

# 点云重构三角网格的生长型神经气算法\*

曾锋<sup>1</sup>, 杨通<sup>2</sup>, 姚山<sup>2</sup>

<sup>1</sup>(嘉应学院 计算机学院, 广东 梅州 514015)

<sup>2</sup>(三束材料改性教育部重点实验室(大连理工大学), 辽宁 大连 116024)

通讯作者: 曾锋, E-mail: zengfeng@jyu.edu.cn

**摘要:** 为克服点云噪声、不均匀分布和复杂拓扑结构对三角网格重构的限制, 改进了生长型神经气重构算法。以样本在网格局部投影作为神经元插入判据, 自适应调节网格增长速度, 保持几何变换与拓扑变换的协调。利用非流形边检测机制删除冗余连接, 保持网格的拓扑有效性。网络学习过程中动态更新三角片结构, 且在孔洞修复阶段扩大近邻查找范围, 连接近邻节点中的边界点, 直到网格收敛, 最终得到正确的欧拉示性数。算例表明, 改进的算法对带噪声点云具有鲁棒性, 可根据非均匀点云的分布自动调整网格密度, 且能重构具有复杂拓扑结构的曲面。重构的三角网格对曲面逼近精度较高, 网格出度均匀, 三角形近似等边。

**关键词:** 点云; 生长型神经气算法; 三角网格

**中图法分类号:** TP391      **文献标识码:** A

中文引用格式: 曾锋, 杨通, 姚山. 点云重构三角网格的生长型神经气算法. 软件学报, 2013, 24(3): 651-662. <http://www.jos.org.cn/1000-9825/4293.htm>

英文引用格式: Zeng F, Yang T, Yao S. From point cloud to triangular mesh by growing neural gas. Ruanjian Xuebao/Journal of Software, 2013, 24(3): 651-662 (in Chinese). <http://www.jos.org.cn/1000-9825/4293.htm>

## From Point Cloud to Triangular Mesh by Growing Neural Gas

ZENG Feng<sup>1</sup>, YANG Tong<sup>2</sup>, YAO Shan<sup>2</sup>

<sup>1</sup>(School of Computer, Jiaying University, Meizhou 514015, China)

<sup>2</sup>(Key Laboratory of Materials Modification by Laser, Ion and Electron Beams (DaLian University of Technology), Ministry of Education, , Dalian 116024, China)

Corresponding author: ZENG Feng, E-mail: zengfeng@jyu.edu.cn

**Abstract:** Triangular surface reconstruction out-of-point clouds suffer from noisy, non-uniform distributed data, and complicated topology structure. Thus, an improved growing neural gas approach is proposed. A point cloud projection on local grid is employed to direct node insertion; therefore, to adaptively control neuron growing rate, the geometric and topologic transforms are synchronized. Redundant links are removed through non-manifold edge detection, that guarantees a topologically validate mesh. The network keeps updating triangular grid and then fills holes in a post phase by the extended neighborhood connection mechanism. After all those steps come to a convergent end, there is a gap free and an Euler characteristic correct mesh was obtained. Case studies invalidate the noise robustness and complex topology adaptability. The algorithm can further adjust mesh size to point cloud distribution. Plus is that reconstructed mesh approximates the surface in high accuracy, and it characterizes uniform equilateral edge share.

**Key words:** point cloud; growing neural gas algorithm; triangular mesh

随着激光、双目视觉、结构光、CT 等多种三维测量技术速度和精度的提高, 点云重构曲面模型的应用在机器视觉、逆向工程、虚拟现实等领域得到拓展。由不附加任何几何与拓扑信息(如法矢、曲面边界、连通性、

\* 基金项目: 广东高校优秀青年创新人才培养计划(LYM10121)

收稿时间: 2012-01-08; 修改时间: 2012-04-18; 定稿时间: 2012-08-10

曲面孔洞数)的散乱点云重构曲面的过程面临多种挑战:

首先,实测点云常包含上百万数据量,部分算法需查找点云  $k$ -近邻<sup>[1]</sup>或者拟合局部平面,限制了大规模点云的处理效率;对于采样不均匀的点云,若采用 Power Crust, Alpha Shape 或 Ball Pivoting 等传统方法,需根据局部点云密度调整算法参数<sup>[2]</sup>;其次,重构曲面之前还需检测和剔除噪声点<sup>[3]</sup>;另外,部分方法对具有复杂拓扑结构的高亏格曲面、开曲面等存在局限性。

因 SOM(self organizing map)神经网络计算时间复杂度与样本数据量无关,具有基于统计学习的噪声鲁棒性,而且可实现拓扑有序映射,基于 SOM 及其变体的曲面重构算法逐渐受到研究者重视.Hoffmann<sup>[4]</sup>在曲面重构中首先引入了 SOM 方法,用于计算样条曲面网格<sup>[5,6]</sup>,利用球面 SOM 网络和保形映射可重构 0 亏格曲面的三角网格近似<sup>[7]</sup>.但 SOM 网络需预定义节点数量以及节点间的连接关系,只适用于简单的拓扑结构<sup>[8]</sup>,可通过网格自适应细分、节点和边操作算子<sup>[9]</sup>改进 SOM 网络.由 Fritzke 提出的 GCS(growing cell structure)<sup>[10]</sup>和 GNG(growing neural gas)<sup>[11]</sup>两种变体各采用了节点动态增长、竞争性连接等机制改善 SOM 法的不足.但 GCS 只能学习和初始网格拓扑等价的曲面<sup>[12]</sup>,而 GNG 并未定义网格结构.Ivrissimtzis<sup>[13]</sup>在 GCS 基础上借助系综平均、网格边界算子、Marching Cube 等方法构造三角网格.在 GCS 学习过程中,也可采用三角片删除、边合并、边交换算子自适应改变初始网格拓扑结构<sup>[14,15]</sup>.Annuth<sup>[16]</sup>在 GCS 基础上加入三角片剪裁与节点分裂等步骤,动态学习点云拓扑结构.Melato<sup>[17]</sup>在 GNG 学习完成后,通过额外的后处理步骤检测三角网格.Holdstein<sup>[18]</sup>在采用 GNG 算法得到初始的网格之后,再利用点云在 GNG 网格上的局部投影构造满足约束的三角网格.Rego<sup>[19]</sup>和 DalleMole<sup>[20]</sup>分别通过修改 GNG 单元插入与删除规则,改进竞争性 Hebbian 连接机制,得到近似等边的三角网格。

以上基于 GNG 和 GCS 的算法都引入了过多控制参数,针对不同特征点云较难确定合适的参数,且无法保证重构结果的拓扑正确性.本文在生长型神经气方法(GNG)的基础上,以样本数据在局部三角网格的投影作为节点插入判据,控制节点数量使网络达到收敛状态,最后对边界节点进行训练,重构三角网格。

## 1 概念与定义

设  $D_n = \{\xi_1, \xi_2, \dots, \xi_n\}$ ,  $\xi_i \in \mathbb{R}^3$ ,  $\xi_i = \mu_i + \delta_i$  是采样于未知流形曲面  $M$  的点集,其中,  $\mu_i \in M$ ,  $\delta_i$  是各点测量误差.原始曲面  $M$  结构未知,可能是开曲面或闭曲面.此外,曲面的亏格也是重要的拓扑特征.若曲面上最多可画出  $n$  条闭合曲线而不将曲面分开,则称该曲面亏格为  $n$ .如图 1 所示,闭曲面亏格数即为其洞眼(柄)的个数.GNG 网络可通过竞争性连接机制学习高亏格结构。



Fig.1 Surfaces of different genus

图 1 不同曲面的亏格数

由点云重构曲面三角网格模型是指:给定散乱数据点  $D$ ,求得一个三角网格  $S = [\{P_i\}, \{L_i\}, \{F_i\}]$ ,  $L_i = (P_m, P_n)$ ,  $F_i = (P_{k1}, P_{k2}, P_{k3})$ ,逼近原曲面  $M$ ,其中,  $P_i \in \mathbb{R}^3$  代表构成三角网格的节点,  $L_i$  为连接两个节点的边,  $F_i$  是 3 个节点构成的三角片.三角网格重构是不适定问题,不存在唯一解,但其可行解需满足如下约束:

- (1) 相交于某节点的边至多比相交于该节点的面的数量大 1;
- (2) 至多存在两个三角片交于一条边;
- (3) 三角片之间无重叠或交叉;
- (4) 构成一个三角片的三点不共线;

- (5) 重构三角曲面满足欧拉定理:定义欧拉示性数  $Euler=card(\{P_i\})+card(\{F_i\})-card(\{L_i\})$  为曲面的一个拓扑不变量,满足  $Euler=2-2\times Genus-Loop$ ,其中,  $card(*)$  为该集合的基数,也就是集合中元素的个数,  $Genus$  为亏格数,  $Loop$  为曲面上边界环的个数.

## 2 重构算法

### 2.1 GNG算法

GNG 网络由节点单元  $A=\{c_1, c_2, \dots, c_k\}$  和单元间连接  $E=\{(i, j) | i, j \in A, (i, j) \equiv (j, i)\}$  组成.单元有各自的参考向量  $\omega_c \in \mathbb{R}^3$  和积累误差  $\varepsilon_c \in \mathbb{R}$ ,参考向量维数与输入样本相同.网络学习过程中,在积累误差较大的部位插入新单元,实现网络动态增长.与某单元相连的单元集合  $N_c=\{i | \forall i \in A, (c, i) \in E\}$  构成其拓扑近邻.每对连接  $(i, j)$  有其年龄属性  $age_{(i, j)}$ ,在输入样本的最近两个单元间添加连接以及依据年龄动态删除无效连接的过程称为竞争性连接,可由此改变单元间拓扑结构.初始 GNG 网络包含少量单元和空连接,学习过程中逐渐插入新单元并且改变连接,从而实现输入空间拓扑结构和空间几何信息的逼近.

以点云数据  $D$  为样本, GNG 网络学习过程如下<sup>[11]</sup>:

Step 1. 初始化  $A$  包含两个单元  $A=\{c_1, c_2\}$ ,从样本  $D$  中随机选择两个点作为  $c_1, c_2$  的参考向量,积累误差设为 0.初始化连接集合  $E=\emptyset$ ,学习次数  $time=0$ .

Step 2.  $time=time+1$ ,从  $D$  中随机选择一点  $\xi$  作为输入样本,查找  $\xi$  在  $A$  中的最近邻和次近邻单元:

$$n_1 = \arg \min_{n_i \in A} \|\xi - \omega_{c_i}\|; n_2 = \arg \min_{n_2 \in A \setminus \{n_1\}} \|\xi - \omega_{c_i}\|.$$

Step 3. 如果  $n_1$  和  $n_2$  之间不存在连接,则添加两个单元间的连接:  $E=E \cup \{(n_1, n_2)\}$  IF  $(n_1, n_2) \notin E$ ;刷新  $(n_1, n_2)$  的年龄  $age_{(n_1, n_2)} = 0$ ,并增加单元  $n_1$  所有连接的年龄:

$$\{(n_1, *)\} = \{(n_1, i) | \forall i \in N_{n_1}\}; age_{(n_1, i)} = age_{(n_1, i)} + 1, \forall (n_1, i) \in \{(n_1, *)\}.$$

Step 4. 更新单元  $n_1$  的积累误差  $\Delta\varepsilon_{n_1} = \|\xi - \omega_{n_1}\|^2$ .

Step 5. 更新  $n_1$  及其拓扑近邻单元的参考向量:  $\Delta\omega_{n_1} = \gamma_b(\xi - \omega_{n_1}); \Delta\omega_i = \gamma_n(\xi - \omega_i), \forall i \in N_{n_1}, \gamma_b, \gamma_n$  分别为获胜单元  $n_1$  及其拓扑近邻的学习率.

Step 6. 若学习次数是单元插入频率  $\lambda$  的整数倍,即  $time \equiv 0 \pmod{\lambda}$ ,则插入新单元:

Step 6.1. 查找积累误差最大的单元  $p$ ,以及  $p$  的拓扑近邻中积累误差最大的单元  $q$ :

$$p = \arg \max_{i \in A} \varepsilon_i; q = \arg \max_{i \in N_p} \varepsilon_i.$$

Step 6.2. 插入新单元  $k$ ,其参考向量和误差按下式确定:

$$\left. \begin{aligned} A &= A \cup \{k\} \\ \omega_k &= \frac{\omega_p + \omega_q}{2} \\ \varepsilon_k &= \frac{\varepsilon_p + \varepsilon_q}{2} \end{aligned} \right\} \quad (1)$$

Step 6.3. 删除连接  $(p, q)$ ,添加新连接  $(p, k)$  和  $(q, k)$ ,并将单元  $p, q, k$  的积累误差降低一个百分比:

$$E = E \setminus (p, q) \cup \{(p, k), (q, k)\}; \Delta\varepsilon_i = -\alpha\varepsilon_i; \forall i \in \{p, q, k\}.$$

Step 7. 所有单元积累误差按一定比例衰减,  $\Delta\varepsilon_i = -\beta\varepsilon_i, \forall i \in A$ .

Step 8. 删除  $\{(n_1, *)\}$  中年龄大于阈值  $\alpha_{max}$  的连接:  $E = E \setminus \{(n_1, i) | age_{(n_1, i)} > \alpha_{max}, \forall i \in N_{n_1}\}$ .

Step 9. 删除连接数为 0 的孤立单元:  $A = A \setminus \{i | N_i = \emptyset, \forall i \in A\}$ .

Step 10. 如果达到设定的终止条件,如网络学习次数  $time > time_{max}$ ,则终止;否则,跳转到 Step 2.

GNG 算法的实质是在神经元的参考向量空间中通过竞争性 Hebbian 学习规则对网络进行变换,使神经元构成的网络在概率上渐进收敛于输入样本空间几何分布和拓扑结构.变换包括两类:几何变换和拓扑变换.几何

变换较为简单,通过 Step 5 移动局部神经元参考向量,从而不断改变神经元在信号空间的感受野,使其收敛为一个局部聚类中心.聚类形成了样本空间的一个向量量化,为减小量化误差,需要通过 Step 6 插入新的聚类中心分配局部误差,Step 9 删除对误差的减小贡献不大的聚类中心.同时,学习过程中的拓扑变换通过 Step 3,Step 4, Step 8 共同完成,具体方式为神经元间连接的插入和删除.通过神经元年龄衰老和更新机制检测应该删除的错误连接,如果一个连接在输入信号的作用下响应频率较高则保留,如果长时间不活跃则删除.

在 GNG 学习过程中应该保持几何变换和拓扑变换过程的协调.神经元位置的变化可能导致拓扑连接失效,若  $\gamma_b, \gamma_n$  两个参数设置得太大,则在训练初始阶段神经元参考向量变化幅度就会较大,而拓扑变换来不及删除由此产生的错误连接,将会导致网络发生自相交叉和大量错误连接.GNG 算法 Step 6 中,新神经元按照固定的频率插入可能产生不合时宜的神经元,如果神经元数量增长过快,不仅增加了查找获胜神经元的负担,而且拓扑变换来不及在新单元附近建立有效的连接,使网络中出现大量空洞.此外,Step 8 通过固定的年龄阈值删除失效的连接,因为无效连接在神经元中的响应频率低,年龄增长快.但是基于年龄的删除机制并不能保证三角网格重构问题的约束(1)和约束(2),若删除的边并不是无效边,就会在三角网格中产生空洞;而且如果后续学习过程中反复插入和删除该边,则会导致神经网络结构震荡,无法收敛.基于以上的分析得出,针对三角网格重构的约束条件,需要对 GNG 算法的神经元插入和连接删除机制进行改进.

### 2.2 重构三角网格

GNG 单元集合  $A$  对应于三角网格节点集  $\{P_i\}$ ,连接集合  $E$  对应边集  $\{L_{ij}\}$ .然而,GNG 没有显式定义三角片结构  $\{F_i\}$ ,需要调整其学习算法或者附加三角片检测步骤重构三角网格.文献[17,18]都是在 GNG 学习完成后通过附加的处理步骤得到满足拓扑约束的三角网格,而本文算法通过修正 GNG 节点插入和连接删除步骤,在学习过程中即保持了拓扑结构的有效性.

网络学习过程中,动态更新一个三角片集合  $T = \{(i,j,k) | (i,j), (i,k), (j,k) \in E\}$ .  $T$  初始化为  $\emptyset$ ,任意三角片  $(i,j,k)$  中不同的单元排列顺序是等价的.与单元  $c$  连接的三角片为该单元的近邻三角片,定义为  $T_c = \{(c,m,n) | \forall m \in A, \forall n \in A, (c,m,n) \in T\}$ ,与连接  $(i,j)$  共边的三角片记为  $T_{ij} = \{(i,j,k) | \forall k \in A, (i,j,k) \in T\}$ .如图 2 所示,根据单元拓扑近邻的数量和近邻三角片的数量,单元分为边界单元  $card(T_c) < card(N_c)$  和内部单元  $card(T_c) = card(N_c)$ ,而根据共边三角片数量可将连接  $(i,j)$  分为流形边  $card(T_{ij}) \leq 2$  和非流形边  $card(T_{ij}) > 2$ .

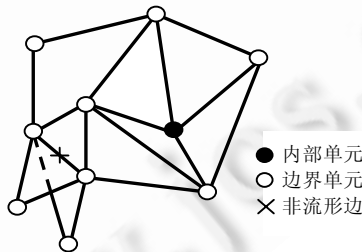


Fig.2 Categorizes of node and edge

图 2 单元类型和边类型

在插入和删除某条边  $(i,j)$  时,同步更新三角片  $T$ ,插入边时有  $T = T \cup \{(i,j,k) | \forall k \in N_i \cap N_j\}$ ,删除边时  $T = T \setminus \{(i,j,k) | \forall k \in A, (i,j,k) \in T\}$ .

为保证  $T$  中的三角片都是由流形边构成,在插入连接时需判断该连接的插入是否会导致非流形边的产生.插入连接  $(i,j)$  后,得非流形边的集合:

$$E_{>2} = \{(m,n) | \forall (m,n) \in \{(i,*)\} \cup \{(j,*)\}, card(T_{m,n}) > 2\} \tag{2}$$

如果  $E_{>2}$  非空,则需删除一条边,从而消除所有非流形边.查找如下的待删除边的集合:

$$\left. \begin{aligned} E_{u,v} &= \bigcup_{\forall (i,j,k) \in T_{u,v}} \{(i,j), (j,k), (i,k)\} \\ E_{del} &= \{(u,v) \mid E_{>2} \neq \emptyset, E_{>2} \subseteq E_{u,v}, \forall (u,v) \in E\} \end{aligned} \right\} \quad (3)$$

其中,  $E_{u,v}$  代表与连接  $(u,v)$  共边的三角片的边的集合,  $E_{del}$  是待删除边的集合. 对于集合中的任意一条边, 与其共边的三角片包含了所有非流形边, 新插入的边  $(i,j)$  也必然属于该集合. 如图 3 所示, 删除集合  $E_{del}$  中年龄最大的一条边, 即可消除所有非流形边. 由于本文的改进算法保存了学习过程中三角片结构  $T$ , 可通过该结构快速查找到集合  $E_{>2}, E_{u,v}$  和  $E_{del}$ , 故通过非流形边立即删除机制代替 GNG 中 Step 8 的年龄阈值删除机制不会增加算法的时间开销. 而且, 基于年龄阈值的连接删除是一种延迟删除机制, 删除的边并不一定是非流形边, 可能因为点云在某处分布密度较低、连接不活跃, 从而导致年龄超过阈值, 这种情况在非均匀分布的点云学习过程中经常出现. 而本文的算法为非流形边原位删除机制, 即在检测到非流形边事件点处删除  $E_{del}$  中年龄最大的边. 由叙述可知, 本文算法仍然保留了年龄机制, 但并不是确定一个年龄阈值来删除非流形边, 而是删除那些能够确保消除非流形边的所有连接中年龄最大的连接, 以此保证训练的结果中没有非流形边.

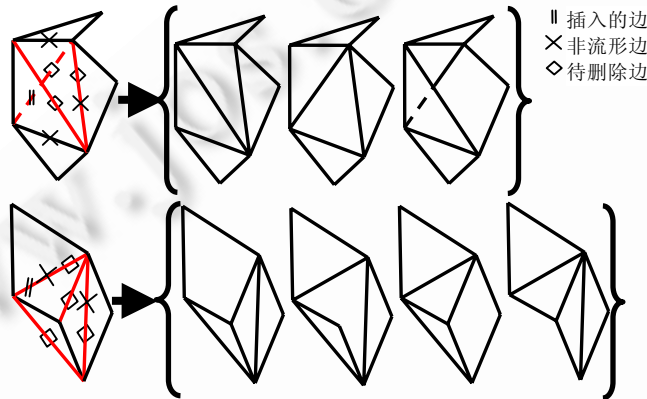


Fig.3 Non-Manifold edge and the deletion strategy

图 3 非流形边的产生及其删除机制

原始 GNG 算法参数包含  $\gamma_b, \gamma_n, \alpha_{max}, time_{max}, \lambda, \alpha$  和  $\beta$ , 后 3 个用于控制单元插入, 而本文改进的算法通过样本点  $\xi$  在单元  $c$  附近局部三角片上的投影判断是否插入新单元, 投影定义为

$$PJ_{\xi,c} = \exists (i,j,k) \in T_c: (N_{i,j,k} \cdot N_{\xi,i,j} \geq 0 \odot N_{i,j,k} \cdot N_{\xi,j,k} \geq 0) \wedge (N_{i,j,k} \cdot N_{\xi,i,j} \geq 0 \odot N_{i,j,k} \cdot N_{\xi,k,i} \geq 0) \quad (4)$$

$$N_{m,n,l} = (n-m) \times (l-n), m,n,l \in A \cup \{\xi\} \quad (5)$$

$N_{m,n,l}$  为  $m,n,l$  这 3 点构成了三角形的法向量,  $\odot$  为与或逻辑运算,  $\wedge$  为逻辑与,  $\cdot$  和  $\times$  分别为矢量点积和叉积.  $PJ_{\xi,c}$  的意义为: 在单元  $c$  的近邻三角片  $T_c$  中是否存在某个三角片  $(i,j,k)$  使点  $\xi$  可垂直投影到该三角片内部. 图 4 所示为  $\xi_1, \xi_2$  两个点在单元  $c_1$  附近三角片的局部投影.  $T_{c_1}$  包含 4 个三角片, 以  $(c_1, c_2, c_3)$  为例, 如图 4(b) 所示, 该三角片的 3 条边及延长线将所在平面分为  $A \sim G$  共 7 个区域,  $A \sim F$  为外部区域,  $G$  为内部区域.  $\xi_1$  投影到内部区域  $G$ , 使得 3 对法向量点积的符号相同, 故  $PJ_{\xi_1, c_1} = \text{TRUE}$ . 如图 4(c) 所示,  $\xi_2$  投影到区域  $C$ , 3 对法矢点积符号相异, 若投影落在其他 5 个外部区域, 也可验证法矢点积相异条件成立. 对于单元  $c$  附近的其他 3 个三角片, 不难验证点  $\xi_2$  都投影到外部区域, 所以  $PJ_{\xi_2, c_1} = \text{FALSE}$ .

原 GNG 算法单元误差更新步骤 Step 4 和 Step 7 不再需要, 单元插入步骤 Step 6 替换为:

Step 6. 若单元数量小于设定的阈值,  $n_1$  为边界单元且当前输入样本不被投影到  $n_1$  所在的局部三角片时, 则插入新单元  $k$ , 并添加  $k$  和  $n_1$  的连接:

$$\text{IF } \text{card}(A) < n_{max} \wedge PJ_{\xi, n_1} = \text{FALSE} \wedge \text{card}(T_{n_1}) < \text{card}(N_{n_1}) \quad (6)$$

$$A = A \cup \{k\}; \omega_k = \xi; E = E \cup \{(n_1, k)\}; \text{age}_{(n_1, k)} = 0$$

由第 2.1 节的分析可知, 固定频率的单元插入机制会导致几何变换和拓扑变换不协调, 而本文的局部投影方法有效地抑制了这一可能. 因为只有在拓扑变换建立了有效的局部三角网格结构时才会计算投影, 判断当前

的局部三角网格是否充分覆盖了点云区域,在点云没有被覆盖的区域加入新的神经元,从而避免 GNG 网络不合时宜的单元插入现象。

GNG 算法 Step 10 中的停止条件修改为:

若三角片数量在经历  $time_{max}$  次学习之后不变或者在小范围内震荡,则认为三角片达到收敛,算法停止.其中,参数  $time_{max}$  代表进行收敛性检验的采样间隔,应随着点云数量变化做适当调整。

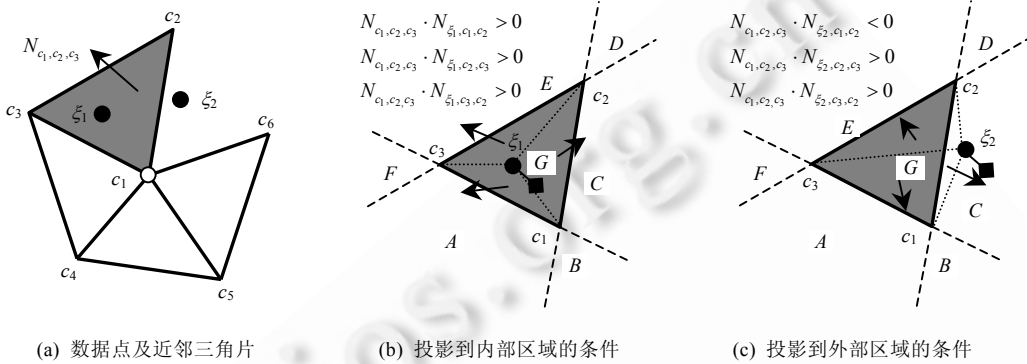


Fig.4 Sample point projection on the local triangles  
图 4 样本点在局部三角片上的投影

### 2.3 三角网格孔洞修补

GNG 采用年龄机制删除单元连接,使三角网格出现孔洞.为控制孔洞的产生,本文算法去除 GNG 中的 Step 8,而采用图 3 所示的非流形边删除机制,只有当新插入的连接导致非流形边产生时,才删除  $E_{del}$  中年龄最大的连接.另外,为使三角网格完全覆盖点云,附加了一个孔洞修补步骤。

孔洞修补学习与前一阶段的步骤类似,学习过程如图 5 所示,近邻查找时除  $n_1, n_2$  外,增加了第 3 近邻  $n_3$ ,连接 3 个近邻中的边界单元并更新单元参考向量,重复学习直到三角片收敛。

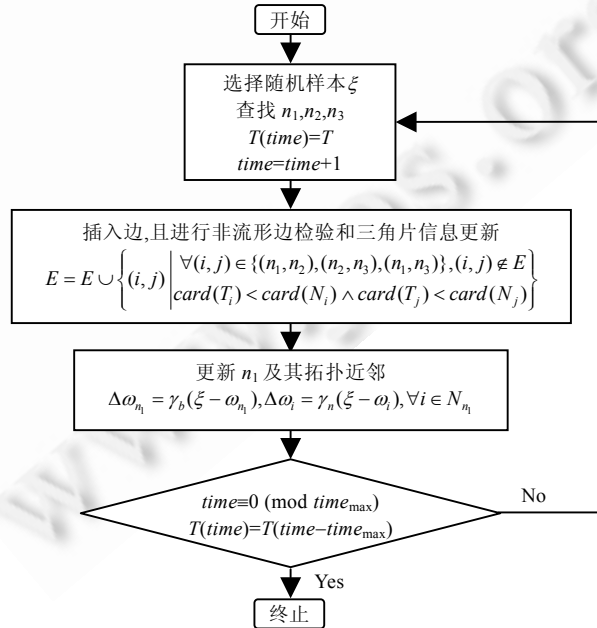


Fig.5 Network learning diagram for gap-fill phase  
图 5 孔洞修补阶段网络学习流程图

由孔洞修补的计算步骤可知,该算法同样是通过 GNG 学习过程修复存在的孔洞,故整个三角网格重构不需要借助除 GNG 之外的其他辅助算法.例如文献[17,18]中,在 GNG 网络学习完成之后还需要进行三角网格检测、无效边删除以及依靠点云在网格上的投影修复存在的孔洞.

### 3 实验与分析

在 Intel Core 3.4GHz CPU,1GB RAM,OS Linux3.0.9,GCC4.6.2 编译器(编译优化级别-O3)平台上采用 C++ 实现本文算法,程序为单线程执行.算例点云来自 AIM@SHAPE 数据库(<http://www.aimatshape.net/>)和斯坦福三维点云库(<http://graphics.stanford.edu/data/3Dscanrep/>).对不同点云进行算例测试,结果见表 1,若无特别说明,计算参数为  $n_{\max}=5000$ ,  $\gamma_b=0.03$ ,  $\gamma_n=0.009$ ,  $time_{\max}=15000$ ,计算时间结果为 3 次平均值.

Table 1 Case study for different point cloud models

表 1 不同点云模型计算结果

| 点云        | 点数量     | 耗时(S) | 三角片    | Euler | 计算参数                               |
|-----------|---------|-------|--------|-------|------------------------------------|
| 1. 兔子     | 34 835  | 8.54  | 9 965  | -2    | $n_{\max}: 10000$                  |
| 2. 恐龙     | 165 955 | 14.59 | 19 979 | 2     |                                    |
| 3. ——     | 25 039  | 8.78  | 9 979  | 2     |                                    |
| 4. 扭结     | 10 001  | 1.15  | 2 998  | 0     | $n_{\max}: 1500$                   |
| 5. 手掌     | 53 055  | 8.27  | 9 943  | 1     |                                    |
| 6. 大象     | 24 955  | 8.61  | 9 972  | -4    | $n_{\max}: 500, time_{\max}: 2000$ |
| 7. ——     | 2 133   | 0.27  | 1 002  | -4    |                                    |
| 8. Mobius | 100 000 | 8.11  | 9 808  | 2     |                                    |
| 9. 头骨     | 20 002  | 9.73  | 9 947  | 2     |                                    |

图 6 是兔子模型重构的结果.如图 6(a)所示,在初始的训练步骤之后,模型表面出现大量微小孔洞区域;而图 6(b)、图 6(c)是在附加的孔洞修复训练步骤之后得到的最终模型,除缺少点云数据的 4 个孔洞区域之外,模型表面不存在其他孔洞.

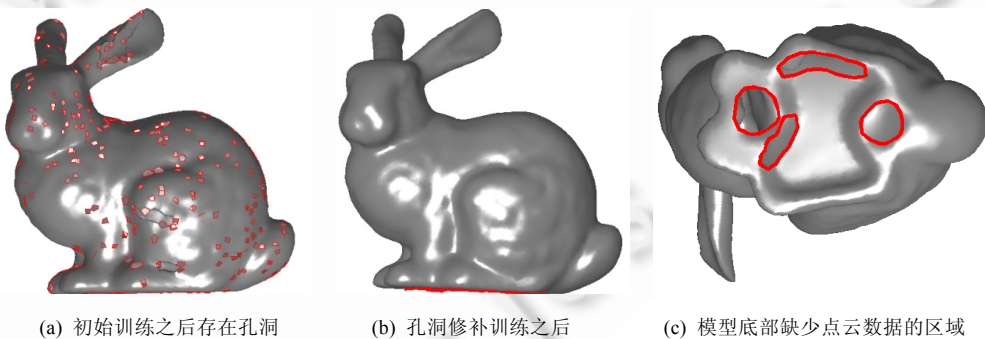


Fig.6 Point cloud 1—triangular mesh reconstruction for Stanford bunny model (light-shading mode)

图 6 点云 1——斯坦福兔子模型三角网格重构结果(光照模型)

图 7(a)~图 7(d)是两个雕塑模型点云数据重构的三角网格,曲面既存在光滑的区域,也存在曲率急剧变化的尖锐部分,采用本文的算法重构出了无任何拓扑错误的三角网格.

图 8(a)是扭结模型重构结果,本文算法正确识别了复杂的扭结缠绕拓扑结构.图 8(b)为一个人体手掌模型,一些重构算法在手指根部容易产生错误的连接<sup>[13]</sup>,而本文算法消除了各个手指之间可能出现的错误拓扑结果.图 8(c)、图 8(d)是两个亏格为 3 的模型重构结果,在大象的象牙和尾部存在 3 个柄.在算法初始化时并未对网格做特殊限定,但重构的网格学习到了正确的亏格数.模型 8 是一个不可定向的 Mobius 环和一个分离球面的组合,如图 8(e)所示,算法可以重构该类不可定向面和分离面.

图 9(a)~图 9(c)的头骨模型中,点云分布不均匀,在牙齿和前额等区域较密集.重构算法在点云密集处自动加

密了网格,如图 9(c)所示,局部特征得到保存.

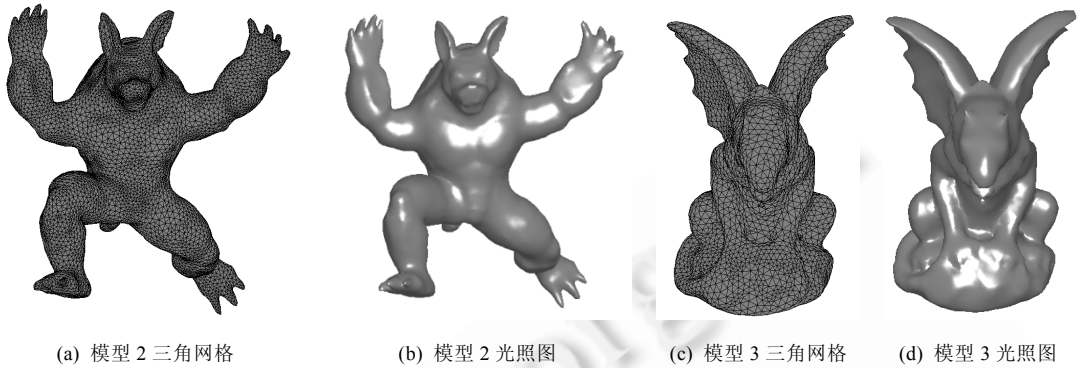


Fig.7 Triangular mesh reconstruction for point cloud of sculpture surface

图 7 雕塑模型点云重构三角网格结果

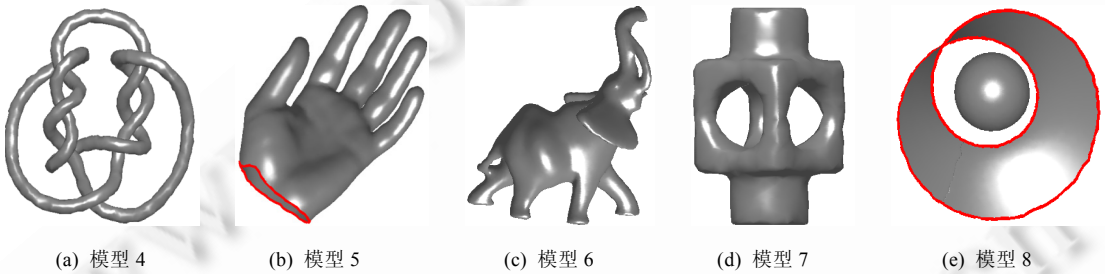


Fig.8 Triangular mesh reconstruction for point cloud of complex surface

图 8 复杂曲面模型点云重构三角网格结果

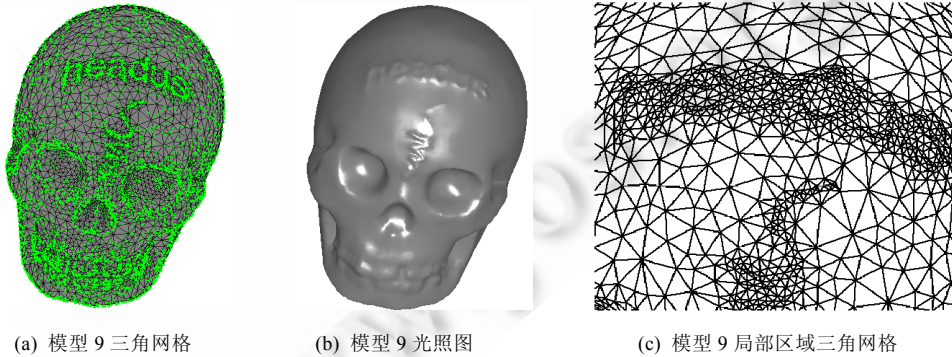


Fig.9 Triangular mesh reconstruction for non-uniform point cloud

图 9 非均匀点云重构三角网格结果

为测试算法对噪声的抑制能力,对兔子模型添加噪声并进行重构,计算参数与表 1 中相同.原始点云  $D_n = \{\xi_1, \xi_2, \dots, \xi_n\}$  含有  $n$  个数据点,对其中  $m$  个点添加噪声,得到新的点云  $D_n$ :

$$D_m = \{\xi_1, \xi_2, \dots, \xi_m\} \in D, m = \alpha \cdot n \tag{7}$$

$$D_n = D \setminus D_m \cup \{\xi_i + \delta_i\}, \forall \xi_i \in D_m, \delta_i \sim U(0, \beta \cdot l) \tag{8}$$

其中,  $\alpha$  代表噪声点的比例;  $\beta$  是噪声强度控制因子;  $l$  是点云最大外包围盒对角线长度;  $\delta_i$  是对各个原始点添加的噪声,在  $[0, \beta l]$  范围内均匀分布.



图 10 所示为不同的噪声参数下的点云、重构结果及局部放大视图.图 10(a)和图 10(e)分别为未添加噪声的点云和重构结果,图 10(i)为图 10(e)中所标志的兔子尾部区域的局部图,比较添加噪声前后的重构可知:当噪声比例为 50%时,在强度为 2%的噪声干扰下,仍然能重构出正确的模型;但是当噪声强度提高到 5%时,如图 10(c)、图 10(g)、图 10(k)所示,重构的三角面存在孔洞、凸起等错误;若减小噪声点比例,如图 10(d)、图 10(h)、图 10(l),则 10%的离群点不会导致拓扑错误.

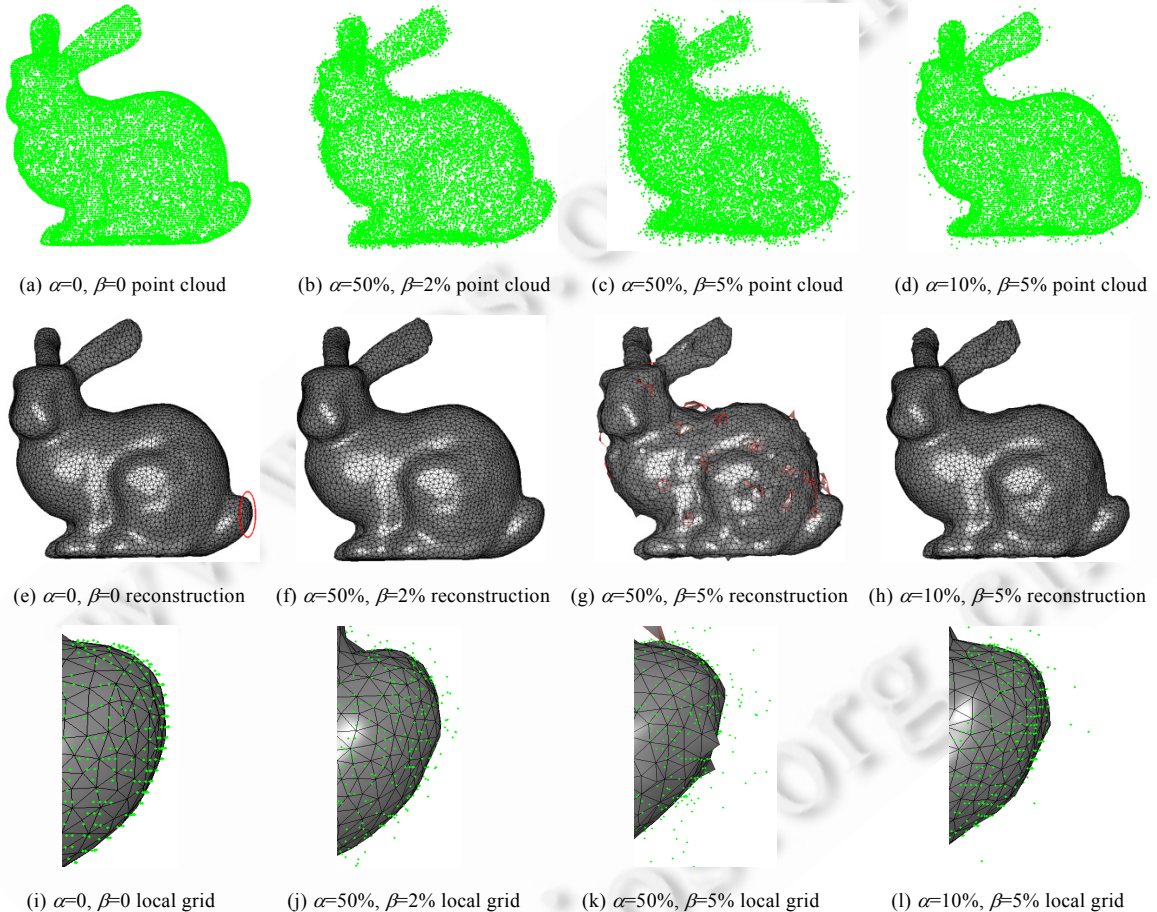


Fig.10 Influence of noisy data on triangular mesh reconstruction

图 10 噪声点云对重构结果的影响

三角网格是曲面的一个近似,其对目标面的逼近程度以及三角网格质量是衡量重构算法的重要指标.本文使用文献[19]中采用的 3 个重构质量评价指标: Hausdorff Distances(HD), Polygon Conformity(PC)和 Valence Distribution(VD). HD 为三角网格到参考面的 Hausdorff 距离,用于评价重构曲面对目标曲面的几何逼近程度,该值越小逼近程度越高; PC 定义为三角片 3 个顶点到其重心距离的最小最大值之比,用于评价三角片的等边性, PC 越大,则越近似于等边三角形; VD 用于评价各个节点连接的均匀性,代表近邻节点数量的分布,是一个拓扑质量评价标准.由表 1 的计算结果可知,本文重构的三角片数量约为节点数量  $n_{\max}$  的两倍,即 VD 的平均值为 6,若投影到平面,则三角片在每个节点的角度约  $60^\circ$ ,近似为等边三角形.

目前常用的三角网格重构方法有基于雕刻的方法如 Alpha Shape、等值面的方法如 Marching Cube、区域增量方法如 Ball Pivoting 等.利用兔子模型比较本文方法和常用算法的计算效率、精度、重构质量,结果见表 2,其中, Alpha Shape, Ball Pivoting 和 Marching Cube 算法来自开源软件 MeshLab.

Table 2 Reconstruction through different methods

表 2 不同算法的重构结果

| 算法                   | 节点数    | 三角片    | 耗时(s) | Euler  | HD        | PC      | VD      |          |
|----------------------|--------|--------|-------|--------|-----------|---------|---------|----------|
|                      |        |        |       |        |           |         | Avg.    | Std dev. |
| GSRM <sup>[19]</sup> | 5 006  | 9 909  | -     | -2     | 0.001 541 | 0.690 2 | 5.959 6 | 0.929 0  |
| Alpha shape          | 33 652 | 48 929 | 4.70  | -8 810 | -         | 0.506 9 | 4.908   | 2.246 6  |
| Ball pivoting        | 34 835 | 68 013 | 7.63  | -19    | -         | 0.606 4 | 5.983 8 | 0.922 8  |
| Marching cube        | 30 313 | 50 848 | 15.14 | -6 290 | 0.002 672 | 0.571 9 | 5.374   | 1.172    |
| 本文算法                 | 5 006  | 9 973  | 8.58  | -2     | 0.001 627 | 0.801 6 | 5.985 2 | 0.750 2  |

由表 2 的结果可知,本文算法和 GSRM 算法重构的三角网格欧拉示性数正确,而其他算法重构结果存在拓扑错误,因为在 AlphaShape 这类算法中并没有特殊的机制保证拓扑有效性,满足欧拉定律是通过对点云密度的先验假设和对算法参数的调整实现的.由于 Alpha Shape 和 Ball Pivoting 在原始点云上生成插值网格,故不能通过 HD 评价.综合比较表 2 中的 5 种算法可知,本文算法的计算精度与 GSRM 都较高,且算法生成的三角网格质量优于其他 4 种算法,PC 均值达到 0.801 6,VD 均值接近于 6,标准偏差最小.

若采用原始 GNG 算法,存在众多调节参数,对不同的点云数据,选择合理的单元插入频率、节点误差降低比例等参数<sup>[19]</sup>较困难.本文算法只包含 4 个参数,其中, $n_{max}$ 用于调节三角片网格节点数量,在计算过程中设置不同的参数  $n_{max}$ ,可以获得不同尺度的三角网格模型.如图 11(a)所示,当单元数量较少时,不能重建曲面的细节特征,随着单元数量增加,如图 11(b)~图 11(d)所示,可以得到更精细的网格.

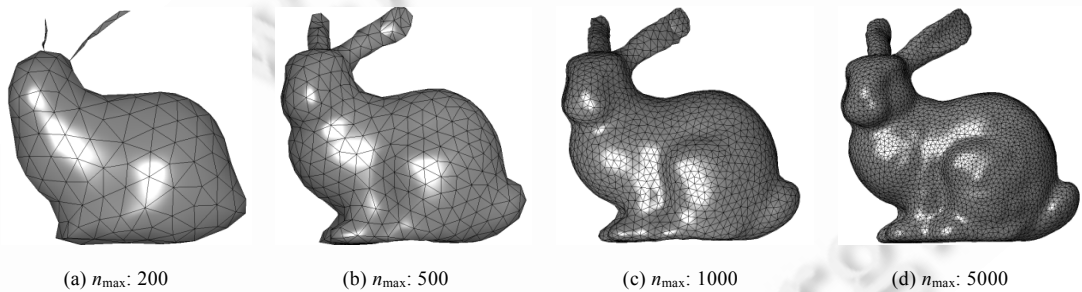


Fig.11 Point cloud 1—triangular mesh reconstruction for Stanford bunny surfaces under different node sizes

图 11 点云 1——兔子模型不同尺度重构结果

为了检验三角片数量对重构质量的影响,对点云 1 在不同的  $n_{max}$  取值下进行测试.从图 12 中可以看出: $n_{max}$  的增加不会对重构结果的 VD 值造成影响,VD 的均值始终保持接近于 6;而几何逼近误差 HD 和三角片的等边性 PC 随着  $n_{max}$  的增大而减小.

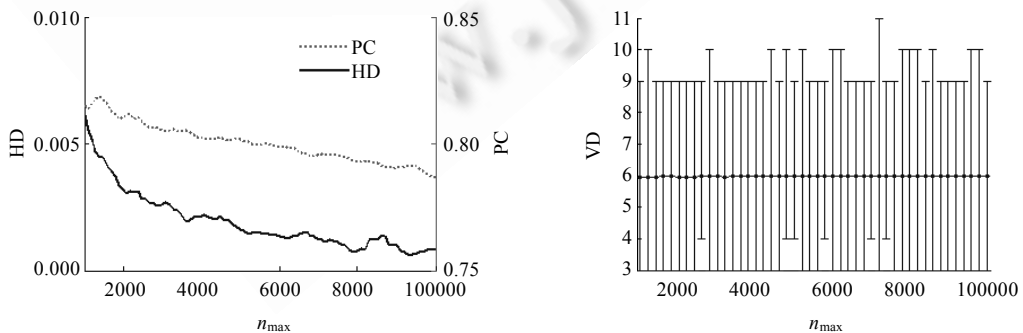


Fig.12  $n_{max}$  influence on reconstructed mesh quality

图 12  $n_{max}$  的取值对重构质量的影响

网络的两个学习率参数 $\gamma_b$ 和 $\gamma_n$ 从SOM网络和GNG网络继承而来,其大小代表激励信号在获胜单元和周边单元引起的变化幅度.直观上,可将神经元构成的网络看做一张弹性网格,输入信号的作用等同于在局部点施加作用力使该点发生形变.但局部点由于连接的存在而相互牵制,受力点无法独立移动,所以由网格形变产生的内应力也带动周围点移动. $\gamma_b$ 和 $\gamma_n$ 越小,网格越不易变形.为讨论学习率对重构结果以及计算时间的影响,在不同取值下对斯坦福兔子模型进行重构,得到如图13所示的结果.由图可知:当 $\gamma_b$ 保持不变、 $\gamma_n$ 相对增加时,会使重构结果的几何精度降低,但值增加,训练时间也相对缩短;若 $\gamma_b$ 与 $\gamma_n$ 取值比例相对恒定,适当提高 $\gamma_b$ 有利于降低几何误差,且三角网格等边性得到改进,但收敛时间有所延长.

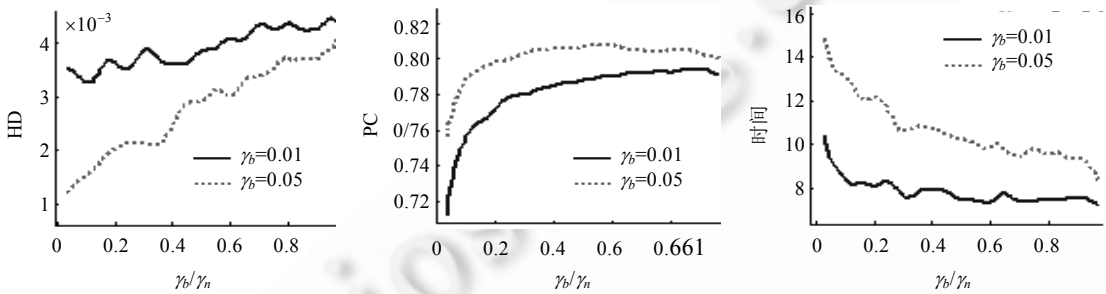


Fig. 13  $\gamma_b$  and  $\gamma_n$  influence on reconstructed mesh quality and time cost

图 13  $\gamma_b, \gamma_n$  对重构质量和计算时间的影响

## 4 结论

本文讨论了一种基于GNG神经网络的点云重构三角网格算法,改进了GNG网络中的神经元插入和连接删除机制.其重构时间复杂度与点云数量无关,只受最大节点数 $n_{\max}$ 控制,该参数的调整可生成不同尺度和精度的网格.通过算例检验了算法对噪声的鲁棒性.同时,本文算法能够在无先验假设的前提下学习高亏格曲面、开曲面、不可定向曲面等复杂的拓扑结构.与文献报道的三角网格重构算法相比较,该方法重构的网格出度均匀,近似为等边三角形,且对待重构曲面的逼近精度较高.在保证重构精度及拓扑正确性的前提下进一步提高算法的时间效率,将是后续研究的主要内容之一.

**致谢** 在此,我们向对本文的工作给予支持和建议的所有审理专家和同行表示由衷的感谢.

## References:

- [1] Zhou RR, Zhang LY, Su X, Zhou LS. Algorithmic research on surface reconstruction from dense scattered points. Ruanjian Xue bao/Journal of Software, 2001,12(2):249–255 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/20010213.htm>
- [2] Wu JJ, Wang QF, Huang YB, Zhou J. Review of surface reconstruction methods in reverse engineering. Journal of Engineering Graphics, 2004,25(2):133–142 (in Chinese with English abstract).
- [3] Nie JH, Hu Y, Ma Z. Outlier detection of scattered point cloud by classification. Journal of Computer Aided Design & Computer Graphics, 2011,23(9):1526–1532 (in Chinese with English abstract).
- [4] Hoffmann M, Varady L. Free-Form surfaces for scattered data by neural networks. Journal for Geometry and Graphics, 1998,2(1): 1–6.
- [5] Hoffmann M. Numerical control of kohonen neural network for scattered data approximation. Numerical Algorithms, 2005,39(1): 175–186. [doi: 10.1007/s11075-004-3628-7]
- [6] Wang HT, Zhang LY, Li ZW, Liu SL, Zhou RR. B-Spline surface reconstruction from scattered data points based on SOM neural network. Journal of Image and Graphics, 2007,12(2):349–355 (in Chinese with English abstract).
- [7] Cheng YL, Yen TK, Jau CH. Conformal self-organizing map on curved seamless surface. Neurocomputing, 2008,71(16): 3140–3149. [doi: 10.1016/j.neucom.2008.04.031]
- [8] Kumar GS, Kalra PK, Dhande SG. Curve and surface reconstruction from points: An approach based on self-organizing maps. Applied Soft Computing, 2004,5(1):55–66. [doi: 10.1016/j.asoc.2004.04.003]

- [9] Brito A, Doria A, de Melo J, Garcia L. An adaptive learning approach for 3-D surface reconstruction from point clouds. *IEEE Trans. on Neural Networks*, 2008,19(6):1130–1140. [doi: 10.1109/TNN.2008.2000390]
- [10] Fritzke B. Growing cell structures—A self-organizing network for unsupervised and supervised learning. *Neural Networks*, 1994, 7(9):1441–1460. [doi: 10.1016/0893-6080(94)90091-4]
- [11] Fritzke B. A growing neural gas network learns topologies. In: *Proc. of the Advances in Neural Information Processing Systems 7*. Cambridge: MIT Press, 1995. 625–632.
- [12] Wang SD, Zhang YS, Ou CS, Xie Y. A rapid algorithm for surface reconstruction based on growing cell structures. *Journal of Hefei University of Technology*, 2006,29(8):984–987 (in Chinese with English abstract).
- [13] Ivrišimtzis I, Lee Y, Lee S, Jeong WK, Seidel HP. Neural mesh ensembles. In: *Proc. of the 2nd Int'l Symp. on 3D Data Processing, Visualization and Transmission*. Los Alamitos: IEEE Computer Society Press, 2004. 308–315. [doi: 10.1109/3DPVT.2004.87]
- [14] Saleem W, Schall O, Patane G, Belyaev A, Seidel HP. On stochastic methods for surface reconstruction. *The Visual Computer*, 2007,23(6):381–395. [doi: 10.1007/s00371-006-0094-3]
- [15] Mari JF, Saito JH, Poli G, Levada ALM. Improving the neural meshes algorithm for 3D surface reconstruction with edge swap operations. In: *Proc. of the 2008 ACM Symp. on Applied Computing*. New York: ACM Press, 2008. 1236–1240. [doi: 10.1145/1363686.1363971]
- [16] Annuth H, Bohn CA. Surface reconstruction with smart growing cells. In: Plemenos D, Miaoulis G, eds. *Proc. of the Intelligent Computer Graphics 2010*, Vol.321. Berlin: Springer-Verlag, 2010. 47–66. [doi: 10.1007/978-3-642-15690-8\_3]
- [17] Melato M, Hammer B, Hormann K. Neural Gas for Surface Reconstruction. Clausthal-Zellerfeld: Clausthal University of Technology, Department of Informatics, 2007.
- [18] Holdstein Y, Fischer A. Three-Dimensional surface reconstruction using meshing growing neural gas (MGNG). *The Visual Computer*, 2008,24(4):295–302. [doi: 10.1007/s00371-007-0202-z]
- [19] Rego RLME, Araujo AFR, de Lima Neto FB. Growing self-reconstruction maps. *IEEE Trans. on Neural Networks*, 2010,21(2): 211–223. [doi: 10.1109/TNN.2009.2035312]
- [20] DalleMole VL, Araujo AFR. The growing self-organizing surface map. In: *Proc. of the Int'l Joint Conf. on Neural Networks*. Hong Kong: IEEE Press, 2008. 2061–2068. [doi: 10.1109/IJCNN.2008.4634081]

#### 附中文参考文献:

- [1] 周儒荣,张丽艳,苏旭,周来水.海量散乱点的曲面重建算法研究.软件学报,2001,12(2):249–255. <http://www.jos.org.cn/1000-9825/20010213.htm>
- [2] 武剑洁,王启付,黄运保,周济.逆向工程中曲面重建的研究进展.工程图学学报,2004,25(2):133–142.
- [3] 聂建辉,胡英,马孜.散乱点云离群点的分类识别算法.计算机辅助设计与图形学学报,2011,23(9):1526–1532.
- [6] 王宏涛,张丽艳,李忠文,刘胜兰,周儒荣.基于 SOM 的散乱数据点集的 B 样条曲面重建.中国图像图形学报,2007,12(2):349–355.
- [12] 王世东,张佑生,偶春生,谢颖.一种基于成长型神经网络的曲面重建快速算法.合肥工业大学学报,2006,29(8):984–987.



曾锋(1980—),男,广东梅州人,副教授,主要研究领域为计算机图形学,逆向工程.  
E-mail: zengfeng@jyu.edu.cn



姚山(1966—),男,教授,博士生导师,主要研究领域为计算机集成制造,计算机图形学.  
E-mail: yaoshan@dlut.edu.cn



杨通(1988—),男,硕士生,主要研究领域为逆向工程,人工神经网络.  
E-mail: ytami@126.com