性的球体模型的简化

Fig.3 Simplifying the sphere model with color

3 结论

本文提出并实现了一种新的递进网格简化处理方式.新的方法采用扩展二次误差测度表示简化网格与初始网格的几何特征和属性信息的匹配程度.利用二次误差值最小原理,指导网格简化操作的进行.同Hoppe提出的基于能量优化的算法和以往大多只能完成几何特征简化的算法相比,该算法不仅效率高,而且可以保证简化模型同初始模型在几何特征和颜色信息上尽可能相似.

在以后的研究中,我们将考虑将该算法与视点和绘制算法结合起来,形成与视点相关的递进

网格,从而支持实时图形绘制.

参考文献:

[1] HOPPE H. Progressive meshes [J]. Computer Graphics, 1996,30(1): 99 ~ 108.

[2] TAUBIN G, GUEZIEC A, HORN W, et al. Progressive forest split compression [A]. Proceedings of Computer Graphics(SIGGRAPH'98)[C], New York: ACM Press, 1998.123 ~ 132.

[3] 陶志良,潘志庚,石教英. 支持快速恢复的可逆递进网格及其生成方法[J]. 软件学报,1999,10(5): 503 ~ 507.

[4] 周振红,张君静,陈峙峰,等."数字黄河"工程及其关键技术支持[J]. 郑州大学学报(工学版),2003,24(2): 59 ~ 62.

[5] 卢章平,赵 泉. 基于"边折叠"的累进网格生成算法的研究[J]. 工程图学学报,2004,24(1): 37 - 41.

[6] 刘焕敏,杨克俭,王玉华. 一种面积加权的半边折叠网格简化算法及其递进网格构造[J]. 武汉理工大学学报,2005,29(2): 76 ~ 78.

[7] GARLAND M. Simplifying surfaces with color and texture using quadric error metrics [A]. Proceedings of IEEE Visualization'98 [C]. New York: ACM Press, 1998.263 ~ 269.

Research on the Algorithm to Generate Progressive Mesh with Color Based on Quadric Error Metric

TAN Tong - de, HAO Qi - hui, CHEN Zheng - yan, ZHAO Hong - ling, LI Run - zhi

(School of Information Engineering, Zhengzhou University, Zhengzhou 450002, China)

Abstract: Current algorithms for generating progressive meshes often have a low efficiency and can only complete the simplification of the geometric characters, without considering the other surface attributes. In order to solve these problems, we present a new algorithm in this paper. This algorithm generalizes the quadric error metric, which is based on the sum of squared distances of the vertex to all the planes in its set, from 3 - dimension to n - dimension including attributes information. Such a quadric error metric denotes the matching degree of the geometric characters and attributes information between the simplified mesh and the original one and then directs the process of mesh simplification under the least quadric error principle. The experimental results prove that this algorithm not only has a high efficiency, but also can guarantee a high resemblance of the geometric characters and color information between the simplified model and the original model.

Key words: quadric error metric; progressive mesh; edge collapse; level of detail