

# 一种基于八邻域深度差的点云边缘提取算法\*

张志佳<sup>1</sup>, 魏 信<sup>1,2</sup>, 周自强<sup>2</sup>, 李嘉兴<sup>1</sup>, 王婷婷<sup>1</sup>

(1. 沈阳工业大学软件学院 沈阳 110020; 2. 江苏省机电产品循环利用技术重点实验室 常熟 215500)

**摘要:**提出了一种基于八邻域深度差(8N-DD)的点云边缘提取算法。算法根据目标特征点云,对每个特征点沿深度方向进行垂直投影并对投影点进行栅格划分,计算出每个栅格内投影点所对应深度的平均值作为该栅格的深度值;然后比较每个栅格与其八邻域栅格的深度差,根据深度差判断该栅格内是否存在边缘点,并采用排序法从栅格内筛选出目标的点云边缘点。针对含有非孔洞和孔洞的两种典型点云数据,利用八邻域深度差算法进行点云边缘提取,验证了算法的有效性。

**关键词:**点云数据;栅格划分;八邻域深度差;边缘提取

**中图分类号:** TH131 **文献标识码:** A **国家标准学科分类代码:** 520.60

## 8N-DD-based point cloud edge extraction algorithm

Zhang Zhijia<sup>1</sup>, Wei Xin<sup>1,2</sup>, Zhou Ziqiang<sup>2</sup>, Li Jiaying<sup>1</sup>, Wang Tingting<sup>1</sup>

(1. School of software, Shenyang University of Technology, Shenyang 110020, China; 2. Jiangsu Key Laboratory of Recycling and Reuse Technology for Mechanical and Electronic Products, Changshu 215500, China)

**Abstract:** A novel point cloud edge extraction algorithm is proposed in this paper based on eight neighbor depth difference. In this algorithm, according to the point clouds of the object features, each feature point is projected vertically along the depth direction, the projection points are divided to grids, and the average of the depths corresponding to the projection points in each grid is calculated and used as the depth of the grid. Then, the depth of each grid is compared with those of its eight neighbor grids; and according to the depth difference, whether edge point exists in the grid or not is determined, and then the sorting method is used to screen out the point cloud edge point of the object in the grid. Aiming at two typical point cloud data containing non-hole and hole, the proposed eight neighbor depth difference algorithm is used to extract the edges of point clouds, which verifies the effectiveness of the algorithm.

**Keywords:** point cloud data; grid meshing; eight neighbor depth difference (8N-DD); edge extraction

## 0 引 言

随着激光测量技术的不断发展,激光测量设备不断推出,使逆求工程得到不断发展。在逆求工程设计中,可以将某个产品表面表达为大量的离散点,即点云。通过点云可进行逆求工程设计,如何从点云信息中提取产品的一些几何特征,成为逆求工程的设计核心。

点云边缘是指能够表达产品特征的一些边缘测量点。点云边缘不仅能表达产品的几何特征,而且在产品识别、重建曲面<sup>[1-2]</sup>模型的品质和精度中起着重要的作用。目前,在基于点云的边缘提取方面,国内外在曲面重构<sup>[3-5]</sup>、

空洞修补<sup>[6-7]</sup>、曲面特征检测<sup>[8-9]</sup>等领域进行了研究。

点云包括孔洞与非孔洞两类。目前,提取非孔洞点云边缘轮廓的方法主要有两类:利用点云分割的凸包算法提取点云边缘<sup>[10]</sup>;建立空间拓扑关系并通过邻近点构成的角度差算法提取点云边缘<sup>[11]</sup>。提取孔洞点云边缘轮廓的方法主要也分为两类: $\alpha$ -shape 算法<sup>[12-13]</sup>和网格拓扑关系算法<sup>[14]</sup>。其中的 $\alpha$ -shape 算法可想象为一个半径为 $\alpha$ 的圆在点云里滚动,当 $\alpha$ 满足一定条件,圆会在点云边缘滚动,其滚动的痕迹就是点云的边缘。

本文提出了一种基于八邻域深度差(eight neighbor depth difference, 8N-DD)的点云边缘提取方法。该方法对含有孔洞和非孔洞点云均能有效进行边缘提取。

# 1 算法描述

八邻域深度差算法首先对点云沿深度方向进行垂直投影,对投影点进行栅格数据组织,然后计算每个栅格的深度值,比较栅格与其八邻域栅格的深度差判断栅格内是否存在边缘点,对存在边缘点的栅格进行筛选并提取点云边缘点。

## 1.1 对点云沿深度方向垂直投影

对沿深度方向获取的单面点云数据,沿深度方向进行投影(相当于垂直投影到  $x, y$  平面上)。

点云数据可表示为:

$$P = \{P_i | P_i = (x_i, y_i, z_i), i \in [1, n]\} \quad (1)$$

式中:  $P_i$  表示空间点  $(x_i, y_i, z_i)$  的位置坐标,  $n$  表示点云个数。

投影点的集合可表示为:

$$P' = \{P'_i | P'_i = (x_i, y_i), i \in [1, n]\} \quad (2)$$

式中:  $P'_i$  表示  $(x_i, y_i)$  处位置坐标。

## 1.2 对投影点集合进行栅格数据组织

对于投影点的集合  $P'$ , 统计其横纵坐标最小与最大的四个值  $X_{\min}, X_{\max}, Y_{\min}, Y_{\max}$ ; 根据栅格的划分次数  $m$ , 计算出单个栅格的大小为  $a \times b$ , 其中:

$$a = \left\lceil \frac{X_{\max} - X_{\min}}{m} \right\rceil \quad (3)$$

$$b = \left\lceil \frac{Y_{\max} - Y_{\min}}{m} \right\rceil \quad (4)$$

式中:  $\lceil \cdot \rceil$  表示取整。

将各投影点分配到各个栅格且进行编号: 首先根据当前投影点坐标  $(x, y)$ , 计算该点所在栅格的行  $r$  和列  $c$ ; 然后将该点添加到第  $r$  行和  $c$  列的栅格中, 且将该栅格编号为  $G(r, c)$ , 如图 1 所示, 其中:

$$r = \left\lceil \frac{y}{b} \right\rceil \quad (5)$$

$$c = \left\lceil \frac{x}{a} \right\rceil \quad (6)$$

式中:  $\lceil \cdot \rceil$  表示取整。

栅格的详细组织方式如图 1 所示, 栅格定义为  $3 \times 3$  栅格, 从栅格左上角对栅格进行编号, 第 1 行第 1 列栅格为  $G(0,0)$ , 第 1 行第 2 列栅格为  $G(0,1)$ , 直到将第 3 行第 3 列栅格编号为  $G(2,2)$ 。

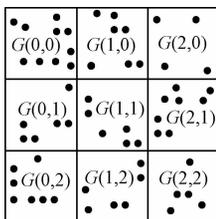


图 1  $3 \times 3$  栅格划分  
Fig. 1  $3 \times 3$  grid division

## 1.3 计算栅格的深度值

对于任意栅格  $G(i, j)$ , 统计栅格内投影点的数目(见图 2), 若投影点数目为零, 则将该栅格的深度值设定为  $Z = 0$ ; 若栅格内投影点数目大于 0, 则将投影点所对应的深度值进行加权平均, 将加权平均值  $Z_{ave}$  作为该栅格的深度值, 栅格深度值可表示为:

$$Z_{ave} = \frac{\sum_{n=1}^L D_m}{L} \quad (7)$$

式中:  $D_m$  表示栅格内投影点所对应的深度值,  $L$  表示栅格内投影点的个数。

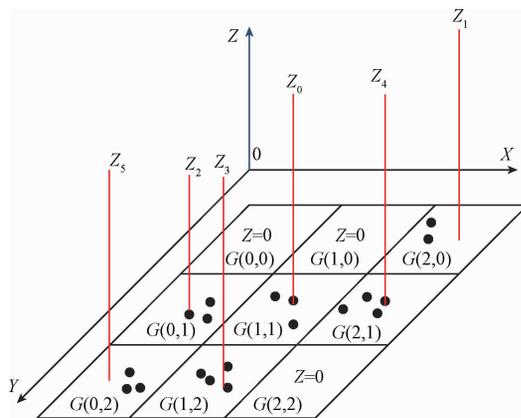


图 2 栅格的深度平均值  
Fig. 2 The average depth of the grid

图 2 给出了  $3 \times 3$  栅格的深度值设定, 栅格  $G(0,0)$ ,  $G(1,0)$  和  $G(2,2)$  中无投影点, 则栅格的深度值设定为零; 其它栅格内都存在投影点, 则栅格的深度值设定为对应的加权平均值, 如  $Z_1, Z_2, Z_3, Z_4, Z_5, Z_6$ 。

## 1.4 栅格间深度比较

对于任意栅格  $G(i, j)$ , 以其八邻域栅格中  $G(i-1, j)$  栅格为例, 其与栅格  $G(i, j)$  的深度差为  $D$ ,  $D$  大于阈值  $T$ , 判断出栅格  $G(i, j)$  存在边缘点, 如图 3 所示。

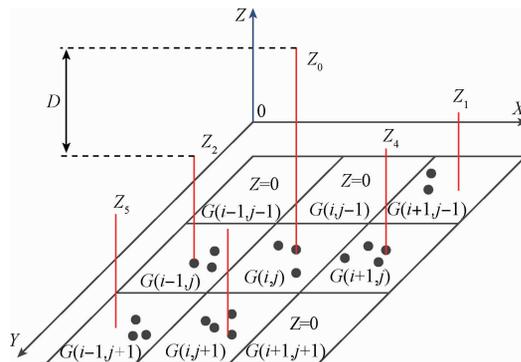


图 3 栅格存在边缘点示意图  
Fig. 3 The grid in which edge point exists

在局部范围内统计深度值,根据深度值出现的频率极大值所对应的深度值确定阈值  $T(T = T_2 - T_1)$ ,其中  $T_1$  和  $T_2$  分别表示深度值在局部范围内的两个连续频率极大值,如图 4 所示。

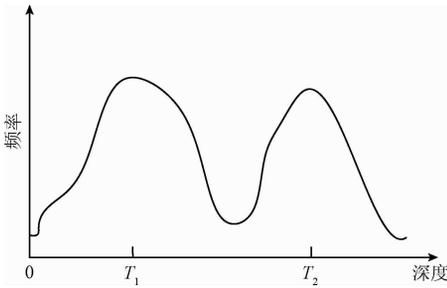


图 4 局部阈值获取示意图

Fig. 4 Schematic diagram of local threshold acquisition

### 1.5 栅格内点云边缘点筛选

对于存在边缘点的栅格  $G(i, j)$ ,统计出栅格内投影点个数  $k$ ,然后对投影点所对应的深度值进行升序排列。当  $k$  为奇数,栅格边缘点为  $k + 1/2$  位置所对应的投影点;当  $k$  为偶数,该栅格边缘点为  $k/2$  或  $k/2 + 1$  位置所对应的投影点,根据投影点提取其所对应的三维点,图 5 所示为边缘点筛选示意图。

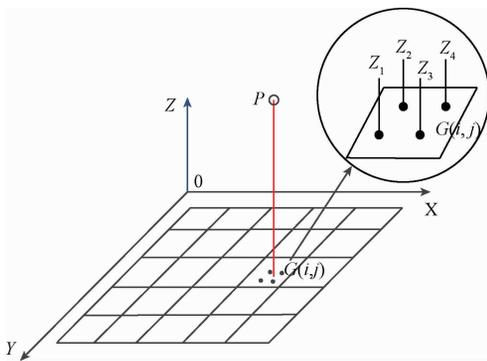


图 5 栅格深度筛选示意图

Fig. 5 Grid depth screening

如图 5 所示,存在边缘点的栅格  $G(i, j)$  中有 4 个投影点,对 4 个投影点所对应的深度值排序为  $Z_1, Z_2, Z_3$  和  $Z_4$ ,所以  $Z_2$  或  $Z_3$  所对应的投影点为边缘点。

## 2 算法比较及实验分析

### 2.1 边缘识别精度

为了检测本文算法的边缘检测精度,设计方形孔洞点云(见图 6),孔洞边缘 4 个顶点的原始坐标为  $(P_1(x_1,$

$y_1, z_1), P_2(x_2, y_2, z_2), P_3(x_3, y_3, z_3), P_4(x_4, y_4, z_4)$ ), 根据该 4 个顶点计算出 4 条边的直线方程是  $(L_1, L_2, L_3, L_4)$ 。用 8N-DD 算法、 $\alpha$ -shape 和网格拓扑算法分别提取边缘,结果如图 7 所示。

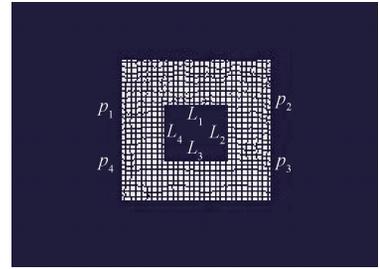


图 6 方形孔洞点云

Fig. 6 Square hole point cloud

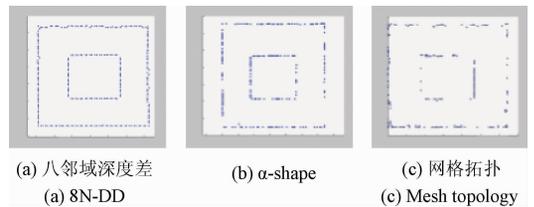


图 7 不同算法边缘提取结果

Fig. 7 Edge extraction results of different algorithms

通过对所提取的孔洞边缘进行直线拟合计算,得出所提取的孔洞边缘 4 条边方程  $(L'_1, L'_2, L'_3, L'_4)$  和 4 个顶点  $(P'_1, P'_2, P'_3, P'_4)$ 。

将原始的直线方程和顶点坐标与边缘提取后得到的直线方程和顶点坐标进行比较和精度分析(为了简化分析,去掉  $Z$  坐标,只考虑二维情况),结果如表 1 与 2 所示。表 1 所示为原始直线方程与经过算法提取的直线方程的相对误差结果,表 2 所示为原始顶点坐标与经过算法提取的顶点坐标的相对误差结果,其中  $\varepsilon_r(x)$  表示竖直线的相对误差,  $\varepsilon_r(y)$  水平线的相对误差:

$$\varepsilon_r(x) = X - X^* \tag{8}$$

$$\varepsilon_r(y) = Y - Y^* \tag{9}$$

表 1 拟合后的直线方程与原始直线方程的误差

Table 1 The errors between original line equation and fitting line equation

	相对误差	8N-DD	$\alpha$ -shape	网格拓扑
$L_1$ 直线	$\varepsilon_r(y)$	-0.000 1	0	0.000 1
$L_2$ 直线	$\varepsilon_r(x)$	0	0	0.000 1
$L_3$ 直线	$\varepsilon_r(y)$	0	0.000 1	0
$L_4$ 直线	$\varepsilon_r(x)$	0	0	0.000 1

表2 提取出的方形孔洞4个顶点与原始方形孔洞4个顶点点云的误差

Table 2 The point cloud errors of 4 vertexes of original and extracted square hole

误差	8N-DD				$\alpha$ -shape				网格拓扑			
	$\Delta P_1$	$\Delta P_2$	$\Delta P_3$	$\Delta P_4$	$\Delta P_1$	$\Delta P_2$	$\Delta P_3$	$\Delta P_4$	$\Delta P_1$	$\Delta P_2$	$\Delta P_3$	$\Delta P_4$
$\Delta x$	0	0.000 2	0	0	-0.000 2	0.000 2	0	-0.000 1	0.000 1	0.000 2	-0.000 1	0.000 1
$\Delta y$	-0.000 1	-0.000 1	0	0.000 1	0	0	0.000 1	0	-0.000 1	-0.000 3	0	-0.000 1

由表1和2可以看出,3种算法提取的边缘直线和边缘坐标与原点云直线和坐标间的误差均较小,所以3种算法的点云边缘提取稳定性较好,但从图7可以看出8N-DD点云边缘提取效果最佳。

## 2.2 边缘识别耗时

### 2.2.1 点云获取

本文为了验证算法的有效性,采用了两种设备提取的点云数据。一种是本文利用 Kinect 设备<sup>[15]</sup>(见图8(a))采集的点云数据;另外一种是从网上下载的 Atos 扫描仪(见图8(b))获取的点云数据<sup>[16]</sup>。

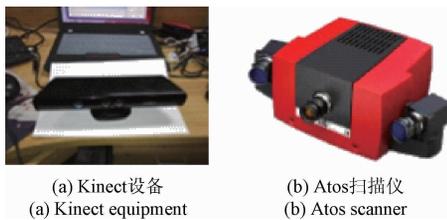


图8 点云获取设备

Fig. 8 The point cloud acquisition device

#### 1) Kinect 获取的点云数据

对于非孔洞点云,利用 Kinect 获取图9(a)所示的竖直木板上六角螺母和方形螺母点云数据,该点云有6 143个三维点,点云如图9(b)所示。

对于孔洞点云,利用 Kinect 提取图9(c)所示钢板点云数据,该点云有9 856个三维点,截取局部点云如图9(d)所示。

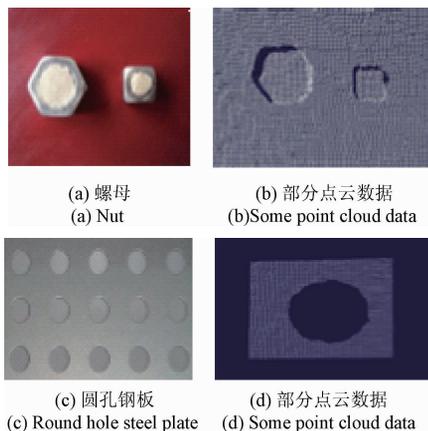


图9 Kinect 获取的点云数据

Fig. 9 Point cloud data acquired by Kinect

#### 2) Atos 获取的点云数据

对于非孔洞点云数据,在网上下载了 Atos 扫描仪对图10(a)获取的图10(b)所示的点云数据,该点云有59 190个三维点。

对于孔洞点云数据,在网上下载了 Atos 扫描仪对图10(c)获取的局部点云数据如图10(d)所示,该点云有9 942个三维点。

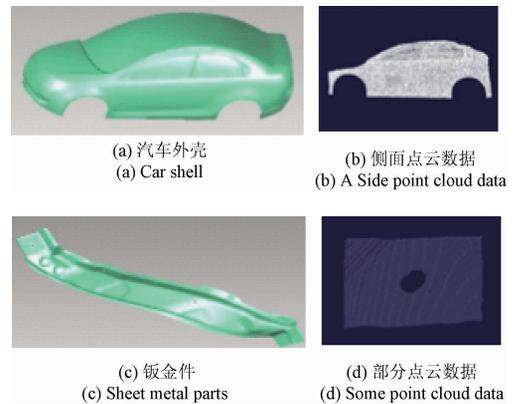


图10 Atos 获取的点云数据

Fig. 10 Point cloud data acquired by Atos

### 2.2.2 算法比较

对于非孔洞点云,分别用凸包算法、交互提取算法和8N-DD算法进行点云边缘提取并进行对比;对于孔洞点云,分别用 $\alpha$ -shape算法、基于拓扑的空间网格法和8N-DD算法进行点云边缘提取并进行对比。

#### 1) 非孔洞点云数据

对图9(b)和图10(b)的非孔洞点云用本文提出的8N-DD算法与凸包算法和交互算法进行比较分析。

利用3种算法对两种非孔洞点云数据进行边缘提取,结果如图11所示。从图11可以看出,8N-DD算法对于孔洞点云边缘提取准确性较高。点云边缘提取实验结果如表3所示,从表3可以看出,本文八邻域深度差算法与另外两种算法边缘提取时间相差不多。

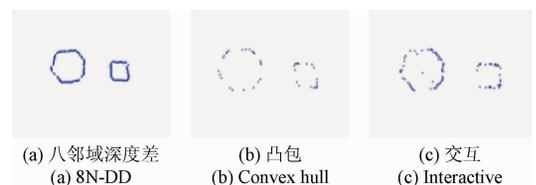


图11 非孔洞点云边缘提取结果比较

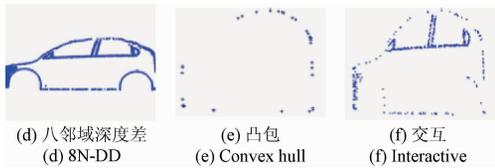


图 11 不同算法边缘提取结果

Fig. 11 Edge extraction results of different algorithms

表 3 非孔洞点云边缘提取时间

Table 3 Edge extraction time for non-hole point cloud

算法	Kinect/s	Atos/s
8N-DD	3.652 0	19.408 0
凸包	0.210 0	19.487 0
交互	9.859 0	50.180 0

2) 孔洞点云数据

对图 9(d) 和图 10(d) 获取的孔洞点云用本文提出的 8N-DD 算法与  $\alpha$ -shape 算法和基于拓扑的网格划分算法进行比较分析。

利用 3 种算法对两种孔洞点云数据进行边缘提取, 结果如图 12 所示。从图 12 可以看出, 8N-DD 算法对于孔洞点云边缘提取准确性较高。算法的时间消耗对比如表 4 所示。在  $\alpha$ -shape 算法建立点云  $K$  维 ( $K$ -dimensional, KD) 树和网格拓扑算法建立点云拓扑关系时需要重复搜索点云数据, 比较耗时, 而本文提出的八邻域深度差算法在栅格数据组织时只需单次搜索点云数据, 能够节省搜索时间。在表 4 中可以看出, 本文提出的方法与  $\alpha$ -shape 和网格拓扑算法相比, 能够节约两个数量级以上的计算时间。

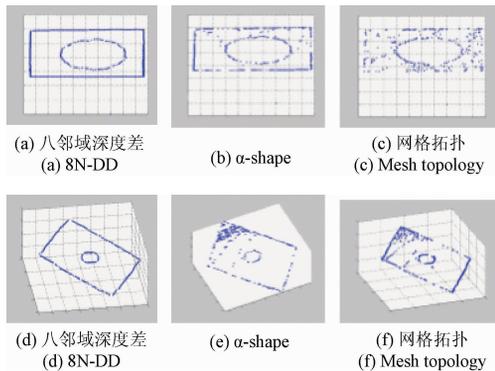


图 12 不同算法边缘提取结果

Fig. 12 Edge extraction results of different algorithms

表 4 孔洞点云边缘提取时间

Table 4 Edge extraction time for hole point cloud

算法	Kinect/s	Atos/s
8N-DD	0.881 0	0.917 0
$\alpha$ -shape	145.420 0	149.136 0
网格拓扑	151.185 0	155.144 0

### 3 算法应用及分析

#### 3.1 算法应用

微软的 Kinect 产品性能稳定、价格低廉, 已经投入市场多年。基于微软的 Kinect 传感器采集点云数据进行典型零部件的识别并定位, 进而获取自动化拆卸知识, 降低了面向自动化拆卸的知识获取平台的系统复杂性, 是自动化拆卸知识获取领域的一个发展方向。

本文提出的 8N-DD 算法已经应用于面向自动化拆卸的知识获取平台中的零部件识别系统。该系统利用 Kinect 传感器获取典型零部件的点云数据, 基于 8N-DD 点云边缘提取和随机抽样一致性 (random sample consensus, RANSAC) 等算法, 可以实现典型零部件的类型识别与空间定位。

零部件识别系统界面如图 13 所示, 界面中“开始检测”按钮集成点云采集、点云边缘提取、点云边缘分割、RANSAC 算法特征提取以识别零部件等部分, 操作人员只需点击“开始检测”即可实现零部件的识别。

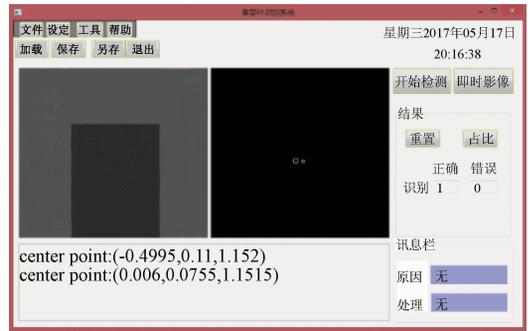


图 13 零部件识别系统界面

Fig. 13 The interface of part identification system

如图 13 所示, 界面中的“加载”按钮能够加载已保存的深度图像; “保存”按钮能够保存深度图像和点云数据; “即时影像”按钮能够实时显示 Kinect 获取的深度图像; “结果”栏里显示识别结果; “讯息”栏里显示一些异常信息和解决办法; 文本框里显示提取的轮廓中心点坐标。

#### 3.2 应用分析

利用零部件识别系统, 以不同的螺母零件为实验对象, 进行了 8N-DD 算法的应用试验, 如图 14 所示。将螺母安装于设备上, 利用 Kinect 传感器对其进行点云提取, 然后利用本文提出的 8N-DD 算法对其进行边缘提取, 最后对点云边缘利用 RANSAC 对点云进行特征提取, 根据特征识别螺母类型。



图14 螺母识别实验平台

Fig. 14 Nut identification experiment platform

以40个不同形状和大小的不锈钢螺母为样本进行实验,结果如表5所示,实现了100%的识别率并准确定位。

表5 螺母实验对象

Table 5 Experiment object of nuts

六角螺母	M22	M24
数量	10	10
方形螺母	M10	M12
数量	10	10

## 4 结 论

本文提出了一种基于8N-DD的点云边缘提取算法。该算法对点云沿深度方向进行投影,对投影点进行栅格数据组织,计算栅格与其八邻域栅格的深度差,从而识别出点云边缘特征点。对于非孔洞点云,用基于点云分割的凸包算法、交互提取算法和8N-DD算法进行了对比;对于孔洞点云,用 $\alpha$ -shape算法、基于拓扑的空间网格法和8N-DD算法进行了对比。实验结果表明,八邻域深度差算法能快速有效的提取点云边缘,能够进一步应用于基于点云的目标识别与应用。

## 参考文献

[1] 刘辉,王伯雄,任怀艺,等. 基于三维重建数据的双向点云去噪方法研究[J]. 电子测量与仪器学报, 2013, 27(1):1-7.  
LIU H, WANG B X, REN H Y, et al. Two-way point cloud denoising method based on three-dimensional reconstruction data [J]. Journal of Electronic

Measurement and Instrument, 2013, 27(1):1-7.

- [2] 李江雄,吴恩启,柯映林. 基于结构光的微细管孔内表面三维重建[J]. 仪器仪表学报, 2006, 27(3): 254-258.  
LI J X, WU E Q, KE Y L. 3D Reconstruction of Small-diameter Pipes Inner Surface Based on Structural Light[J]. Chinese Journal of Scientific Instrument, 2006, 27(3):254-258.
- [3] 叶井飞,高志山,刘晓莉,等. 基于Zernike多项式和径向基函数的自由曲面重构方法[J]. 光学学报, 2014, 34(8):233-241.  
YE J F, GAO ZH SH, LIU X L, et al. Freeform surfaces reconstruction based on Zernike polynomials and radial basis function[J]. Acta Optical Sinica, 2014, 34(8): 233-241.
- [4] 杨贵军,柳钦火,武文波. 三维激光扫描仪曲面重构算法研究[J]. 仪器仪表学报, 2005, 26(11): 1181-1183.  
YANG G J, LIU Q H, WU W B. Research on 3D laser scan surface reconstruction algorithms [J]. Chinese Journal of Scientific Instrument, 2005, 26(11): 1181-1183.
- [5] QI C, PENG G, DENNIS B, et al. Filtering airborne laser scanning data with morphological methods [J]. Photogrammetric Engineering & Remote Sensing, 2007(73): 175-185.
- [6] 蒋刚. 基于SVM和空间投影的点云空洞修补方法[J]. 计算机工程, 2009, 35(22):269-271.  
JIANG G. Point cloud hole filling method based on SVM and space projection[J]. Computer Engineering, 2009, 35(22): 269-271.
- [7] 唐煜程,张明君,王浩宇,等. 基于GPU的三维人脸数据动态线性快速修复[J]. 电子测量与仪器学报, 2016, 30(6):959-967.  
KANG Y CH, ZHANG M J, WANG H Y, et al. Fast linear recovering algorithm for low quality 3D face data based on GPU[J]. Journal of Electronic Measurement and Instrument, 2016, 30(6):959-967.
- [8] 王凯,穆志纯. 基于沟回结构特征的三维人耳识别方法[J]. 仪器仪表学报, 2014, 35(2): 313-318.  
WANG K, MU ZH CH, 3D human ear recognition method based on auricle structural feature[J]. Chinese Journal of Scientific Instrument, 2014, 35(2): 313-318.
- [9] 张志佳,尹秀萍,苑玮琦. 碗形塞加工误差双目视觉在线检测方法[J]. 红外与激光工程, 2016, 45(12): 261-268.  
ZHANG ZH J, YIN X P, YUAN W Q. An online detection method of mechanical error for bowl plug based

- on binocular vision[J]. *Infrared and Laser Engineering*, 2016,45(12):261-268.
- [10] 徐伟恒,冯仲科,苏志芳,等.一种基于三维激光点云数据的单木树冠投影面积和树冠体积自动提取算法[J].*光谱学与光谱分析*,2014,34(2):465-470.  
XU W H, FENG ZH K, SU ZH F, et al. Automatic extraction algorithm of single tree crown projection area and canopy volume based on 3D laser point cloud data[J]. *Spectral and Spectral Analysis*, 2014,34(2):465-470.
- [11] 刘增益,江开勇,林俊义.散乱点云特征边缘交互提取[J].*计算机工程与应用*,2016,52(6):186-190.  
LIU Z Y, JIANG K Y, LIN J Y. Interactive extraction of boundary of specified target feature on scattered point cloud [J]. *Computer Engineering and Applications*, 2016, 52(6):186-190.
- [12] 范启忠,陈是扞,刁燕,等.基于 Alpha Shapes 三维重建原理的机器人工作空间求解方法[J].*机械传动*,2014,38(9):27-29,59.  
FAN Q ZH, CHEN SH Q, DIAO Y, et al. Solution method of robot workspace based on alpha shapes 3D reconstruction principle [J]. *Mechanical Drive*, 2014, 38(9):27-29,59.
- [13] 王宗跃,马洪超,徐宏根,等.海量点云的边缘快速提取算法[J].*计算机工程与应用*,2010,46(36):213-215.  
WANG Z Y, MA H CH, XU H G, et al. Novel algorithm for fast extracting edges from massive point clouds [J]. *Computer Engineering and Applications*, 2010, 46(36):213-215.
- [14] 吕书明,张明磊,孙树立.基于简化和细分技术的三角网格拓扑优化方法[J].*计算机辅助设计与图形学学报*,2014,26(8):1225-1231.  
LV SH M, ZHANG M L, SUN SH L. Topological optimization for triangular mesh based on simplication and subdivision [J]. *Journal of Computer-Aided Design & Computer Graphics*, 2014,26(8):1225-1231.
- [15] 张志佳,魏信,周自强,等.基于 Kinect 的拆卸工具运

动轨迹识别方法 [J]. *制造业自动化*,2016,38(12):140-145.

ZHANG ZH J, WEI X, ZHOU Z Q, et al. Recognition of removal tool trajectory based on Kinect [J]. *Manufacturing Automation*, 2016,38(12):140-145.

- [16] 张潇予,徐熙平.基于 ATOS 的异形件逆向工程实现技术分析[J].*电子测试*,2014(10):44-45.  
ZHANG X Y, XU X P. The shaped pieces ATOS-based reverse engineering to achieve technical analysis [J]. *Electronic Test*,2014(10):44-45.

## 作者简介



张志佳,分别在1996年和2002年于东北大学获得学士和硕士学位,2006年于中国科学院自动化研究所获得博士学位,现为沈阳工业大学副教授、硕士生导师,主要研究方向为机器视觉检测技术、图像处理与模式识别。

E-mail: zzjsut@126.com

**Zhang Zhijia** received his B. Sc. and M. Sc. degrees both from Northeastern University in 1996 and 2002, and respectively, received his Ph. D. degree from Institute of Automation, Chinese Academy of Sciences in 2006. Now, he is an associate professor and master student supervisor in Shenyang University of Technology. His main research interest includes machine vision, image processing and pattern recognition.



周自强(通讯作者),1996年于安徽工程大学获得学士学位,2000年于东北大学获硕士学位,2005年于中国科学技术大学获博士学位,现为常熟理工学院机械学院副教授,主要研究方向为智能制造和机电一体化。

Email: zzq\_rrme@163.com

**Zhou Ziqiang** (Corresponding author) received his B. Sc. degree from Anhui Polytechnic University in 1996, M. Sc. degree from Northeastern University in 2000 and Ph. D. degree from University of Science and Technology of China in 2005. Now, he is an associate professor in School of Mechanical Engineering, Changshu Institute of Technology. His main research interest includes intelligent manufacturing and mechatronics.