

# 大交叠区域的三维网格的融合与拼接

邹北骥,周浩宇,王 磊,梁毅雄

(中南大学信息科学与工程学院,湖南长沙 410083)

**摘 要:** 通过融合重叠区域的顶点以及带边约束的基于点邻域平坦度的网格重构算法,实现了大交叠区域的三角网格的拼接,避免了因删除重叠区域的顶点而需要的补洞操作以及裁剪而产生的细碎三角形.首先,基于有向包围盒的思想,对网格的重叠区域进行快速定位;针对重叠区域的网格顶点,利用移动最小二乘的方法进行融合,避免了由于配准不精确而产生的噪音;而后,利用带边约束的基于点邻域平坦度的网格重构算法,对剩下的重叠区域的顶点进行三角化;最后,通过合并网格中的共同边界边实现网格的拼接.实验表明,该方法对于存在大交叠区域的三角网格的拼接快速而有效.

**关键词:** 网格拼接; 网格优化; 网格重构; 移动最小二乘; 有向包围盒

**中图分类号:** TP391.41      **文献标识码:** A      **文章编号:** 0372-2112 (2012) 05-1005-06

**电子学报 URL:** <http://www.ejournal.org.cn>

**DOI:** 10.3969/j.issn.0372-2112.2012.05.023

## 3D Mesh Merging and Stitching with Large Overlaps

ZOU Bei-ji, ZHOU Hao-yu, WANG Lei, LIANG Yi-xiong

(School of Information Science and Engineering, Central South University, Changsha, Hunan 410083, China)

**Abstract:** Via merging vertices in the overlapping regions, mesh stitching with large overlaps was achieved to avoid the filling-hole operations and small triangles produced by clipping edges. Firstly, the overlapping regions were detected by oriented bounding box. With the method of moving least squares, the vertices in the overlapping regions were smoothed in order to reduce the noise produced by inaccurate alignment. Then, we merged the vertices with distance less than given tolerance and triangulated the remaining vertices with the constraint of common boundary edges. The final mesh was created by stitching the several meshes via the common boundary edges. Experiments show that this method can create merged mesh with good quality and the performance is acceptable in practice.

**Key words:** mesh stitching; mesh optimization; mesh reconstruction; moving least squares; oriented bounding box

## 1 引言

基于扫描数据的逆向产品建模因能够快速获得与实物一致的 CAD 模型,在产品创新设计、角色建模中得到了广泛应用.在数字化扫描设备获取物体表面信息的过程中,由于模型复杂性、设备扫描角度的局限性,往往需要在不同侧面、不同位置进行多次扫描才能获得物体的全部表面信息.因此,扫描设备输出的数据,经常包含多片、有重叠的三角网格.在利用这些数据重建三维网格模型时,首先需要对这些网格片进行对齐、配准<sup>[1,2]</sup>,而后执行网格的融合、拼接、简化等操作<sup>[3~6]</sup>,将有重叠的多片网格合并成一个完整的、无重叠的有效网格模型,而后进行实时显示<sup>[7]</sup>.

为了提高两次扫描结果的配准精度,往往希望在两次扫描结果中存在尽可能多的重叠区域,从而通过最小二乘的方式减少配准的误差.然而,增加的重叠区域对于现有的网格融合与拼接方法<sup>[3~5]</sup>而言,会带来表示精度以及三角网格质量等方面的问题.因此,本文针对存在大交叠区域的网格融合与拼接,提出了一种基于顶点融合的方法,很好的解决了这方面的问题.

## 2 相关工作

网格的融合与拼接已有很多相关的研究工作.这些方法大体上可以分成三类:一类是基于裁剪的方法<sup>[3,5]</sup>,一类就是基于补洞的方法<sup>[4]</sup>,一类是基于微分网格变形的的方法<sup>[8~10]</sup>.基于裁剪的方法,即在重叠区域

利用一片网格的边界去裁剪另外一片网格,然后在公共的边界上生成新的三角形,使得两片网格拼接在一起.该方法会因为裁剪,从而在公共边界处产生大量的细小的三角形;并且该方法只是利用了重叠区域中的一片网格中的顶点,另一片网格上的顶点则完全被抛弃,对于存在大交叠区域的网格而言,未能利用两片网格的重叠区域进行顶点的校正;同时,由于多片网格在配准过程中存在的误差,在进行裁剪之前还需将一片网格的边界投影到另一片网格才能进行裁剪.基于补洞的方法,则直接将重叠区域的三角形全部删掉,然后通过补洞的方法重新生成重叠区域的三角形.这种方法对于小交叠区域的网格非常适用.但对于存在大交叠区域的网格而言,虽然可以通过径向基函数<sup>[11]</sup>等方法重新生成重叠区域的顶点,但新生成的顶点很难反映物体的实际形状.基于微分网格变形的的方法,如利用泊松方程<sup>[8]</sup>, Laplace 坐标<sup>[9]</sup>的方法等,需要指定一个精确的边界.边界指定的好坏,对融合结果会产生很大的影响,并且难以控制融合的精度.而对于大交叠区域的网格拼接而言,难以确定一个精确的融合边界.

针对以前方法对于大交叠区域的网格拼接存在的问题,本文提出了一种基于顶点融合的网格拼接方法.该方法通过融合重叠区域的顶点,并利用带边约束的基于点邻域平坦度的三角网格重构算法,最终实现大交叠区域的网格拼接,避免了基于裁剪方法中出现的大量细碎三角形,并且可以借助多次扫描提高重叠区域三角网格顶点的表示精度;其次,由于利用的是原有网格的顶点,避免了补洞的操作,不会因为重新生成顶点而降低顶点的表示精度.

### 3 整体框架与流程

对于配准后的网格  $M_1$  和  $M_2$ , 网格的融合与拼接主要包括三个步骤:重叠区域的检测,重叠区域的顶点融合与重叠区域的拼接.其整体流程如图 1 所示.

#### (1) 重叠区域的检测

为了快速定位重叠区域的三角网格,基于有向包围盒的思想,判断网格的有向包围盒是否相交,若相交,计算出相交区域的三角网格的顶点,并以相交区域的顶点分别构建有向包围盒,如此迭代,直至相交区域的三角网格的顶点不再发生变化为止.由于网格拼接是基于顶点融合的思想,重叠区域的顶点定位不用非常精确,额外的顶点有助于重叠区域边界处的平滑拼接.

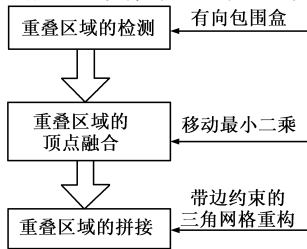


图1 网格拼接的整体流程图. 右边箭头表示的为当前步骤中所使用的主要方法

#### (2) 重叠区域的顶点融合

对于重叠区域的顶点,首先利用移动最小二乘的方法对这些顶点进行平滑,减少由于扫描过程和配准过程中产生的诸如薄层和相互穿插等噪音;而后合并距离小于给定阈值的顶点,剔除重叠区域的重复顶点.

#### (3) 重叠区域的拼接

将  $M_1$  和  $M_2$  中删除重叠区域后新生成的边界顶点加入到融合后的顶点集合中,并以新生成的边界边作为约束,利用带边约束的基于点邻域平坦度的网格重构算法对该顶点集进行三角化.新生成的三角网格由于包含共同的边界,合并该共同的边界即实现三角网格的拼接.

由上可以看出,对于存在大交叠区域的三角网格的融合与拼接,本文方法相较于其他方法的优点在于:

(1) 提出了一种快速定位重叠区域的方法.通过构建有向包围盒并判断是否相交,快速定位重叠区域的顶点;

(2) 通过顶点融合,利用了多次扫描的结果,减少了扫描过程和配准过程所产生的噪音,又无需通过补洞的方法重新生成新的顶点;

(3) 提出了一种带边约束的基于点邻域平坦度的网格重构算法,使得网格的拼接可以利用公共边界,简化了拼接操作.

## 4 重叠区域的检测

为了实现重叠区域的网格融合与拼接,首先需要检测与定位网格的重叠区域.对于配准好后的两片三角网格  $M_1$  和  $M_2$ , 其重叠区域的顶点集合  $\Omega$  可以由式 (1) 加以描述,

$$\Omega = \{v_i | v_i \in M_1 \text{ \&\& } d(v_i, M_2) < \epsilon \\ \vee v_i \in M_2 \text{ \&\& } d(v_i, M_1) < \epsilon\}, \quad (1)$$

其中  $d(v, M)$  表示顶点  $v$  到网格  $M$  的距离,  $\epsilon$  为用户给定的允许误差.

由式(1)可以看出,要确定网格的重叠区域,需要求取顶点到网格的距离.顶点到网格的距离计算可以由 k-d 树进行加速,但是对于拥有大量顶点的网格而言,计算依然非常耗时.文献[4]中提出了首先利用栅格六面体进行粗略判断,然后利用 k-d 树进行确认的方法,从而减少距离计算的次数.这里的栅格六面体,实质上就是物体的轴向包围盒.由于轴向包围盒并不能很好的体现网格的形状,如图 2(a) 所示,本文采用有向包围盒(如图 2(b))的方法进行判断,从而更准确的定位到网格重叠区域,减少判断的次数.虽然利用该方法并不能十分精确的定位到网格的重叠区域,但由于本文采用的是基于顶点融合的方法,而非删除重叠区域的顶点,如 5 节中所述,额外的顶点并不会影响网格的

拼接,相反可以提高网格拼接区域的光顺性.

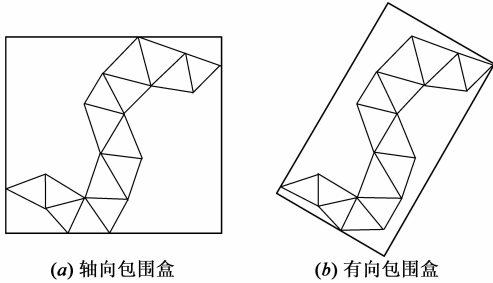


图2 轴向包围盒和有向包围盒的构建(二维情况)

有向包围盒技术最先应用于碰撞检测的问题中<sup>[12]</sup>,用于判断两个物体是否相交,而后被应用到点云的绘制<sup>[13]</sup>、变形<sup>[14]</sup>等众多应用中.本文基于该方法快速定位重叠区域的顶点集合  $V_{inter}$ .其整体算法描述如下:

(1)初始化顶点集合  $V_1, V_2$  分别为网格  $M_1$  和  $M_2$  的顶点集合.

(2)分别计算顶点集合  $V_1$  和  $V_2$  的有向包围盒,记为  $B_1$  和  $B_2$ .

(3)若有向包围盒  $B_1$  和  $B_2$  相交,则更新顶点集合  $V_1, V_2$  的顶点.  $V_1$  更新为集合  $V_1$  中落在有向包围盒  $B_2$  中的顶点集,  $V_2$  更新为集合  $V_2$  中落在包围盒  $B_1$  中的顶点集合,即:

$$V_1 = \{v_i | v_i \in V_1 \ \&\& \ v_i \in B_2\},$$

$$V_2 = \{v_i | v_i \in V_2 \ \&\& \ v_i \in B_1\}.$$

若不相交,结束整个算法,相交区域的顶点集合  $V_{inter} = \emptyset$ .

(4)若顶点集合  $V_1$  和  $V_2$  中的顶点较前一次循环发生变化,则循环执行第(2)步;否则,结束整个算法,相交区域的顶点集合  $V_{inter} = V_1 \cup V_2$ .

顶点集合的有向包围盒的构建与文献[12]中的方法相同.一个有向包围盒由中心点,三个轴向及其对应的轴向半径确定.三个轴向由该顶点集合所构成的协方差矩阵的三个特征向量组成,轴向半径即为所有顶点在对应轴向上投影最小值  $l_k$  与最大值  $m_k$  的中值  $(m_k - l_k)/2$ ,中心点对应顶点集合的质心,具体的构建过程可以参考文献[12].

有向包围盒的相交测试基于的是轴分离原理,即:如果存在这样的一条分离轴,使得两个有向包围盒各顶点在该轴上的投

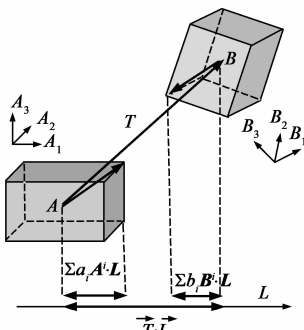


图3 有向包围盒相交测试的轴分离原理

影互不相交,则这两个有向包围盒不相交,否则相交.如图3所示,设有向包围盒  $B_1, B_2$  的轴向分别为  $A^i, B^i$ ,轴向半径分别为  $a_i, b_i, T$  为连接两个包围盒中心的线段,若  $L$  为其一条分离轴则:

$$|T \cdot L| > \sum_{i=0}^3 |a_i A^i \cdot L| + \sum_{i=0}^3 |b_i B^i \cdot L| \quad (2)$$

由轴分离原理知,若某个向量为分离轴,则该向量或者平行于某个有向包围盒的某条边,或者垂直于分别来自不同包围盒的两条边所在的平面.因此,如果两个有向包围盒相交,只需按式(2)做15次测试即可;如果不相交,则所需的测试更少.实际测试中,通常先将一个有向包围盒的局部标架变换到另一个有向包围盒的局部标架中,这样可以简化式(2)中的计算.

### 5 重叠区域的融合

由于扫描过程以及配准过程的不精确,配准后的重叠区域会出现薄层或者是相互穿插的现象.对于大交叠区域的三角网格尤为明显,为了减少这些噪声造成的影响,利用移动最小二乘的方法对第4节中检测出的重叠区域的顶点进行融合,使得融合后的顶点比较平滑.

为了增加重叠区域在新边界处与原始网格拼接的光顺性,将原始网格在重叠区域新形成的边界点  $V_{bdry}$  加入到待平滑的顶点集合中.这样也可以简化第6节中的网格拼接操作.

移动最小二乘方法最先由 Lancaster 和 Salkauskas 提出<sup>[15]</sup>,Levin 等人率先将其应用于点云曲面的重建与绘制中<sup>[16,17]</sup>.本文通过将重叠区域中的顶点投影到其所在的移动最小二乘曲面而起到平滑的效果.

移动最小二乘曲面的构建主要包括两个步骤:局部参考平面的计算和基于局部参考平面的多项式曲面拟合.对于点集  $V = \{v_i | i = 1, \dots, n\}$  中的待投影点  $r$ ,其局部参考平面  $\{x | \langle n, x \rangle - D = 0\}$  可由最小化式(3)所示的距离函数得到:

$$\sum_{i=1}^n (\langle n, v_i \rangle - D)^2 \theta(\|v_i - q\|) \quad (3)$$

其中,  $q$  为待投影点  $r$  在该参考平面上的投影,如图4所示,  $\theta(\cdot)$  为权重函数,通常取为高斯函数,即  $\theta(d) = e^{-d^2/h^2}$ ,  $h$  用于控制权重函数的衰减速度,本文中取为  $4 * avgLen$ ,  $avgLen$  为网格的平均边长.局部参考平面得到后,基于点  $q$  以及该参考平面可以建立一个局部参考标架.在该局部参考标架下,离散点集可以通过一个双变量多项式函数  $g$

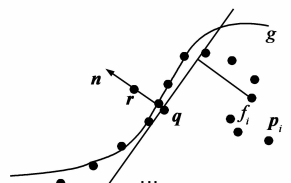


图4 最小二乘投影示意图

$(x, y)$ 进行拟合. 记  $f_i$  为顶点  $v_i$  到局部参考平面的距离, 即  $f_i = \mathbf{n} \cdot (\mathbf{v}_i - \mathbf{q})$ , 则目标函数  $g(x, y)$  可以通过最小化式(4)所示的误差函数得到:

$$\sum_{i=1}^n (g(x_i, y_i) - f_i)^2 \theta(\|\mathbf{v}_i - \mathbf{q}\|) \quad (4)$$

目标函数  $g(x, y)$  得到后, 对于待投影点  $r$ , 其在目标函数上的投影点可表示为  $\mathbf{q} + g(0, 0)\mathbf{n}$ .

由上可以看出, 移动最小二乘曲面的构建中, 主要的计算在于局部参考平面的计算, 参考平面计算好后, 在局部参考标架下的曲面拟合, 即式(4)的优化, 只是一个线性优化的问题, 可以通过求解线性方程组得到. 而参考平面的计算, 即式(3)的优化, 是关于法向  $\mathbf{n}$  以及平面参考点  $\mathbf{q}$  的非线性优化, 计算起来非常耗时. 本文通过对移动最小二乘进行近似, 采用了一个迭代的快速计算方法, 其主要思想为: 在每步迭代过程中, 将平面参考点  $\mathbf{q}$  取为点集中各点的加权平均, 即:

$$\mathbf{q}(\mathbf{r}) = \frac{\sum_i \theta(\|\mathbf{v}_i - \mathbf{r}\|) \mathbf{v}_i}{\sum_i \theta(\|\mathbf{v}_i - \mathbf{r}\|)} \quad (5)$$

从而将式(3)所示的问题转化为一个线性优化问题, 得到当前条件下的平面法向  $\mathbf{n}$ , 即:

$$\mathbf{n}(\mathbf{r}) = \operatorname{argmin} \sum_i \langle \mathbf{n}, \mathbf{v}_i - \mathbf{q} \rangle^2 \theta(\|\mathbf{v}_i - \mathbf{q}\|) \quad (6)$$

整体算法描述如下:

(1) 对于待投影点  $r$ , 计算  $\mathbf{q} \leftarrow \mathbf{q}(\mathbf{r}), \mathbf{n} \leftarrow \mathbf{n}(\mathbf{r})$ ;

(2) 如果  $\langle \mathbf{n}, \mathbf{q} - \mathbf{r} \rangle < \epsilon$ , 则  $\mathbf{n}$  和  $\mathbf{q}$  为所求, 结束迭代;

(3) 将待投影点  $r$  投影到该参考平面上, 即  $\mathbf{r} \leftarrow \mathbf{r} + \langle \mathbf{n}, \mathbf{q} - \mathbf{r} \rangle \mathbf{n}$ , 返回到(1)继续迭代.

重叠区域经过平滑后, 需删除顶点集合中重复的顶点. 本文通过合并距离小于给定阈值的顶点实现. 对于距离小于  $\epsilon$  的顶点  $v_i, v_j$ , 合并后的顶点为  $0.5 * (v_i + v_j)$ . 本文中  $\epsilon$  取为  $0.2 * \text{avgLen}$ ,  $\text{avgLen}$  为网格的平均边长. 为了合并距离小于给定阈值的顶点对, 需要查找给定点的最近点, 本文通过 ANN 算法<sup>[18]</sup>进行快速实现.

## 6 网格拼接

若重叠区域三角化后的网格  $M_{\text{overlap}}$  与原始网格  $M_1, M_2$  去除重叠部分以后的子网格  $M'_1, M'_2$  在新形成的边界处有着共同的边界边, 如图 5 所示, 那么这三部分网格的拼接只需合并共同的边界顶点和边界边即可. 为此, 本文通过采用带边约束的基于点邻域平坦度的网格重构算法, 对融合后的顶

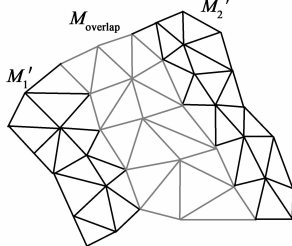


图5 含共同边界的网格拼接

点集合进行三角化, 最后合并共同的边界边实现网格的拼接.

记融合后的重叠区域的顶点集合为  $V_{\text{merge}}$ , 如第 5 节中所述, 该顶点集合已包含原始网格在重叠区域形成的新的边界点, 另外, 记原始网格在重叠区域形成的新的边界边集合为  $E_{\text{bdry}}$ , 带边约束的网格重构算法即在  $E_{\text{bdry}}$  的约束下, 对顶点集合  $V_{\text{merge}}$  进行三角化. 此时新形成的网格  $M_{\text{overlap}}$  与  $M'_1, M'_2$  在重叠区域的边界处将形成共同的边界, 网格的拼接通过合并共同的顶点和边即可实现.

带边约束的基于点邻域平坦度的网格重构算法是在文献[19]中提出的基于点邻域平坦度的网格重构算法的基础上, 引入边约束实现的, 即三角化后的网格经过指定的约束边集. 算法的整体过程如下:

(1) 对数据点进行初始化处理, 计算每个点邻域的平均度, 并确定每个点的平坦级别;

(2) 针对约束边集中的每条边, 根据其所属于的平均度区域, 放入相应的优先队列;

(3) 从优先级最高的活动边队列中选取活动边进入区域扩张过程. 区域扩张过程包括建立有效区域、确定候选点、在候选点中寻找一个最佳点, 从而生成新的三角面片加入网格, 同时更新边和点的状态, 并将新生成的边按照所属区域平坦度放入对应的队列. 如此扩张过程不断往复, 直至所有数据点都被处理.

(4) 经过简单的后续处理, 构造形成流形网格.

由上描述可以看出, 本文算法与文献[19]中的算法的主要区别在于: 初始化过程中, 首先将约束边集中的边根据其所属于区域的平坦度级别放入到相应的优先队列中, 因此满足了边界约束; 同时, 由于有了这些初始的约束边, 无需构造种子三角形, 简化了种子点的选取. 平坦度的计算以及区域扩张的具体过程可以参考文献[19].

## 7 实例与算法分析

本文算法使用 VC++ 6.0 在 AMD AthlonTM 7850 Dual Core 2.6GHz CPU, 1GB 内存的 PC 机上实现. 图 6 和图 7 分别是针对两个真实扫描的人头数据进行的交叠网格的融合和拼接, 图 8 是经过人工裁剪后得到的数据. 由图 6(a) 和图 7(a) 中矩形框放大部分可以看出, 真实扫描数据往往由于扫描过程的精度问题以及配准过程的误差问题, 交叠区域会存在穿插或者薄层的现象, 本文通过对重叠区域的顶点融合, 能够得到较好的拼接效果, 如图 6(c)、图 7(c) 所示. 虽然本文中采用的基于有向包围盒进行重叠区域快速定位的方法, 并不能精确定位到重叠区域的顶点, 如图 6(b)、图 7(b) 所示, 但由于使用的是基于重叠区域的顶点融合的方法,

并不会对网格的拼接带来问题,反而可以增加网格连接处的光顺性.此外,重叠区域的顶点融合会造成尖锐特征的平滑,不过对于实际中的一般特征而言损失较少,如图 7(c)中鼻子处的融合效果,图 8 中的花纹特征.

表 1 是对图 6 至图 8 中各交叠网格进行融合拼接所需要的时间的统计,其中 # overlap 表示交叠区域的顶点数目,  $T_{\text{detect}}$  表示快速定位重叠区域所需的时间,  $T_{\text{merge}}$  表示重叠区域的顶点融合所需的时间,  $T_{\text{reconstruct}}$  表示重

构重叠区域的顶点所需的时间.从表中可以看出,本文算法的效率对于工程实际而言还是可以接受的.

表 1 各模型进行融合与拼接所需的时间统计

模型	# overlap	$T_{\text{detect}}$	$T_{\text{merge}}$	$T_{\text{reconstruct}}$
人头 1	29,227	0.328 s	1.11 s	2.875 s
人头 2	31,800	0.312 s	1.078 s	4.5 s
花瓶	112,771	0.718 s	1.875 s	10.75 s



图 6 人头 1 的交叠区域检测与拼接



图 7 人头 2 的交叠区域检测与拼接



图 8 花瓶模型的交叠区域监测与拼接

## 8 结论

本文通过融合重叠区域的顶点以及带边约束的基于点邻域平坦度的网格重构算法,实现了大交叠区域的网格融合与拼接,避免了利用边界裁剪方法而出现的大量细碎三角形,以及补洞算法而出现的数据失真.由于融合了多次采集的重叠区域的顶点,从统计意义上减少了采集过程中的误差,提高了数据表示的精度.但由于对重叠区域的融合同样也会造成尖锐特征的平滑,因此,研究一种特征保持的顶点融合算法将是下一步工作需要处理的主要问题.

## 参考文献

[1] Besl P J, McKay N D. A method for registration of 3-D shapes

[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, 14(2): 239 - 256.

[2] 戴静兰, 陈志杨, 叶修梓. ICP 算法在点云配准中的应用 [J]. 中国图象图形学报, 2007, 12(3): 517 - 521.

Dai Jinglan, Chen Zhiyang, Ye Xiuzi. The application of ICP algorithm in point cloud alignment [J]. Journal of Image and Graphics, 2007, 12(3): 517 - 521. (in Chinese)

[3] Turk G, Levoy M. Zippered polygon meshes from range images [A]. Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques [C]. New York: ACM, 1994. 311 - 318.

[4] 陈志杨, 叶建华, 沈瑛, 等. 交叠网格的检测与合并 [J]. 中国机械工程, 2008, 19(17): 2064 - 2068.

Chen Zhiyang, Ye Jianghua, Shen Ying, et al. Detection and merge of overlap meshes [J]. China Mechanical Engineering,

- 2008, 19(17): 2064 – 2068. (in Chinese)
- [5] 刘晖, 向世明, 陈睿, 等. 三维扫描网格的合并和优化[J]. 计算机工程与应用, 2004, 40(29): 28 – 31.  
Liu Hui, Xiang Shiming, Chen Rui, et al. Merged and regularized polygon meshes from range images[J]. Computer Engineering and Applications, 2004, 40(29): 28 – 31. (in Chinese)
- [6] 邹北骥, 申煜湘, 彭群生. 局部包络片控制误差简化三角网格模型[J]. 电子学报, 2005, 33(5): 798 – 803.  
Zou Bei, Shen Yuxiang, Peng Qunsheng. Triangulated surface simplification using part offset-paths as errors tolerance[J]. Acta Electronica Sinica, 2005, 33(5): 798 – 803. (in Chinese)
- [7] 袁友伟. 采用 GPU 加速的三维实体模型绘制[J]. 电子学报, 2008, 36(12A): 144 – 146.  
Yuan Youwei. 3D solid models rendering based on GPU acceleration[J]. Acta Electronica Sinica, 2008, 36(12A): 144 – 146. (in Chinese)
- [8] Yu Y, Zhou K, Xu D, et al. Mesh editing with Poisson-based gradient field manipulation[J]. ACM Transaction on Graphics, 2004, 23(3): 644 – 651.
- [9] Sorkine O. Differential representations for mesh processing[J]. Computer Graphics Forum, 2006, 25(4): 789 – 807.
- [10] Sharf A, Blumenkrants M, Shamir A, et al. SnapPaste: An interactive technique for easy mesh composition[J]. The Visual Computer, 2006, 22(9): 835 – 844.
- [11] 陈飞舟, 陈志杨, 丁展, 等. 基于径向基函数的残缺点云数据修复[J]. 计算机辅助设计与图形学学报, 2006, 18(9): 1414 – 1419.  
Chen Feizhou, Chen Zhiyang, Ding Zhan, et al. Filling holes in point cloud with radial basis function[J]. Journal of Computer-Aided Design & Computer Graphics, 2006, 18(9): 1414 – 1419. (in Chinese)
- [12] Gottschalk S, Lin M C, Manocha D. OBBTree: A hierarchical structure for rapid interference detection[A]. Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques[C]. New York: ACM, 1996. 171 – 180.
- [13] 陈为, 马瑞金, 郑文庭, 等. 基于 OBB 树的无网格几何数据处理[J]. 计算机学报, 2007, 30(2): 330 – 335.  
Chen Wei, Ma Ruijin, Zheng Wenting, et al. Efficient processing of meshless geometrical data based on OBB tree structure[J]. Chinese Journal of Computers, 2007, 30(2): 330 – 335. (in Chinese)
- [14] 谭光华, 许秋儿, 吴双卿, 等. 特征保持的大规模点云曲面自由变形[J]. 浙江大学学报(工学版), 2010, 44(1): 34 – 40.  
Tan Guanghua, Xu Xiu'er, Wu Shuangqing, et al. Detail-preserving deformation for large-scale point set surface[J]. Journal of Zhejiang University (Engineering Science), 2010, 44(1): 34 – 40. (in Chinese)
- [15] Lancaster P, Salkauskas K. Surfaces generated by moving least squares methods[J]. Mathematics of Computation, 1981, 37(155): 141 – 158.
- [16] Alexa M, Behr J, Cohen-Or D, et al. Point set surfaces[A]. Proceedings of the Conference on Visualization[C]. Washington DC: IEEE Computer Society, 2001. 21 – 28.
- [17] Alexa M, Behr J, Cohen-Or D, et al. Computing and rendering point set surfaces[J]. IEEE Transactions on Visualization and Computer Graphics, 2003, 9(1): 3 – 15.
- [18] Mount D M, Arya S. ANN: A library for approximate nearest neighbor searching[CP/OL]. <http://www.cs.umd.edu/~mount/ANN/>, 2010-1-27.
- [19] 李根, 陈志杨, 张三元, 等. 基于点邻域平坦度的网格重构算法[J]. 计算机辅助设计与图形学学报, 2008, 20(4): 482 – 487.  
Li Gen, Chen Zhiyang, Zhang Sanyuan, et al. Mesh reconstruction algorithm based on neighborhood smoothness measure[J]. Journal of Computer-Aided Design & Computer Graphics, 2008, 20(4): 482 – 487. (in Chinese)

#### 作者简介



邹北骥 男, 1961 年生于湖南邵阳, 博士, 教授, 博士生导师. 分别于 1982 年、1984 年和 2001 年在浙江大学、清华大学和湖南大学获得工学学士、工学硕士和工学博士学位. 现为中南大学信息科学与工程学院副院长. 主要研究方向: 计算机视觉、机器学习、虚拟现实技术、计算机图形学、数字图像处理、CAD 技术及软件工程技术.



周浩宇 男, 1979 年生于湖南长沙, 博士研究生. 分别于 2002 年和 2005 年在湖南大学获得工学学士和工学硕士学位. 现就读于中南大学信息科学与工程学院. 研究方向: 计算机图形图像处理.

E-mail: zhou\_haoyu@163.com

王磊 男, 1978 年生于河南平顶山, 博士, 讲师. 分别于 2002 年、2004 年和 2007 年在湖南大学获得工学学士、工学硕士和工学博士学位. 现就职于中南大学信息科学与工程学院. 研究方向: 计算机图形图像处理.

梁毅雄 男, 1977 年生于湖南涟源, 博士, 副教授. 分别于 1999 年、2002 年和 2005 年在重庆大学获得工学学士、工学硕士和工学博士学位. 现就职于中南大学信息科学与工程学院. 研究方向: 计算机图形图像处理.