

基于测地距离的多边形网格模型约束变形*

周 艳^{1,2+}, 刘圣军², 金小刚², 冯结青², 叶晓丰^{1,2}

¹(温州大学 计算机科学与工程学院, 浙江 温州 325000)

²(浙江大学 CAD&CG 国家重点实验室, 浙江 杭州 310027)

Geodesic-Based Constrained Deformations for Polygonal Mesh Models

ZHOU Yan^{1,2+}, LIU Sheng-Jun², JIN Xiao-Gang², FENG Jie-Qing², YE Xiao-Feng^{1,2}

¹(School of Computer Science and Engineering, Wenzhou University, Wenzhou 325000, China)

²(State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310027, China)

+ Corresponding author: Phn: +86-571-88206681 ext 507, Fax: +86-571-88206680, E-mail: zhouyan@wzu.edu.cn

Zhou Y, Liu SJ, Jin XG, Feng JQ, Ye XF. Geodesic-Based constrained deformations for polygonal mesh models. *Journal of Software*, 2007,18(6):1543–1552. <http://www.jos.org.cn/1000-9825/18/1543.htm>

Abstract: This paper presents a geodesic-based constrained deformation method for polygonal mesh models. After a user specifies a series of constraints (which may consist of points, lines and faces), their effective radii, maximum displacements, scaling, and rotation angles, the method creates the deformation weights for the vertices of the mesh surfaces using the field values determined by the geodesic distances between the vertices and the constraint sources. Compared with the traditional constrained deformation model based on Euclidean distance, this method can avoid the undesired deformation on the regions near the constraints. Experimental results show that the new deformation model is both efficient and intuitive.

Key words: constrained deformation; geodesic path; geodesic distance; geometric modeling; polygonal mesh model

摘 要: 提出了一种基于测地线的多边形网格模型的约束变形方法. 首先给定一系列的变形约束源(可以是点、线或者面)以及约束源的有效半径及变形目标(偏移量、缩放比例、旋转轴和旋转角度), 然后通过计算三角形网格的各顶点到约束源的测地距离来确定各顶点的场值, 这个场值将作为变形的权值. 在基于欧氏距离的传统约束变形中, 对某一约束区域的变形往往导致对约束源附近区域不需要的变形结果, 而利用测地距离来计算各点的变形权值, 可以很好地避免这种现象的出现. 实验结果表明, 这种变形方法是直观而且有效的.

关键词: 约束变形; 测地线; 测地距离; 几何建模; 多边形网格模型

中图法分类号: TP391 文献标识码: A

三维形状的变形是几何建模和计算机动画的一个重要工具. 三维变形方法主要包括基于物理的变形和空

* Supported by the National Natural Science Foundation of China under Grant No.60573153 (国家自然科学基金); the Program for New Century Excellent Talents in University of China under Grant No.NCET-05-0519 (新世纪优秀人才支持计划); the Natural Science Foundation of Zhejiang Province of China under Grant No.R105431 (浙江省自然科学基金)

Received 2006-06-06; Accepted 2006-08-16

间变形这两种,其中使用比较多的是空间变形方法.现在应用比较广泛的空间变形技术是由 Sederberg 和 Parry 提出的自由变形(free form deformation,简称 FFD)技术^[1].之后,针对自由变形技术又有了许多改进的算法.如 Hsu 等人提出了直接操纵的 FFD 变形方法^[2],但该方法的计算复杂度非常高.FFD 的方法及其各种变种为空间变形提供了高度的控制,但使用这类方法的局限性主要在于,用户首先必须在被变形物体的周围定义一个控制网格并规定一些控制点,然后通过操纵这些控制点去完成变形的过程.而且,这种方法只适合于对物体做粗略的变形,不适用于对物体进行精细的变形.冯结青等人针对 B 样条曲线和曲面控制的空间变形方法,提出了面向多边形物体的空间变形反走样方法^[3].Llamas 等人用一对三维轨迹球去控制一个虚拟的变形条,可以直观地控制物体的变形^[4,5].利用拉普拉斯表面编辑的方法可以在变形的过程中保持物体的细节^[6,7].

Borrel 等人提出了一种基于任意多个约束点的简单约束变形方法^[8].这种简单约束变形方法具有局部性,而且对用户来说又很直观.用户可以把这些约束点直接放置在被变形物体的表面,然后通过控制这些点对物体进行变形.但是这种约束变形方法,其约束源只能是点.而对一种通用的模型来说,很自然地希望能把约束源扩展到线、面甚至体.在 Jin 等人提出的基于广义元球的一般约束变形方法中^[9],约束变形的约束源被扩展为可以包括点、线、面和体.首先,在被变形物体上指定变形的约束源(点、线、曲面或体)、约束的有效半径和变形目标,然后,根据这些约束源计算出空间各点的场值作为变形的权重对物体进行变形.张慧等人分别用一个参考面片和一个目标面片来定义约束变形^[10].基于卷积曲面的约束变形方法^[11]进一步克服了基于元球的变形中所产生的鼓包现象.

但是在上述约束变形方法中,对于约束源的有效范围的计算都是基于欧氏距离的,这样,在变形过程中对某一约束区域进行变形时,往往会对附近不需要变形的区域产生一定的影响.如图 1(b)所示,当对小狗的耳朵进行变形时,由于采用的是欧氏距离,小狗的耳朵离脸部和身体很近,不可避免地引起了脸部和身体的变形,而这样的变形结果并不是我们所希望的.在计算机动画和几何造型领域,待变形的模型中经常会包含这些空间上直线距离很近但是曲面上的距离又并不近的区域,比如兔子的两只耳朵之间的距离,或者一只手的 5 个手指之间的距离.如果在进行变形时直接采用欧氏距离,必然会给周围部分带来不需要的变形.考虑到曲面上两点之间的测地线是连接这两点的最短路径,如果采用测地距离来计算约束源的有效范围以及各顶点的场值,将可以很好地克服在变形过程中对非变形区所产生的扭曲现象.如图 1(c)所示,对小狗耳朵的变形并不会对脸部和身体产生影响.

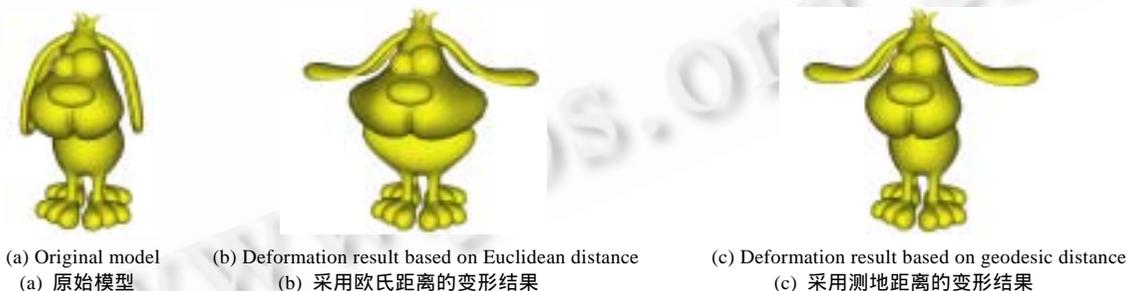


Fig.1 Deformation results based on Euclidean distance and geodesic distance

图 1 基于欧氏距离和测地距离的约束变形结果

测地线是连接曲面上给定的两点之间的最短路径.根据计算结果的精确性,可以将网格上的测地距离的计算方法分为两大类:精确的方法和近似的方法.

Chen 和 Han 提出了一种精确计算测地线的方法^[12],并且该算法由 Kaneva 等人部分实现^[13].Kapoor 提出了一种计算网格上给定的两顶点间的测地路径的方法^[14].肖春霞等人实现了点采样曲面上的测地线的计算^[15].

测地线的近似计算方法大都采用迭代法^[16-19],因此,其结果与初始的近似路径的精确性有很大关系.Kimmel 和 Sethian 使用快速跟踪法(fast-marching method)近似计算网格上的测地线^[20].在这种方法的基础上,Novotni 和 Klein 改进其更新步骤以获得更高的精确度^[21].

Surazhsky 等人在文献[22]中实现了精确计算测地线的方法,其计算结果比文献[13]快几倍.而他们所提出的测地线的近似算法的精确性也比文献[20]要高.但是,对于我们的应用来说,文献[22]中的方法速度还不够快.

我们希望在控制一定误差的基础上获得尽可能快的速度.在实验中发现,使用文献[21]中的方法计算出的测地距离比文献[20]的方法更精确,而速度比文献[22]中的方法更快,比较适合我们的需要.但是,文献[21]只能计算单个源点到网格上各顶点的距离,而我们希望源的形状可以包括点、线或面.所以,我们对文献[21]进行了一定的扩展,使其可以有效地计算出网格上的各个顶点到不同形状的约束源的测地距离值.

在我们的变形方法中,用户只要指定约束源(可以是点、线或者面)、约束源的有效半径及变形目标(偏移距离、缩放比例以及旋转轴和旋转角度),就可以通过计算多边形网格(这里,我们采用了三角形网格,其他多边形网格可以很容易地转换为三角形网格)的各顶点到约束源的测地距离确定变形的有效范围和各顶点的变形权值来进行变形.实验结果表明,这种变形方法是有效而且直观的.由于所采用的测地距离的计算是基于三角形网格的,所以我们的约束变形方法是与物体表示相关的.

本文第 1 节详细讨论网格模型上的测地距离的计算方法.第 2 节介绍基于网格的约束变形模型.第 3 节给出一些实验结果.第 4 节是结论.

1 三角形网格表面的测地距离计算

测地线在科学与工程的许多领域都非常有用,比如机器人的运动规划、地形漫游、模型表面的参数化等.按照应用领域的不同,对测地线的定义也各不相同^[18].其中使用最广泛的定义是将测地线看作三维模型表面两点之间的最短路径,该路径被约束在模型的表面.曲面上两点之间的测地线的长度就是这两点之间的测地距离.

利用以上定义,我们可以计算出模型上的一个源点到其余各点之间的测地距离.但是,在我们的应用中,希望其源可以是点、线或面.下面给出一个扩展后的测地距离映射的定义.

1.1 测地距离映射的定义

假设 S 是网格模型上的源(这里,源的形状可以是点、线或曲面),设 $q \in S$ 为源上的点,则模型上的任意点 p 到该源 S 的测地距离就是从 p 到 S 上的最近点之间的距离.对应于一个网格表面上的某一个源 S 的测地距离映射 ψ 是一个 $R^3 \rightarrow R$ 的映射,我们可以把 ψ 定义为

$$\psi(p) = \min \{ \|p - q\|_{Geo} : q \in S \} \quad (1)$$

这里, $\|p - q\|_{Geo}$ 表示点 p 和 q 之间的非负测地距离值.

三角形上其他点的测地距离值可以通过插值三角形顶点处的值得到.在图 2 中给出了同一个模型上的不同形状的源(用较粗的线条表示)及其根据式(1)所定义的测地距离映射求出来的等测地线.

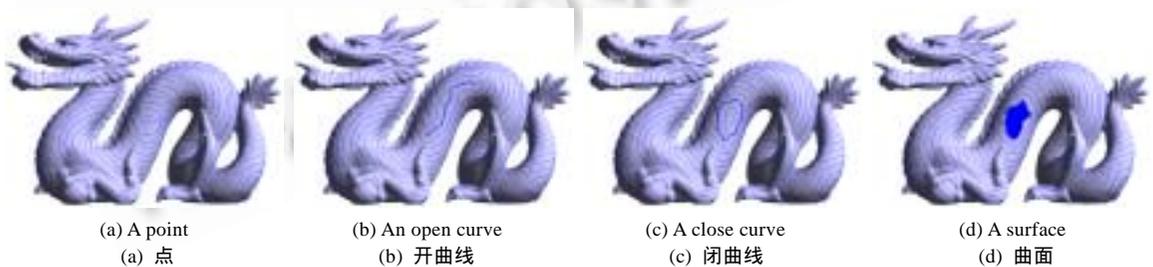


Fig.2 Isolines of geodesic distance generated from sources of different shapes

图 2 各种不同形状的源及其等测地线

1.2 测地距离的快速计算方法

Kimmel 和 Sethian 使用快速跟踪法(fast-marching method)来近似计算网格上的测地线^[20].将整个模型的点分成 3 个集合,已经计算好测地距离的点集 G_{fixed} ,当前活动的点集 G_{active} ,还没有处理的点集 $G_{unprocessed}$.由于算法每次都是选择具有最小距离值的点开始计算,所以整个算法能保证向整个模型传递.下面是算法的伪代码.

Algorithm FastMarching(G).

Input: A graph $G(v, f)$ of the given mesh surface M and a given shape S ;

Output: The updated weight number W_{v_i} of every triangular node.

```

1.  for every node  $v_i \in G$  {
2.       $W_{v_i} \leftarrow +\infty$ ;
3.      Set the flag of  $v_i$  —  $fhp_{v_i}$  to unprocessed;
4.      if ( $v_i \in S$ )
5.          Add  $v_i$  to the minimal heap  $H_v$ ,  $W_{v_i} \leftarrow 0$ , and set  $fhp_{v_i}$  to active;
6.      }
7.  do {
8.       $v_k \leftarrow$  pop the primitive with minimal value from  $H_v$ ;
9.      Set  $fhp_{v_k}$  to fixed;
10.     for every node  $v_j$  adjacent to  $v_k$  {
11.         if ( the triangle  $f \in G$  incident to  $v_j$  and  $v_k$  contains a vertex  $v_m$  which  $fhp_{v_m}$  is fixed){
12.             Compute the distance value  $W'_{v_j}$  for node  $v_j$ ;
13.             if ( $W'_{v_j} < W_{v_j}$ ){
14.                  $W_{v_j} \leftarrow W'_{v_j}$ ;
15.                 if ( $fhp_{v_j}$  is unprocessed){
16.                     Set  $fhp_{v_j}$  to active;
17.                     Add  $v_j$  to  $H_v$ ;
18.                 }
19.                 Update  $H_v$ ;
20.             }
21.         }
22.     }
23. }while ( $H_v = \emptyset$ )

```

Novotni 和 Klein 在此基础上改进了算法的更新步骤(算法伪代码中的第 12 步)^[21],得到更加精确的结果.他们所采用方法的更新步骤是:若三角形的 3 个顶点为 v_1, v_2 和 v_3 ,假设已经求得顶点 v_1 和 v_2 到源点的测地距离分别为 $T(v_1)$ 和 $T(v_2)$,接下来需要求出 $T(v_3)$ 的值.不失一般性,假设顶点 v_1, v_2 和 v_3 位于 xy 平面上.分别以 v_1 和 v_2 为圆心、 $T(v_1)$ 和 $T(v_2)$ 为半径作圆,可以得到两个交点 o_1 和 o_2 .在这两个交点中选取离 v_3 较远的点为伪源点,此时,伪源点处于三角形所在的 xy 平面上.这样, v_3 到该伪源点的距离就是 v_3 到源点的近似测地距离 $T(v_3)$,如图 3(a) 所示.

在文献[21]中有一种特殊情形没有被讨论,即如果 v_1 和 v_2 中有一点是鞍点或边界点,则上述的更新规则可能会不正确.例如,在图 3(b)中,顶点 v_1 是边界点(边界已经用灰色虚线标出),这时 o_1 和 v_1 之间的直线就不是曲面上的有效地测地路径.对于这种情形,我们规定 $T(v_3)$ 为 $\min\{T(v_1)+\|v_1v_3\|, T(v_2)+\|v_2v_3\|\}$ (这里, $\|\dots\|$ 代表两顶点之间的欧氏距离).

另外,文献[21]中所提出的算法仅局限于单个源点的情形.如果要将源的形状扩展为包括线和曲面上的多边形区域等情形,此时在源中就包含了多个点.在特殊情况下, $T(v_1)$ 和 $T(v_2)$ 之和会小于顶点 v_1 和 v_2 之间的边长,如图 3(c)所示.这时,两个圆之间没有交点,无法确定伪源点,则上面所采用的更新方法就会失效.在这种情况下,我们采用以下方法来近似估算 $T(v_3)$:

- 如果 v_1v_2 是边界边,则 $T(v_3)=\min\{T(v_1)+\|v_1v_3\|, T(v_2)+\|v_2v_3\|\}$.
- 如果 v_1v_2 有两个相邻三角形 $\Delta v_0v_1v_2$ 和 $\Delta v_1v_2v_3$,如图 3(d)所示,则将这两个三角形都变换到 xy 平面上. $T(v_3)$

用下式计算:

$$T(v_3) = \min\{T(v_0) + \|v_0v_3\|, T(v_1) + \|v_1v_3\|, T(v_2) + \|v_2v_3\|\}.$$

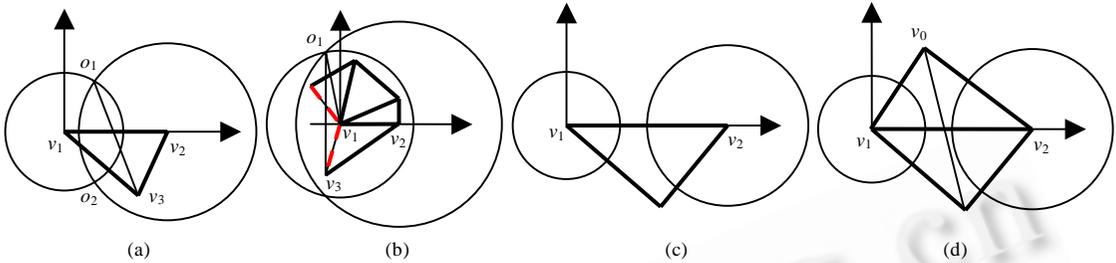


Fig.3 Computing the geodesic distance of $T(v_3)$

图 3 计算测地距离 $T(v_3)$

从以上论述也可以看出,由于我们对测地距离的计算采用的是迭代的方法,所以只适合于模型为流形的情形.当模型为非流形时,在计算测地距离时将不能遍历所有的三角形.对于这种情形,我们可以先对模型进行预处理,将模型封闭.

2 三角形网格模型的约束变形

三角形网格模型的约束变形的基本思想是,首先由用户给出变形的约束源(可以是模型上的点、线或者面)、约束的有效半径 R 及变形目标(偏移、缩放以及旋转),然后根据模型上各顶点到约束源的距离(这里为测地距离)来确定各点处的变形权值进行变形.在第 1 节我们已经介绍了模型上各点到约束源的测地距离的计算方法,利用测地距离值来计算约束的有效半径 R 可以很好地避免在非变形区产生的扭曲现象.下面我们将介绍如何计算各顶点的基于测地距离的场值,并给出变形模型的定义.

2.1 势函数的确定

首先需要为约束源 S 定义一个势函数.由于已经确定了约束的有效半径 R ,也就是说,只有到 S 的测地距离值小于 R 的点才会产生变形,而到 S 的测地距离值大于 R 的点将不受影响.所以从理论上来说,任何帽子函数都可以定义为所需要的势函数.在这里,我们采用了 Myvill 的六次多项式作为势函数^[23],

$$f(r, R) = \begin{cases} -\frac{4}{9}\left(\frac{r}{R}\right)^6 + \frac{17}{9}\left(\frac{r}{R}\right)^4 - \frac{22}{9}\left(\frac{r}{R}\right)^2 + 1, & 0 \leq r \leq R \\ 0, & r > R \end{cases} \quad (2)$$

在上式中, r 表示模型上的点与约束源 S 的测地距离, R 是约束的有效半径, $f(r, R)$ 是相对于 S 的势函数.对于网格模型上的任意点 p , 利用式(2)可以计算出其场值 $f_p(r, R)$, 作为该点处变形的权值.

2.2 变形模型的定义

给定了的势函数定义之后,我们就可以定义约束变形模型,包括平移、旋转和缩放.

2.2.1 平移变形

假设 $p=(x, y, z)$ 是模型上的点, $d(d_x, d_y, d_z)$ 是相对于约束源 S 的最大偏移量, $f_p(r, R)$ 是约束源 S 在点 p 处的势函数值. $Displacement(p)$ 是平移变形映射, 则有以下计算公式:

$$Displacement(p) = p + d \times f_p(r, R) \quad (3)$$

根据式(2)的定义,显然,当点 p 与约束源 S 的测地距离大于或等于有效半径 R 时, $Displacement(p)=p$, 即点 p 不受变形影响; 当点 p 恰好处于约束源 S 上时, $Displacement(p)=p+d$, 即在该点处获得最大的偏移量.如图 4 所示是几种不同形状的约束源对同一个长方体模型的平移变形结果.

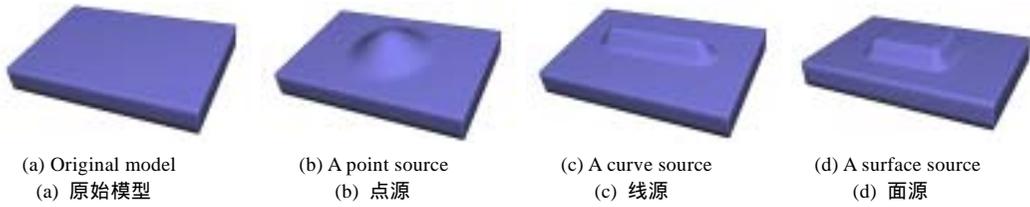


Fig.4 Deformation results by displacement

图4 平移变形的结果

2.2.2 旋转变形

要想让模型能够对任意旋转轴做旋转变形,比较简单的方法是先在约束源 S 处建立一个局部坐标系,然后将模型变换到该局部坐标系中进行旋转变换,最后再将局部坐标系变换回全局坐标系.

要表示物体的旋转,可以采用欧拉变换.但是采用欧拉变换可能会带来万向节死锁(gimbal lock)的问题,所以我们采用了四元数法.假设要让空间点 p 绕单位轴 u 旋转 2θ 角度,则有单位四元数 $q=(\sin\theta u, \cos\theta)=(x, y, z, w)$, 旋转矩阵 $R(2\theta)$ 可以由下式决定^[24]:

$$R(2\theta) = \begin{bmatrix} 1-2y^2-2z^2 & 2xy-2wz & 2xz+2wy \\ 2xy+2wz & 1-2x^2-2z^2 & 2yz-2w \\ 2xz-2wy & 2yz+2wx & 1-2x^2-2y^2 \end{bmatrix}$$

对于约束变形的问题,假设约束源为 S ,要绕轴 u 旋转 2θ 角度进行变形,则对模型上点的旋转变形公式如下:

$$Rotation(p)=R(2\theta \times f_p(r,R)) \times p \quad (4)$$

其中 $f_p(r,R)$ 是约束源 S 在点 p 处的场值.

图 5(a)~图 5(c)所示是针对同一个长方体模型在不同方向上的旋转变形的结果.从图 5(b)可以看出,当有效半径 R 足够大时,整个物体都会受到变形的影响.

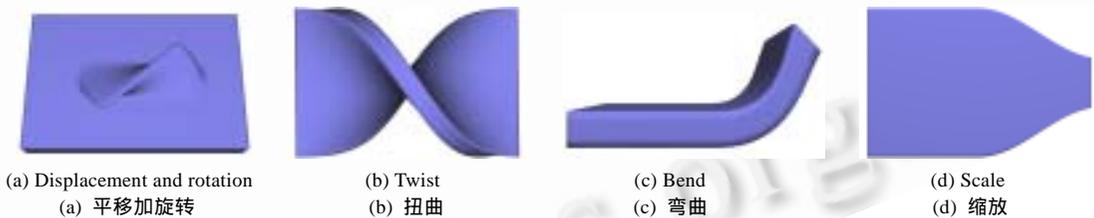


Fig.5 Deformation results by rotation and scale

图5 旋转和缩放变形的结果

2.2.3 缩放变形

缩放变形是比较简单的变形方式.假设在约束源处的最大缩放比例为 $s=(s_x, s_y, s_z)$, 则对模型上的任意点 p 的缩放变形公式如下:

$$Scale(p)=p+(s_x-1, s_y-1, s_z-1) \times f_p(r,R) \times p \quad (5)$$

图 5(d)所示是对长方体模型的半边进行缩小的结果.

前面我们分别讨论了针对平移、旋转和缩放的约束变形的计算.这 3 种变形可以各自独立进行,也可以相互复合以得到用户所需要的变形效果.实际上,可以将各种变换都统一表达为矩阵的形式.这样,如果要进行复合变形,即在一次变形操作中同时包含不同类型的变形,则可以通过先计算变形的复合矩阵来提高计算的效率.

2.3 算法的加速

前面我们在计算测地距离时,针对约束源计算了所有顶点的测地距离.实际上,由于约束变形具有局部性,模型上在有效影响范围之外的点将不受变形影响,所以,可以考虑只计算局部范围(变形的有效范围)的测地距

离.我们采用以下方法来实现测地距离的局部计算:首先由用户在模型中指定变形的边界,在计算中就将该边界作为测地距离的计算边界.如果在计算过程中,算法传递到用户指定的边界,则不再向外传递.这个加速的实现只需在测地距离算法的第 15 步加上适当的判断条件即可.从表 1 中给出的数据可以知道,使用局部测地距离的计算方法可以极大地减少计算所需的时间,从而加速变形的过程.

Table 1 Deformation time

表 1 变形时间

Model	Times of deformation		Triangle number	Time cost for calculating geodesic distance (s)		Time cost for deformation (s)
				Global calculation	Local calculation	
Dragon	1		118 732	0.609	0.062	0.031
Bunny	Stretch	1	69 451	0.359	0.015	0.015
	Bend	1				0.016
	Twist	1				0.015
Hand	7	Thumb rotation 1	12 750	0.062	0.016	0.000
		Thumb rotation 2				0.000
		Forefinger rotation 1		0.062	0.000	0.000
		Forefinger rotation 2				0.000
		Forefinger rotation 3				0.000
		Forefinger displacement 1		0.063	0.000	0.000
		Forefinger displacement 2		0.063	0.000	0.000

注:表 1 中时间数据为 0 的地方是由于程序无法检测到时间的消耗.对兔子模型的 3 种变形都采用了同一约束源和约束半径,所以只给出一组测地距离的计算时间.在对手指的变形过程中,如果几次变形采用了同一约束源,则对测地距离的计算只需做 1 次.

3 实验结果

我们在奔腾 IV 2.4GHz,512M 内存的 PC 机上实现了基于测地距离的多边形网格模型的约束变形方法.实验表明,当模型的三角片数目在 10 万个以下时,系统基本上能达到实时的效果.使用局部计算来加速则速度更快.图 6(d)所示是对龙模型的上颚进行局部变形的结果,在图 6(b)中显示出约束源为点源,位于龙模型的鼻尖上,在图 6(c)中用深色标出了约束的有效范围.

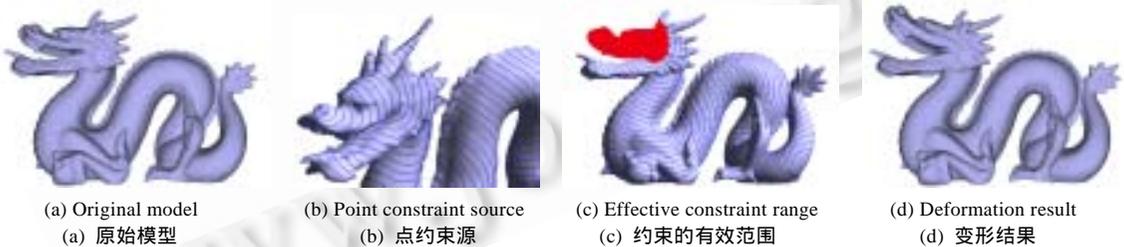


Fig.6 Constrained deformation results for the Dragon model

图 6 对龙模型的局部约束变形结果

图 7 所示是对兔子模型进行约束变形的结果,其中,图 7(a)所示是原始模型,图 7(b)显示了约束源的位置及约束的有效范围,图 7(c)~图 7(e)所示是对兔子模型的左耳分别进行拉伸、弯曲和扭曲的结果.从图中可以看出,左耳的变形对右耳不产生影响.图 8 所示是将一个手模型的大拇指和食指进行弯曲变形的结果.虽然可以很简单地对大拇指和食指各做一次旋转让其弯曲,但是,为了能够很好地表现出指关节的存在,实际上我们对手指的各指节所采用的旋转角度是不同的.图 8(b)所示是对原始模型经过 7 次变形才达到的效果.其中,对拇指取同一约束源(约束源为点源,在大拇指的指尖上)、不同有效范围和旋转角度做了 2 次旋转变形;对食指取同一约束源(约束源为点源,在食指的指尖上)、不同有效范围和旋转角度做了 3 次旋转变形;另外,又对食指中节的上下两面(取约束源在各面的中心)各做了 1 次极小偏移量的平移操作,以避免整个食指中节呈弧状弯曲.从图 8(b)中可以

看出,对大拇指和食指的变形并没有影响到其余的手指,这在其他的基于欧氏距离的变形方法中是很难达到的.在表 1 中统计了对各模型进行变形和计算测地距离所花费的时间.

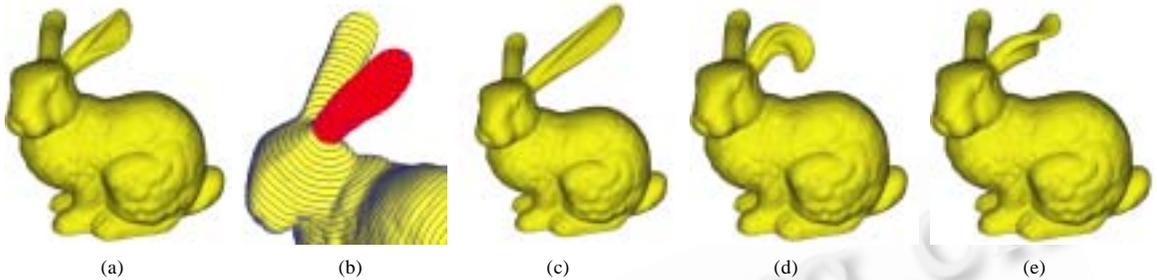


Fig.7 Constrained deformation results for the Bunny model

图 7 对兔子模型的局部约束变形结果

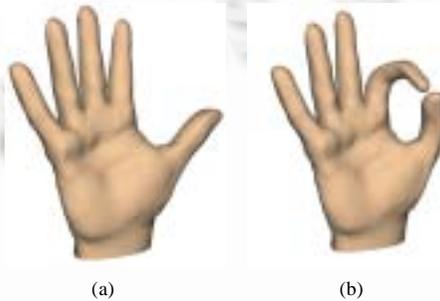


Fig.8 Constrained deformation result for the hand

图 8 对手模型的局部约束变形结果

4 结 论

本文介绍了利用测地距离对多边形网格模型进行约束变形的办法.给定约束源,我们首先计算网格上的点到约束源的测地距离;然后根据定义的变形模型确定网格上的点的变形权值,这里所使用的变形模型是基于测地距离的.在变形过程中,由于约束变形具有局部性质,我们可以使用局部计算测地距离的方法来加速计算的过程.我们计算测地距离的方法是在文献[21]的基础上进行扩展的,改进后的算法能够处理一些原始算法所不能解决的问题,如计算过程中出现边界点和鞍点的问题、多源问题等.我们使用的测地距离计算方法是一种近似计算方法,在保证一定精确性的基础上可以达到较快的速度.

我们的约束变形模型包括了偏移、旋转和缩放等各种情形,变形结果非常直观.在我们完成的系统中,用户可以直接用鼠标在模型上选择变形的约束源、输入约束的有效半径及变形目标,系统自动产生变形的结果.实验结果表明,利用这种方法进行变形能够很好地避免经典变形方法在非变形区产生的扭曲现象.

虽然约束变形的办法本身与物体表示无关,但由于我们所采用的测地线的计算是基于网格的,所以变形对象受到了限制.在下一步的工作中,我们将尝试把该变形模型扩展到其他类型的曲面(如参数曲面)上.

References:

- [1] Sederberg TW, Parry SR. Free-Form deformation of solid geometric models. Computer Graphics, 1986,20(4):151-160.
- [2] Hsu WM, Hughes JF, Kaufman H. Direct manipulation of free-form deformations. Computer Graphics, 1992,26(2):177-184.
- [3] Feng JQ, Zhao YH, Wan HG, Guo JM, Jin XG, Peng QS. Antialiasing of polygonal object deformation controlled by curves and surfaces. Chinese Journal of Computers, 2005,28(1):60-67 (in Chinese with English abstract).

- [4] Llamas I, Kim B, Gargus J, Rossignac J, Shaw CD. Twister: A space-warp operator for the two-handed editing of 3D shapes. *ACM Trans. on Graphics*, 2003,22(3):663–668.
- [5] Llamas I, Powell A, Rossignac J, Shaw CD. Bender: A virtual ribbon for deforming 3D shapes in biomedical and styling applications. In: Kobbelt L, Shapiro V, eds. *Proc. of the 2005 ACM Symp. on Solid and Physical Modeling*. New York: ACM Press, 2005. 89–99.
- [6] Sorkine O, Cohen-OR D, Lipman Y, Alexa M, Ross C, Seidel HP. Laplacian surface editing. In: Scopigno R, Zorin D, eds. *Proc. of the Eurographics Symp. on Geometry Processing*. Aire-la-Ville: Eurographics Association, 2004. 179–188.
- [7] Ohtake Y, Kanai T, Kase K. A Laplacian based approach for free-form deformation of sparse low-degree implicit surfaces. In: *Proc. of the IEEE Int'l Conf. on Shape Modeling and Applications*. Washington: IEEE Computer Society Press, 2006.
- [8] Borrel P, Rappoport A. Simple constrained deformations for geometric modeling and interactive design. *ACM Trans. on Graphics*, 1994,13(2):137–155.
- [9] Jin XG, Li YF, Peng QS. General constrained deformations based on generalized metaballs. *Computers & Graphics*, 2000,24(3): 219–231.
- [10] Zhang H, Sun JG. Patch constrained deformation technique. *Journal of Computer-Aided Design & Computer Graphics*, 2003,15(4): 389–392 (in Chinese with English abstract).
- [11] Zhou Y, Jin XG, Feng JQ. Constrained deformations based on convolution surfaces. *Journal of Computational Information Systems*, 2006,2(2):513–520.
- [12] Chen J, Han Y. Shortest paths on a polyhedron, part I: Computing shortest paths. *Int'l Journal Computational Geometry and Application*, 1996,6(2):127–144.
- [13] Kaneva B, O'Rourke J. An implementation of Chen & Han's shortest paths algorithm. In: Bremner D, ed. *Proc. of the 12th Canadian Conf. on Computational Geometry*. Fredericton: University of New Brunswick, 2000. 139–146.
- [14] Kapoor S. Efficient computation of geodesic shortest paths. In: *Proc. of the 31st Annual ACM Symp. on Theory Computation*. New York: ACM Press, 1999. 770–779.
- [15] Xiao CX, Feng JQ, Miao YW, Zheng WT, Peng QS. Geodesic path computation and region decomposition of point-based surface based on level set method. *Chinese Journal of Computers*, 2005,28(2):250–258 (in Chinese with English abstract).
- [16] Lanthier M, Maheshwari A, Sack JR. Approximating weighted shortest paths on polyhedral surfaces. In: *Proc. of the 13th Annual ACM Symp. on Computational Geometry*. New York: ACM Press, 1997. 274–283.
- [17] Kanai T, Suzuki H. Approximate shortest path on a polyhedral surface and its applications. *Computer-Aided Design*, 2001,33(11): 801–811.
- [18] Martinez D, Velho L, Carvalho PC. Geodesic paths on triangular meshes. In: *Proc. of the SIBGRAPI 2004*. Washington: IEEE Computer Society, 2004. 210–217.
- [19] Wang CCL, Wang Y, Tang K, Yuen MMF. Reduce the stretch in surface flattening by finding cutting paths to the surface boundary. *Computer-Aided Design*, 2004,36(8):665–677.
- [20] Kimmel R, Sethian JA. Computing geodesic paths on manifolds. *Proc. of the National Academy of Sciences*, 1998,95(15): 8431–8435.
- [21] Novotni M, Klein R. Computing geodesic distances on triangular meshes. In: Skala V, ed. *Proc. of the 10th Int'l Conf. in Central Europe on Computer Graphics, Visualization and Computer Vision*. Plzen: University of West Bohemia, 2002. 341–347.
- [22] Surazhsky V, Surazhsky T, Kirsanov D, Gortler SJ, Hoppe H. Fast exact and approximate geodesics on meshes. *ACM Trans. on Graphics*, 2005,24(3):553–560.
- [23] Nishita T, Nakamae E. A method for displaying metaballs by using Bezier clipping. *Computer Graphics Forum*, 1994,13(3): 271–280.
- [24] Akenine-Moller T, Haines E. *Real-Time Rendering*. 2nd ed, Wellesley: A K Peters Ltd., 2002.

附中文参考文献:

- [3] 冯结青,赵豫红,万华根,郭建民,金小刚,彭群生.基于曲线和曲面控制的多边形物体变形反走样. *计算机学报*,2005,28(1):60–67.
- [10] 张慧,孙家广.一种面片约束的变形技术. *计算机辅助设计与图形学学报*,2003,15(4):389–392.

[15] 肖春霞,冯结青,缪永伟,郑文庭,彭群生.基于 Level Set 方法的点采样曲面测地线计算及区域分解.计算机学报,2005,28(2): 250-258.



周艳(1970 -),女,浙江乐清人,硕士.主要研究领域为计算机动画,空间变形技术.



冯结青(1970 -),男,博士,研究员,博士生导师.CCF 高级会员,主要研究领域为计算机图形学,几何造型,科学数据可视化.



刘圣军(1979 -),男,博士生,主要研究领域为计算机动画,隐式曲面建模.



叶晓丰(1973 -),男,硕士,主要研究领域为计算机动画,多媒体技术.



金小刚(1969 -),男,博士,研究员,博士生导师,主要研究领域为计算机动画,曲面造型,纹理设计,空间变形技术.

2007 中国计算机大会征文通知

2007 China National Computer Conference (CNCC 2007)
2007 年 10 月 18-20 日, 苏州
<http://ccf.org.cn/cncc2007>

主办: 中国计算机学会
苏州市人民政府
承办: 苏州市科学技术协会

2007 中国计算机大会(2007 China National Computer Conference, CNCC 2007)将于 2007 年 10 月 18 日~20 日在苏州举行。它将为我国计算机界提供一个交流最新研究成果的舞台。CNCC 2007 是继 CNCC 2003,CNCC 2005 和 CNCC 2006 之后的中国计算机界又一次盛会。

议题内容 (但不限于此):

- | | | | | | |
|---------|----------|-------|--------|-------|---------|
| 高性能计算机 | 高性能计算机评测 | 传感器网络 | 嵌入式系统 | 对等计算 | 生物信息学 |
| 网络计算 | 网络存储系统 | 编译系统 | 虚拟现实 | 多核处理器 | 人工智能 |
| 理论计算机科学 | 软件工程 | 多媒体技术 | 信息安全技术 | 普适计算 | 数据库技术 |
| 搜索引擎技术 | 图形学与人机交互 | 中文处理 | 互联网络 | 模式识别 | 计算机应用技术 |

征稿截止: 2007 年 7 月 30 日

论文处理结果通知: 2007 年 8 月 30 日