

# 航空叶片打磨余量分析及轨迹生成

兰旭东<sup>1</sup>, 谢望<sup>1</sup>, 赵新航<sup>1</sup>, 韩飞<sup>1</sup>, 朱明一<sup>1</sup>, 孙天亮<sup>2</sup>, 徐昕<sup>1</sup>

(1. 上海航天电子技术研究所, 上海 201109; 2. 上海联影医疗科技股份有限公司, 上海 201109)

**摘要:** 针对航空发动机叶片打磨加工前, 叶身余量分布不均且较小的问题, 提出一种基于毛坯点云配准的加工余量分析和自适应打磨轨迹生成方法; 利用精确扫描测绘技术获取毛坯三维点云, 形成了毛坯/零件数模二者点云配准方法; 通过基准对齐、点云轮廓包含等条件约束, 实现了毛坯三维加工余量分析; 在余量分布点云基础上, 通过截面获取加工点云轨迹, 对轨迹点云进行珠链排序, 将轨迹排序点有效化, 计算出连续合适的打磨路径, 实现自适应余量打磨; 最后在 Vericut 软件中进行了打磨仿真, 验证了该方法的有效性。

**关键词:** 航发叶片; 点云配准; 余量分析; 轨迹规划; 自适应打磨

## Grinding Allowance Analysis and Path Generation of Aeroengine Blades

LAN Xudong<sup>1</sup>, XIE Wang<sup>1</sup>, ZHAO Xinhang<sup>1</sup>, HAN Fei<sup>1</sup>, ZHU Mingyi<sup>1</sup>, SUN Tianliang<sup>2</sup>, XU xin<sup>1</sup>

(1. Shanghai Aerospace Electronic Technology Institute, Shanghai 201109, China;

2. Shanghai United Imaging Healthcare Co., Ltd., Shanghai 201109, China)

**Abstract:** In order to solve the problems of uneven distribution and small allowance before aeroengine blade grinding and machining, a method of machining allowance analysis and adaptive grinding path generation based on blank point cloud registration is proposed. The accurate scanning and mapping technology is used to obtain the 3D point cloud of the blank, and form the point cloud registration method of the blank or part mathematical model; Through the constraints of datum alignment and point cloud contour inclusion, the 3D machining allowance analysis of the blank is realized; On the basis of the allowance distribution, the machining point cloud track is obtained through the section of the blank, the bead chain is sequenced for the track point cloud, and the continuous and appropriate grinding path is calculated to achieve the adaptive allowance grinding. Finally, the Vericut software is used to simulate the grinding of aeroengine blades, the results verify the effectiveness of the proposed method.

**Keywords:** aeroengine blade; point cloud registration; margin analysis; trajectory planning; adaptive grinding

## 0 引言

为提高航空发动机的推进性能, 叶片设计和制造需满足流体力学要求, 叶片的表面形状呈现为空间任意积叠的自由曲面<sup>[1-3]</sup>, 叶片的型面形状和加工质量会直接影响到发动机的燃油效率<sup>[4]</sup>。三维点云采集设备的高速发展, 零件的表面轮廓采集精度迅速提高, 可以准确采集叶片坯料数字模型。发展方向为“毛坯轮廓精密测量—毛坯加工余量数字化分析和自动路径优化—精密数控加工”为特色的数字化、自动化、集成化精密加工技术。

三维激光扫描技术, 指的是使用线结构光或其它的激光扫描仪等设备, 在中小距离的情况下对目标物体进行扫描, 自动采集并计算出被测量物体表面的点云数据, 凭借编程方式或者相关软件, 生成并可视化被测物体三维模型的技术<sup>[5]</sup>。测量数据中包含被测物体空间坐标、纹理等多种关键信息。三维激光扫描技术与传统测量方法相比, 优点在于效率高、无接触、自动化程度高, 可连续重复采集<sup>[6]</sup>。点云到三维曲面重建的主要流程为滤波去噪、拼接配准、重复点去除、压缩简化、曲面重建、孔洞修复等

步骤<sup>[7]</sup>。

目前, 基于数字化的模型的匹配对准多以迭代最近点算法(ICP, interactive closest point)及其改进算法为主<sup>[8-12]</sup>。韩杰龙等<sup>[13]</sup>针对薄壁舱体先用PCA法进行粗配准, 后用ICP进行精配准, 提高了配准精度和效率。王等<sup>[14]</sup>针对非圆、薄壁等结构件的余量提取, 提出立体模型匹配对准为核心的余量分析法。因此, ICP算法在航空航天中的关键零部件的机械加工中, 有着很好的适用性。在轨迹规划研究方面, 国外学者 Erkorkmaz<sup>[15]</sup>根据加工精度的要求, 刀具路径必须要准确地描述而且还必须平滑, 以保持较高的精度, 通过预设轨迹行距改进并简化了等参数截面算法, 获得平滑的加速度和减速, 缩短了加工时间。Tournier<sup>[16]</sup>提出复杂曲面等间距法, 计算得到曲面的残高面和刀位面, 得到它连续计算刀具路径, 从而提高了打磨轨迹的精度。但是这些都没有考虑打磨件的打磨余量信息对加工的影响, 导致加工质量和效率不是很理想。

粗配准主要是为精确配准提供较好的数据集。Jiang等<sup>[17]</sup>以该点与相邻点的法线方向夹角为当前点的K维特征

收稿日期: 2023-03-03; 修回日期: 2023-03-09。

作者简介: 兰旭东(1997-), 男, 河北张家口人, 硕士, 主要从事三维测量、数据分析和测控通信方向的研究。

引用格式: 兰旭东, 谢望, 赵新航, 等. 航空叶片打磨余量分析及轨迹生成[J]. 计算机测量与控制, 2023, 31(6): 238-245, 251.

的粗配准方法, 其配准效果由法线计算的精度决定。Lei 等<sup>[18]</sup>采用多尺度特征值的描述符算法, 可以提高配准效率和优化计算结果。张顺利等<sup>[19-20]</sup>提出自适应邻域配准法, 对点进行 GAN 定义后 Delaunay 三角剖分, 通过几何信息匹配快速得到配准结果。

精配准主要是通过迭代算法求解出准确的变换矩阵, 精配准的核心是点与点的计算, 即尽可能地将多次测量的重叠部分点重叠到一起, 即迭代最近点 (ICP, iterative closest point)<sup>[21]</sup>; Yang 等<sup>[22]</sup>提出一种 ICP 和全局优化相结合直接进行精配准的方法, 提高精度和效率; Servos 等<sup>[23]</sup>认为当下拼接配准利用的信息局限于坐标值, 于是自行编写算法将局部特征如颜色、强度等信息引入计算框架中, 提高了拼接精度; Xiong 等<sup>[24]</sup>在 ICP 迭代过程中, 对子数据集进行细致划分, 并更新控制状态, 提高了 ICP 算法的鲁棒性和收敛性; Bac 等<sup>[25]</sup>通过均值曲率限制, 可以较好地去除噪声, 提高了鲁棒性; 王育坚等<sup>[26]</sup>建立 KD-tree, 通过点的索引快速计算各个点对距离差值, 提高精配准的收敛效率; Chen<sup>[27]</sup>面对矢量双边滤波的大计算量问题, 提出基于 GPU 的双边滤波算法, 来提高滤波效果。

针对传统打磨路径生成没有考虑实际模型余量分布情况, 难以获得理想的加工效果, 所以本文采用三维测量技术, 实测模型与理论模型快速对比, 分析打磨余量分布情况, 实现自适应打磨。

## 1 基于点云配准的余量分析

### 1.1 点云的获取与配准

点云是物体三维信息的一种表达形式, 核心是空间坐标点的如何表示。基本的点云数据为 XYZ 三坐标存储, 扩展信息可包含当前点的曲率、RGB 值和法向量坐标等。点云数据的存储格式优劣会直接影响读取时间的长短。

流行的点云存储格式包含 PLY、PCD、LAS、TXT 等, 不同的领域常用的点云格式不同。PLY 格式是由斯坦福大学 Turk 等人在 The Digital Michelangelo Project 中存储扫描立体结果时研发的存储文件格式。每个 PLY 文档中都包含档头, 格式如表 1 所示, 其设定网格模型的“元素”和“属性”, 紧接着是文档的各种数据统计。“元素”是顶点、面、边、三角带等信息。其存储形式分为 ASCLL 与 binary 两种, 但是档头总是 ASCLL 编码, 便于使用者直接查看整体信息。

表 1 PLY 格式信息

信息	描述
FORMAT	PLY 版本信息
COMMENT	编写注解
ELEMENT	描述元素与属性
PROPERTY	依次描述元素数值字段意义
ELEMENT FACE	面元素
END_HEADER	头列表结束

通性, 形成一种开放格式标准。LAS 格式是美国摄影测量协会 (ASPRS) 发布的机载雷达的标准数据格式<sup>[28]</sup>。其包含头文件区、变长区和记录区三部分。头文件区主要包含公共声明、版本信息等内容。编程进行数据处理时, 主要是对点记录个数、坐标值缩放因子、坐标偏移量、点坐标范围等。变长区中包含的信息主要为, 用户 ID、记录 ID、拓展信息等内容。记录区中存储每个点的坐标信息, 版本 LAS1.0 中定义了具体的文件数据格式。后续更新的版本均在此格式上不断地添加改进。

随着三维扫描技术的不断发展, 单幅点云的数据量也在急剧增加。当下海量数据存储方式基本采用分而治之的方法。即将海量数据利用 Hash 映射到不同的地址空间中。通过散列算法, 将大量的数据压缩分割为同等大小的数据空间, 分别存储。需要使用时, 通过 Hash 表开展查找。

目前 PCD 格式的点云数据存储主要分为 ASCLL 码和 binary 两种, 当点云数据以 ASCLL 格式存储是可以直接通过 Word 等文本软件直接打开, 在点云数据传输时, 接收者可以直接打开查看内部数据, 缺点内存消耗大。二进制 (binary) 的编码方式可以大幅的节约存储空间。二进制的存储也可以通过预先读取, 减少读取时显示的时间。本文对不同格式的点云从所占空间和读取耗时两个方面来分析, 对比优劣。结果如表 2~3 所示。从表格中可以看出 binary 方式存储可以节省大量空间, 在显示过程中可以极大地缩短时间, 提高程序交互性。

表 2 不同方式存储优劣对比表

点云所含点数	2 千	5 千	1 万	10 万	100 万
ASCLL 存储所占空间/kB	78	156	326	3 290	31 731
Binary 存储所占空间/kB	29	58	121	1 209	11 855
ASCLL 存储显示时间/s	0.183	0.307	0.491S	4.484	49.716
Binary 存储显示时间/s	0.003	0.004	0.006	0.019	0.189

表 3 叶片实测模型部分余量表

点索引号	X	Y	Z	R(余量)
0	27.764	6.735	360	0.124
1	27.871	6.784	360	0.092
2	27.932	6.813	360	0.073
3	28.450	5.437	360	0.101
4	28.969	5.891	360	0.076
5	28.029	5.902	360	0.153
...	...	...	...	...

激光扫描获取毛坯初始点云数据, 经过点云滤波、去噪获取毛坯精确点云数据。

初始采集点云为可用点和噪点的合集, 噪点的存在将会导致后续曲面重建的精度不高、产生孔洞、生成杂边, 严重时会导致曲面重建算法失效。为此, 必须进行点云降噪。点云噪点的产生原因众多, 基本分为两类: 三维扫描

LAS 的文件均采用二进制存储, 主旨是提高文件的流

系统自身的系统误差,采集时环境因素如被测物体反光等。

噪点主要分为以下几类:散乱噪点、离群聚集噪点、内部重复点、冗余噪点。噪点的去除方法核心是根据噪点的分布特点,构造一定的数学函数,针对噪声特点,筛选并记录点后统一去除。噪点去除方法如下:

1) 选择滤波:点云在以 PCD 格式存储时会创建专有的结构体,结构体是一种数据类型,在此类型下有各个分支分别存储点云的三坐标、曲率等信息。选择滤波时会专门留有条件选择的接口,供给开发人员使用,可以依据的条件为 X、Y、Z 三坐标等,通过按照自己的选择,过滤原始数据。此方法通常需要将点云可视化,通过界面交互软件了解噪点位置,再通过此法过滤点云。

2) 统计滤波:此方法利用的是概率论中的  $3\sigma$  理论,计算点与其领域内的点之间距离的平均值,点的领域划分为就近原则。统计整个点云距离的计算值,统计结果可构成高斯分布。开放均值与方差两个函数接口,按照不同的输入值剔除相关的点云数据。此方法的去噪适合均匀采样的点云集,去除的噪点类型主要为散乱噪点。

3) 半径滤波:首先按照给定的数据作为半径,在每个点以此半径画球,当球内的数量满足要求时,该点就会保留,反之就会删除。此法滤波后会根据点云数据的疏密进行筛选,当某部分点云数据的密集程度没有达到要求时,便会被删除。在图 1 中所中,当 K 值设定为一时,橘黄色点就会被删除,其它两点则会保留。

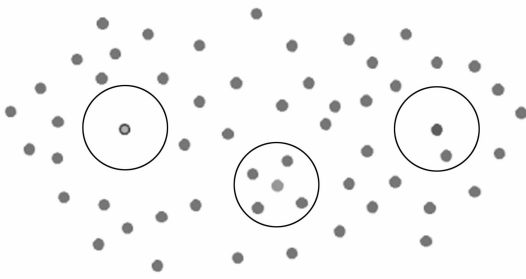


图 1 半径滤波原理示意图

4) 体素滤波:体素概念的提出是从像素中学习而来。使用空间立体网格划分点云数据,每一个网格可以理解为正方体,正方体中包含的点云数据会被其质心所代替。不同的体素网格划分会得到不同的处理结果。此方法处理会很大程度上保存原始点云的原貌,此方法的点云数据处理,可以成倍数的压缩点云数据,此方法也可以使原始点云数据整体密度均匀化。

逆向重建需要完整的点云数据。由于现有的三维扫描设备,只能得到物体的单面点云,想要得到完整物体的三维点云数据,必须通过不同的角度,采集物体数据点。得到的数据点一般都有重叠部分,用于后续的点云拼接和配准,有时为了计算方便,会单独放置特殊形状的物体,作为标记点,计算时专门计算标记点,得到点云数据的旋转平移矩阵,再将其余点云带入,通过旋转平移矩阵完成

拼接。

点云拼接的研究最早开始于 1983 年, Faugeras 首次通过四元数法计算拼接中的旋转矩阵。点云拼接算法从计算特性上大致可分为两个方向,通过提取两块点云中特征点群,比较计算两块特征点云中的法向量和坐标值,得出旋转平移矩阵。根据此矩阵实现点云拼接。另一方向为:通过对两块点云进行整体计算,统一比较分析两者重叠部分点云的特征值,可以得到当前两幅点云之间的旋转平移矩阵,实现点云拼接。两种方法在本质上均是坐标变换,即以某一个坐标系为基准将另一幅点云旋转平移到基准坐标系内。

点云拼接后重叠部分的点会发生重复,即会发生冗余。随着拼接的不断延续,冗余的点会越来越多。导致点云整体文件过大,大量的冗余点不仅会导致点云数据量的迅速增大,拼接的计算时间延长,还会导致曲面重建计算中微观邻域内曲面拟合时,拟合点的重复选择,导致曲面重建精度下降,甚至产生孔洞。拼接导致的冗余点数量多少和两幅点云重叠部分的大小有直接关系,部分学者为了提高点云拼接的精度,从而提高拼接点云的重叠部分,导致冗余点数量增多。此类冗余点,根据两点之间的距离划分,很容易理解为重复点,即区域特征信息,本来有一个点代表即可,这时反而多出了一个点,导致区域信息特征过表示。

针对此类冗余点,提出重复点去除算法,专门针对拼接或其他处理导致的同一位置,有两点或者多点产生的冗余问题。此类问题,有部分学者会称之为双层点云问题,即同一区域有不止一层点云表示,两者之间的细微差别在于,拼接导致的冗余点,几乎与原始点在同一平面,距离很小,而双层点云则是,距离过大,导致同一区域有两幅点云表示。

算法思想是首先通过 KD-tree 构建快速邻域查找的数据结构。设定查找半径,遍历点云数据中的点,对每个遍历点都开展近邻搜索。结果分为未发现近点和发现单点或不止一点两种情况,当近邻查找结构为未发现点时,则证明此点不是冗余点,将此点略过。当发现最近点时,这时实际情况可能不止存在一个最近点,为了提高程序的鲁棒性,只记录最近点的索引。被记录的每一个索引对应着原始点云中的每一个点。整个数据点云遍历完成以后,会将记录索引对应的点统一删除掉。

点的遍历过程中会发生同一个点多次重复记录现象。具体原因如图 2 所示,原始遍历搜索时,搜索半径设置不合理,超出了非重叠点之间的最小距离。当以搜索点 1 搜索时,重复记录点会被记录索引,当以搜索点 2 为球心开始搜索时,重复记录点依旧会被记录索引,如此会导致重复记录点被重复记录,造成索引点重复记录现象。处理方法是在完成整体点云检索遍历以后,将记录索引值按照从小到大排序,完成以后去除重复索引值,防止重复删除。

实际使用中,数据点全员遍历也会导致局部区域点云

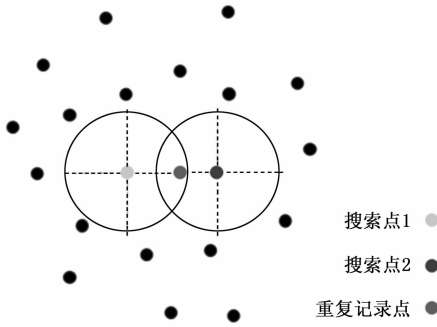


图 2 半径搜索中单点重复记录图

被连锁记录。如图 2 所示, 当搜索半径设置过大时, 以搜索点 1 搜索时, 会记录搜索点 2 的索引, 以搜索点 2 搜索时, 会记录重复记录点的索引, 则此微观区域点都被登记, 后续删除记录索引点。这样会删除那些应该保留的原始点, 导致区域无点存在, 成为孔洞。处理此问题的方法是, 在处理过程中登记需要删除的冗余点, 实时对比, 防止连锁记录。具体做法是, 在以重复记录点检索时, 记录搜索点 2, 会将记录的搜索点 2 进行登记, 等遍历到以搜索点 2 进行搜索前, 会与登记的点集进行对比, 看是否有与登记点相同, 如果相同则表明此点, 已经被记录索引, 需要删除, 便无需搜索最近点, 直接跳过。如此便可以防止连锁记录。如此, 去除重复点算法编写完成, 此方法的应用对象, 主要是针对十万级点云的重复点去除, 当数据量上升时, 计算时间会不断延长, 导致效率下降。

点云配准是将实测毛坯模型与理论模型统一同一个坐标系下。毛坯模型包络理论模型。ICP 迭代配准误差函数:

$$E(\mathbf{R}, \mathbf{t}) = \frac{1}{n} \sum_{i=1}^n \|x_i - (\mathbf{R}y_i + \mathbf{t})\|^2 \quad (1)$$

其中:  $x_i, y_i$  分别是坯料点云与标准点云中的对应点,  $\mathbf{R}$  为旋转矩阵,  $\mathbf{t}$  为平移矩阵。计算求出使误差函数最小的  $\mathbf{R}$  和  $\mathbf{t}$ 。坯料点云通过当前的  $\mathbf{R}$  和  $\mathbf{t}$  变换坐标信息, 第一次迭代结束。变换后的坯料点云再找出对应点对, 进行第二次迭代。迭代一定次数后 (根据原始的点云差别大小迭代次数浮动), 找出最优转换矩阵。

### 1.2 点云中加工余量分析

通过坐标对齐, 实测模型与设计模型点之间的偏差距离即为加工余量, 现有的 NC 刀路生成软件求解基础理论是理论设计模型, 没有实际余量信息。所以, 首先提取加工余量, 然后进行余量分析。本文采取的思路如下:

首先确定标准点云中点的法向量, 最小二乘法拟合平面得到法向量。然后根理论模型点和法向量寻找毛坯件上的最近点, 方法使用基于 KdTree 的近似查询的算法, 基于这一数据结构的查找算法可以快速而准确地找到给定点的最近点。计算求出不少于三个的最近点, 然后通过该点群拟合成一个平面, 记点的领域为:

$$N = \{(p_i, y_i, z_i) \mid i = 1, 2, \dots, n\} \quad (2)$$

拟合平面的一般方程:

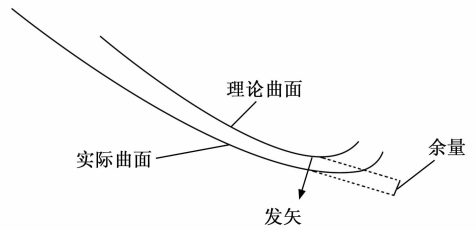


图 3 叶片余量示意图

$$Ax + By + Cz + D = 0 \quad (3)$$

$$A^2 + B^2 + C^2 = 1 \quad (4)$$

采用最小二乘法进行平面的拟合, 即求解如下的最小值问题:

$$\begin{cases} \min \sum_{i=1}^N (Ax_i + By_i + Cz_i + D)^2 \\ A^2 + B^2 + C^2 = 1 \end{cases} \quad (5)$$

计算完成后标准件上的点沿着法向量延长, 通过直线与坯料曲面交于一点, 得到交点的坐标。平面是以标准件上的点为基准邻近搜索坯料上的点拟合。

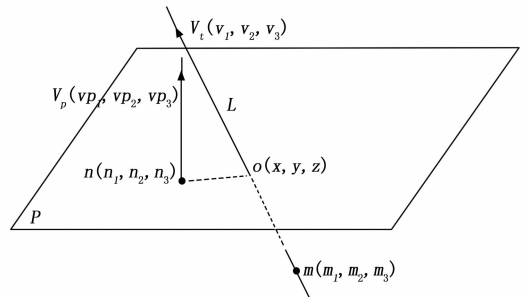


图 4 余量提取分析图

将直线方程改写成参数方程形式, 即为:

$$x = m_1 + v_1 \times t \quad (6)$$

$$y = m_2 + v_2 \times t \quad (7)$$

$$z = m_3 + v_3 \times t \quad (8)$$

将平面写成点法式方程:

$$v_1 \times (x - n_1) + v_2 \times (y - n_2) + v_3 \times (z - n_3) = 0 \quad (9)$$

联立求解得:

$$t = \frac{(n_1 - m_1) \times v_1 + (n_2 - m_2) \times v_2 + (n_3 - m_3) \times v_3}{(v_1 \times v_1 + v_2 \times v_2 + v_3 \times v_3)} \quad (10)$$

求出坐标后再计算交点与标准件的欧式距离, 便为加工余量。

针对以上的分析, 余量提取具体步骤:

- 1) 建立 KdTree 数据结构 (多维空间数据结构), 把扫描处理之后的点云模型数据集合 Q 输入;
- 2) 计算标准模型点集合 P 中  $p_i$  的 K 个最近邻点集合;
- 3) 基于搜最近的 K 个最邻近点, 用最小二乘法拟合局部平面;
- 4) 根据  $p_i$  的法向量, 计算到拟合平面的距离;
- 5) 根据设定的误差, 确定最终的距离即打磨余量。

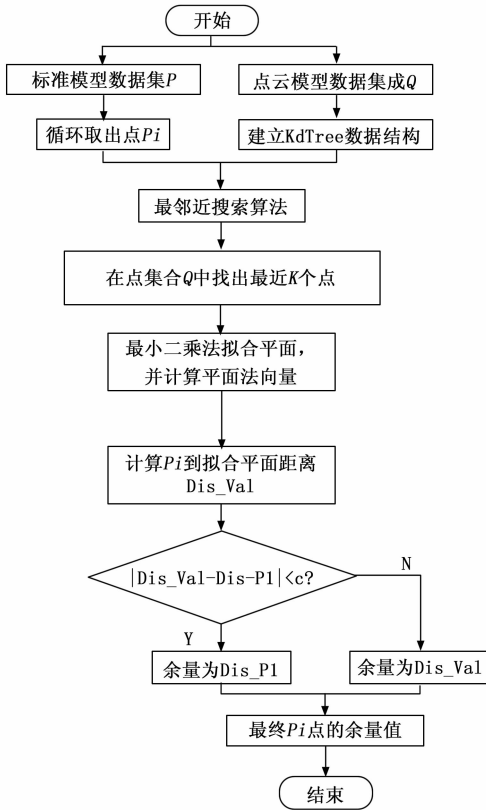


图 5 零件/毛坯的打磨余量提取基本流程图

## 2 线性轨迹点云数据有序化

### 2.1 排序算法原理

线性点云的有序化主要体现在对点的内部排序。此排序主要应用，在于轮廓提取、轨迹规划等线性点云处理中。线性点云有序化的实质是对杂乱分布点的排列整合。本文提出基于 KD-tree 数据结构的珠链排序算法的原理流程图 6 所示。

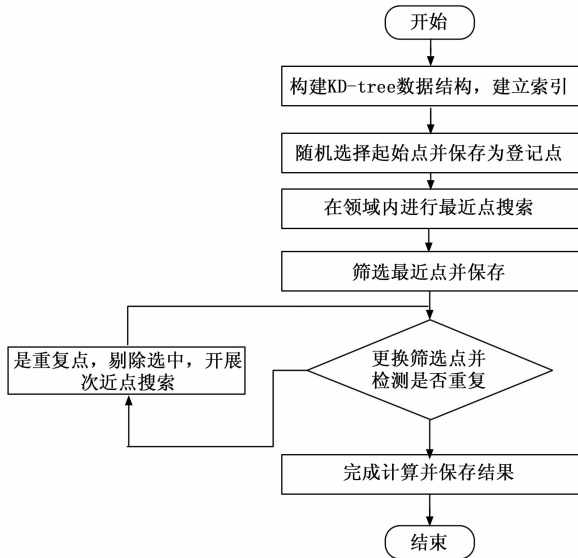


图 6 珠链排序算法流程图

假设，未排序的数据点云为点云 A，排好序后的存储点云为点云 B。首先，在线性点云 A 中随机选取一点，将此点定义为点云 B 的最初起点，并保存到新开辟内存空间点云 B 的结构体中。然后，以此点为球心，开始在点云 A 中开展近邻搜索，检索到距离此点最近一点后停止搜索。此点便为点云 B 中的第二个保存点，于此同时，也决定了线性点云的整体排序方向。此排序方向是随机的，即没有限定每一次点云排序的方向。如果需要限定排序方向，需要在第一次检索时，同时输出检索点的坐标。坐标输出以后，通过两点之间的坐标计算，可以得出当前点之间的坐标差值。通过判断 X、Y、Z 三坐标值的增减，即正负值，便可以限定增长方向。

将第一个搜索点保存后，便将搜索得到的第二个点转换为搜索点开始计算，第二个点保存到点云 B 后，将其替换原来的搜索点，作为新的搜索点，开始搜索最近点。得到最近点后，再将其保存到点云 B 中，更换搜索点。如此往复，直到点云 B 的点的个数等于点云 A 为止，此算法图解如图 7 所示。

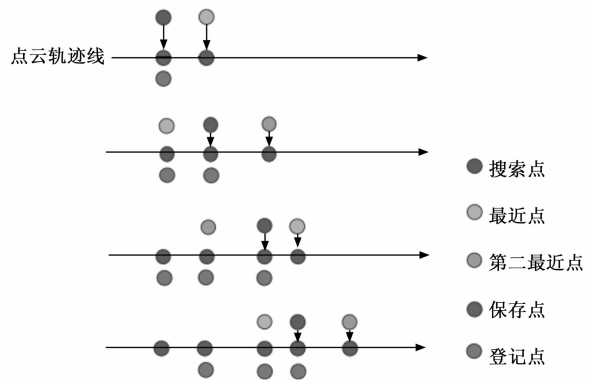


图 7 珠链排序算法原理解释图

如此计算，容易出现回返现象。杜绝回返，需要添加暂时登记点。用于对比剔除检索最近点时的回返点。在起点保存的同时，也开辟内存将其保存为登记点。第二个点保存的同时也添加为登记点。第三个点在保存的同时要与登记点进行对比，如果它是已经登记过的点，那么就证明出现了回返现象，需要提高搜索领域的半径寻找下一个点。再进行与登记点的对比，如果没有在登记点中找到相同点。那么这个就不是回返点。可以将其作为下一个搜索点，继续搜索，而原来的搜索点则保存到点云 B 中，并同时保存为登记点。

登记点的存在就是为了防止点云检索时发生回返，它的存在可以想象为一条不断向前移动的细小曲线，在沿着搜索点的不断更换而向前移动。正如算法抽象形如步进。每一次搜索点要更换为检索出的最近点，都是在细小曲线前进的方向中寻找到的近点，搜索点是绝对不可以更换为已有的登记点，也就是绝对不能落到细小曲线中去。

面对不同密度的线点云，需要调整登记点的个数，防

止在搜索最近点的时候, 搜索到的点超出了登记点的范围, 落回到保存点区域内。此情况如图 8 所示, 所以此算法面对疏密不均的点云时, 要增加登记点的个数, 从而延长登记保护线的长度, 防止回返现象发生。



图 8 珠链排序算法中回返现象发生示意图

### 2.2 排序算法实例化验证分析

本文以球体为例, 验证网格点云的排序结果。网格点云经过排序后, 会在某一坐标轴向呈现连续分布。本文对球形点云进行排序后, 会使点云按照层间进行连续排序。如图 9 所示, 排序后的点云会出现层间递增, 为了更加直观地体现排序后效果, 将球型点云按照排好序的 Z 值进行渲染, 即不同的 Z 值对应不同的颜色, 直观地显示球型点云排序后结果。

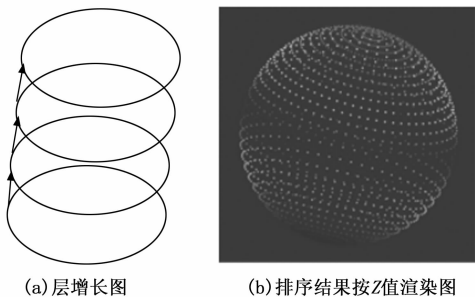


图 9 网格点云排序结果渲染

线性点云的排序结果, 本文将测试圆形点云, 检测排序结果是否为点间连续的有序点云。测试方法为将原始的轮廓点云输入到软件中进行逐个点遍历并依次连接为曲线。通过连接的曲线直观的显示线性点云排序结果。

图 10 (a) 中依次连接点, 出现点间跨越, 不能画成完整圆, 图 10 (b) 为排好序的点云依次连接图。点与点排列有序, 则依次连接结果为圆形。

### 3 自适应刀路规划

基于三维点云模型的刀具路径规划, 需先构造曲面或网格模型, 再基于点云三维曲面或网格曲面生成加工刀轨。由于航空叶片的属于复杂曲面, 这里采用旋转磨削的走刀方式, 这样能更好地贴合打磨表面。

本文采用砂带打磨, 砂带打磨是一种弹性磨削, 所以要建立砂带磨削材料去料模型, 某些学者<sup>[14-15]</sup>从宏观角度

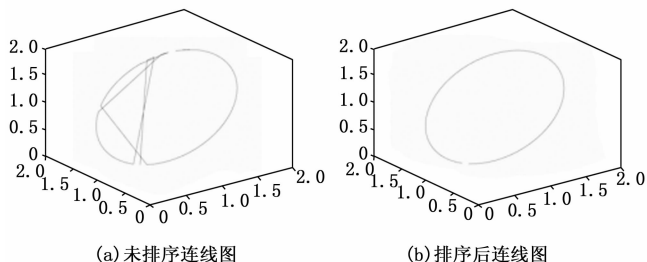


图 10 排序结果验证

探究, 材料去除量是含有多个参数的  $N$  元函数, 是多个参数共同影响的, 基于此提出一个线性的磨削模型公式:

$$r = C_A \cdot K_A \cdot K_t \cdot \frac{V_n}{V_w \cdot L_w} \cdot F_A \quad (11)$$

式中,  $r$  表示材料的瞬时材料去除量,  $C_A$  表示磨削过程的修正常数,  $K_A$  表示由打磨工具和被加工件材料阻力系数,  $K_t$  参数表示砂带耐用度,  $V_b$  表示砂带的线速度,  $V_w$ 、 $F_A$ 、 $L_w$  分别是打磨零件的进给速度、工件表面上的法向接触压力和打磨宽度。由于材料去除量与影响因素之间不是简单的线性关系, 应是呈指数关系。一些学者提出新的切削模型<sup>[29-30]</sup>:

$$r = K_A C_g \cdot V_b^{\epsilon_1} \cdot V_w^{\epsilon_2} \cdot F^{\epsilon_3} \quad (12)$$

其中:  $C_g = C_A \cdot K_A \cdot K_t$ ,  $F$  为磨削区域上各磨削点的法向接触正压力。当打磨沙袋型号确定时, 其中  $C = C_g \cdot K_A$  是个确定的值, 上式可以简化为:

$$r = C \cdot V_b^{\epsilon_1} \cdot V_w^{\epsilon_2} \cdot F^{\epsilon_3} \quad (13)$$

当参数  $F^{\epsilon_3}$  和  $C$  确定时, 下述反映进给速度和切削速度与加工余量之间的关系。

$$V_b = \sqrt{\frac{r}{C \cdot V_w^{\epsilon_2} \cdot F^{\epsilon_3}}} \quad (14)$$

$$V_w = \sqrt{\frac{r}{C \cdot V_b^{\epsilon_1} \cdot F^{\epsilon_3}}} \quad (15)$$

结合前面打磨余量提取和分析, 即使同一打磨区域的余量薄厚不一样, 切削量大的区域可以调整磨削速度与进给速度, 确保打磨区域的加工质量。

走刀行距即相邻轨迹的间距直接影响加工质量和效率, 合理规划加工行距很重要。行距主要与残留高度有关。设两加工路径之间的残留高度为  $h$ , 行距为  $L$ , 如图 11 所示的相邻轨迹行距与残留高度和曲率半径的关系:

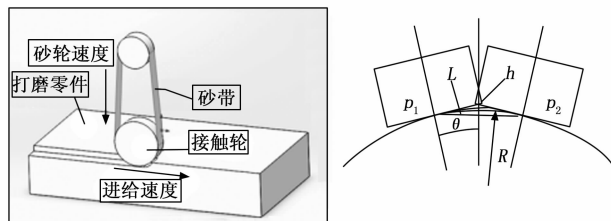


图 11 砂轮打磨残留高度与行距关系图

$$h = \frac{1}{2} \tan \theta - R + \sqrt{R^2 - \left(\frac{L}{2}\right)^2} \quad (16)$$

表 4 叶片部分磨削加工点信息表

点索引号	X	Y	Z	X_N	Y_N	Z_N	Curvature(曲率)	R(余量)
0	0.494 9	22.455 4	361	-0.984 1	0.176 8	-0.014 0	0.277 1	0.201 7
1	0.498 6	22.356 5	361	-0.980 7	0.194 8	-0.011 2	0.277 6	0.512 1
2	0.499 8	22.554 2	361	-0.980 8	0.194 2	-0.015 5	0.285 4	0.328 8
3	0.510 7	22.258 3	361	-0.977 3	0.208 9	-0.016 7	0.304 7	0.277 6
4	0.513 2	22.161 4	361	0.978 3	0.205 6	-0.021 1	0.357 1	0.198 5
5	0.530 9	22.066 5	361	-0.973 3	0.228 3	-0.013 3	0.390 1	0.263 2
...	...	...	...	...	...	...	...	...

$$\tan\theta = \frac{L}{\left(2\sqrt{R^2 - \left(\frac{L}{2}\right)^2}\right)} \quad (17)$$

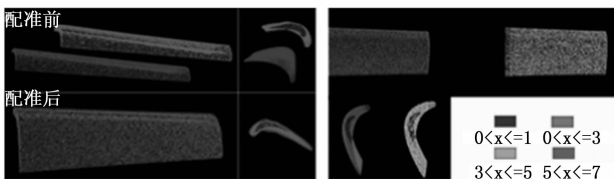
把  $\tan\theta$  带入化简得:

$$h = \frac{-L + 4R^2 + L^2}{4\sqrt{R^2 - \left(\frac{L}{2}\right)^2}} - R \quad (18)$$

式中,  $R$  是叶片两个接触点之间的平均曲率, 上式反映出行程与曲率半径  $R$  和残留高度  $h$  之间的关系, 由此可以求出加工路径行程, 当满足叶片表面加工精度情况下, 随着曲率半径变化, 可以自适应规划相邻轨迹行程, 这样可以提高整体加工效率和保证表面加工质量。

#### 4 基于 Vericut 的仿真

本文采用 C++ 语言, 集成开发环境 VS2013, window7 操作系统等完成软件开发。首先进行标准点云和坯料点云的迭代配准, 配准后的效果如图 12 (a) 所示。迭代配准后, 将结果带入余量计算程序中直观显示加工余量的分布, 不同颜色的彩点聚集代表着加工余量的变化, 整体分布效果如图 12 (b) 所示。



(a) 点云配准结果 (b) 余量提取色彩偏差图

图 12 坐标对齐及余量偏差图

表 3 是基于配准技术快速对齐实测模型与理论模型坐标系, 提取部分打磨余量信息表。表 4 是叶片模型通过软件生成的部分打磨点信息。

通过 vericut 仿真首先需要选择机床, 叶片打磨选用的是 vericut 中自带的机床系统。构建的 vericut 项目如图 14 所示。

由于航空工业的叶片加工要求的是叶片打磨, 在 vericut 的刀具库中需要自己制作一个专用的砂带打磨刀具。在 vericut 建立如图 15 所示的砂带打磨刀具。刀具按照软件生成的加工轨迹, 走刀平滑、切除余量均匀。

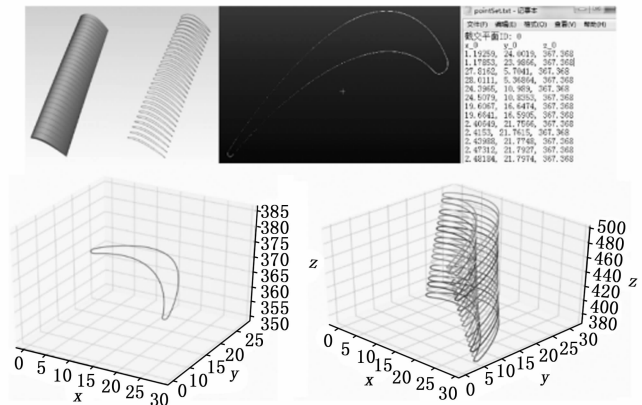


图 13 生成的打磨路径

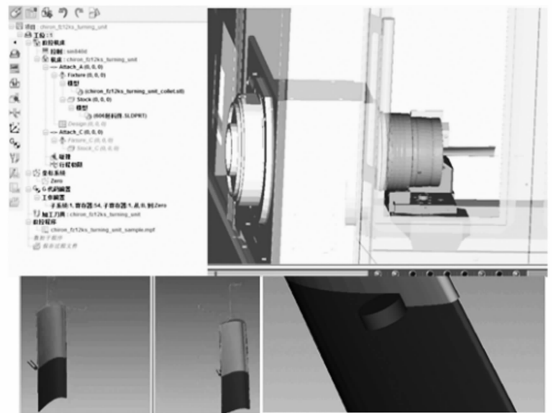


图 14 vericut 仿真项目界面图

仿真中从软件导出仿真加工数据, 绘制余量与进给速度关系表, 从图 15 中直观地看出软件生成的走刀速度与余量自适应, 呈现类似反比关系。对打磨后的叶片模型用截面平面分别在不同的位置截取得到打磨后的数据, 每个截面轮廓点数为 30 个, 和理论模型数据进行加工误差对比分析, 结果如图 11 (b), 有三个截面数据采集点的打磨误差在均  $\pm 0.08$  mm 之内, 达到自由曲面打磨技术要求。

#### 5 结束语

本文使用 C++ 语言通过 VS 编写软件, 实现实测坯料模型与标准点云之间余量精准提取, 通过 Vericut 软件仿真

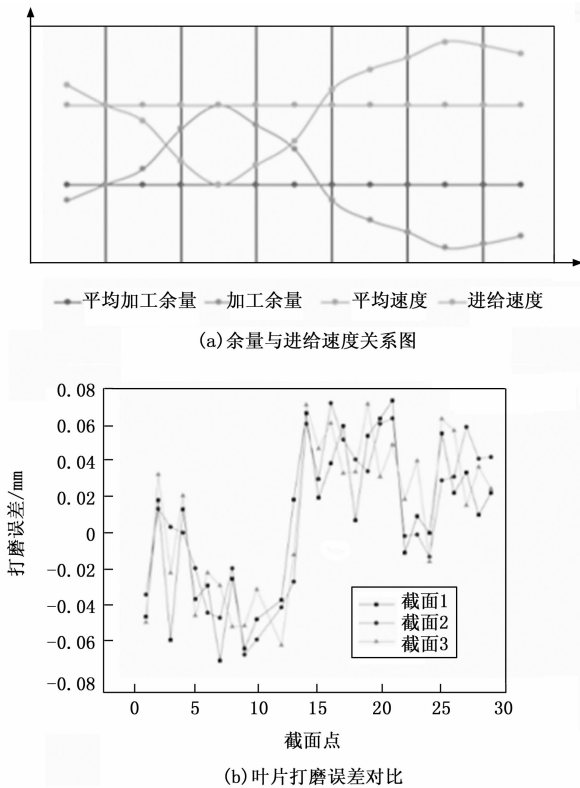


图15 叶片仿真结果

加工,加工后叶片模型打磨误差均在控制范围内,显著提高加工精度和加工质量一致性。

#### 参考文献:

- [1] 刘威,朱淑梅,周琛,等. 点云模型行距自适应数控加工刀轨生成方法研究[J]. 制造业自动化, 2017, 39(11): 105-108.
- [2] 张宝峰. 航空发动机叶片加工中二次造型的研究及应用[J]. 中国新技术新产品, 2016(20): 116.
- [3] 孙方成,于群,石宏. 航空发动机叶片及其发展趋势[J]. 山东工业技术, 2018(3): 51.
- [4] 李杰,巨亚斌. Leap-x发动机和tech-x发动机先进技术分析[J]. 航空制造技术, 2013, 19(10): 56-59.
- [5] GHAREMANI K, SAFA M, YEUNG J, et al. Quality assurance for high-frequency mechanical impact treatment of welds using handheld 3D laser scanning technology[J]. Welding in the World, 2015, 59(3): 391-400.
- [6] 孙瑜,周浩,赵炳巍,等. 多自由度叶片测量机的研究现状与发展[J]. 电子测试, 2020, 18: 20-21, 58.
- [7] QIN X J, HU Z T, ZHENG H B, et al. Surface reconstruction from unorganized point clouds based on edge growing[J]. Advances in Manufacturing, 2019, 7(3): 343-352.
- [8] 胡赤兵,欧志英,严克明. 水轮机叶片修复机器人运行轨迹的研究[J]. 兰州铁道学院学报, 2000(4): 91-93.
- [9] 朱淑梅,刘威,朱俊俊,等. 点云模型粗加工刀轨高效生成方法研究[J]. 制造业自动化, 2017, 39(7): 32-35, 42.

- [10] SZYMON R, MARC L. Efficient variants of the ICP algorithm [C] //Proceedings of the 3<sup>rd</sup> International Conference on 3D Digital Imaging and Modeling. Washington, D. C. :USA IEEE Computer Society, 2001: 145-152.
- [11] 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.
- [12] ZHANG Z Y. Iterative point matching for registration of free-form curves and surfaces [J]. International Journal of Computer Vision, 1994, 13(12): 119-152.
- [13] 韩杰龙. 基于测量的薄壁舱体零件加工余量适应性分配研究[D]. 南京: 南京航空航天大学, 2019.
- [14] 王辉,周明星,郑维珍,等. 高速飞行器薄壁异形零件加工余量的数字化分析与优化[J]. 计算机集成制造系统, 2016, 22(6): 1456-1464.
- [15] KANAN, E. High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation [J]. International Journal of Machine Tools & Manufacture, 2001, 41(9): 1323-1345.
- [16] TOURNIER C, DUC E. A surface based approach for constant scallop height tool-path generation [J]. The International Journal of Advanced Manufacturing Technology, 2002, 19(5): 318-324.
- [17] JIANG J, CHENG J, CHEN X L. Registration for 3D point cloud using angular-invariant feature [J]. Neurocomputing, 2009, 72(16): 3839-3844.
- [18] HUAN L, GUANG J, LONG Q. Fast descriptors and correspondence propagation for robust global point cloud registration [J]. IEEE Transactions on Image Processing, 2017, 26(8): 3614-3623.
- [19] 张顺利,徐艳芝,周明全,等. 基于自适应邻域匹配的点云配准方法[J]. 计算机学报, 2019, 42(9): 2114-2126.
- [20] DEBAYLE J, PRESLES B. Rigid image registration by general adaptive neighborhood matching [J]. Pattern Recognition, 2016, 55: 45-57.
- [21] BESL P J, MCKAY N D. A method for registration of 3D shapes [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, 14(2): 239-256.
- [22] YANG J, LI H, CAMPBELL D, et al. Go-ICP: a globally optimal solution to 3D ICP point set registration [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(11): 2241-2254.
- [23] SERVOS J, WASLANDER S L. A robust framework for multichannel scan registration [J]. Robotics and Autonomous Systems, 2017, 87: 247-257.
- [24] XIONG L, WU L Y, CUI W T, et al. Robust non-rigid registration based on affine ICP algorithm and part-based method [J]. Neural Processing Letters, 2018, 48(3): 1305-1321.
- [25] BAE K H, LICHTI D D. A method for automated registration of unorganized point clouds [J]. Journal of Photogrammetry and Remote Sensing, 2008, 63(1): 36-54.

(下转第251页)