

虚拟试衣系统中的模型变形

杨晨晖*, 陈 辰

(厦门大学信息科学与技术学院, 福建 厦门 361005)

摘要: 在虚拟试衣系统中,所用的衣物模型都是预先制作好的三维网格模型.为了使虚拟衣服针对不同用户进行“量体裁衣”,就必须对其进行网格模型变形.提出了一种基于人体轮廓图驱动的三维模型变形方法来生成与用户体型相符的衣物模型.该方法包括对网格模型的编码、选择和移动锚点,求解能量优化问题,最终得到变形后的网格,并通过网格简化操作加速网格变形,使得模型变形算法效率大大提升,论证了将模型变形应用于虚拟试衣系统的可能性.

关键词: 虚拟试衣系统; 三维网格; 模型变形

中图分类号: TP 319

文献标志码: A

文章编号: 0438-0479(2014)01-0046-06

在虚拟试衣环境中,输入的数据都是通过摄像头等视频设备得到的二维图像或者图像序列.这些图像是仅有的包含了用户信息的数据,利用这些数据得到用户参数并生成个性化的三维人体模型的过程与基于图像建模的方法非常相似.基于图像的建模研究是使用二维图像生成或者修改三维模型,通过这样一种更自然的方式,使得三维建模的过程更加直观和高效.

根据建模过程中二维图像的不同种类,可以分为基于草图/手绘图的建模^[1]和基于单张/多张图片的建模.其中基于草图/手绘图的建模主要关注通过用户使用笔型输入设备以传统绘图的方式构建三维模型,例如文献[2]中提到的利用手绘笔画对模型进行局部调整的方法,该方法能够快捷自然地修改模型,但是此方法需要人工标注模型轮廓线,且必须事先定义影响区域(ROD),操作复杂;文献[3-4]中介绍的草图建模方法使用了手绘线条的方式生成复杂模型,但是要使生成的模型准确,则需要绘制大量不同角度的线条来控制生成过程.基于单张/多张图片的建模目的在于从图像中重构出物体的三维模型,文献[5]中提出了一种从单张图片中生成三维模型的方法,但是生成的模型细节程度并不高;文献[6]使用了高斯过程隐变量模型方法从一组不同姿势的同一种物体的三维模型中训练得到人体模型姿势和体型的参数,并用来合成新的人体

模型.

综合来说,基于草图的建模方法因其大量的交互操作并不能直接应用于虚拟试衣中,而基于图片的建模过程需要使用大量的同类模型作为训练集,而虚拟试衣中的人体模型是通过特殊的建模工具得到并绑定了衣物信息,难以得到同类的训练样本集.本文提出的变形算法通过均值坐标保存模型表面细节,使用人体轮廓图像与三维人体模型的匹配作为变形的约束,整个变形过程只需求解一个能量函数的最小二乘.无需任何人工交互,也不需要前期的预计算和学习过程.

1 虚拟试衣中的模型处理

在虚拟试衣系统中,我们使用工具制作身着衣物的人体模型,如图 1(a)所示.这个人体模型被映射上用户的体型以及动作并在物理仿真阶段用作支撑衣物模型,但在渲染时并不被送入管线,所以最后用户只会看见衣物,如图 1(b)所示.简单的说,人体模型就是用户的“替身”.为了使衣服能匹配用户体型,需要对人体模型进行变形,使其尽量能与用户体型对应.

2 三维模型变形方法流程

如图 2 所示,输入为原始三维网格模型 M 和二维轮廓图 C ,输出为新的网格 M' .期间经过了模型编码、对应关系寻找、模型简化、模型变形和顶点恢复等阶段.

收稿日期: 2013-07-16

基金项目: 福建省高校产学研合作重大项目(2012H6024)

* 通信作者: chyang@xmu.edu.cn

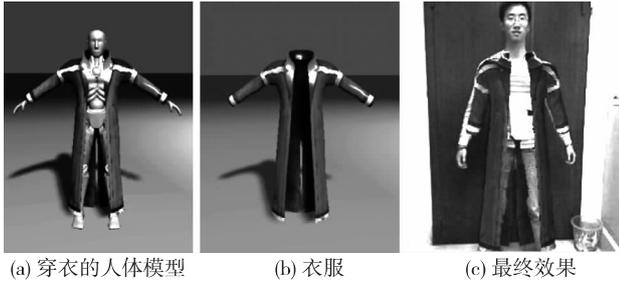


图 1 用户模型替换人体模型
Fig. 1 The user's body replace human model

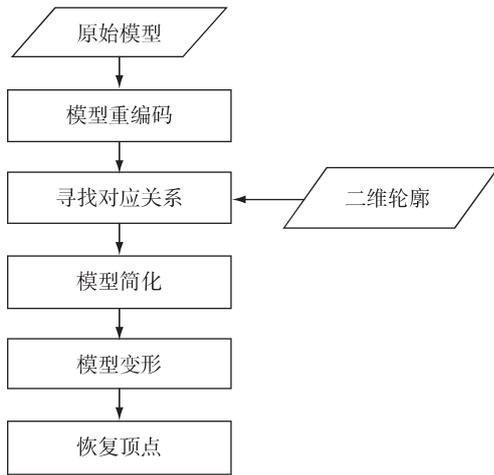


图 2 模型变形算法流程
Fig. 2 The process of model deformation

3 网格模型的编解码与轮廓点的匹配

3.1 网格模型编码与解码

本文使用一种基于均值坐标的形状编码对原始模型进行重新编码^[7].

锚点是网格上由用户指定的少数几个特殊顶点,通过移动锚点来控制整个网格的变形效果,如图 3.

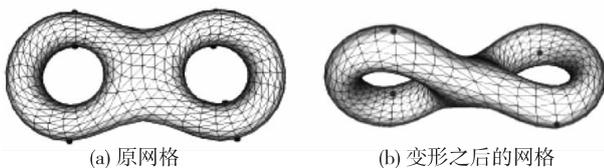


图 3 原网格与变形后的网格
Fig. 3 Original model and twisting model

网格 M 由顶点集 V 和边集 E 组成. 为了计算这个编码,使用这个顶点 v_i 和它的邻接顶点 v_{j_1}, \dots, v_{j_m}

建立一个局部坐标系. 其中 v_{j_1}, \dots, v_{j_m} 按顺时针排列. 定义一个投影平面 P_i :

$$P_i = n_x + n_y + n_z + d_i,$$

其中平面的法线为 $n_i = (n_x, n_y, n_z)$, d_i 表示了从这些顶点到局部坐标系原点的平均距离.

均值编码中,每个顶点的编码包括两部分:正切部分 w_{ij} 和法线部分 b_{ij} . 其中正切部分是基于顶点在投影平面投影点,法线部分基于顶点对投影平面的距离偏移.

给定了这个投影平面后,将顶点 v_i 和其邻接点 v_j 都投影到此平面上:

$$v'_i = v_i - (d_i + (v_i \cdot n_i))n_i,$$

$$v'_j = v_j - (d_j + (v_j \cdot n_j))n_j.$$

然后计算 v'_i 相对于 v'_j 的均值权重:

$$w'_{ij} = \frac{\tan(\frac{\gamma_{ij}}{2}) + \tan(\frac{\delta_{ij}}{2})}{l_{ij}},$$

最后得到

$$w_{ij} = \frac{w'_{ij}}{\sum_{(i,k) \in E} w'_{ik}},$$

其中 γ_{ij} 为角 $v'_j v'_i v'_{j+1}$, δ_{ij} 为角 $v'_j v'_i v'_{j-1}$, l_{ij} 为 $\|v'_j - v'_i\|$.

法线部分 b_{ij} 首先计算边 (v'_i, v'_j) 与法线 n_i 的余弦值得到:

$$c_{ij} = \frac{(v_i - v_j) \cdot n}{\|v_i - v_j\|}.$$

然后计算角度的余切值即该顶点编码的法线部分:

$$b_{ij} = \frac{c_{ij}}{\sqrt{1 - c_{ij}^2}}.$$

综上,得到了一个顶点均值编码的两部分: $\{w_{ij} | (i, j) \in E\}$ 和 $\{b_{ij} | (i, j) \in E\}$.

通过顶点 v_i 的均值编码和 v_i 所有邻接顶点 v_j 的坐标可以还原出 v_i 的位置. 根据文献^[7]中的推导, v_i 可由以下解码公式得到:

$$v_i = F_i(V) = \sum_{(i,j) \in E} w_{ij} (v_j + \|N_i\| \sum_{(i,k) \in E} w_{ik} (v_k - v_j) \|b_{ij} n_i),$$

其中 N_i 是一个 3×3 的矩阵:

$$N_i = I_3 - n_i n_i^T (I_3 \text{ 为单位矩阵}).$$

在给出了单个顶点的解码方法后,对于整个网格,可以通过设置一个能量函数,并求解非线性最小二乘来得到变形后的网格:

$$\operatorname{argmin}_V G(V') = \frac{1}{2} \sum_{v_i \in V} (v_i - F_i(V))^2,$$

其中 $V' = V \setminus V_a \cdot V_a$ 表示网格中锚点的集合.

以上阐述了如何为网格编码,并通过解码得到变

形后的网格. 变形过程由网格上的锚点来控制, 下面将说明如何得到锚点.

3.2 寻找轮廓点与三维模型匹配关系

在虚拟试衣系统中从外部输入设备提取用户人体轮廓作为人体三维模型变形的依据, 将原始的人体三维模型变形为与用户体型相符的人体模型, 为此必须将离散的二维轮廓点与三维网格模型的顶点对应起来, 将这些对应后的网格顶点作为锚点向轮廓点移动以带动三维人体模型整体变形, 如图 4. 本文使用基于隐马科夫链模型的笔画与模型匹配算法来建立人体轮廓与人体模型的对应关系.



图 4 轮廓点与模型顶点的对应关系

Fig 4 The correspondence of outline and model's vertex

3.2.1 局部相似性的量度

首先我们定义一些衡量的手段, 使用两个量来界定轮廓点与模型顶点的性质:

$$d_p = (p^x - v^x)^2 + (p^y - v^y)^2, d_N = n^p \cdot n^v,$$

其中 n^p 为轮廓点的法线, 因为轮廓点在二维图像空间中建立时并没有法线信息, 所以定义轮廓点的法线为 $n^p = ((p_i - p_{i-1}) + (p_{i+1} - p_i))^\perp$ (图 5). n^v 为三维模型顶点的法线. d_p 定义了一种轮廓点与模型顶点在距离上的量度, 而 d_N 从朝向上衡量点之间的相似性.

3.2.2 基于隐马科夫链模型的匹配

在给出了衡量轮廓与模型局部相似度方法之后, 我们的目标就是求出使这些度量值的和最小的匹配. 为了得到最优解, 我们使用一种动态规划的方式来测试所有的匹配可能. 这种动态规划解法建立在隐马科夫链模型的基础之上^[8].

在隐马科夫链模型中, 输入是一组观察结果的状态序列, 而目标是根据状态间的发散概率和转移概率找到一组最有可能产生这组观察序列的隐藏状态序

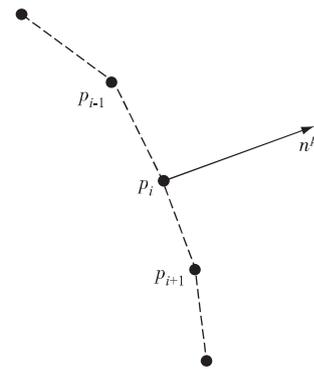


图 5 轮廓点的法线

Fig 5 The normal of outline vertex

列. 在轮廓点与模型顶点匹配时, 将轮廓顶点作为观察结果

$$O = \{p_1, p_2, \dots, p_m\}.$$

某组模型顶点序列作为隐藏状态

$$Q = \{v'_1, v'_2, \dots, v'_m\}.$$

同时用局部相似性的量度定义发散概率和转移概率:

$$\text{发散概率 } P(p_j | v_i) = e^{-\frac{1}{2}(\frac{d_p}{\delta_p})^2} e^{-\frac{1}{2}(\frac{d_N - 1}{\delta_N})^2}, \text{ 转移}$$

概率 $P(v_i | v_{i-1}) = e^{-\frac{1}{2}(\frac{d_c - 1}{\delta_c})^2}$, 其中 $\delta_p, \delta_N, \delta_c$ 是根据三维模型和轮廓的尺寸设定的参数. 发散概率定义了轮廓点 p_j 与模型顶点 v_i 匹配的可能性; 转移概率定义了 v_{i-1} 已经与 p_{j-1} 匹配的条件下, v_i 与 p_j 匹配的可能性. 此隐马科夫链模型问题需要解决的就是找到一个与轮廓点 p_1, p_2, \dots, p_m 一一匹配模型顶点的序列 v'_1, v'_2, \dots, v'_m , 且总体匹配的概率最大.

本文采用基于动态规划的 Viterbi 算法来求解. 从第一个轮廓点 p_1 开始, 计算每一个隐藏状态 v_i 产生当前的观察结果 p_j 的概率 $P(p_j | v_i)$, 然后迭代计算在此基础上下一个隐藏状态 v_{i+1} 产生下一个观察结果 p_{j+1} 的概率 $P(v_{i+1} | v_i) \times P(p_{j+1} | v_{i+1})$, 最后回溯求得最优的隐藏状态序列. 设 $\psi_c(i)$ 为迭代至轮廓点 p_c , 且状态序列为 v'_1, v'_2, \dots, v_i 时产生观察结果 p_1, p_2, \dots, p_c 的概率值; $\varphi_c(i)$ 则记录了迭代至 p_c 且状态序列为 v'_1, v'_2, \dots, v_i 时 v_i 的前一个状态, 记录了迭代的每一步并在最后用于回溯得到最优的状态序列. 算法的步骤如下.

1) 初始化:

$$\psi_1(i) = \max_{1 \leq j \leq n} P(p_j | v_i), 1 \leq i \leq n$$

$$\varphi_1(i) = 0$$

2) 迭代:

$$\psi_c(i) = \max_{1 \leq j \leq n} (\psi_{c-1}(j) P(p_c | v_i)), 1 \leq i \leq n, 2 \leq c \leq m$$

$$\varphi_c(i) = \operatorname{argmax}_{1 \leq j \leq n} (\psi_{c-1}(j))$$

3) 回溯:

$$v'_m = \operatorname{argmax}_{1 \leq j \leq n} (\psi_c(j))$$

$$v'_i = \varphi_{i+1}(v'_{i+1}) \quad i = c-1, c-2, \dots, 1$$

由于模型顶点的数量可能非常巨大,在算法中需要直接过滤那些发射概率小于某个阈值的模型顶点,防止循环次数过多.而算法最后得到的轮廓点与模型顶点的匹配可能出现多对一的情况,因此在结束算法前还需进行后处理,根据发射概率将多对一简化为一对一.例如: p_2 和 p_3 同时匹配到了 v_1 ,这时需比较 v_1 对 p_2 和 v_1 对 p_3 的发射概率,假设对 p_2 发射概率为 0.5 而对 p_3 概率为 0.4,则选取概率值高的 p_2 作为与 v_1 的匹配轮廓点,剔除 p_3 .造成多对一匹配的原因是网格顶点密度与轮廓点密度不相称,轮廓点的密度过大,所以此时并不需要将所有的轮廓点都匹配到模型顶点之上,而是将多余的轮廓点剔除.回溯之后得到的模型顶点序列 v'_1, v'_2, \dots, v'_m 就是与轮廓点序列匹配的最优结果,如图 6.

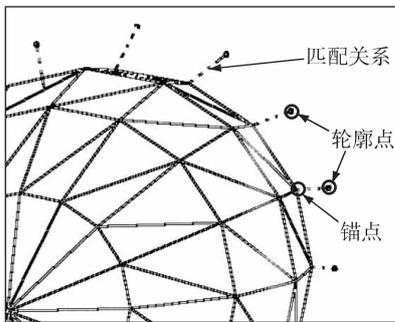


图 6 匹配结果(局部)

Fig. 6 The result of match (partial)

4 网格模型简化

解码实际上是对一个非线性最小二乘问题进行求解,其解空间的维度为 $3 \times$ 模型的顶点数,当需要变形的人体模型顶点数为 1 000 时,求解过程中使用的雅克比矩阵就高达上万维,直接导致算法效率低下.为了使解码过程更有效率,本文采用一种类似于文献[7]中使用的层次编码技术在编解码过程中引入网格简化操作,大幅减少解码阶段求解非线性最小二乘问题的规模.

4.1 三维模型的网格简化

在建立轮廓点与模型顶点的匹配后,得到 $V' = \{v'_1, v'_2, \dots, v'_m\}$ 为网格上与轮廓匹配的顶点,同时也是控制整个网格变形用的锚点, $\bar{V} = V - V'$ 为非锚顶

点.然后通过移除部分非锚顶点生成一个简化的网格.本文中使用了较为简单的网格简化方法,该方法基于文献[9]的顶点移除网格简化算法.流程如下.

- 1) 计算顶点误差 $E_v = \{e_1, e_2, \dots, e_n\}$,
- 2) 移除顶点 v_i , 其中 $i = \operatorname{argmax}_i (E_v)$,
- 3) 更新网格三角布局,更新 v_i 邻接点的误差值,
- 4) 如果 $\max_i (E_v)$ 或者 $\|E_v\|$ 小于某个阈值时,则结束,否则执行步骤 2).

顶点的误差值代表了移除这个顶点之后的新网格与原网格的误差大小,而误差值可以根据顶点包含的局部细节程度定义,某个顶点越“尖锐”则其包含的细节程度越高,移除该顶点后造成的网格几何信息的损失也越大.每次移除时都选择误差值最小的顶点,以此在简化过程中保持模型的几何特征.如文献[10]中所提到的,可以根据顶点与其邻接三角面的平均距离计算该顶点的误差值,通过为每个顶点计算一个误差矩阵,可以很快地从某个顶点与其邻接顶点的误差矩阵中得到顶点误差值.具体的计算过程见文献[10].

4.2 网格变形过程的改进

在通过移除顶点的方式来简化网格的同时,每次移除操作之前先计算该顶点的均值编码并储存起来.移除的模型顶点为 $V_r = \{V_{r1}, V_{r2}, \dots, V_{rm}\} \in \bar{V}$,最后剩余的顶点为 $\bar{V}UV'$,包括小部分非锚顶点和全部的锚点,且 $\|\bar{V}UV'\| < \|V\|$,简化后的网格模型为 M' .然后对网格 M' 编码解码,其中解码过程由于简化了大量的顶点,因此计算速度得以提升.加入网格简化之后的模型变形详细流程如图 7.

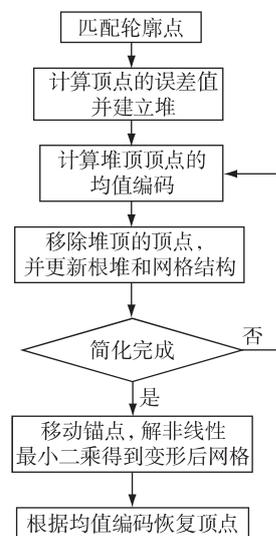


图 7 结合网格简化的变形流程

Fig. 7 The deformation process with simplified mesh

在变形过程中加入模型简化操作,虽然使得编码过程需要更多的时间,但是大大加快了解码时计算非线性最小二乘优化的速度,且对于每个模型,只需要进行一次编码并保存均值编码和顶点移除的顺序,每次变形均可使用,这样整体变形过程将大大加快。

5 实验结果

本实验使用 Visual Studio 2008 软件环境,运行平台硬件环境为 Pentium (R) Dual-Core 3.20 GHz/DDR2 2.0 G/Intel G41 集成显示芯片组. 实验结果如图 8~10. 对于球体模型,通过均值编码配合模型简化,最终能够很好得到纺锤状的三维模型,并保持了球体特征;而对于人体模型,使用变形算法生成了不同体型的模型. 图 10 展示了不同身材的人穿着绿色衣服的效果. 表 1 分别对比了球体与人体模型在未使用简化操作和使用简化操作的情况下变形算法执行情况,可以发现使用了简化操作之后算法效率大大提高。



(a) 衣物模型 (b) 身材瘦小的人试穿 (c) 身材高大的人试穿

图 10 不同身材的人试穿同款衣服

Fig. 10 Different men wear the same clothes

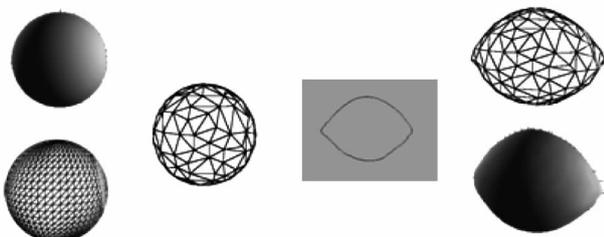
表 1 网格变形算法结果

Tab. 1 The result of mesh deformation

网格模型	原始顶点数	简化后顶点数	变形时间(无简化)/s	变形时间(简化)/s
球体	1 432	232	151	7
人	6 950	950	632	31

试衣中的应用. 首先描述了网格模型的编码与解码,然后建立轮廓点与模型点的匹配关系,并利用轮廓点控制网格的变形,最后通过网格简化操作提高变形算法的效率。

在以后的工作中,我们将提高模型变形的精度,并将此模型变形算法应用到其他领域。



(a) 原模型及网格 (b) 简化后网格 (c) 纺锤状轮廓图 (d) 变形后的网格

图 8 原模型的网格经过简化以及变形

Fig. 8 The mesh of original model have been simplified and deformed

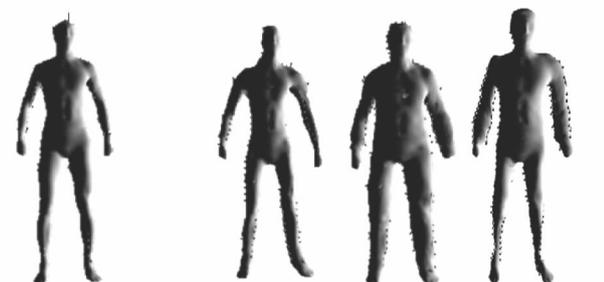


图 9 对人体模型变形后得到的不同体型的人体模型

Fig. 9 Different shape model of human body after deformation

6 讨论

本文介绍了三维网格变形的过程以及其在虚拟

参考文献:

[1] Olsen L, Samavati F F, Sousa M C, et al. Sketch-based modeling: a survey [J]. Computers&Graphics, 2009, 33 (1):85-103.

[2] Nealen A, Sorkine O, Alexa M, et al. A sketch-based interface for detail-preserving mesh editing[C]// ACM Siggraph 2007 Courses. New York: ACM, 2007:42.

[3] Cherlin J J, Samavati F, Sousa M C, et al. Sketch-based modeling with few strokes[C]// Proceedings of the 21st Spring Conference on Computer Graphics. New York: ACM, 2005:137-145.

[4] Schmidt R, Wyvill B, Sousa M C, et al. shapeshop: sketch-based solid modeling with blobtrees[C]// ACM Siggraph 2006 Courses. New York: ACM, 2006:14.

[5] Cashman T, Fitzgibbon A. What shape are dolphins? Building 3D morphable models from 2D images[J]. IEEE Trans Pattern Anal Mach Intell, 2013, 35(1):232-244.

[6] Chen Y, Kim T K, Cipolla R. Inferring 3D shapes and deformations from single views[C]// Computer Vision—ECCV 2010. Berlin Heidelberg: Springer, 2010:300-313.

[7] Kraevoy V, Sheffer A. Mean-value geometry encoding

- [J]. International Journal of Shape Modeling, 2006, 12 (1):29-46.
- [8] Kobbelt L, Campagna S, Vorsatz J, et al. Interactive multi-resolution modeling on arbitrary meshes[C]//Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, New York: ACM, 1998:105-114.
- [9] Botsch M, Kobbelt L. A remeshing approach to multiresolution modeling[C]//Proceedings of the 2004 Eurographics/ACM Siggraph Symposium on Geometry Processing, New York: ACM, 2004:185-192.
- [10] Garland M, Heckbert P S. Surface simplification using quadric error metrics[C]//Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, New York: ACM Press/Addison-Wesley Publishing Co, 1997:209-216.

Model Deformation in Virtual Dressing System

YANG Chen-hui* , CHEN Chen

(School of Information Science and Engineering, Xiamen University, Xiamen 361005, China)

Abstract: In virtual dressing system, all the clothes are three-dimensional (3D) mesh model which was made previously. To make virtual clothes match different users more accurately, we need to deform these mesh models. In this paper, an image-drive model deformation method was proposed to get a model which was adjusted to match user's body shape. This method includes coding, selection and movement of anchors, solving energy optimization, and finally deformed mesh was got. The algorithm is made more efficient by simplifying mesh. The possibility of model deformation using in virtual dressing system was proved.

Key words: virtual dressing system; 3D meshes; model deformation