

---

---

# 基于侧影轮廓的网格模型交互式编辑\*

南亮亮<sup>1</sup>, 刘伟军<sup>2</sup>

<sup>1</sup>(中国科学院深圳先进技术研究院 先进计算与数字工程研究所 可视计算中心 深圳 518055)

<sup>2</sup>(中国科学院沈阳自动化研究所 现代装备研究室 沈阳 110016)

## Silhouette-based Interactive Editing for Mesh Models

NAN Liang-Liang<sup>1</sup>, LIU Wei-Jun<sup>2</sup>

<sup>1</sup>(Visual Computing Center, Institute of Advanced Computing and Digital Engineering, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China)

<sup>2</sup>(Advanced Equipment Research and Design Center, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China)

+ Corresponding author: Phn: +86-755-8639-2391, E-mail: liangliang.nan@gmail.com

Received 2010-00-00; Accepted 2010-00-00

**Abstract:** A silhouette based manipulation method was proposed to improve the interactivity in mesh editing. The editing process contains two stages. First, the user selects the area to be edited and then the model is deformed by modifying the silhouette of the selected region. During the editing process, the boundary vertices of the selected region are fixed and the positions of the silhouette vertices are known beforehand via direct manipulation. These can be used as constraints for the editing computation. The objective function is the difference between the differential property before and after editing, with the above constraints, which has to be minimized to reconstruct the region to be deformed by solving a sparse linear system. Experimental results show that the method is straightforward and intuitive. Moreover, the most important property is that the local details of the mesh are preserved.

**Key words:** geometric modeling; differential properties; silhouette; interactive editing; mesh models

**摘要:** 本文提出了一种基于侧影轮廓修改的交互式编辑方法。首先指定待编辑区域, 然后通过交互式修改模型的侧影轮廓曲线实现模型的变形。算法以待编辑区域边界顶点作为固定顶点, 以修改后的侧影轮廓曲线顶点作为约束, 将变形前和变形后网格模型的微分属性之差最小作为目标函数, 通过求解一个稀疏的线性方程组恢复变形后的网格模型。实验表明, 该方法交互过程简单直观, 是一种有效的网格模型编辑算法, 并且变形后模型的局部特征得以保持。

**关键词:** 几何造型; 微分属性; 侧影轮廓; 交互式编辑; 网格模型

中图法分类号: TP391 文献标识码: A

---

\*Supported by the Natural Science Foundation of China under Grant No. 60902104(国家自然科学基金 60902104); National High-tech R&D Program of China under Grant No. 2009AA01Z302(国家高技术研究发展计划 2009AA01Z302).

**作者简介:** 南亮亮(1981—), 男, 陕西渭南人, 博士, 助理研究员, 主要研究领域为基于点云的城市场景建模, CAD/CAM等; 刘伟军(1969—), 男, 博士, 研究员, 博士生导师, 主要研究领域为逆向工程, CAD/CAM。

多边形网格模型是几何造型、科学计算可视化和计算机动画等领域三维几何形体流形的表示方法，网格模型的形状编辑是几何造型和计算机图形学研究的热点问题之一。Zorin 等人采用细分曲面的表示方法，开发了第一个网格模型多分辨率编辑系统<sup>[1]</sup>，随后有大批学者针对网格模型的编辑进行了研究。文献[2, 3]的工作都是对 Zorin 方法的改进，其局限主要在于当变形程度较大时，模型的细节可能发生扭曲。

近几年出现了许多新的变形编辑的方法。Yu 等人提出了网格曲面的梯度编辑方法<sup>[4]</sup>，该方法通过操作网格梯度场，将梯度信息还原到空间网格上实现曲面形状的改变。该方法具有变形直观的优点，但需要求解一个离散泊松方程来重建编辑后的曲面。Sorkine 等将三角网格上的 Laplacian 坐标定义为网格顶点邻域质心到该顶点的向量，提出了基于 Laplacian 坐标的网格曲面编辑方法<sup>[5]</sup>，该方法以变形前后网格顶点 Laplacian 坐标之差最小作为约束条件，建立最小二乘目标函数，通过求解一个稀疏的线性方程组实现网格的变形、曲面移植和几何细节迁移等操作。Lipman 提出的网格模型的线形旋转不变坐标的表示方法<sup>[6,7]</sup>，很好的解决了 Laplacian 坐标旋转的问题。但该方法需要求解两个线性方程组才能重建变形后的网格模型。并且，在变形量较大的情况下，模型将会出现一定程度的扭曲<sup>[8]</sup>。周艳等人提出一种基于测地距离的网格模型约束变形方法<sup>[9]</sup>，根据模型上各顶点到约束源的测地距离定义各点的变形权值，实现了模型的平移、旋转和缩放等多种变形方式，避免了模型的扭曲现象。然而测地距离的计算较为复杂，并且在变形过程中各点到约束源的测地距离是在不断变化的，因此对于多步变形需要多次计算测地距离，较为耗时，不适用于交互式变形。

上述方法都是通过选择模型部分顶点作为“手柄”，通过操纵该手柄实现模型的变形。文献[10, 11]的方法都是勾画式的编辑，变形灵活。然而这两种方法对勾画出的轮廓缺乏编辑环节，交互性有待提高。文献[12]给出了一种勾画式的自由曲面造型系统，其交互过程首先勾画一组三维曲线作为控制曲面形状的基础，然后通过一系列编辑操作和优化生成复杂的几何形体。该系统可用于拓扑复杂的几何模型的构建，然而对构建出的模型，其交互式变形能力有限。曲面变形的目的是为用户提供直观且高效的编辑工具，实现复杂几何模型的构造，因此所采用的交互方式必须直接和灵活。本文利用网格模型微分属性近似表示所具有的平移不变的特点，给出了一种基于侧影轮廓修改的网格模型交互式变形编辑方法。实验证明，该方法提高了变形的交互性和可控性，并且模型的细节能够得以保持。

本文第 1 节详细讨论网格模型的微分属性表示。第 2 节介绍网格模型上侧影轮廓的计算和本文变形方法的实现。第 3 节给出一些实验结果，并进行讨论。第 4 节总结全文。

## 1 网格模型的微分属性表示

### 1.1 问题的数学描述

对于三角网格模型  $M = (V, E)$ ，其中  $V = \{v_i\}$  为模型顶点坐标， $E = \{(i, j) | i \neq j\}$ ， $i, j \in \{0, 1, \dots, n-1\}$  表示模型的边， $n$  为模型的顶点个数。网格模型的变形编辑是指对模型  $M$  进行某种变换，使得模型变为  $M'$ ， $M'$  和  $M$  具有相同的邻接关系，即  $E(M) = E(M')$ 。对于变形前后模型的顶点， $V'$  和  $V$  可以完全不同，此时模型发生整体变形；也可以部分不同，此时模型发生局部变形。

### 1.2 网格模型的微分属性表示

曲面与电场、引力场和流场等物理对象具有某些相同的特性，因此在微分几何中，曲面以场的形式加以描述。Laplacian 方程

$$\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} + \frac{\partial^2 \varphi}{\partial z^2} = 0$$

以势函数的形式描述了上述物理对象的性质。对于连续曲面  $r = r(u, v)$ ，考虑 Laplacian 算子<sup>∇<sup>2</sup></sup>

$$\nabla^2 r = \frac{\partial^2 r}{\partial u^2} + \frac{\partial^2 r}{\partial v^2},$$

$\nabla^2$ 是一个二阶微分算子。如图1所示,  $r_{ij}$  是曲面上一点,  $r_{i-1,j}$  和  $r_{i+1,j}$ ,  $r_{i,j-1}$  和  $r_{i,j+1}$  分别是  $r_{ij}$  在曲纹坐标网  $u$  向和  $v$  向的相邻点, 称为曲面上点  $r_{ij}$  的1-邻域点。  $r_{ij}$  处的二阶偏导数可由有限差分近似表示为

$$\begin{cases} \frac{\partial^2 r(u,v)}{\partial u^2} \approx \frac{1}{\Delta u^2} (r_{i-1,j} - 2r_{ij} + r_{i+1,j}) \\ \frac{\partial^2 r(u,v)}{\partial v^2} \approx \frac{1}{\Delta v^2} (r_{i,j-1} - 2r_{ij} + r_{i,j+1}) \end{cases} .$$

$\nabla^2 r$  的误差取决于  $o(\Delta u^2)$  和  $o(\Delta v^2)$  中的较大者, 取  $\Delta u = \Delta v = \delta$ , 可以得到Laplacian算子  $\nabla^2$  的有限差分逼近表示形式

$$\nabla^2 r \approx \frac{1}{\delta^2} (r_{i-1,j} + r_{i+1,j} + r_{i,j-1} + r_{i,j+1} - 4r_{ij}) .$$

令  $N_i = \{j | (i,j) \in E\}$  表示曲面上一点  $v_i$  的一阶邻域点的索引, 上式可以表示为如下形式

$$\nabla^2 r \approx \frac{1}{\delta^2} \sum_{j \in N_i} (v_j - v_i) . \quad (1)$$

这是连续曲面的Laplacian算子, 从(1)式可以看出, 曲面的Laplacian算子仅仅取决于曲面上各点邻域的形状, 揭示的是曲面的局部几何属性。

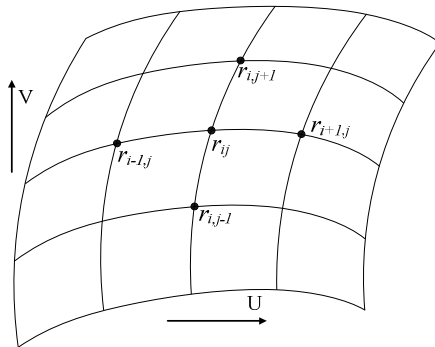


Fig.1 The mesh coordinate on continuous surfaces  
图1 连续曲面的曲纹坐标网

网格模型是连续曲面的离散表示形式, 我们可以将连续曲面的Laplacian算子近似用于三角网格模型。对于离散的三角网格曲面, 顶点  $v_i$  处的Laplacian算子定义为

$$L(v_i) = \sum_{j \in N_i} w_{ij} (v_i - v_j), \quad \sum_{j \in N_i} w_{ij} = 1 . \quad (2)$$

称  $\delta_i = L(v_i)$  为顶点  $v_i$  的Laplacian坐标, 本文将其所表示的向量称为Laplacian向量。  $L$  是一个二阶微分算子, 称  $\Delta(M) = \{\delta_i\}$  为网格模型  $M$  的微分属性表示, 用矩阵形式可以表示为  $\Delta = LV$ 。其中  $L = I - D^{-1}A$ , 称为网格模型  $M$  的Laplacian矩阵,  $I$  为  $n$  阶单位方阵,  $A = (a_{ij})$  是网格模型的邻接矩阵表示,  $D = (d_{ij})$  称为网格的度数矩阵, 是一个对角矩阵。其中

$$a_{ij} = \begin{cases} 1, (i,j) \in E \\ 0, (i,j) \notin E \end{cases}, \quad d_{ij} = \begin{cases} d_i, i = j \\ 0, i \neq j \end{cases} .$$

$d_i = |N_i|$ , 表示顶点  $v_i$  的度数, 即  $v_i$  的1-邻域顶点个数。  $w_{ij}$  是一种权值, 可以有多种选取方法, 特别地, 当取  $w_{ij} = 1/d_i$  时, 式(2)可以改写为

$$L(v_i) = v_i - \frac{1}{d_i} \sum_{j \in N_i} v_j \quad (3)$$

可以看出, 当采用这一特殊的权值时, Laplacian向量表示的是顶点的1-邻域中心指向该顶点的向量(图2)。

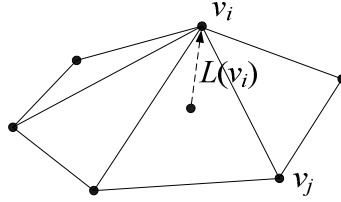


Fig.2 The definition of Laplacian coordinate  
图2 网格顶点的 Laplacian 坐标

从微分几何我们知道, 式(3)可以看作是对以下曲线积分的离散近似<sup>[13]</sup>

$$\frac{1}{|\gamma|} \int_{v \in \gamma} (v - v_i) dl(v),$$

其中  $\gamma$  是曲面上包围  $v_i$  的一条封闭曲线, 曲线的长度为  $|\gamma|$ 。对于曲率连续的曲面, 当  $\gamma$  向  $v_i$  收缩时, 上述积分收敛于  $v_i$  处的平均曲率  $k(v_i)$  和法向量  $n_i$  的乘积, 即

$$\lim_{|\gamma| \rightarrow 0} \frac{1}{|\gamma|} \int_{v \in \gamma} (v - v_i) dl(v) = k(v_i) n_i.$$

可见, 对于三角网格模型, Laplacian坐标所表示向量的方向是模型在该顶点处法向量的逼近, 其长度与该顶点处的曲面平均曲率成比例。由于曲面的法向量与曲面的一阶导数有关, 而平均曲率取决于曲面的二阶导数, 所以Laplacian坐标同时反映了曲面的一阶和二阶微分性质, 它描述的是曲面的内蕴几何属性。

## 2 本文方法及实现

为了提高变形的交互性, 我们首先修改待编辑区域的侧影轮廓曲线, 以变形后的侧影轮廓曲线作为约束对网格中的感兴趣区域进行编辑。

### 2.1 侧影轮廓定义与计算

对于处在三维空间中特定姿态的任意光滑曲面, 如图3所示, 其侧影轮廓定义为满足  $\mathbf{n}(p) \cdot \mathbf{v} = 0$  的曲面上点的集合, 其中  $\mathbf{n}(p)$  为曲面上点  $p$  处的法向量,  $\mathbf{v}$  为曲面处在该姿态时的视线方向。

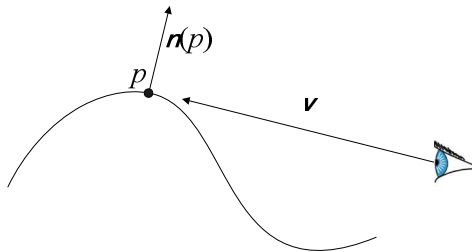


Fig.3 The definition of silhouette  
图3 侧影轮廓的定义

对于网格模型, 侧影轮廓可以近似看作是一系列边的集合, 这些边所关联的两个多边形面片中的一个可见的, 另一个是不可见的。因此我们可以通过一条边相关联的两个三角形的法向和视线方向的关系来判断其是否在侧影轮廓上。对于模型上的任意一条边  $e$ , 它所关联两个三角形的法向量分别为  $\mathbf{n}_1$  和  $\mathbf{n}_2$ , 如果  $\langle \mathbf{n}_1, \mathbf{v} \rangle \cdot \langle \mathbf{n}_2, \mathbf{v} \rangle < 0$ , 则  $e$  在侧影轮廓上。这是一种近似的计算方法, 然而对本文的变形问题, 侧影轮廓作为用

户交互的手段，变形后侧影轮廓上顶点的坐标仅作为变形的约束条件，我们寻求的变形后的模型是最小二乘意义上的解(见2.2节)。实验证明，通过这种近似计算得到的侧影轮廓已经能够很好的满足本文变形方法需要，而且计算效率高，更精确的侧影轮廓计算方法可参考文献[14, 15]。

## 2.2 变形方法实现

对于网格模型  $M$ ，我们希望寻找到某种几何量  $\Phi$ ，使得模型在变形前和变形后该几何量保持不变，即  $\Phi' = \Phi$ 。同时，为了计算简单，要求该几何量和顶点坐标具有线性关系，即  $\Phi = LV$ 。如果能找到这样的几何量，就可以通过求解如下方程组计算出变形后网格模型各顶点的坐标：

$$LV' = LV \quad (4)$$

在2.0节中，我们指出网格模型的Laplacian坐标表示同时反映了曲面的一阶和二阶微分属性，它描述的是曲面的内蕴几何属性。由于其具有平移不变的特性，同时该量又和模型的顶点坐标具有线性关系，因此可以近似作为我们所要寻找的几何不变量。

s网格模型可以看作是一幅图，如果模型含有  $k$  个连通分量，那么矩阵  $L$  的秩为  $n-k$ 。对于一般待编辑模型，其连通分量为1，因此  $L$  的秩为  $n-1$ 。可见， $L$  是奇异的，无法通过求解(4)式获得变形模型各顶点的坐标，必须引入至少一个固定顶点或已知顶点作为约束方程组才可求解。编辑过程中，我们希望待编辑区域的边界固定，以保证编辑区域和未编辑区域边界的光滑过渡，这正好可以解决  $L$  矩阵奇异的问题。对于本文算法，模型形状的改变是以变形后的侧影轮廓作为约束的。变形后侧影轮廓上顶点的坐标是已知的，这些顶点可以看作已知顶点，作为约束条件引入。设变形后已知顶点为  $V_j' = \{v_i\}$ ， $i \in \{m, \dots, n\}$ ， $m < n$ ，可以通过下式求解出经过变形编辑后的模型  $M'$  上各顶点的坐标

$$\begin{cases} LV' = \Phi' \\ v_i = v_i, i \in \{m, \dots, n\}, m < n \end{cases} \circ$$

上式可以记为  $Ax = b$ ，这是一个超定的线性方程组，我们计算它在最小二乘意义下的解，即使得  $\|Ax - b\|$  取最小值的  $x$  向量。于是问题转化为一个可解的线性最小二乘问题，简化后待求解的方程组为  $(A^T A)x = A^T b$ ，记  $A' = A^T A$ ， $b' = A^T b$ ，上式改写为

$$A'x = b' \quad (5)$$

交互式编辑必须满足实时性的要求。通常情况下，待编辑区域顶点个数可能多达上万甚至更多，因此需要快速稳定的求解方法。不难证明，矩阵  $A'$  是对称且正定的，因此可以先对其进行Cholesky分解<sup>[16]</sup>，即  $A' = CC^T$ ， $C$  是一个对角线元素为正的下三角矩阵，具有如下形式

$$C = \begin{pmatrix} c_{11} & & & & \\ c_{21} & c_{22} & & & 0 \\ c_{31} & c_{32} & c_{33} & & \\ \vdots & & & \ddots & \\ c_{N1} & c_{N2} & c_{N3} & \cdots & c_{NN} \end{pmatrix} \circ$$

其中  $c_{ii} = \sqrt{a'_{ii} - \sum_{k=0}^{i-1} c_{ik}^2}$ ， $c_{ji} = \frac{1}{c_{ii}}(a'_{ij} - \sum_{k=0}^{i-1} c_{ik}c_{jk})$ ， $j = i+1, i+2, \dots, N-1$ ， $N$  为(5)式所表示方程组的行数。如果按照  $i=0, 1, \dots, N-1$  顺序依次进行求解，就会确定右端出现的  $C$  的元素。一旦系数矩阵被分解，可以通过回代求解该方程组，获得网格模型各顶点的坐标。

## 2.3 用户界面

编辑过程分为以下几个步骤：

- 1) 选择待编辑区域，同时选择一系列固定顶点。本文将待编辑区域的边界顶点作为固定顶点；
- 2) 修改待编辑区域的侧影轮廓，然后以2.2节中的方法使模型发生变形。侧影轮廓上的顶点必须是待编辑

区域顶点的子集，作为用户鼠标交互的对象。

### 3 实验结果与讨论

在本节，我们提供若干实验结果，证明本文算法的有效性，并对影响编辑效果的若干因素进行分析和讨论。

利用OpenGL图形库，我们实现了本文提出的交互式变形算法。算法在Windows XP操作系统上使用C++在Visual Studio 2005开发环境中实现，所有实验均在一台中央处理器为AMD Athlon 3000+ 1.8GHz，内存为1GB的微型计算机上完成。对若干具有代表性的三角网格模型进行编辑，取得了预期的效果(图4)。

#### 3.1 变形效果和交互性比较

与文献[10, 11]的变形效果类似，本文的方法也能够保持模型的局部几何细节信息。然而文献[10, 11]所采用的勾画式的编辑方法缺乏对勾画出的轮廓进行编辑的环节，每当进行完一次变形后必须重新勾画出新的轮廓才可再次进行变形。相比之下，本文方法可以对侧影轮廓进行多次修改，交互性更加灵活。

#### 3.2 网格模型质量对编辑的影响

网格模型的微分属性表示是建立在网格模型是连续曲面模型逼近的基础上的，离散Laplacian坐标同时反映了网格曲面上各顶点处的一阶微分和二阶微分属性，因此要求待编辑的模型必须拓扑正确，且是光滑的。对于这一类模型，微分属性表示能够很好的反映模型的局部几何形状特征，变形效果理想，如图4所示。某些三角网格模型并不是严格正则的，存在错误的连接，这类模型必须经过预处理，包括拓扑错误的修正、孔洞的修补和光滑等。另外一类模型虽然拓扑正确，但模型上各三角面片的边长度差异很大，这种情况下，Laplacian坐标无法反应真实曲面模型的几何形状特征，变形效果不理想。对于这一类模型，可以首先经过重新网格化改善其质量，这方面已有很多研究成果，请参考文献[17-19]。

#### 3.3 方程组的求解效率

对于待编辑模型，模型上各顶点的邻接关系已知，虽然系数矩阵的规模可能很大，但由于常见三角网格模型各顶点的1-邻域顶点个数一般不超过10，因此待求解的方程组(5)是一个稀疏的线性方程组，这为本文算法的快速求解提供了可能。另一方面，在构建线性方程组时，只需考虑待编辑区域的顶点，这样方程的规模可大大减小。同时，对于特定区域的编辑，系数矩阵在交互前进行分解，在交互变形过程中只需通过回代求解方程组即可。实验表明，通过回代求解的速度很快，能够满足交互变形实时性的要求。表1给出了图4中三个模型变形过程中求解方程组所耗费的时间。

**Table 1** The running times of solving the linear systems for the different models in Fig 4

**表 1** 图 4 中各模型对应线性方程组的求解时间

模型	顶点数	求解时间/(s)
小鹿(图 4-a)	181	0.012
金鱼(图 4-b)	529	0.038
人头(图 4-c)	757	0.067

注：所列时间为 5 次求解的平均时间

#### 3.4 拓扑保持特性

本文的变形方法是一种拓扑保持的编辑方法，在变形过程中，模型各顶点的邻接关系保持不变，因此该方法适用于对已有几何形体进行局部调整和修改。实验中我们发现，在对模型局部区域进行大距离的变形后，编辑区域的三角面片不再均匀，此时微分属性的近似表示失效，导致变形区域发生一定的扭曲。网格的不均匀可以通过重新网格化加以改善，然而已有的重新网格化算法都比较耗时，满足不了交互的要求。另一方面，重新网格化后模型的邻接关系将发生改变，先前的线性方程组需要重新构建，系数矩阵需要重新分解，这严

重影响了变形的实时性。因此，重新网格化的方法不适于本文的交互式编辑。

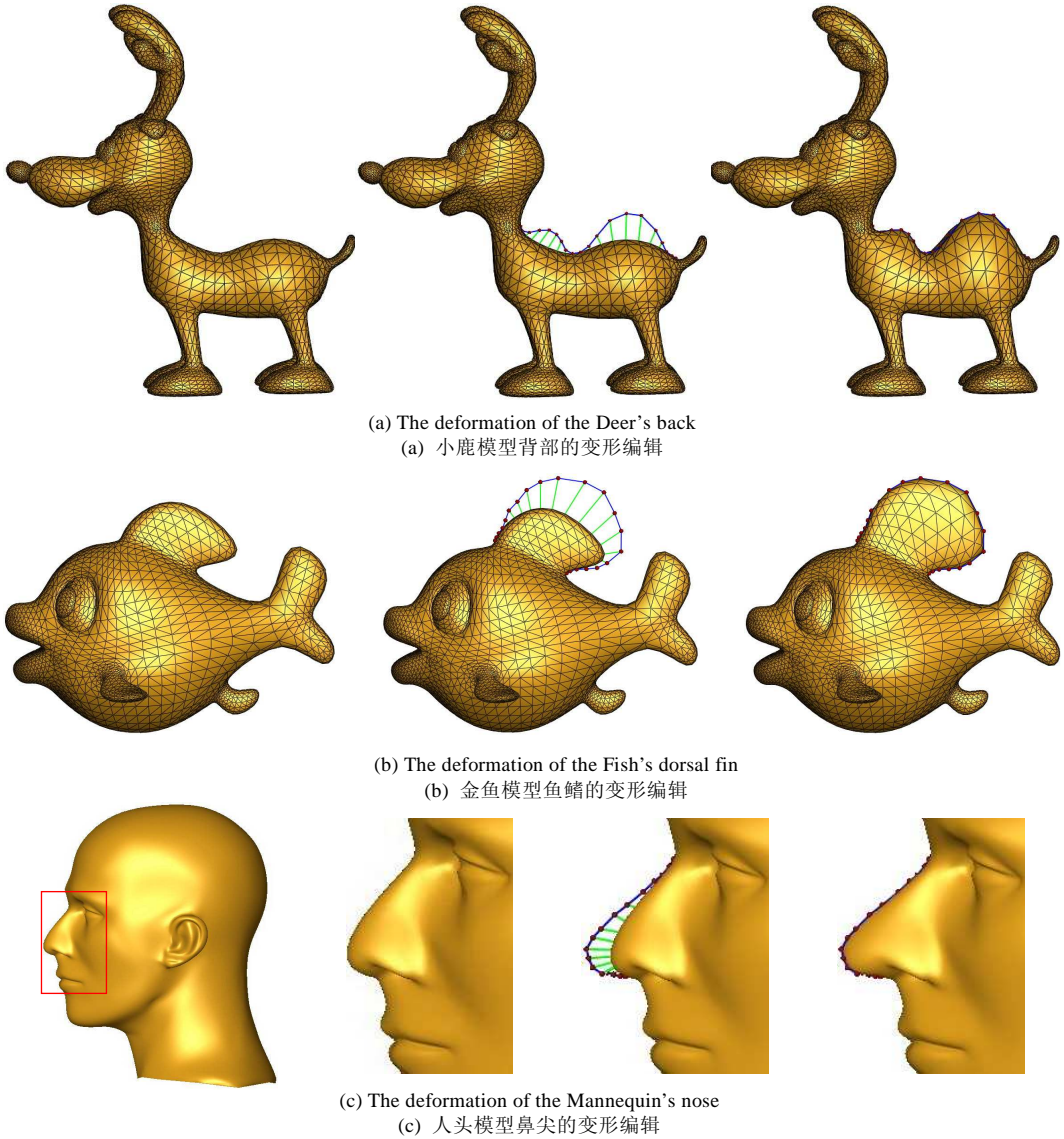


Fig.4 The interactive deformation of several mesh models

图4 若干模型的变形效果

#### 4 结论与展望

本文讨论了网格模型的交互式变形编辑，提出了一种基于侧影轮廓修改的变形方法。采用网格模型的微分属性表示，通过交互式的修改模型的侧影轮廓曲线，从模型的微分属性恢复变形后的顶点坐标。实验证明，该方法是一种有效直观的编辑方法，能够保持模型的细节特征，既适用于局部变形，也适用于大面积变形。经过本文变形方法编辑获得的模型可以作为STL模型输出，用于快速原型加工，也可以作为通用CAD系统的输入进行更复杂的编辑造型。本文的方法还有以下两方面值得研究：(1) 为了提高求解效率，充分发挥图形处理硬件的功能，研究利用GPU求解本文算法所涉及的线性稀疏方程组；(2) 编辑过程中，模型各部分(主要是固定部分和变形部分)之间可能发生碰撞并交叉现象，需要研究变形过程中的碰撞检测并加以避免。

**References:**

- [1] Zorin D, Schröder P, Sweldens W. Interactive multiresolution mesh editing. In: Proc. of SIGGRAPH 1997, 1997, 259-268.
- [2] Kobbelt L, Campagna S, Vorsatz J, Seidel HP. Interactive multi-resolution modeling on arbitrary meshes. In: Proc. of SIGGRAPH 1998, 105-114.
- [3] Guskov I, Sweldens W, Schröder P. Multiresolution signal processing for meshes. In: Proc. of SIGGRAPH 1999, 325-334.
- [4] Yu YZ, Zhou K, Xu D, Shi XH, Bao HJ, Guo BN, Shum HY. Mesh editing with Poisson-based gradient field manipulation. In: Proc. of SIGGRAPH 2004, 2004, 644-651.
- [5] Sorkine O, Cohen-Or D, Lipman Y, Alexa M, Rossl C, Seidel HP. Laplacian surface editing. In: Proc. of the Eurographics/ACM SIGGRAPH symposium on Geometry processing, 2004, 175-184.
- [6] Lipman Y, Sorkine O, Cohen-Or D, Levin D, Rössl C, Seidel HP. Differential coordinates for interactive mesh editing. In: Proc. of the Shape Modeling International 2004, 181-190.
- [7] Lipman Y, Sorkine O, Levin D, Cohen-Or D. Linear rotation-invariant coordinates for meshes. Proc. of SIGGRAPH 2005, 24(3): 479-487.
- [8] Lipman Y, Cohen-Or D, Gal R, Levin D. Volume and shape preservation via moving frame manipulation. ACM Trans. on Graphics, 2007, 26(1): 1-14.
- [9] 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 (in Chinese with English abstract).
- [10] Nealen A, Sorkine O, Alexa M, Cohen-Or D. A sketch-based interface for detail-preserving mesh editing. ACM Trans. on Graphics, 2005, 24(3): 1142-1147.
- [11] Wang J, Zhang HX, Xu D, Bao HJ. Sketch-based Poisson mesh editing. Journal of computer-aided design & computer graphics, 2006, 18(11): 1723-1729 (in Chinese with English abstract).
- [12] Nealen A, Igarashi T, Sorkine O, Alexa M. FiberMesh: designing freeform surfaces with 3D curves. ACM Trans. on Graphics, 2007, 26(3): 41.
- [13] Taubin G. A signal processing approach to fair surface design. In: Proc. of SIGGRAPH 1995, 351-358.
- [14] Hertzmann A, Zorin D. Introduction to 3D non-photorealistic rendering: Silhouettes and outlines. In: SIGGRAPH 1999 Course Notes, 1999.
- [15] Hertzmann A, Zorin D. Illustrating smooth surfaces. In: Proc. of SIGGRAPH 2000, 517-526.
- [16] Press WH, Teukolsky SA, Vetterling WT, Flannery BP. Numerical Recipes in C++. 2nd ed, Beijing: Publishing House of Electronics Industry, 2003.
- [17] Alliez P, Meyer M, Desbrun M. Interactive geometry remeshing. ACM Trans. On Graphics, 2002, 21(3): 347-354.
- [18] Alliez P, Cohen-Steiner D, Devillers O, Lévy B, Desbrun M. Anisotropic polygonal remeshing. Proc. of SIGGRAPH 2003, 22(3): 485-493.
- [19] Surazhsky V, Gotsman C. Explicit surface remeshing. In: Proc. of the Eurographics/ACM SIGGRAPH symposium on Geometry processing, 2003, 20-30.

**附中文参考文献:**

- [9] 周艳, 刘圣军, 金小刚, 冯结青, 叶晓丰. 基于测地距离的多边形网格模型约束变形. 软件学报, 2007, 18(6): 1543-1552.
- [11] 王隽, 张宏鑫, 许栋, 鲍虎军. 勾画式泊松网格编辑. 计算机辅助设计与图形学学报, 2006, 18(11): 1723-1729.