

反求工程中消除 CT 切片轮廓奇异点研究

陈树越 王红亮

(中北大学 仪器科学与动态测试教育部重点实验室, 太原 030051)

摘 要 在基于工业 CT 的反求工程中, 轮廓跟踪是其重要的步骤之一。本文分析了边缘检测后可能存在的断点、毛刺、双像素宽边界等奇异点, 给出了 8 邻域和 5 邻域相结合的跟踪算法。该算法能够有效地消除奇异点, 实现了连续单像素链序列表示的封闭轮廓曲线。通过对某车辆发动机的实验, 表明采用该算法能够准确表达实物的边界, 具有较好的实用性。

关键词 反求工程 工业CT 轮廓跟踪 图像处理

中图分类号: TP391.72 **文献标识码:** A **文章编号:** 1006-8961(2006)08-1110-04

Study of Removing Singular Pixels of CT Slice Contour in Reverse Engineering

CHEN Shu-yue, WANG Hong-liang

(Key Laboratory of Instrumentation Science & Dynamic Measurement, Ministry of Education, North University of China, Taiyuan 030051)

Abstract The contour tracking is one of the important steps in reverse engineering based on industrial computed tomography. In this paper, the singular pixels of CT slice contour after edge detection, such as discontinuity, redundant, double pixels, are analyzed and the contour tracking algorithm combined eight neighborhood calculation with five neighborhood is presented, which removes the singular pixels effectively and gives a close contour with single-pixel connection. It is shown by the experiments of a vehicle engine that the contour can be obtained exactly and the method is practical.

Keywords reverse engineering, industrial CT, contour tracking, image processing

1 引言

随着工业 CT (ICT, industrial computed tomography) 测量精度的不断提高及其无损测量的特点, ICT 断层数据的 CAD 建模已经成为国内外研究的热点^[1-3], 并成为反求工程的重要研究方向之一。

实物产品或样件经前端 ICT 装置处理后, 获得该实物的一系列 2 维断层图像。为了准确地获得实物的结构尺寸, 要求对 ICT 切片中物体的轮廓几何特征进行准确提取^[4,5]。但在 ICT 切片数据的获取过程中不可避免地会产生各种各样的噪声, 如伪迹

等, 这将对实物的边缘检测和跟踪产生不利后果, 直接影响快速成形的精度。其中, 图像轮廓可能存在着断点、毛刺、双像素宽边界等奇异点。

本文在轮廓跟踪时, 针对边缘检测后可能存在的断点、毛刺、双像素宽边界等情况, 进行相应的处理, 得到连续单像素链序列表示的封闭轮廓曲线, 可以较精确地表达实物的边界。从而反求出实物或产品的原型, 最后送入快速成型机, 制造出实物样件。

2 ICT 切片轮廓奇异点消除方法

边界轮廓上的点, 原则上只有且只有两个邻接点,

基金项目: 山西省自然科学基金项目(20031033)

收稿日期: 2005-03-11; 改回日期: 2005-09-12

第一作者简介: 陈树越(1963 ~), 男, 教授。2001 年于南京理工大学获测试计量技术及仪器专业博士学位。主要研究方向为图像处理、无损检测、测试技术。E-mail: csyue2000@yahoo.com.cn

一个为入点,一个为出点。凡是满足这个原则的点就称为非奇异点,否则称为奇异点。奇异点包括断点、毛刺和重叠点。

2.1 断点

断点是指只有入点,没有出点的轮廓点。实验表明一条轮廓线出现断点的次数较少(只有个别轮廓线出现一次),而且两个断点之间的间隔较少(一般为 1~3 个像素点),所以比较容易处理。以断点为圆心,设定一个半径范围 ϵ ,若在此范围内存在若干个尚未跟踪过的边界轮廓上的点,则将这些点中与该断点距离最近的点与断点进行直线连接,线段上经过的点均为补偿的轮廓点。如图 1 所示,点 1 为跟踪路径上的一个断点,在以点 1 为圆心、 $\epsilon = 2$ 为半径的圆范围内存在点 1 与点 3 之间的断点,因此将点 1 和点 3 连接起来,点 2 为补偿点。这样就实现了两断点的连接,使跟踪能继续进行。

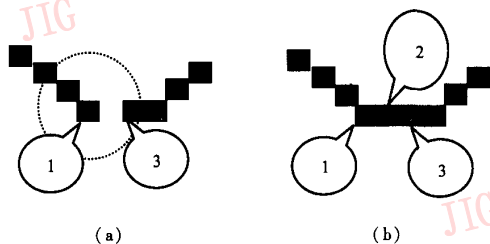


图 1 断点处理

Fig. 1 Broken point processing

2.2 毛刺

毛刺点是指出点不唯一且不是双像素宽的边界点。在毛刺点处,需要寻找和确定跟踪路径的下一个点,并去除毛刺。先将毛刺点和其邻域各点暂置为背景点,判断其邻域各点的连接性,若邻域点为孤立点,则将邻域点去除,否则保留该点。然后,恢复毛刺点和剩余邻点的原值,再判断毛刺点的出点数,若出点数为一个,则毛刺点已去除,可按一般跟踪进行;否则,采用启发式进行,即沿某一邻域点跟踪,若跟踪出错,则回溯到毛刺点,去掉该邻域点,并判断是否仍是毛刺点,若是,则再沿另一邻域点跟踪,如此进行,直到成为非毛刺点,或者启发跟踪成功。以图 2(a)为例,当跟踪到点 2 时,点 2 有两个与它距离相等的出点,即点 3 和点 4,点 2 成为毛刺点;于是将点 2 及其所有邻点(点 1,3,4)都暂置为背景点,这时点 4 仍有边界点与其相连,而点 3 成为



图 2 毛刺处理

Fig. 2 Burr point processing

孤立点,因此删除点 3,点 2 只剩下一个出点,从而毛刺被去除,结果如图 2(b)。

有时会出现将毛刺点和其邻域各点暂置为背景点后,其邻域各点都不为孤立点的情况。此时,可选择其中一邻域点作为毛刺点的出点,按一般跟踪继续进行。若跟踪出错,则回溯到毛刺点,再沿另一邻域点继续跟踪。

2.3 双像素宽边缘的奇异点

对于两像素长、两像素宽的边界,需要将其变为单像素宽。双像素宽边缘奇异点如图 3(a)所示。先将构成双像素宽边的 4 个像素点看成一个整体,用去毛刺方法处理其邻域各点,使其出点和入点都为一个,然后在 4 个像素点中选取分别与入点和出点距离最近的两个像素点为边界轮廓点,并去掉其余两点,从而使边界宽度变为单像素宽。以图 3(a)为例,点 2,3,4,5 组成双像素边界,将这 4 点看成一个整体,则它的入点为 1,出点为 6。保留与点 1 距离最近的点 2、与点 6 距离最近的点 5,删除其余的两点 3 和 4,就使得边界变为单像素宽,如图 3(b)所示。

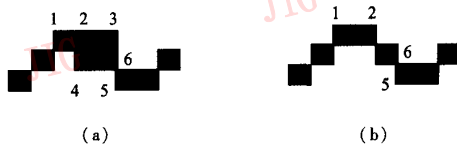


图 3 双像素宽边缘处理

Fig. 3 Double-pixel contour processing

当双像素宽的边界比较长时也可做类似的处理,先将构成双像素宽边界的 N 个像素点看成一个整体,用去毛刺方法处理其邻域各点,使其出点和入点都为一个,然后在 N 个像素点中选取与入点距离最近的像素点 P_i 为边界轮廓点,把 P_i 作为毛刺点,用去毛刺方法处理其邻域各点;然后用同样的方法处理剩余的双像素宽边界,直至其变为单像素宽边界。

3 轮廓跟踪算法

由于实际的 ICT 实物切片图像边缘复杂且很不规则,若只采用 8 邻域跟踪方法,在双像素宽边缘处可能出现死循环,而 5 邻域跟踪则可以避免这一现象,5 邻域如图 4 所示。

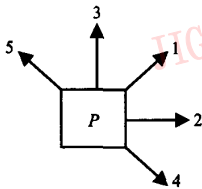


图 4 5 邻域跟踪

Fig. 4 Five neighborhood contour tracking

本文采用一种 8 邻域和 5 邻域相结合的跟踪方法,其跟踪步骤如下:

(1) 逐行扫描整幅图像,若搜索到某边界点,则将该点作为首起点,放入轮廓数据链;若搜索不到边界点,则搜索结束;

(2) 按 8 邻域搜索规则(顺时针或逆时针)搜索首起点的下一个邻点,若存在则把这个邻点作为当前点,放入轮廓数据链,并把其置为背景点,转下一步继续搜索;若不存在,则说明该点是一孤立点,删除该点,转上一步重新开始搜索;

(3) 按 5 邻域搜索规则(方向与 8 邻域一致)继续搜索,此时可能出现一种意外情况,因为未把首起点置为背景点,按 8 邻域搜索到首起点的下一邻点 P 后,再按 5 邻域搜索点 P 的下一邻点,搜索到的可能是首起点,但此时不能认为已得到了一条封闭轮廓。处理办法是:把首起点和其后继邻点交换位置,并把原首起点置为背景点,其后继邻点(现首

起点)恢复为轮廓点。调整后,转下一步;

(4) 继续按 5 邻域搜索规则搜索;

(4.1) 若搜索到一个 5 邻域后继点,将其放入轮廓数据链,并把其置为背景点;若该点是首起点,转第 5 步;否则转第 4 步继续搜索;

(4.2) 若搜索到两个 5 邻域后继点,如果这两个后继点相邻,则进行双像素宽边缘的处理;如果这两个后继点不相邻,则进行毛刺点处理;处理后转第 4 步继续搜索;

(4.3) 若搜索不到 5 邻域后继点,则按 8 邻域搜索规则搜索;

(4.3.1) 若搜索到一个 8 邻域后继点,将其放入轮廓数据链,并把其置为背景点,转第 4 步继续搜索;

(4.3.2) 若搜索到两个 8 邻域后继点,如果这两个后继点相邻,则进行双像素宽边缘的处理;如果这两个后继点不相邻,则进行毛刺点处理;处理后转第 4 步继续搜索;

(4.3.3) 若搜索不到 8 邻域后继点,按欧氏距离计算轮廓链的长度 S ,若 S 小于某一域值 L ,则认为是一虚假轮廓,将其删除,转第 1 步重新开始搜索;若 S 大于等于 L ,而且这条轮廓还没有被逆置(按相反顺序存放)过,则将该轮廓数据链逆置,如果这条轮廓已被逆置(按相反顺序存放)过,则进行断点处理;处理后转第 4 步继续搜索;

(5) 把首起点置为背景点,至此得到一条封闭的单像素链表示的轮廓曲线。转第 1 步搜索下一轮廓曲线。

4 实际应用

对某车辆发动机 ICT 图像进行实验,图 5(a)是



(a) 实物切片边缘图像中的典型边缘



(b) 处理后的边缘

图 5 实验结果

Fig. 5 Experimental result

从某 ICT 切片图像的边缘图像中提取出的典型边缘,可以看出,ICT 切片图像经边缘检测处理后,存在断点、毛刺、双像素宽边界,采用本文的轮廓跟踪方法,去除了断点、毛刺、双像素宽边界,得到图 5(b)所示的单像素数据链表示的边缘。根据本算法对该发动机实物 ICT 切片进行处理,然后通过快速成形机进行加工得到了令人满意的结果。

5 结 论

对反求工程中 ICT 切片轮廓数据中出现的断点、毛刺、双像素宽等奇异点问题进行了分析,给出了消除这些奇异点的轮廓跟踪算法,实现了封闭轮廓的单像素数据链表达。对某车辆发动机 ICT 切片数据进行了处理实验验证,取得了较好的效果,表明该算法具有实际应用价值,同时也可应用于其他相关领域的轮廓跟踪。

参考文献 (References)

- 1 Zheng Wei-guo, Yan Yong-nian, Lu Qing-ping, *et al.* A modeling method for rapid prototyping based on CT images [J]. *China Mechanical Engineering*, 2002, **13**(20): 1734 ~ 1737. [郑卫国, 颜永年, 卢清萍等. 基于 CT 图像的快速成形数据建模方法[J]. *中国机械工程*, 2002, **13**(20): 1734 ~ 1737.]
- 2 Akira Hirakimoto. Microfocus X-ray computed tomography and it's industrial applications [J]. *Analytical Sciences*, 2001, **17**(4): 123 ~ 125.
- 3 Dastarac D. Industry computed tomography at TOMO ADOUR: control and digital [J]. *Computerized Tomography for Industrial Application and Image Processing in Radiology*, 1999, **15**(3): 193 ~ 201.
- 4 Wang Gao-fei, Ning Tao, Xi Ping, *et al.* Edge detection and pattern recognition for model reconstruction on base of the ICT image [J]. *Manufacturing Automation*, 2003, **25**(5): 8 ~ 11. [王高飞, 宁涛, 席平等. ICT 图像实体重建技术中的边缘检测和特征识别[J]. *制造业自动化*, 2003, **25**(5): 8 ~ 11.]
- 5 Yuan Ye, Ou Zong-ying, Hou Jian-hua. Edge detection, tracking, and compensation in CT image surface render [J]. *Mechanical Science and Technology*, 2001, **20**(6): 962 ~ 963. [袁野, 欧宗瑛, 侯建华. CT 图像表面重建技术中的边缘检测和跟踪补偿[J]. *机械科学与技术*, 2001, **20**(6): 962 ~ 963.]

1 Zheng Wei-guo, Yan Yong-nian, Lu Qing-ping, *et al.* A modeling