

## 基于向量场距离函数的网孔模型修复

王 俐, 吴禄慎

(南昌大学机电工程学院, 江西 南昌 330031)

**摘要:**提出了一种基于向量场距离函数的网孔模型修复算法,用于修复各种多边形网孔模型。可用于激光三维建模等应用中,作为一种体修复算法,其鲁棒性好,对于任意带缺陷的输入网孔模型,经修复后均能得到无缺陷的封闭且法向一致的三角形网孔输出。修复过程中,体模型用向量场距离函数表征,该表征方法比传统的标量场距离函数更能保留模型的原始特征。此外,在提取输出表面轮廓时,对传统MC算法中求取立方体网格与形面交点的插补算法作了改进。理论和实践证明,改进后插补点的位置更接近网格与形面交点的实际位置,因而提取出来的网孔模型更为精确,比传统方法精确约16.8%。

**关键词:**模型修复;向量场距离函数;插补算法;体元化

**中图分类号:**TP391 **文献标识码:**A **DOI:**10.3969/j.issn.1001-5078.2013.12.18

## Mesh model repair based on vector field distance function

WANG Li, WU Lu-shen

(School of Mechatronic Engineering, Nanchang University, Nanchang 330031, China)

**Abstract:** A mesh model repairing algorithm based on vector field distance function is presented. It is fit for all kinds of polygonal mesh. As a volumetric repair method, it is very robust. For every flawed input mesh model, it can output an error-free and closed triangular mesh with coherent orientations. In the algorithm, the transformed volumetric model is represented by vector field distance function instead of scalar field distance function, it can keep much more features of the original mesh than the traditional scalar field distance function. In addition, when contouring output surface, the interpolation algorithm in the traditional MC method is improved to adapt it to new vector field distance function. Proved by theory and practice, the calculated interpolation position is much closer to the actual position of the mesh model and 3D grids. The extracted triangular mesh is 16.8% more accurate than traditional method.

**Key words:** model repair; vector field distance function; interpolation algorithm; voxelization

### 1 引言

逆向工程数据采集过程中,由于测量技术(三坐标测量仪、激光扫描仪等)及被测物体自身形面的复杂性和材质等原因,采集到的物体表面数据往往是不完整的,存在着孔洞、间隙、重叠等多种缺陷或数据丢失现象,这将给后续的曲面重构带来极大地麻烦。

为此,研究者们提出了许多修复不完整数据模型的方法,这些方法主要分成两大类:基于网孔模型的修复法(Mesh-based model repair)和体模型修复法(Volumetric model repair)。基于网孔模型的修复法直接在多边形网孔模型上进行操作以修复模型中

各种各样的几何与拓补缺陷,比如孔洞的填补<sup>[1-2]</sup>、缝隙的缝合<sup>[3-4]</sup>等。由于只要对有缺陷的局部网孔进行修复,网孔模型修复法的优点是能够最大限度的维持原物体表面的几何形貌。其缺点是鲁棒性差,有时会出现拓补结构上的歧义性,且难以确保其输出网孔模型的封闭性。体模型修复法将被修复网

**基金项目:**国家自然科学基金资助项目(No. 51065021)资助。

**作者简介:**王 俐(1968-),女,副教授,博士研究生,主要从事三维光学图像检测及处理的研究。E-mail:1239285141@qq.com

**收稿日期:**2013-05-17

孔模型置于某个 3D 网格中,得到网孔模型的相应体模型数据,之后再用 MC 算法<sup>[5]</sup>从该网格的体数据中重构原网孔模型。经过这一系列步骤后,原网孔模型中的缺陷(包括几何体的重叠和嵌套)均得以自动修。体修复法鲁棒性好,且能确保输出多边形网孔模型的封闭性和各网孔方向的一致性。其缺点是处理速度慢,体元转换需要大量的存储空间;可能会失去原网孔模型中一些尖锐的形貌特征,且输出的三角形网孔数量庞大,常常需要做后续的简化处理。无论使用哪种修复方法,鲁棒性和精确性都是两重要的修复指标。

本文提出了一种基于向量场距离函数的网孔模型修复算法,该算法属于体修复方法,但它在保证算法鲁棒性的同时,通过使用向量场距离函数及对 MC 算法中插补方法的改进,最大限度地保留了被修复网孔模型的原始特征。算法以物体的三角形网孔模型作为输入,将其置入三维体网格中,创建其体网格模型——向量场距离函数模型,替代传统的标量场距离函数模型。最后,采用改进的 MC 算法从网格体模型中提取物体表面轮廓(三角形网孔)作为输出。对于任意带缺陷的输入网孔模型,该算法总能输出无缺陷的准确、封闭且方向一致的三角形网孔模型,鲁棒性好,特别适合于 CAD 模型的修复。

## 2 体修复算法

### 2.1 修复原理

体模型修复法的基本思想如图 1 所示:修复过程中关键是输入模型的体元化,即将多边形网孔模型转换为体模型。体模型通常用一个三变量标量值场函数表示,该函数表征了 3D 空间中某点到输入多边形网孔模型的最短距离。建立体模型后,再用诸如 Marching Cube 类的等值面提取方法从该体模型重构多边形网孔模型作为输出。原输入多边形网孔模型中存在的各种缺陷会在体元化及网孔重构的过程中得以自动修复。

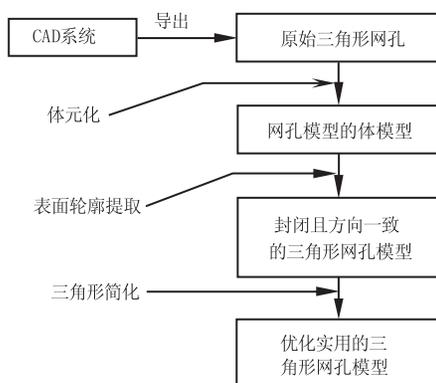


图 1 体模型法修复过程

### 2.2 输入模型的体元化

体元化就是将多边形网孔模型转化为体模型。体模型通常用符号场距离函数来表征,空间任意一点的场距离函数定义为该点到网孔模型的最短距离,而场距离函数的符号决定了该点在被表征网孔模型的内部还是外部,这样,符号场距离函数的零水平集即为被表征多边形网孔模型的逼近。为了能得到一个封闭的输出表面,关键是确定 3D 空间中各网格顶点场距离函数的符号,体元化方法也往往取决于场距离函数符号的确定方法。

符号产生的方法很多,较经典的是 Nooruddin and Turk 的奇偶计数法和投影法<sup>[6]</sup>,他们从各网格顶点向模型投射光束,根据光束与模型交点个数的奇偶性或位置来决定各网格顶点的符号。但是该符号产生方法只适合于均匀网格,而且投影是全局性的操作,因此,体模型的分辨率会受到限制。最普遍的方法是空间划分法,根据各网格顶点在网孔模型的内部还是外部决定距离函数的符号,Friskin et al<sup>[7]</sup>计算各网格顶点到输入多边形的最短符号距离来实现网孔模型的体元化。近几年,已有人开始使用计算机图形硬件来实现模型的体元化。G. Passalis, I. A. Kakadiaris<sup>[8]</sup>利用计算机图形硬件中的深度和模板缓存来实现网孔或参数表面的体元化。Thiago Bastos<sup>[9]</sup>、Llamas, Ignacio<sup>[10]</sup>等利用 GPU 产生符号场距离函数实现网孔模型的体元化。这些方法增加了硬件开销,而且与 z 缓冲区垂直的那些三角形面片处会产生一些小瑕疵,尤其是在网格分辨率较低的情况下。

本文使用空间划分法建立网孔的体模型,不同于传统方法的是,表示体模型的场距离函数不再是标量场距离函数,而是向量场距离函数。与标量场数据函数一样,常常在 3D 立方体网格上定义向量场距离函数。网格中的每个立方体称作一个单元,立方体的各个顶点则称作像素,如图 2 所示。对于网格中的每个像素,寻找三角形网孔模型中离它最近的点,并计算两者间的距离。传统的标量场距离函数表示中,每个像素只保存该像素到三角形网孔模型最近点间的距离,而在向量场距离函数表示中,保存的是该像素指向三角形网孔模型上离它最近点的 3D 向量。此外,还用逻辑量来表征各个像素在隐含物体表面的内部还是外部,这对之后用 MC 算法提取物体表面尤为重要。

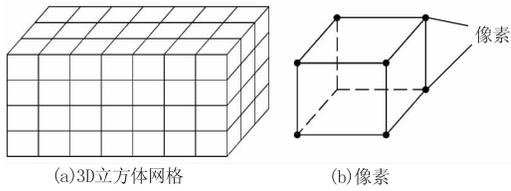


图2 3D立方体网格及其像素

算法中,通过求取网孔模型上距某像素最近点所在三角形法向量与之前保存的该像素到网孔模型最近点的3D向量的点积,来判断该像素位于隐含表面的内部还是外部:如果该点积的值为正,则该像素位于表面内部(比如图3(b)中的 $V_5$ ),否则,位于表面外部(比如图3(b)中的 $V_1$ )。

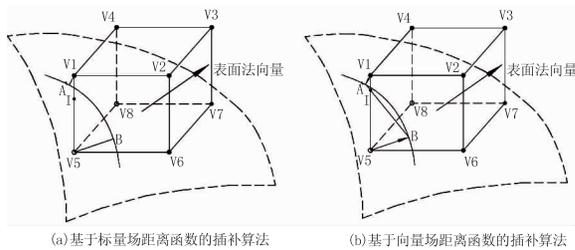


图3 确定立方体边与物体表面交点的插补算法

向量场距离函数不仅记录了各像素到隐含表面的最短距离,还记录了各像素到该最短距离点的确切方向,因而比传统的标量场距离函数更加精确、完善。

### 2.3 网孔重构

理论上,距离函数的零水平集  $f(x, y, z) = 0$  即为所表征的曲面。在标量场距离函数中,以距离为0来提取物体表面,而在向量场距离函数中则是以向量的幅值为0来提取物体表面。在实际应用中,由于离散采样及网格分辨率的缘故,往往要设定一阈值  $\delta$  来表征场距离函数的零水平集。若  $|\hat{f}(x, y, z)| \leq \delta$ , 则点  $(x, y, z)$  属于物体表面。

网孔重构就是通过轮廓提取算法从带符号体模型中提取其零等值面作为原三角形网孔模型的逼近。MC算法是最经典的轮廓提取算法,但它是基于标量场距离函数设计的。本文对MC算法中相应立方体边上的线性插补算法稍作了改动,成功地用它从向量场距离函数中提取出了体模型的三角形网孔逼近。

根据MC算法,哪些立方体网格单元需要三角化,其中的哪些边需要进行插补运算,取决于立方体网格的各个顶点(像素)在被重构物体表面的内部还是外部。对于那些8个顶点都位于表面内部或外部的立方体网格单元,物体表面是不经过的,无需三

角化。只有那些8个顶点部分位于隐含表面内部,部分位于隐含表面外部的立方体网格才要进行插补运算以确定三角化三角形的顶点。具体方法是:对于该立方体网格中的每条边,若其两个顶点都在隐含表面的内部或都在隐含表面的外部,则无需插补;若其中一个顶点位于隐含表面内部,另一个顶点位于隐含表面外部,则该条边上一定存在一零水平集的点,即该边与隐含表面一定存在一个交点,需要用插补算法确定该交点的位置,作为后续三角化时的顶点之一。

图3给出了基于标量场距离函数的插补算法(a)与基于向量场距离函数的插补算法(b),并将两者进行了对比。图3中 $V_1 \sim V_8$ 是需要插补的某立方体网格的8个顶点,其中 $V_5, V_8$ 位于待重构表面的内部,其余顶点位于待重构表面的外部,因此,物体表面必经过边 $V_1V_5, V_5V_6, V_4V_8$ 和 $V_7V_8$ 。以下以 $V_1V_5$ 边为例,介绍两种不同体模型表征下边的插补算法。在标量场距离函数表示中, $D(V_1), D(V_5)$ 分别为像素 $V_1, V_5$ 到隐含表面的最短标量距离,插补算法用两者的比值来粗略地确定 $V_1V_5$ 边上插补点 $I$ 的位置,即  $\frac{D(V_1)}{D(V_5)} = \frac{V_1I}{IV_5}$ ;在向量场距离函数表示中, $A, B$ 分别是隐含表面上距离像素 $V_1, V_5$ 最近的点,向量 $\vec{V_1A}, \vec{V_5B}$ 即为向量场距离函数表示中像素 $V_1, V_5$ 中所保存的向量。与基于标量场距离函数的MC算法相同,从向量场距离函数中提取物体表面,也要在边 $V_1V_5$ 上寻找一个插补点 $I$ ,不同的是,此处用 $AB$ 与 $V_1V_5$ 的交点作为插补点(如果 $AB$ 与 $V_1V_5$ 在同一平面上的话),如图3(b)所示。但在实际的3D情形中,边 $V_1V_5$ 与线段 $AB$ 往往是异面的,因此,插补算法在边 $V_1V_5$ 上寻找距离线段 $AB$ 最近的点作为最终的插补点 $I$ 。

求以上插补点 $I$ 的方法很多,典型的方法是投影法,投影法需要确定投影平面,还要进行叉积运算。下面给出一种更容易实现的方法,该方法仅需使用向量 $\vec{V_1V_5}, \vec{AB}, \vec{V_1A}$ 以及向量间一些简单的点积运算,便可算得插补点 $I$ 分有向线段 $V_1V_5$ 所成的比例 $\lambda$ ,从而算得插补点 $I$ 的坐标。直线 $V_1V_5$ 与 $AB$ 异面,若用 $\vec{\alpha}$ 表示两线段上距离最近的点所构成的向量,则 $\vec{\alpha}$ 既垂直于 $V_1V_5$ ,也垂直于 $AB$ ,故 $\vec{\alpha} \cdot \vec{V_1V_5} = 0, \vec{\alpha} \cdot \vec{AB} = 0$ ,又考虑到 $V_1V_5, AB$ 为线段,根据文献[11]推得 $V_1V_5$ 上距 $AB$ 最近的点(即插补点 $I$ )分有向线段 $V_1V_5$ 所成的比例:

$$\lambda = \frac{(\overrightarrow{V_1V_5} \cdot \overrightarrow{AB})(\overrightarrow{AB} \cdot \overrightarrow{V_1A}) - (\overrightarrow{AB} \cdot \overrightarrow{AB})(\overrightarrow{V_1V_5} \cdot \overrightarrow{V_1A})}{(\overrightarrow{V_1V_5} \cdot \overrightarrow{V_1V_5})(\overrightarrow{AB} \cdot \overrightarrow{AB}) - (\overrightarrow{V_1V_5} \cdot \overrightarrow{AB})^2}$$

该公式也适用于一些特殊的情形,比如向量  $\overrightarrow{V_1V_5}$  与  $\overrightarrow{AB}$  平行时,式中分母为 0,  $\lambda$  为无穷大,此时,插补点  $I$  取为  $V_5$ 。对于其他  $\lambda \notin [0, 1]$  的情形,  $I$  既可取为  $V_1$ , 也可取为  $V_5$ 。

图 3 从理论上表明,在同样的网格分辨率下,基于向量场距离函数的插补算法算得的插补点  $I$  比基于标量场距离函数的插补算法算得的插补点更接近原始曲面与立方体网格边  $V_1V_5$  的交点,因此更加精确。

为证实这点,针对图 4(a) 中阶梯函数的三角形网孔,在同样地网格分辨率及相同的空间坐标下,分别采样、计算其标量场距离函数数学模型与向量场距离函数数学模型,然后用标准的 MC 算法提取该两场距离函数模型所表征的曲面,得到图 4(b) 基于标量场距离函数提取的三角形网孔、图 4(c) 基于向量场距离函数提取的三角形网孔所示结果。由图可见:由阶梯函数标量场距离函数提取的三角形网孔,图 4(a) 中阶梯函数的四个特征点 A、C、D、F,只有 A 点被保留下来,另外,在采样点 B、E 处函数也有些变形;而由阶梯函数向量场距离函数提取的三角形网孔,除特征点 C 处由于分辨率低的原因稍有变形外,基本保持了阶梯函数的原样。可见,向量场距离函数能更精确地表征物体的形面特征。最后,用误差估算函数  $\varepsilon = \sqrt{\frac{1}{3} \sum_i \text{dist}(v_i, R)^2 \cdot \frac{\sum A(v_i)}{A(M)}}$  计算两提取结果 (b)、(c) 相对于原始阶梯函数 (a) 的重构误差,分别为 0.433 和 0.360。显然,基于向量场距离函数提取的网孔比基于标量场距离函数提取的网孔要精确约 16.8%。

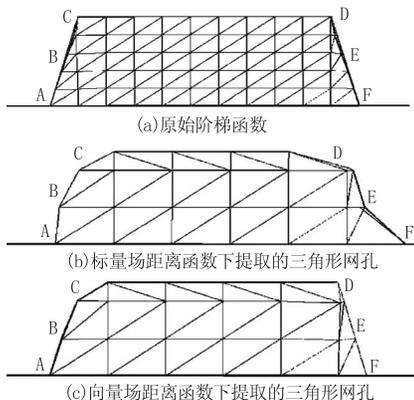


图 4 阶梯函数及其三角化网孔

### 3 算法的实现

第一步,体元化,即将三角形网孔模型转换为用向量场距离函数表示的体模型。具体做法如下:将三角形网孔模型置于具有一定分辨率的均匀 3D 立方体网格中,对于网格中各立方体的顶点,寻找三角形网孔模型中距离它最近的一点,用有向线段(由各立方体顶点指向三角形网孔模型中距离它最近的点)连接该两点,便得到立方体网格各顶点处的向量场距离函数值,从而实现了三角形网孔模型的体元化。

第二步,网格顶点符号的确定。网格顶点符号的确定是确保输出表面封的中至关重要的一步,通过求取输入网孔模型上距某像素最近的点所在三角形面片的表面法向量与之前保存的该像素到输入网孔模型最近点的 3D 向量的点积,来判断该像素位于隐含封闭表面的内部还是外部:如果该点积的值为正,则该像素位于表面内部,否则,位于表面外部。

第三步,确定了 3D 网格各立方体顶点的符号后,一个封闭的将不同符号的立方体网格顶点分在内外两边的隐含表面也就确定下来了。此时,可用线性插补法计算各立方体的边(顶点有符号变化的)与三角形网孔模型的交点,作为后续输出表面三角化的顶点。

第四步,以上步中计算出的各个插补点位顶点,用 MC 算法提取隐含表面的轮廓得到系统的最终输出——消除了各种缺陷的、封闭且法向量一致的三角形网孔表面模型。

### 4 实例(结果)

图 5 为用文中算法对兔 CAD 模型修复的结果。图 5(a) 为用 3D 扫描仪获取的兔模型的原始网孔模型,由于模型自身的缺陷和扫描仪的局限性,原始网孔模型中通常都存在间隙、孔洞及自交三角形网孔等缺陷,分别如图 5(b)、图 6(a) 和图 5(c) 细节所示。图 5(d)、(e) 为用基于向量场距离函数的体修复算法对兔模型间隙和自交网孔修复后的结果,与图 5(a)、(c) 相比,原始网孔中的间隙与自交现象都得以消除,得到的是封闭、无自交的三角形网孔。本算法不仅可以消除间隙,对于大的孔洞也有很好的修复效果。图 6(a) 为原始兔模型的底部,存在着几个较大的孔洞,图 6(b) 为用文中算法修复后的结果,不仅孔洞被填补上了,而且修复后的封闭网孔在网格分辨率的允许下尽可能地保存了模型的原貌特征。

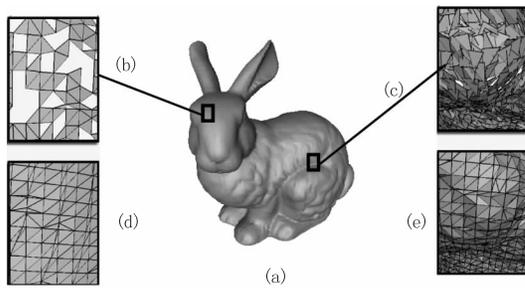


图5 间隙、自交及其修复

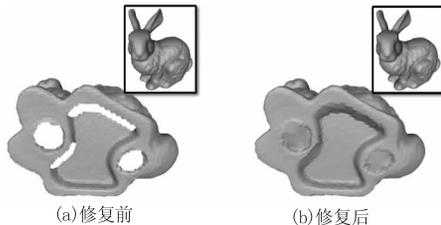


图6 孔洞的修复

## 5 结论

本文给出了一种基于向量场距离函数的三角形网孔模型体修复算法,该方法鲁棒性好,对于任意给定的三角形网孔模型,经本方法修复后,总能够得到封闭且法向量一致的表面模型。由于算法中体模型使用向量场距离函数来表征,相对于传统的标量场距离函数而言,修复的结果更能体现形面的原貌特征。

## 参考文献:

- [1] Jun, Yongtae. A piecewise hole filling algorithm in reverse engineering[J]. CAD Computer Aided Design, 2005, 37(2):263 - 270.
- [2] Chen haoho, Sung Chunyi, Chen Tsongyi, et al. An efficient hole-filling approach using adaptive rendering[C]. Proceedings - 2012 6th International Conference on Genetic and Evolutionary Computing, 2012:336 - 339.
- [3] Borodin P, Novotni M, Klein R. Progressive gap closing for mesh repairing[J]. Advances in Modelling, Animation and Rendering, 2002:201 - 213.
- [4] H T Yau, C C Kuo, C H Yeh. Extension of surface reconstruction algorithm to the global stitching and repairing of STL models[J]. Comput. Aided Design, 2003, 35: 477 - 486.
- [5] Lorensen W, Cline H. Marching cubes: A high resolution 3d surface construction algorithm[J]. Computer Graphics (Proceedings of ACM SIGGRAPH 87), ACM Press, 1987, 21:163 - 169.
- [6] NOORUDDIN F S, TURK G. Simplification and repair of polygonal models using volumetric techniques[J]. IEEE Transactions on Visualization and Computer Graphics. 2003, 9(2):191 - 205.
- [7] Frisken S F, Perry R N, Rockwood A P, et al. Adaptively sampled distance fields: A general representation of shape for computer graphics[C]. Proceedings of ACM SIGGRAPH 2000, ACM SIGGRAPH, New York. E. Fiume, Ed., Computer Graphics Proceedings, Annual Conference Series, ACM, K. Akeley, Ed., 2000:249 - 254.
- [8] Passalis G, Kakadiaris I A, Theoharis T. Efficient hardware voxelization[C]. Proceedings of Computer Graphics International Conference, CGI. 2004:374 - 377.
- [9] Bastos Thiago, Celes Waldemar. GPU-accelerated adaptively sampled distance fields[C]. IEEE International Conference on Shape Modeling and Applications 2008, Proceedings, SMI. 2008:171 - 178.
- [10] Llamas, Ignacio. Real-time voxelization of triangle meshes on the GPU[C]. ACM SIGGRAPH 2007 Sketches, SIGGRAPH'07, 2007.
- [11] Math Encyclopedia Compiler Committee. Math encyclopedia[M]. Beijing: Science Publishing Press, 1995. (in Chinese)  
数学百科全书编译委员会. 数学百科全书[M]. 北京: 科学出版社, 1995.