

一种可见外壳生成算法*

刘 钢, 王 锐, 鲍虎军, 彭群生

(浙江大学 CAD&CG 国家重点实验室, 浙江 杭州 310027)

E-mail: gliu@cad.zju.edu.cn

http://www.zju.edu.cn

摘要: 提出了一种可见外壳(visual hulls)生成算法.该算法充分发挥了自适应采样距离场(adaptively sampled distance fields,简称 ADFs)表示物体时在逼近精度和内存消耗方面的优越性.与传统的通过体素求交生成可见外壳的方法不同,该算法引入 ADFs 作为描述可见外壳的采样距离场的空间数据组织形式,从而极大地简化了求交运算,提高了算法的效率.最后通过改进的 SurfaceNets 算法,将可见外壳的 ADF 表示转化为面片表示.实验结果表明,该算法可以快速地生成具有较高精度的复杂模型.

关键词: 可见外壳;距离场;ADFs(adaptively sampled distance fields);等值面抽取;三维造型

中图法分类号: TP391 文献标识码: A

在计算机动画、游戏和虚拟现实等领域中都涉及到大量的三维模型,在这些领域中,简单、快速地获取三维模型具有极其重要的意义.目前,人们主要通过 3 种方式来获取三维模型:(1) 利用传统几何造型技术直接构造模型;(2) 利用三维扫描设备对真实物体进行扫描,进而重建出模型;(3) 利用从各个视角拍摄的真实物体的多幅图像重建模型.由图像重建三维模型技术又可分为两类:一类是通过多幅深度图像重建模型,另一类是通过多幅照片生成物体的可见外壳.第 2 类方法与第 1 类方法相比,算法较为复杂,但所需设备较少且对被造型物体的规模和位置限制也较少.本文给出的算法即属于第 2 类.

Martin 和 Aggarwal 最早提出了用多幅照片生成可见外壳的方法^[1].他们利用真实物体在照片上的侧影轮廓线和相关的相机参数实现了物体的三维重建.Chien 等人采用八叉树结构来表示物体的可见外壳^[2].这种八叉树结构是通过预先在平行投影照片上生成表示物体的四叉树的基础上建立的.Potmesil^[3]和 Szeliski^[4]同样建立了可见外壳的八叉树表示,但输入改为从任意视角拍摄的透视投影照片.Garcia 等人也提出了一种改进的八叉树表示方法^[5].20 世纪 90 年代以来,人们提出了众多基于从多视角拍摄的视频图像生成可见外壳的方法,例如,Moezzi 等人构建了一个由放置在不同位置的 17 架摄像机组成的重建动态场景的造型系统^[6].

在上述方法中,在从每一个视角拍摄的照片上都要进行物体的侧影轮廓线与所有八叉树节点投影的求交测试.由于八叉树节点在图像平面上的投影为多边形,所以上述求交测试是一项非常耗时的工作.为了简化求交测试,上述方法中采取了一些措施,例如,用包围盒来取代投影多边形参与求交测试,或者用多边形来逼近侧影轮廓线.前者显然会极大地降低重建精度,后者则会导致求交运算的复杂和不稳定,尤其是在物体的形状较为复杂或是有空洞时更是如此.Matusik 等人相继提出了两种方法来克服上述问题:一种是利用基于图像的方法^[7],但是这种方法并不能真正生成物体的模型;另一种是利用多边形求交的方法生成模型^[8],但是这种方法在输入图像数量较多时运算量很大.

* 收稿日期: 2002-03-01; 修改日期: 2002-06-19

基金项目: 国家自然科学基金资助项目(60033010;69925204;60021201)

作者简介: 刘钢(1975 -),男,辽宁铁岭人,博士生,主要研究领域为计算机图形学,隐式曲面,基于图像造型;王锐(1978 -),男,四川成都人,硕士生,主要研究领域为计算机图形学,基于图像造型;鲍虎军(1966 -),男,浙江乐清人,博士,研究员,博士生导师,主要研究领域为计算机图形学,虚拟现实,几何造型;彭群生(1947 -),男,湖南新化人,博士,教授,博士生导师,主要研究领域为计算机图形学,虚拟现实,红外仿真成像技术.

本文给出了一种新的可见外壳生成算法.与上述通过体素求交生成可见外壳的方法不同,算法引入了自适应采样距离场(adaptively sampled distance fields,简称 ADFs)作为描述可见外壳的采样距离场的空间数据组织形式,充分发挥了 ADFs 表示物体时在逼近精度和内存消耗方面的优越性,避免了上述复杂的求交过程.最后,通过改进的 SurfaceNets 算法将物体的 ADF 表示转化为面片表示,从而可以将转换结果直接导入其他图形管道.

1 基本概念

1.1 可见外壳

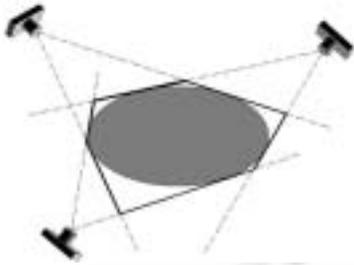


Fig.1 An object (hatched) and its corresponding visual hull (bold lines)

图 1 物体(阴影)和它对应的可见外壳(实线)

Laurentini 最早提出了可见外壳(visual hulls)的概念^[9].简单地说,可见外壳就是由空间物体的所有已知侧影轮廓线决定的该物体的凸包.当利用透视投影的方式从多个视角观察某一空间物体时,在每一个视角上都会得到一条该物体的侧影轮廓线.这条侧影轮廓线和对应的透视投影中心将共同决定一个三维空间中一般形状的锥体.显然,该物体必将落于这个锥体之内;而由所有已知侧影轮廓线和对应的透视投影中心决定的锥体的交最终将决定一个包含该物体的凸包,这个凸包就是物体的可见外壳.在多数情况下,空间物体的可见外壳是该物体的合理逼近.图 1 给出了物体和它所对应的可见外壳的二维情形.

1.2 距离场

距离场是一个标量场,它反映了空间中任意一点到一个给定物体表面的最短距离.这个距离可以是有符号的,从而可以表征该点是位于物体之外还是物体之内.这样,相对于任一封闭曲面 $S \subset R^3$,可以定义空间中任意一点 $p \in R^3$ 的有符号距离场值 $f(p) = dist(p, S)$.一般而言,可以设定该点位于物体之外时场值为正,位于物体之内时场值为负.显然, $f(p) = 0$ 就表示了物体的表面 S .将物体表示为距离场形式可以大大简化物体间的布尔运算、点相对物体位置的判断等操作.例如,对于用距离场表示的物体 $f_i(p) = 0, i = 1, 2, \dots, n$,它们的交和并可以分别表示为 $f_i(p) = \max(f_1(p), \dots, f_n(p)) = 0$ 和 $f_U(p) = \min(f_1(p), \dots, f_n(p)) = 0$.

相对于上述距离场的连续表示,距离场的离散表示在实际应用中也很普遍.距离场的离散表示是指在表达物体时只保留在离散采样点上的距离场值.一种常用的采样方法是均匀采样,即将物体所在空间剖分为均匀的三维网格,并保存各个网格顶点上的距离场值.这样,空间中任意一点的距离场值就可以通过对该点所在网格的 8 个顶点的场值进行三线性插值得到.均匀采样距离场虽然是物体的一种有效表达方式,但由于采用统一的剖分精度而难以表达物体表面的细节.这是因为即使物体表面只有很小的一部分具有较高的细节信息,但用均匀剖分的方法也需要将整个空间剖分为较高的精度.另外一种常用的采样方法是非均匀采样,例如常见的八叉树空间剖分采样.这类方法虽然在一定程度上起到了数据压缩的作用,但同样也面临着上述问题.自适应采样距离场(ADFs)则很好地解决了这一问题.

1.3 ADFs

ADFs 是一种基本的图形数据结构^[10,11].它可以表示复杂的物体,并支持布尔运算、点相对物体位置判断等大量的操作.ADFs 可以根据物体表面的局部细节对距离场进行自适应采样,即在物体表面存在细节处采用较高的采样精度,而在物体表面的平滑部分则取较低的采样精度,并将采样得到的距离场值保存到八叉树空间数据结构中.这样,ADFs 就可以按照预先设定的任意逼近精度表示物体,而且节省了大量不必要的内存开销.

在传统的八叉树空间剖分技术中,所有位于物体边界上的叶节点都被剖分到某一预先设定的剖分层次,而在 ADFs 表示中,边界叶节点是否被继续剖分取决于该节点内部距离场值的变化情况,只有当边界叶节点内部的点的场值不能按照预设精度通过三线性插值有效逼近时,该节点才被继续剖分.因此,在物体表面的平滑部分

就可以使用较大的网格,从而极大地压缩了数据量.

1.4 图像表示

一般而言,相机拍摄到的图像由两个要素组成:一个是分布在图像平面上的一系列二维像素点,每一个像素点包含了物体表面某一点在当前视线方向上的亮度信息;另一个是一个透视投影变换矩阵,这个矩阵表明了空间物体到二维图像的映射关系.若分别用齐次坐标系 $O(x, y, z, w)$ 表示物体空间,用 $I(u, v)$ 表示图像空间,则这一变换可表达为 $[uw', vw', w']^T = T[x, y, z, w]^T$, 其中 $T = (t_{ij})_{3 \times 4}$ 即为透视投影变换矩阵.对于实拍图像,变换矩阵 T 一般通过相机定标技术得到^[3].具体做法是,将已知物体表面上的点及其投影到图像平面上的对应点(至少 6 对)的坐标代入上述变换矩阵,进而求解联立的线性方程组.本文主要关心可见外壳的重建算法,因而采用了由造型软件生成的图像来代替实拍图像的做法,故不存在定标问题.

2 新的可见外壳生成算法

2.1 算法概述

本算法主要包含如下几个步骤:

- (1) 图像获取:首先得到从各个视角拍摄的用于重建的物体照片,并对这些照片做预处理操作;
- (2) 构造距离场函数:构造一个距离场函数,使其能够返回物体包围盒内任意一点相对于最终生成的可见外壳的距离场值;
- (3) 构造 ADFs 表示:将 ADFs 作为距离场值的空间数据组织形式,对包围盒进行自适应的剖分,直到可见外壳的满意逼近为止;
- (4) 生成面片表示:利用改进的 SurfaceNets 算法将可见外壳的 ADF 表示转化为面片表示.

2.2 图像预处理

在本算法中,需要正确获取物体在每幅照片上的侧影轮廓线,进而将物体和背景分离开并分别做出标记.本文由于采用了由造型软件生成的具有单一背景的照片,故分离方法较为简单,只要按照某一亮度阈值对图像上的所有像素进行分类即可.另一方面,为了求取空间点的距离场值,需要找出图像平面上任意一点在侧影轮廓线上的最短距离点.为此,我们预先建立图像空间相对于侧影轮廓线的平面距离场表示,即求出图像上任意一个像素点到侧影轮廓线的最短距离.具体地,我们将侧影轮廓线上的点沿着它们的四连通或是八连通方向同时向内、外扩展,从而形成了一系列平行于侧影轮廓线的近似等距曲线.这样,对于图像平面上的任意一点,我们只需沿着平面距离场的梯度方向即可容易地找到该点在侧影轮廓线上的最短距离点.需要指出的是,采用较高分辨率的图像虽然可以明显地改善本算法的重建效果,但同时也增大了此处预处理步骤和后续最短距离点查找工作的负担.

2.3 距离场函数的构造

对于每一幅给定的照片,物体在照片上的侧影轮廓线和对应的透视投影中心都决定了一个空间锥体,这一锥体可以用一族直线 $\{s(t), r(t)\}$ 来定义,其中 $s(t)$ 为物体在图像平面 P 上的侧影轮廓线, $r(t)$ 则由 $s(t)$ 上的一点和透视投影中心 $p \in P$ 共同决定.为了得到物体可见外壳的 ADF 表示,我们需要构造一个距离场函数,使其能够返回物体包围盒内任意一点 V 相对于所有上述锥体的交的距离场值.为此,可以先求出 V 点相对于其中某一个锥体的距离场值,即求出 V 点到直线簇 $\{s(t), r(t)\}$ 的最短距离.一种直接的办法是计算出 V 点到直线簇中每一条直线的距离,并找出其中的最短距离,但这无疑会降低算法的

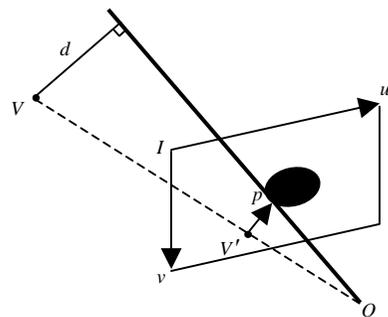


Fig.2 Calculation of the distance value at a random vertex in the bounding volume
图 2 包围盒内任意一点距离场值的计算

效率.为此,如图2所示,我们将 V 点投影到图像平面上,得到点 V' ,利用图像预处理步骤中的结果,容易确定 V' 点在侧影轮廓线上的最短距离点 p ,这样, V 点到由投影中心 O 和最短距离点 p 决定的直线的距离即为 V 点的距离场值,场值的符号则可以由 V' 点相对于侧影轮廓线的位置决定.

重复以上步骤,我们就能得到 V 点相对于每一个锥体的距离场值,利用距离场表示便于进行布尔运算的特点,我们只需保留这些距离场值中的最大值,就得到了该点相对于最终生成的可见外壳的距离场值.由此可见,我们通过引入距离场表示避免了以往算法中复杂的求交运算,大大提高了算法的效率.

2.4 ADF表示的构造

若已知包围盒内任意一点相对于可见外壳的距离场值,就可以很容易地得到可见外壳的 ADF 表示.首先,我们将物体的包围盒作为八叉树空间剖分的根节点,并计算出根节点的 8 个顶点的距离场值,然后对根节点进行递归剖分,进而计算出各个子节点中新生成顶点的距离场值.若八叉树节点满足如下几个条件之一,就可以对其停止剖分:(1) 没有零等值面通过该节点;(2) 有零等值面通过该节点,但利用该节点的 8 个顶点的距离场值插值得到的等值面满足预先设定的逼近精度;(3) 达到了预先设定的最大剖分层次.对于上述第(2)种情况,可以选择多种评测方法,一种简单、常用的方法是,通过三线性插值计算出若干比较点的场值(这些比较点包括节点的中心、节点每条边的中心和每个面的中心),并把这些场值与通过距离场函数计算得到的这些点的场值相比较,若其中某一个值大于预设的逼近精度,则继续剖分,否则停止剖分.当八叉树结构中的每一个节点都满足上述剖分终止条件之后,我们就得到了物体的可见外壳的 ADF 表示.事实上,为了加速 ADF 表示的构造,我们无须计算出所有八叉树节点的顶点的距离场值,而只需处理位于物体边界上的节点;在将八叉树节点投影至图像平面之后,我们就可以方便地将位于边界上的节点与其他节点区分开来.

2.5 面片抽取

在获取了 ADF 表示以后,距离场的零等值面就表示了物体的可见外壳.当从各个视角拍摄的物体的照片数量足够多时,可见外壳就能够较好地逼近原物体.注意到 ADF 只是对原物体可见外壳的隐式表示,为了使重建结果易于被其他造型和绘制系统所采用,需要将可见外壳的 ADF 表示转化为面片表示.对于均匀采样距离场,通常采用 Marching Cubes 算法来抽取距离场的等值面^[12].该算法在相关剖分网格的边界上插值生成新的三角面片顶点,从而产生物体的面片表示.但是,考虑到 ADF 表示中含有自适应剖分得到的大小不同的空间网格,利用 Marching Cubes 算法抽取面片往往会产生空洞.我们以二维情形为例来说明这一问题.在图3中,在两个大小不同的剖分网格交界处,其中一个网格中插值得到的顶点位置可能与在另一个网格中插值得到的顶点位置不同,这就使得抽取结果在该处发生断裂.为了产生可见外壳的拓扑连续的面片表示,我们采用了改进的 SurfaceNets 算法^[11,13].该算法可以快速地抽取高质量的面片表示,并有效地克服了产生空洞的问题.

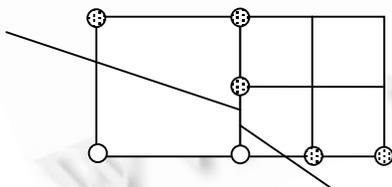


Fig.3 The marching cubes algorithm may cause cracks on adaptive grids

图3 利用 Marching Cubes 算法处理自适应剖分网格可能会产生空洞

该算法主要包括以下 3 个步骤:

(1) 在 ADF 的每一个边界叶节点内放置一个网格顶点,这个顶点的初始位置在边界叶节点的中心.

(2) 当符合下述几个条件时,3 个相邻边界叶节点内的网格顶点将被连接起来,构成一个三角面片: 3 个边界叶节点有一条公共边(因此,每个三角面片与一条节点边线相对应);

在这条公共边上有等值面通过; 因为每个边界叶节点有 12 条边,所以最多可以有 12 个三角面片与其对应,为了避免产生多余的三角面片,对于每一个边界叶节点,只允许在它的

6 条边上产生新的三角面片,这 6 条边分别位于边界叶节点的右上方、左下方、前上方、后下方、右前方和左后方.对于每一个新生成的三角面片,其法向可由其对应公共边两端点的场值推出.

(3) 按上述要求连接好所有的三角面片之后,我们就得到了一个拓扑连续可见外壳面片网格.为了提高网格的逼近精度,可以将每个预置于边界叶节点中心的网格顶点向零等值面方向移动,即沿着 ADF 梯度方向移动,移动距离就等于该点的距离场值;对于较大的三角面片,则可以在面片内插入新的网格顶点,并同样将其向

零等值面方向移动.最后,为了改进网格质量,可将网格顶点沿零等值面向相邻网格顶点的平均位置移动.

在上述改进的 SurfaceNets 算法中,没有像 Marching Cubes 算法那样在网格边界上产生新的三角面片顶点,而是在每个边界叶节点的中心位置设置顶点,即三角面片可以跨越大小不同的叶节点的面和边,从而有效地避免了产生空洞的问题.另外,从上述面片抽取过程可知,改进的 SurfaceNets 算法可以自然地产生物体的 LOD 模型.

3 实验结果与结论

我们已经在 P 600 微机上用 C++ 语言实现了上述的可见外壳生成算法,并对若干实例进行了测试.其中的一个实例是利用造型软件生成的茶壶模型,其包围盒是一个大小为 $20 \times 20 \times 20$ 的立方体.我们从各个角度对其拍摄了 20 幅照片作为输入,其中每幅照片的分辨率为 400×300 .图 4 给出了原始照片的第 1、3 和第 5 幅,图 5 给出了在不同逼近精度下的重建结果,表 1 对这些重建结果进行了比较,图 6 则给出了另外两个较为复杂的模型,其中每个模型均采用分辨率为 400×300 的 20 幅照片作为输入,模型包围盒大小为 $80 \times 80 \times 80$,预设逼近精度为 0.5.从以上重建结果可以看出,由于采用了 ADFs 表示,本算法可以处理具有较高精度的复杂模型,同时减少了大量不必要的内存开销,算法效率较高.



Fig.4 Three of the twenty original images
图 4 20 幅原始照片中的 3 幅



Fig.5 Reconstruction results with different approximation precisions
图 5 不同逼近精度下的重建结果

Table 1 Comparison of the reconstruction results in Fig.5

表 1 图 5 中重建结果的比较

Model	Number of images	Image resolution	Approximation precision	Run time (s)
Left	20	400×300	0.6	7
Middle	20	400×300	0.4	16
Right	20	400×300	0.25	35

模型, 输入图像数量, 图像分辨率, 预设的逼近精度, 运行时间, 左, 中, 右.

此外,我们将若干基于本算法的重建结果与基于均匀采样距离场的重建结果进行了比较.结果表明,本算法对于表面相对光滑并且拓扑简单的物体,在速度上优势明显;而对于拓扑复杂的物体,例如物体上存在狭长部分或有空洞时,则与均匀采样比较并无明显优势,在某些情况下甚至需要更长的处理时间.这主要是因为在上述情形时,为了不改变物体的拓扑结构,我们不得不将物体的包围盒均匀剖分到某一层次后再开始自适应采样,而自适应采样与均匀采样相比又需要作额外的判断,以决定是否对网格继续剖分.



Fig.6 Another two complex examples: for the both examples, the left is one of the original 20 images and the right two are the reconstructed models

图 6 两个较为复杂的实例.左侧一幅为 20 幅原始照片中的一幅,右侧两幅为重建结果

我们下一步的工作是将上述算法应用于由放置在不同视角上的摄像机获取的视频图像,进而构建一个分布式的实时造型系统.

References:

- [1] Martin, W.N., Aggarwal, J.K. Volumetric descriptions of objects from multiple views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1983,5(2):150~158.
- [2] Chien, C.H., Aggarwal, J.K. Identification of 3D objects from multiple silhouettes using Quadrees/Octrees. *Computer Vision, Graphics, and Image Processing*, 1986,36:256~273.
- [3] Potmesil, M. Generating Octree models of 3D objects from their silhouettes in a sequence of images. *Computer Vision, Graphics, and Image Processing*, 1987,40(1):1~29.
- [4] Szeliski, R. Rapid Octree construction from image sequences. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 1993,58(1):23~32.
- [5] Garcia, B., Brunet, P. 3D reconstruction with projective Octrees and epipolar geometry. In: Chandran, S., Desai, U., eds. *Proceedings of the 6th International Conference on Computer Vision*. New Delhi: Narosa Publishing House, 1998. 1067~1072.
- [6] Moezzi, S., Katkere, A., Kuramura, D., *et al.* Reality modeling and visualization from multiple video sequences. *IEEE Computer Graphics and Applications*, 1996,16(6):58~63.
- [7] Matusik, W., Buehler, C., Raskar, R., *et al.* Image-Based visual hulls. In: Akeley, K., ed. *Proceedings of the SIGGRAPH 2000*. New York: ACM Press, 2000. 369~374.
- [8] Matusik, W., Buehler, C., McMillan, L. Polyhedral visual hulls for real-time rendering. In: Myskowski, K., Gortler, S., eds. *Proceedings of the 12th Eurographics Workshop on Rendering*. Wien: Springer-Verlag, 2001. 115~125.
- [9] Laurentini, A. The visual hull concept for silhouette based image understanding. *IEEE PAMI*, 1994,16(2):150~162.
- [10] Frisken, S., Perry, R., Rockwood, A., *et al.* Adaptively sampled distance fields: a general representation of shape for computer graphics. In: Akeley, K., ed. *Proceedings of the SIGGRAPH 2000*. New York: ACM Press, 2000. 249~254.
- [11] Perry, R., Frisken, S. Kizamu: A system for sculpting digital characters. In: Fiume, E., ed. *Proceedings of the SIGGRAPH 2001*. New York: ACM Press, 2001. 47~56.
- [12] Lorensen, W.E., Cline, H.E. Marching cubes: a high resolution 3D surface construction algorithm. In: Stone, M.C., ed. *Proceedings of the SIGGRAPH'87*. New York: ACM Press, 1987. 163~169.

