

## 有效保留模型特征的自适应分层算法

李文康<sup>1,2</sup>, 陈长波<sup>1\*</sup>, 吴文渊<sup>1</sup>

(1. 中国科学院 重庆绿色智能技术研究院, 重庆 400714; 2. 重庆邮电大学 计算机科学与技术学院, 重庆 400065)

(\* 通信作者电子邮箱 chenchangbo@cigit.ac.cn)

**摘要:**针对3D打印中已有自适应分层算法不能有效保留模型特征的问题,提出了一种新的识别和保留模型特征的自适应分层算法。首先,扩展了模型特征的定义,引入了模型特征丢失和偏移的概念;然后,提出了一种特征识别的方法,其识别模型特征的关键在于利用了模型特征出现的地方必然伴随着模型表面复杂度或切片轮廓数的变化这一性质;最后,在已有自适应分层算法的基础上,通过在特征附近用最小的分层厚度处理模型来保留模型的特征。在自主开发的软件Slicer3DP上实现了均匀分层、自适应分层和所提的分层算法,对比发现所提算法能有效解决模型特征的丢失和偏移,兼顾了分层精度和效率。仿真表明该算法可以用于对模型精度要求较高的3D打印中。

**关键词:**3D打印;模型特征;特征识别;分层算法;自适应分层

**中图分类号:** TP391.7 **文献标志码:** A

## Adaptive slicing algorithm to retain model characteristics

LI Wenkang<sup>1,2</sup>, CHEN Changbo<sup>1\*</sup>, WU Wenyuan<sup>1</sup>

(1. Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing 400714, China;

2. College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

**Abstract:** To resolve the problem that the existing adaptive slicing algorithm in 3D printing cannot retain effectively model characteristics, a new adaptive slicing method for recognizing and retaining model characteristics was proposed. Firstly, the definition of model characteristic was extended, and the concept of loss and offset of model characteristic was introduced. Secondly, a characteristic recognition method was proposed, the key point of which is to make use of the fact that the surface complexity and number of contours must change around the model characteristics. Finally, based on existing adaptive slicing algorithms, this algorithm retained model characteristics by slicing the model with minimum layer thickness near the model characteristics. On the self-developed software Slicer3DP, the following algorithms were implemented: the uniform slicing algorithm, the adaptive slicing algorithm and the proposed slicing algorithm. By comparing these algorithms, it is found that the proposed slicing algorithm resolves effectively the loss and offset of model characteristics while maintaining both slicing precision and efficiency. The result shows that the proposed method can be used for 3D printing with high precision requirement.

**Key words:** 3D printing; model characteristic; characteristic recognition; slicing algorithm; adaptive slicing

## 0 引言

3D打印也称增材制造(Additive Manufacturing, AM)<sup>[1]</sup>,它的基本原理是:在某一方向上,用足够多的分层平面去切三维模型,那么在每一层上就得到分层平面与模型相交的所有轮廓信息,利用轮廓彼此的包含关系判断出要打印的部分,再通过对打印材料的逐层堆积粘结,最终打印出想要打印的三维实体(3D打印的方法不仅限于分层制造,也有基于体素的3D打印<sup>[2]</sup>等方法,但本文讨论的3D打印采用分层制造的方法)。3D打印中的三维模型可以是计算机辅助设计(Computer Aided Design, CAD)模型、通过求反工程得到的点云数据模型和由CAD模型转变而来的立体光刻(STereoLithography, STL)文件格式的模型。其中STL由于格式简单、通用性良好,目前已成为增材制造的工业标准<sup>[3]</sup>。STL文件是用许多空间小三角形来逼近CAD模型的表面,是

CAD模型经过三角网格化后得到的。从上面快速成型的原理可以看出,对三维模型的分层处理是快速成型中的重要步骤之一,因此,设计出良好的分层算法是非常有必要的。

目前基于STL文件的3D打印分层算法主要有均匀分层和自适应分层两种。均匀分层<sup>[4]</sup>即采用统一的分层厚度对模型进行分层处理,由于3D打印逐层累积打印的固有缺陷,分层算法会带来阶梯误差<sup>[5]</sup>,均匀分层会造成阶梯误差的不均匀,使得在打印所得实体的有些地方误差过大。自适应分层<sup>[5]</sup>主要是为了降低3D打印中的阶梯误差提出来的,它会根据模型表面的复杂情况,自动调节分层厚度。自适应分层的算法很多,如文献[6]采用了根据模型的表面复杂情况直接决定分层厚度的直接自适应分层算法;文献[7]采用了逐次应用均匀分层对模型进行切片的方法;文献[8]采用逐步细分分层厚度的方法来处理复杂的模型;文献[9]提出的自适应分层算法应用了一种新的度量表面复杂度的方法;文献

收稿日期:2015-03-25;修回日期:2015-05-05。

基金项目:国家自然科学基金青年项目(11301524);重庆市科技攻关重点项目(cstc2012ggB40004)。

作者简介:李文康(1990-),男,湖北石首人,硕士研究生,主要研究方向:3D打印分层算法、可视化软件; 陈长波(1981-),男,山东济宁人,副研究员,博士,主要研究方向:计算机代数; 吴文渊(1976-),男,四川成都人,副研究员,博士,主要研究方向:符号数值混合计算。

[10]通过建立 STL 文件中三角面片的拓扑结构来加快自适应切片的过程;文献[11]提出了一种具有高鲁棒性的自适应分层算法;文献[12]对由功能梯度材料组成的模型提出了一种自适应分层算法;文献[13]提出了一种适用于金属 3D 打印的自适应分层算法。

通过文献调研发现,针对 STL 文件切片算法的研究主要集中于如何在分层数少的情况下降低模型的阶梯误差,如何减少切片的时间,如何增加切片的可靠性以及切片算法在特定领域中的应用,还没有发现文献对模型特征的讨论,只有在文献[8]中提到了模型特征的概念。受此启发,本文提出 3D 打印中的误差除了阶梯误差外,还有模型特征的丢失和偏移。首先对文献[8]中给出的三维模型特征的定义进行扩展,然后引入模型特征丢失和偏移的概念,并为降低 3D 打印中的误差,提出一种新的识别和保留模型特征的自适应分层方法。

本文提出的自适应分层算法与文献[8]所提的分层算法在对模型特征处理上的不同之处在于:1)文献[8]在特征的定义中并未给出特征点的完整定义,只是给出了特征点可能存在的地方;而本文则给出了特征的完全定义方法。2)文献[8]对于特征的识别采用穷举法,此方法的代价非常高,虽然有提到模型特征出现的地方必然伴随着相邻分层平面上切片轮廓数的变化,但此说法是不完全的,因为相邻分层平面上和相邻分层平面间模型表面复杂度的变化也可能是模型特征出现的地方;本文则给出了特征识别的具体方法。3)文献[8]对特征的处理方法是在特征所在的地方插入一分层平面,此方法能解决模型特征的丢失,但是由于阶梯误差的存在,该方法不能有效地解决模型特征的偏移;本文则通过在有特征所在的相邻分层平面间用最小的分层厚度来处理模型,能有效地解决模型特征的丢失和偏移,从而更有效地保留模型的特征。

## 1 3D 打印误差解决方案

### 1.1 模型特征的定义

三维模型的特征是模型中的重要部分,针对 STL 文件的特点(用三角面片来表示模型),把模型的特征分为特征面、特征线和特征点,并作如下定义:

1) 特征面:是平行于分层平面的表面。在这里及下面的叙述中都假设分层轴为  $Z$  轴,分层平面平行于  $XY$  平面。分层方向的选取会影响阶梯误差和模型特征的数量,进而影响制造的复杂度。但新的分层方向可能导致新的特征的出现,且分层方向的优化是一个综合考虑模型表面精度、支撑体积、构建时间和打印总消耗的过程<sup>[14]</sup>,因而本文只考虑给定分层方向下的保留特征的分层问题。特征面由那些三顶点的  $Z$  坐标值都相等的三角面片组成,如图 1 中的 1、3、7 都为特征面。

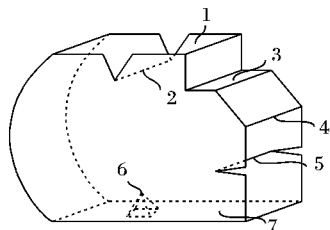


图1 STL 特征模型

2) 特征线:由两三角面片的交线组成,此交线平行于分层平面,两三角面片的夹角大于一阈值,且两三角面片都不是特征面,如图 1 中的 2、4、5 都是特征线。

3) 特征点:由多个三角面片的交点组成,且这些三角面

片的除交点外的其他点的  $Z$  坐标值全部大于或者全部小于交点的  $Z$  坐标值,如图 1 中的 6。

### 1.2 模型特征的丢失和偏移

三维模型分层后,每一层按打印过程重新累积后的模型我们称之为打印实体。模型特征的丢失(或偏移)是指三维模型分层后,原三维模型的特征(特征点、特征线、特征面)在打印实体中消失(或位移)。下面用例子来说明模型特征是如何丢失和偏移的。

图 2 显示了图 1 中的模型过特征点 6 的截面。对该模型进行均匀分层,如图 3(a)所示(虚线为分层平面所在的地方),得到打印实体,如图 3(b)所示。

由图 3(b)可得,打印实体是不光滑的,这是由于逐层累积造成的阶梯误差。除去阶梯误差带来的模型特征形变的影响,可以看到特征大多产生了偏移,特征 2、3、4、6 分别偏移到了图 3(a)的  $o$ 、 $p$ 、 $q$ 、 $r$  处,同时特征 5 由于没被分层处理到,丢失掉了。总的来说,3D 打印中的误差来源主要有阶梯误差及模型特征的丢失和偏移。对于图 3(b),模型的阶梯误差和形变在真实的 3D 打印中并没有这么大,这里主要为了演示的方便而加大了分层厚度。

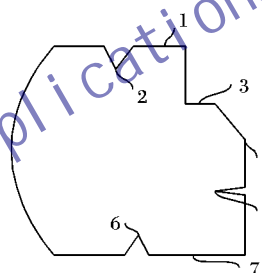


图2 过特征6的模型截面

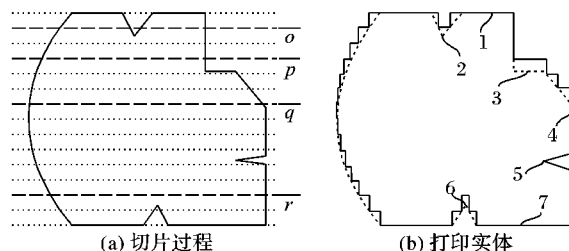


图3 均匀分层切片过程

### 1.3 误差解决方案

由 3D 打印的过程,可以很自然地看到减小均匀分层的分层厚度肯定可以减小阶梯误差,但是由于减小分层厚度、增加分层数必然会导致打印时间的增加,考虑到三维模型中阶梯误差的分布是不均匀的,于是提出了自适应分层<sup>[5]</sup>,它能根据模型表面的复杂程度自动调节分层厚度,从而使得阶梯误差均匀化,并能将阶梯误差控制在一定范围之内,下面提到的自适应分层算法均为直接自适应分层算法。

模型表面复杂度是根据 STL 模型表面的三角面片相对于分层轴的倾斜程度而定,三角面片倾斜度越大(即图 4 中的  $|\theta|$  越小,  $\theta$  为分层轴与三角面片法向量的夹角,三角面片的法向量与分层轴正向的夹角为正,负向的夹角为负),模型表面的复杂度越高,平行于分层轴的表面具有最小的复杂度,垂直于分层轴的表面具有最大的复杂度,对某个分层平面而言,它的模型表面复杂度用此分层平面上倾斜度最高的那个三角面片来表征,令此三角面片为表征三角面片。自适应分层中度量阶

梯误差的方法有很多,但目的都是让复杂的表面具有的分层厚度较小,简单的表面具有的分层厚度较大。本文选取尖端高度(即图4中的 $c$ 值)来度量阶梯误差,尖端高度的值取得越小,阶梯误差就越小。确定分层厚度的具体方法如图4所示: $Z$ 方向为分层方向, $h$ 为当前分层平面,在某一层上找到 $\theta$ 的绝对值最小的三角面片,即表征三角面片,它的法向量为 $N = (x_N, y_N, z_N)$ ,在STL文件中法向量的模长为1, $c$ 为尖端高度,那么分层厚度 $l = c / |\cos(\theta)|$ ,又因为 $\cos(\theta) = z_N$ ,所以 $l = c / |z_N|$  ( $l \in [l_{\min}, l_{\max}]$ ),那么下一分层平面的高度 $h' = h + l$ 。图5为自适应分层对模型的处理结果。

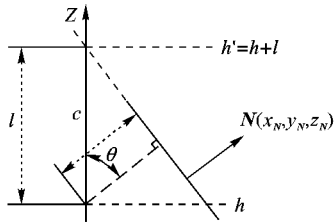


图4 STL模型分层厚度的确定方法

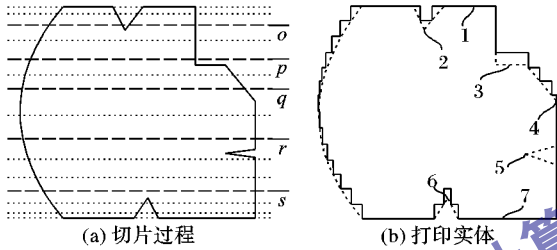


图5 自适应分层过程

由图5得:相比均匀分层,自适应分层在分层数一样的情况下能有效减小阶梯误差,但是仍不能处理模型特征的丢失和偏移。

文献[8]所提的分层算法用到了模型特征识别的思想,方法首先识别模型特征所在的地方,然后在特征所在处增加一分层平面,处理过程如图6,图中的虚线为识别特征后添加进的分层平面。

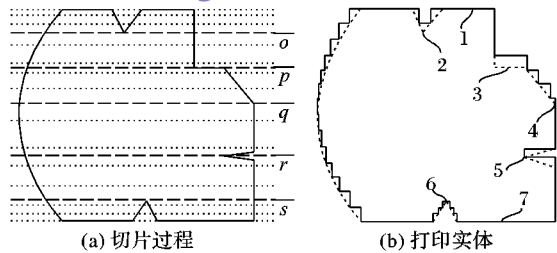


图6 文献[8]的自适应分层过程

由图6得:文献[8]所提的分层算法能保留所有的模型特征,使特征不丢失,但是特征还是发生了较大的偏移,如特征2、3、4、5分别偏移到了图6(a)的 $o$ 、 $p$ 、 $q$ 、 $r$ 处。这是因为在识别模型特征后(如特征4),在此处添加进一个分层平面,那么根据自适应分层,它的下一个分层平面是由特征所在处的分层平面确定,由于特征所在处的模型的表面复杂度不一定很大,因此此时的分层厚度可能会较大,使得下一个分层平面离特征所在处的分层平面较远,使特征具有较大的偏移,特征所在处也会由于模型的阶梯误差产生较大的形变。由此可知,此分层算法在对模型特征的处理上有两个缺陷:

1) 此方法模型特征的识别是困难的。精确地识别模型

特征所在处,需要对STL模型中的三角面片遍历很多次,算法复杂度很高。虽然文献[8]提出在模型特征出现的地方必然伴随着相邻分层平面上切片轮廓数的变化,但此说法是不完全的,具体的完全识别模型特征的方法见1.4节叙述。

2) 方法能处理模型特征的丢失,但是不能很好地处理特征的偏移。由于3D打印逐层累积打印的固有缺陷,使得模型特征或多或少会有一定的偏移和形变,解决方法是在特征附近用尽可能小的分层厚度来处理模型,这样就可以尽量小地减少模型特征的偏移和形变。

本文提出一种能有效保留模型特征的自适应分层算法:算法先用直接自适应方法对模型进行分层,得到切片信息;然后通过特征识别的方法来判断相邻分层平面间是否有特征存在,如果有特征在这两层平面间,则在这两层平面间再用最小的分层厚度对模型进行二次分层,具体的特征识别的方法见1.4节描述。图7为用本文算法对模型进行分层的结果,虚线为与图5的自适应分层方法相比,本文算法添加进的分层平面。可以看到用本文算法对模型进行分层处理后,模型特征也会偏移,如特征2、3、4、5、6分别偏移到了图7(a)的 $o$ 、 $p$ 、 $q$ 、 $r$ 、 $s$ 处,但此时的偏移非常小;同时本文算法也能保证模型特征不丢失。当然用分层厚度为最小值的均匀分层对模型进行分层也能保留模型的特征,但是此种方法所用的分层数将远大于本文提出的自适应分层算法,特别是对特征数较少且特征分布不集中的模型。

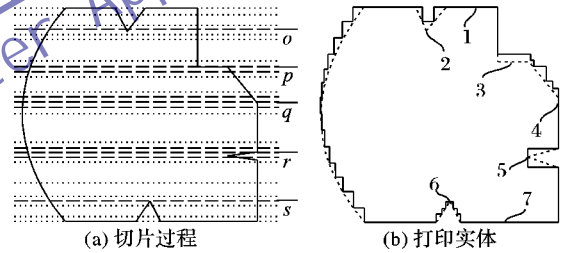


图7 保留特征的自适应分层过程

#### 1.4 模型特征识别的方法

模型特征识别的关键在于特征出现的地方必然伴随着模型表面复杂度的变化,或分层平面间切片轮廓数的变化。本文用下面三种方法来识别模型的特征,假设已经先用自适应分层对模型进行分层处理,并得到切片信息。

1) 如果相邻分层平面上切片的轮廓数发生变化,那么这两分层平面间必然存在着模型特征。如图2中模型的特征2和6:对于特征6,它的出现伴随着包含此特征的相邻分层平面上的切片轮廓个数由两个变为一个,如图8(a)所示;对于特征2,它的出现伴随着包含此特征的相邻分层平面上的切片轮廓个数由一个变为两个,如图8(b)所示。

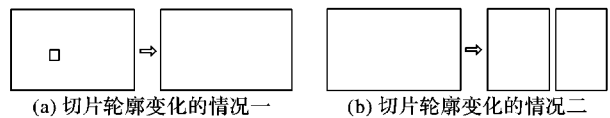


图8 特征出现的情况一

2) 如果在相邻分层平面上,模型的表面复杂度发生变化,且变化到一定程度,即这两分层平面上的表征三角面片的法向量与分层轴夹角的差值的绝对值大于一阈值(这个阈值及下面所提的阈值都和定义特征线时所使用的阈值一致),那么在这两层平面间可能存在着模型的特征。如图2



中特征3和4,与图5(a)进行对比可以很好地发现此特性。

3)如果在相邻分层平面间,模型的表面复杂度发生变化,且变化到一定程度,即在这两分层平面间的三角面片中,找到法向量与分层轴夹角最大和最小的那两个三角面片,如果这两个夹角的差值大于一阈值,那么在这两分层平面间可能存在着模型的特征。如图2中的特征5。

根据上面的三种方法能够识别模型所有特征所在的地方。对给定的两相邻的分层平面,按照上面所提方法的顺序,先判断此两分层平面上切片的轮廓数是否发生变化,如果不发生变化,再比较此两分层平面上的模型表面复杂度,如果不满足特征存在的条件,再比较此两分层平面间的模型表面复杂度,如还不满足条件,则判断这两分层平面间没特征存在,否则可能有特征在此两分层平面间。对于特征在分层平面上的情况,处理方法与上面的一样。

## 2 算法描述

图9为本文的自适应分层算法处理流程,主要包括:读取STL文件、对STL文件进行预处理、切片。

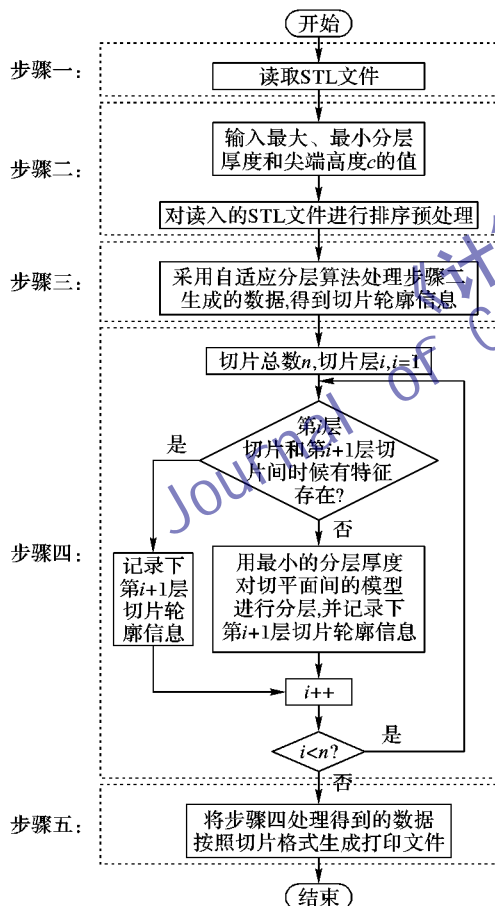


图9 本文分层算法的流程

### 2.1 STL文件的读取

STL文件的读取是将STL文件中三角面片的信息读入到计算机,并用一定的数据结构来存储读进来的信息。本文针对STL的二进制和ASCII码两种不同的格式,采用不同的方法读取<sup>[15]</sup>,并用如下的数据结构来存储和处理STL文件:

1)顶点、法向量及三角面片的边与分层平面的交点都用Point类型来表示;

2)三角面片的信息包含3个顶点和一法向量,用Facet类型来表示;

3)三角面片与分层平面交相的交线包含两顶点信息,用Line类型来表示;

4)STL文件中所有三角面片的信息由Facet类型的容器来表示(vector<Facet>model);

5)每层上的交线信息由Line类型的容器来表示(vector<Line>lines);

6)一个轮廓的信息由Point类型的容器来表示(vector<Point>contour);一层上的轮廓信息由contour类型的容器来表示,即顶点的容器的容器(vector<vector<Point>>contours);所有的轮廓信息由contours类型的容器来表示,即顶点的容器的容器的容器(vector<vector<vector<Point>>>allcontours)。

### 2.2 对STL文件的预处理

对读取的三角面片进行预处理的目的是为了便于切片时查找与分层平面相交的三角面片<sup>[16]</sup>。本文采用对三角面片进行排序处理的方法。分两次对读入的三角面片进行排序,首先遍历所有的三角面片,让每个面片中的三个顶点按Z坐标值的大小进行升序排序;第二次,再次遍历所有的三角面片,让所有的三角面片按面片中顶点Z坐标的最小值升序排序。这样做的好处是方便我们通过比较来判断某个三角面片是否与分层平面相交,且在依次读取三角面片判断它是否与分层平面相交时,当读到一个三角面片,如果它的Z坐标的最小值都在分层平面之上,那么就可以断定这个三角面片之后的所有三角面片都不与分层平面相交。

### 2.3 对STL文件的切片

切片处理是分层算法的核心,本文先用自适应分层算法处理STL模型,得到切片信息,然后在相邻分层平面间用特征识别的方法判断是否有特征存在于这两分层平面间:如果有,则用最小的分层厚度来处理分层平面间的模型数据;如果没有,则不作处理,并记录切片轮廓信息。图9中步骤二和步骤三为对STL文件切片的过程。图9步骤三中自适应分层算法的具体描述见图10。图10中对STL文件自适应分层的详细处理,包括如何确定三角面片是否和分层平面相交,如何求三角面片与分层平面的交点、交线信息,以及首尾相连的方法等,可以参考文献[17]。

假设第*i*层和第*i+1*层切片的分层高度分别为*h*和*h'*,它们上的切片轮廓信息保存在contours和contours'中,那么图9步骤四中判断这两切片间是否存在特征的实现方法为:

1)如果contours'.size()!=contours.size()(特征存在的情况1);

2)或分层高度*h*和*h'*所在分层平面上的表征三角面片的法向量与Z轴的夹角(即表征三角面片法向量Z值的反余弦函数值)之差的绝对值大于阈值(特征存在的情况2);

3)或在这两分层平面间找到法向量与Z轴的夹角最大和最小(即法向量Z值最小和最大)的那两个三角面片,这两个夹角之差的绝对值大于阈值(特征存在的情况3)。方法中,相邻分层平面间的三角面片,即三角面片的顶点的Z坐标值都在*h*和*h'*内的那些三角面片,由于之前已经对三角面片经过排序处理,所以如果假设在*h*高度上与分层平面相交的

第一个三角面片的位置为 $f$ ,在 $h'$ 高度上与分层平面相交的最后一个三角面片的位置为 $f'$ ,那么遍历三角面片列表 $model$ 中从 $f$ 到 $f'$ 间的三角面片即可找到这两分层平面间的所有三角面片。

如果上面的三种情况有一个满足,则这两分层平面间存在特征;否则,这两分层平面中不存在特征。

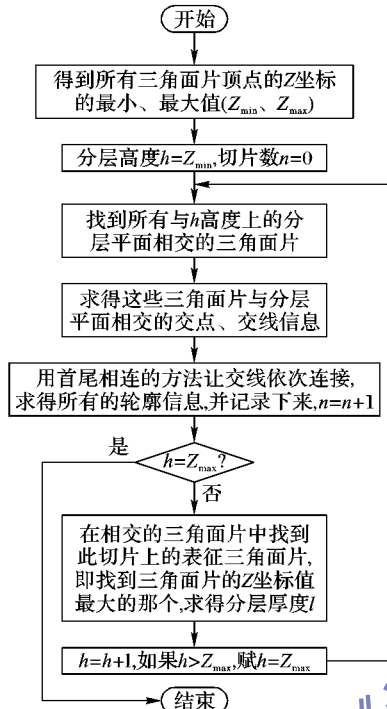


图10 自适应分层算法的流程

### 3 应用实现

基于Qt和OpenGL开发了跨Windows和UNIX平台的3D打印分层软件Slicer3DP。Slicer3DP实现了三种分层算法:均匀分层、自适应分层和本文所提的有效保留模型特征的自适应分层算法,并利用一些模型对比这三种方法的效果。

实现中,确定模型特征线的阈值由用户自己输入,这里选取为 $30^\circ$ ;为了使均匀分层能和本文所提的分层算法具有同样的保留特征的性质,取均匀分层的分层厚度为最小分层厚度,自适应分层和本文所提的分层算法具有同样的最小、最大分层厚度及相同的尖端高度,分层结果如表1所示。

表1 3种分层算法的分层数比较

例子	三角面片数	特征数	均匀分层算法	自适应分层算法	改进的自适应分层算法
1	3 246	0	101	58	58
2	504	1	61	46	49
3	10 714	3	173	63	71
4	35 342	11	248	141	174
5	34 860	33	252	134	197

图11为对表1中的第4个例子的展示,其中:图(a)为一个具有11个特征的奖杯 STL 模型,它的Z坐标的最大、最小值分别为247 mm和0 mm,给定的最大、最小分层厚度和尖端高度的值分别为10 mm、1 mm、0.8 mm;图(b)~(d)分别为用均匀分层、自适应分层和本文所提的有效保留模型特征的自适应分层方法的切片结果,这三种切片算法的分层数分别为248、141、174;图(e)是用分层厚度为1.42 mm的均匀分层算法对图(a)所示模型切片的结果,它的分层数为174。

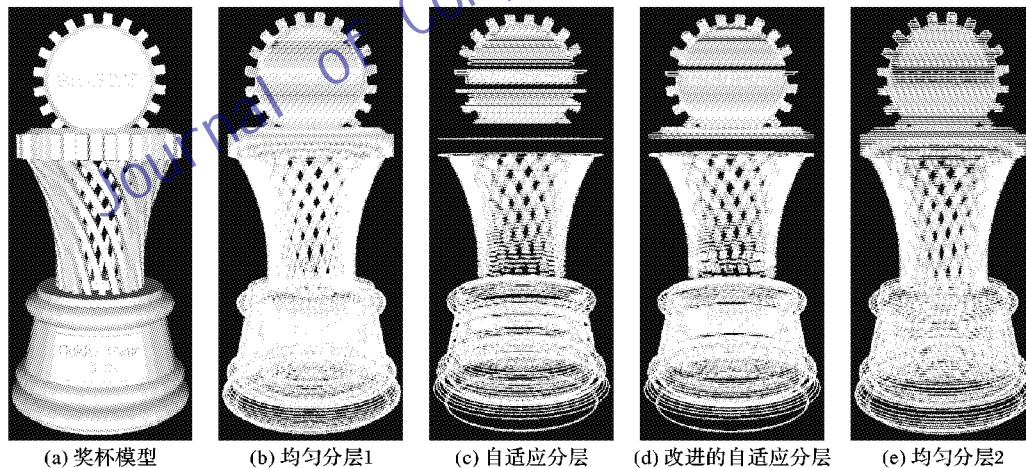


图11 三种分层算法的比较实例

对比表1数据和图11中的效果可以看出,当STL模型中不含特征时,本文所提的自适应分层算法和直接自适应分层算法切片的结果相同,且随着特征的出现和增多,本文所提的自适应分层算法所用的切片数比直接自适应分层算法要有所增加,但还是远少于具有同样特征保留能力的均匀分层算法;在均匀分层算法与所提的分层算法具有相同的分层数时,所提的分层算法显然具有更小的阶梯误差,且能保留模型特征,而均匀分层算法对特征的处理是随机的。可得出结论:本文所提的自适应分层算法与保留同样特征精度的均匀分层算法相比,切片数要少很多;而与直接自适应分层算法相比,虽然多用了一些切片数,但却保留了模型的特征,提高了3D打印

的精度。

### 4 结语

基于3D打印中的误差来源除了阶梯误差外,还有模型特征的丢失和偏移,本文提出了一种能有效保留模型特征的自适应分层算法。文中给出了STL文件模型特征的具体定义,此定义也可以扩展到其他格式的三维模型中;给出了模型特征识别的方法,该方法基于模型特征出现的地方必然伴随着模型表面复杂度或分层平面间切片轮廓数的变化,在需要详细确定特征所在的地方时,也可用此方法来缩小特征查找的范围;本文所提的分层算法在自适应分层算法的基础上,通

过在特征附近用最小的分层厚度来处理模型,能够有效地解决模型特征的丢失和偏移,从而更有效地保留模型特征。在Linux环境下,基于Qt和OpenGL开发了一款分层软件Slicer3DP, Slicer3DP实现了均匀分层、自适应分层和本文所提的能有效保留模型特征的自适应分层算法,对比结果验证了本文算法的有效性。

#### 参考文献:

- [1] WANG X. Development and prospects analysis of 3D printing technology and industrialization [J]. Chinese High-tech Enterprises, 2012(26): 3-5. (王雪莹. 3D打印技术与产业的发展及前景分析[J]. 中国高新技术企业, 2012(26): 3-5.)
- [2] HILLER J, LIPSON H. Design and analysis of digital materials for physical 3D voxel printing [J]. Rapid Prototyping Journal, 2009, 15(2): 137-149.
- [3] WEN P, HUANG W, WU C. Modified fast algorithm for STL file slicing [J]. Journal of Computer Applications, 2008, 28(7): 1766-1768. (温佩芝, 黄文明, 吴成柯. 一种改进的STL文件快速分层算法[J]. 计算机应用, 2008, 28(7): 1766-1768.)
- [4] ZHAO J, LIU W. Recent progress in slicing algorithm of rapid prototyping technology [J]. Computer Integrated Manufacturing Systems, 2009, 15(2): 209-221. (赵吉宾, 刘伟军. 快速成型技术中分层算法的研究与进展[J]. 计算机集成制造系统, 2009, 15(2): 209-221.)
- [5] PANDEY P M, REDDY N V, DHANDE S G. Slicing procedures in layered manufacturing: a review [J]. Rapid Prototyping Journal, 2003, 9(5): 274-288.
- [6] DOLENC A, MÄKELÄ I. Slicing procedures for layered manufacturing techniques [J]. Computer-Aided Design, 1994, 26(2): 119-126.
- [7] SABOURIN E, HOUSER S A, BØHN J H. Adaptive slicing using stepwise uniform refinement [J]. Rapid Prototyping Journal, 1996, 2(4): 20-26.
- [8] TATA K, FADEL G, BAGCHI A. Efficient slicing for layered manufacturing [J]. Rapid Prototyping Journal, 1998, 4(4): 151-167.
- [9] ZHONG S, YANG Y. Precise slicing method based on STL in RE/RP integrated system [J]. Computer Integrated Manufacturing Systems, 2012, 18(6): 1145-1150. (钟山, 杨永强. RE/RP集成系统中基于STL的精确分层方法[J]. 计算机集成制造系统, 2012, 18(6): 1145-1150.)
- [10] PAN X, CHEN K, CHEN D. Development of rapid prototyping slicing software based on STL model [C]// Proceedings of the 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design. Piscataway: IEEE, 2014: 191-195.
- [11] ZHANG Z, JOSHI S. An improved slicing algorithm with efficient contour construction using STL files [J/OL]. The International Journal of Advanced Manufacturing Technology, 2015 [2015-04-19]. <http://link.springer.com/article/10.1007%2Fs00170-015-7071-9>.
- [12] WANG S, WANG Y, CHEN C-S, et al. An adaptive slicing algorithm and data format for functionally graded material objects [J]. The International Journal of Advanced Manufacturing Technology, 2013, 65(1/2/3/4): 251-258.
- [13] VERMA A, TYAGI S, YANG K. Modeling and optimization of direct metal laser sintering process [J]. The International Journal of Advanced Manufacturing Technology, 2015, 77(5/6/7/8): 847-860.
- [14] LUO N, WANG Q, LIU H. A fast determination algorithm for slicing direction of 3D printing [J]. Journal of Xi'an Jiaotong University, 2015, 49(5): 1-7. (罗楠, 王泉, 刘红霞. 一种快速3D打印分层方向确定算法[J]. 西安交通大学学报, 2015, 49(5): 1-7.)
- [15] YAN Z, ZHONG Y. STL files reading and visualization based on VC++ and OpenGL [J]. Computer Systems & Applications, 2009, 18(3): 172-175. (严桂铭, 钟艳如. 基于VC++和OpenGL的STL文件读取显示[J]. 计算机系统应用, 2009, 18(3): 172-175.)
- [16] ZHAO C. Design and implementation of 3D printers layered software [D]. Nanjing: Southeast University, 2009: 12-13. (赵晨. 3D打印分层软件的设计与实现[D]. 南京: 东南大学, 2009: 12-13.)
- [17] CHOI S H, KWOK K T. A tolerant slicing algorithm for layered manufacturing [J]. Rapid Prototyping Journal, 2002, 8(3): 161-179.

(上接第2294页)

- [4] NARASIMHAN S G, NAYER S K. Contrast restoration of weather degraded images [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(6): 713-724.
- [5] FATTAL R. Single image dehazing [J]. ACM Transactions on Graphics, 2008, 27(3): 721-729.
- [6] TAREL J P, HAUTIERE N. Fast visibility restoration from a single color or gray level image [C]// Proceedings of the 12th International Conference on Computer Vision. Washington, DC: IEEE Computer Society, 2009: 2201-2208.
- [7] HE K, SUN J, TANG X. Single image haze removal using dark channel prior [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(12): 2341-2353.
- [8] HE K, SUN J, TANG X. Guided image filtering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(6): 1397-1409.
- [9] WANG Y, YIN C, HUANG Y, et al. Image haze removal using a bilateral filter [J]. Journal of Image and Graphics, 2014, 19(3): 386-392. (王一帆, 尹传历, 黄义明, 等. 基于双边滤波的图像去雾[J]. 中国图象图形学报, 2014, 19(3): 386-392.)
- [10] WANG W, DAI S. Fast haze removal method based on image fusion and segmentation [J]. Journal of Image and Graphics, 2014, 19(8): 1155-1161. (王伟鹏, 戴声奎. 结合图像融合与分割的快速去雾[J]. 中国图象图形学报, 2014, 19(8): 1155-1161.)
- [11] NAYAR S K, NARASIMHAN S G. Vision in bad weather [C]// Proceedings of the 7th IEEE International Conference on Computer Vision. Piscataway: IEEE, 1999: 820-827.
- [12] TAN R T. Visibility in bad weather from a single image [C]// Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2008: 2347-2354.
- [13] YU M, ZHANG H. Single-image dehazing based on dark channel and incident light assumption [J]. Journal of Image and Graphics, 2014, 19(12): 1812-1819. (於敏杰, 张浩峰. 基于暗原色及入射光假设的单幅图像去雾[J]. 中国图象图形学报, 2014, 19(12): 1812-1819.)
- [14] LI D, YU J, XIAO C. No-reference quality assessment method for defogged images [J]. Journal of Image and Graphics, 2011, 16(9): 1753-1757. (李大鹏, 禹晶, 肖创柏. 图像去雾的无参考客观质量评测方法[J]. 中国图象图形学报, 2011, 16(9): 1753-1757.)