

DOI: 10.19650/j.cnki.cjsi.J2209126

# 基于梯度聚类的有序点云边缘优化提取方法<sup>\*</sup>

陈浩<sup>1,2</sup>, 丁其川<sup>1</sup>, 潘磊<sup>1</sup>

(1. 东北大学机器人科学与工程学院 沈阳 110000; 2. 东北大学信息科学与工程学院 沈阳 110000)

**摘要:** 应用传统的2D边缘检测器检测低分辨率深度图中物体边缘时,边缘检测精度较差,召回率低;而当前基于3D点云的边缘提取方法也存在实时性差、抗干扰能力弱等缺点。为此,提出一种基于梯度聚类的边缘优化提取方法,实现从有序点云中快速、稳定地检测物体的边缘。首先,通过邻域点距离分析滤除飞行像素噪声,消除边缘误检;其次,提出一种基于梯度聚类的边缘点/非边缘点分离方法,快速获取物体的粗边缘;最后,结合快速平行细化算法与掩膜滤波,优化粗边缘,获得物体精确边缘。在公共数据集和TOF相机实测数据上进行实验验证。结果表明,提出方法的实时性与检测精度均优于现有方法,在实测数据中的边缘检测精度达89%,FPS达28 fps。

**关键词:** 点云;深度相机;边缘检测;邻域点距离分析;梯度聚类

**中图分类号:** TH164 **文献标识码:** A **国家标准学科分类代码:** 520.60

## Edge optimized extraction from the organized point-cloud data base on the gradient clustering

Chen Hao<sup>1,2</sup>, Ding Qichuan<sup>1</sup>, Pan Lei<sup>1</sup>

(1. Faculty of Robot Science and Engineering, Northeastern University, Shenyang 110000, China;

2. College of Information Science and Engineering, Northeastern University, Shenyang 110000, China)

**Abstract:** When the traditional 2D edge detectors are applied to detect object edges in low-resolution depth images, the detection accuracy is poor and the recall rate is low. At present, the existing edge extraction methods based on the 3D point-cloud data have poor real-time performance and weak anti-interference ability. To address these issues, an edge optimized extraction method based on the gradient clustering is proposed to fast and stably detect the 3D edges of objects from the organized point-cloud data. First, the flying pixel noise is filtered to eliminate false detection on the edge by analyzing the distance between neighborhood points. Secondly, an edge/no-edge point separation method based on the gradient clustering is proposed to fast extract the rough edges of objects. Finally, the combination of the fast parallel thinning and the mask filtering is employed to optimize the rough edge. In this way, the precise edges are obtained. Experiments are implemented on the public datasets and a dataset collected by a TOF depth camera to evaluate the proposed method. Results show that the proposed method is superior to the existing methods in the real-time and detection accuracy. With the real data, the edge detection is accuracy 89%, and the FPS achieves 28 fps.

**Keywords:** point cloud; depth camera; edge detection; neighborhood point distance analysis; gradient clustering

## 0 引言

利用深度相机可以获得场景中物体表面点的3D坐标( $x$ 、 $y$ 和 $z$ ),而3D坐标点的集合则构成了描述场景内物体属性的有序点云<sup>[1]</sup>。目前深度视觉已广泛应用于场

景语义分割<sup>[2]</sup>、人体姿态识别<sup>[3]</sup>、缺陷检测<sup>[4]</sup>等领域,从有序点云中快速、准确地提取物体特征是各项应用的基本需求,其中边缘提取是广受研究者关注的主要任务。尽管基于有序点云或深度图的边缘提取算法层出不穷,但真正应用到智能化工业设备上的寥寥无几,其主要原因在于,受深度相机低分辨率及飞行像素干扰的影响,现

收稿日期:2022-01-03 Received Date: 2022-01-03

<sup>\*</sup> 基金项目:国家自然科学基金面上项目(61973065,61973063)、辽宁省科技厅联合开放基金机器人学国家重点实验室开放基金资助项目(2020-KF-12-02)、中央高校基本科研业务费专项基金项目(N2226002)资助

有的点云边缘提取方法无法满足实时性、稳定性及高精度等综合要求<sup>[5]</sup>。

现有方法通常采用两种途径从有序点云中提取物体边缘:1)直接利用 2D 边缘检测器检测深度图中的物体边缘<sup>[6-7]</sup>;2)通过近邻点特征分析提取边缘<sup>[8-9]</sup>。但上述途径在实际应用中都存在一定的弊端。首先途径 1 中的 2D 边缘检测器对深度图的检测精度会因为分辨率降低而下降<sup>[6]</sup>,其次途径 2 中近邻点的特征提取(如法向量<sup>[8]</sup>、邻域点夹角<sup>[9]</sup>等)涉及复杂的计算过程,影响算法的实时性;并且易受孤立点、飞行像素等噪声干扰,导致边缘定位精度低。

针对 2D 边缘检测器用于低分辨率深度图时,边缘检测精度差、召回率低的问题。部分学者尝试采用形态学方法<sup>[10]</sup>、引导滤波<sup>[11]</sup>等方法对深度图优化处理,降低噪声影响,并改进 Canny<sup>[12]</sup>、Laplace<sup>[13]</sup>、DLBP (depth local binary pattern)<sup>[14]</sup>等传统检测算法的算子,以实现深度图中边缘精确检测,但是这些方法仅适用于单一或小噪声干扰的简单场景,且引入的阈值或算子约束限制了方法的适应性与拓展性,当深度图出现孤立点、飞行像素等多干扰时,检测精度较低、稳定性差。

针对近邻点特征分析影响 3D 边缘检测实时性的问题,优化邻域栅格划分及自适应调整网格大小与方向是研究者普遍采用的提升实时性的方法<sup>[15-17]</sup>。文献[15]提出基于网格的可变半径 A-Shapes 方法提取点云中的物体轮廓,通过自适应调整半径,优化网格构建,提升边缘检测速度,但该方法仅在有较多拐角特征的场景中才

能发挥作用。通过栅格划分策略替代单点法向量计算<sup>[16]</sup>,或者平面邻域夹角替代法向量夹角<sup>[17]</sup>等方法也可以有效提升检测算法的实时性能,但这些方法也都存在适应场景受限,无法克服孤立点、飞行像素等噪声导致的 3D 边缘误检问题。

为实现从低分辨率深度图中快速检测出物体的边缘,并克服孤立点、飞行像素等干扰导致的误检问题,提出一种基于梯度聚类的边缘优化提取方法,实现从有序点云中快速、稳定地检测物体的边缘。本文主要贡献是:首先,通过对有序点云进行邻域点距离分析,实现飞行像素噪声的检测与滤除,以提高边缘检测的稳定性;其次,提出一种基于梯度聚类的边缘点/非边缘点分离方法,通过对有序点云全局梯度特征自动聚类,解决了仅使用深度图导致的几何特征丢失问题以及逐点计算特征的耗时问题,并根据场景自适应调节阈值,以快速获取物体的粗边缘;最后,结合快速平行细化算法与掩膜滤波,优化粗边缘、滤除孤立点噪声,进一步提高边缘检测的稳定性。在公共数据集和 TOF (time-of-flight) 深度相机实测数据上进行了大量实验验证。实验结果表明,本文提出方法的实时性及检测精度都优于现有方法。

1 算法分析

图 1 最后一行给出了本文算法的流程图,同时也给出了途径 1(图 1(a)),途径 2(图 1(b))的算法流程图。

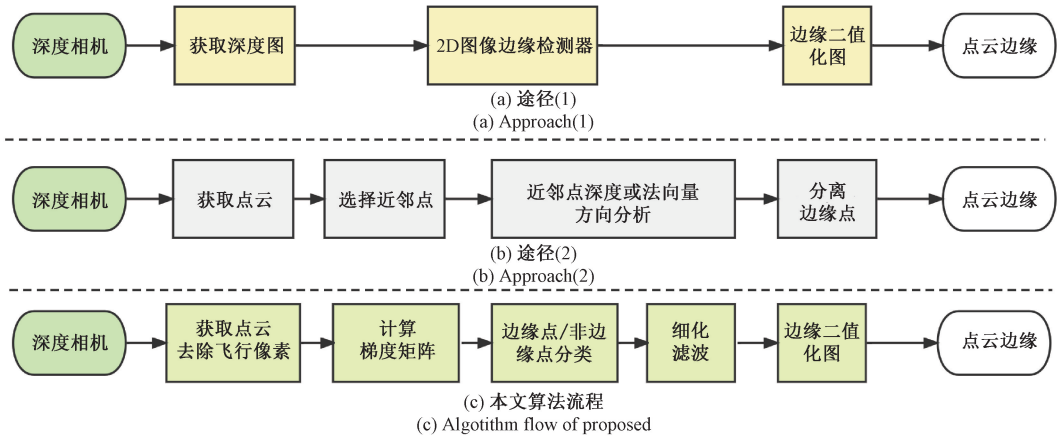


图 1 边缘提取方法流程图

Fig. 1 Flow chart of the edge extraction method

1.1 邻域点距离分析去除飞行像素

由深度相机测量得到前景与背景之间的深度值,在 3D 点显示的时候会产生“飞行像素”,飞行像素(图 2(a)中的方形点)通常与其他普通像素点(图 2(a)中的圆形点)分离,并且出现在不存在实体表面的地

方<sup>[18]</sup>。由于飞行像素直接影响物体边缘的准确定位,针对这一问题,本文采用一种基于相机角分辨率 $\beta$ 的飞行像素检测与滤除方法,其原理如图 2(b)所示,其中点 $w_1$ 、 $w_2$ 为相机平面上两相邻像素点 $c_1$ 、 $c_2$ (图 2(b)中的阴影区域)在世界坐标系中的 3D 投影点, $d$ 为 $c_1$ 像素点测得

的距离,则  $w_1$  和  $w_2$  之间的距离阈值为:

$$\eta_{thre} = k \times 2d \times \sin(\beta/2) \quad (1)$$

其中,  $k$  为比例因子,本文设置  $k=4$ ,如果  $w_1$  与其邻域 3D 投影点之间的欧式距离大于  $\eta_{thre}$ ,则  $w_1$  被认为是飞行像素点。

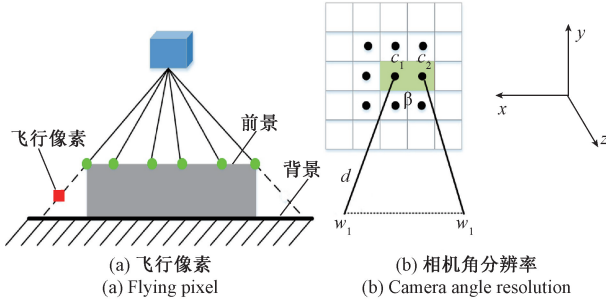


图2 飞行像素和相机角分辨率示意图

Fig. 2 Diagram of flying pixels and camera angle resolution

图3说明了飞行像素滤除的效果,从单一的箱体到复杂的遮挡场景中,飞行像素的滤除性能保持稳定。

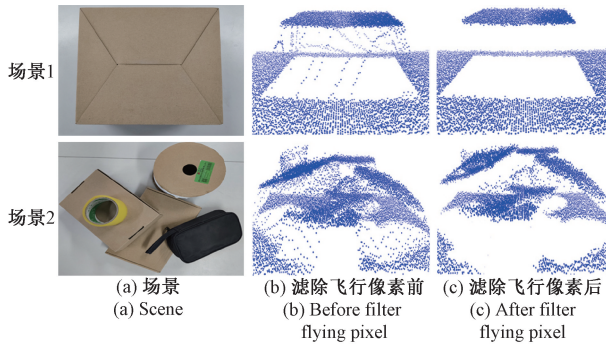


图3 飞行像素删除前后对比图

Fig. 3 Comparison of before and after deletion of flying pixels

## 1.2 梯度聚类分离边缘点

深度相机捕获的有序点云数据由3个  $m \times n$  的矩阵或图像组成,即  $X_{map}$ ,  $Y_{map}$  和  $Z_{map}$ ,其中  $m \times n$  是深度相机的分辨率。 $X_{map}$ ,  $Y_{map}$  和  $Z_{map}$  的元素值分别代表相机在  $x$  轴、 $y$  轴和  $z$  轴方向测量的距离。这里  $z$  轴为主光轴,在  $X_{map}$ ,  $Y_{map}$  和  $Z_{map}$  中相同位置的元素值构成的3D向量  $(x, y, z)$ ,是捕获场景点的3D坐标。

通常方法是对深度图  $Z_{map}$  进行处理<sup>[10-14]</sup>,如图1中的途径1,由于深度图中包含的深度值与其在图中位置间没有直接的几何关系,深度图仅体现了相邻点之间的深度差距,无法表达有序点云数据的稀疏性和3D几何结构,导致途径1方法提取的边缘精度低。为此本文利用3D数据,提出一种新的梯度特征提取方法。

如图4所示,对于有序点云中的邻域点  $P_1(x_1, y_1, z_1)$ 、 $P_2(x_2, y_2, z_2)$  与  $P_3(x_3, y_3, z_3)$ ,相应方向的距离如下。

$$\begin{cases} \Delta x_{12} = |x_1 - x_2|, \Delta x_{23} = |x_2 - x_3| \\ \Delta y_{12} = |y_1 - y_2|, \Delta y_{23} = |y_2 - y_3| \\ \Delta z_{12} = |z_1 - z_2|, \Delta z_{23} = |z_2 - z_3| \end{cases} \quad (2)$$

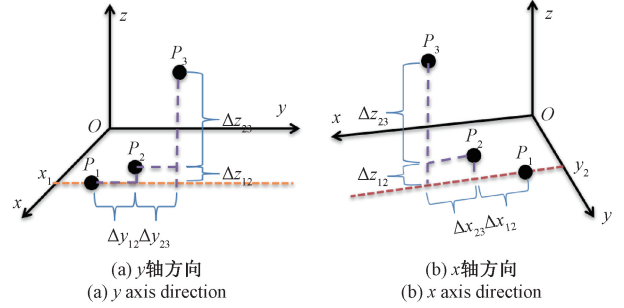


图4  $x$  方向,  $y$  方向平面点和非平面点示意图,其中  $P_1, P_2$  为平面点,  $P_3$  为非平面点

Fig. 4 Diagram of plane and non-planar points in  $x, y$  directions,  $P_1$  and  $P_2$  are plane points, and  $P_3$  is a non-planar point

若  $P_1, P_2$  位于同一平面内,  $P_3$  与  $P_1, P_2$  不在同一平面,如图4所示,则它们的梯度差满足以下关系,

$$\begin{cases} \Delta z_{12} < \varepsilon \\ |\Delta x_{12} - \Delta x_{23}| < |\Delta z_{12} - \Delta z_{23}|, \text{ or} \\ |\Delta y_{12} - \Delta y_{23}| < |\Delta z_{12} - \Delta z_{23}| \end{cases} \quad (3)$$

其中,  $\varepsilon$  为一个小的正数(注意这里仅为了分析平面点与非平面点特性,后续算法中不会使用式(2)和(3),无需给出  $\varepsilon$  的值)。通常边缘点位于非平面点附近,因此边缘检测时需重点关注非平面点。依据上述平面点/非平面点的梯度差异分析,可将3D点云视为空间曲面,定义为  $z=f(x, y)$ ,于是可以利用偏导  $\partial z/\partial x$  和  $\partial z/\partial y$  区分平面点和非平面点。

由于  $f(\cdot)$  的显式表达式未知,首先利用 Sobel<sup>[19]</sup> 计算  $X_{map}$ ,  $Y_{map}$  和  $Z_{map}$  的差分矩阵,然后计算  $\partial z/\partial x$  和  $\partial z/\partial y$  的近似值。卷积核如下式所示:

$$X_{kernel} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, Y_{kernel} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (4)$$

计算  $Z_{map}$  相对于  $x$  和  $y$  方向的差分矩阵:

$$GZ_x = Z_{map} * X_{kernel}, GZ_y = Z_{map} * Y_{kernel} \quad (5)$$

其中,“\*”表示卷积,  $GZ_x$  和  $GZ_y$  是  $m \times n$  矩阵。类似地,计算  $X_{map}$ ,  $Y_{map}$  的差分矩阵。

$$G_x = X_{map} * X_{kernel}, G_y = Y_{map} * Y_{kernel} \quad (6)$$

于是获得  $x, y$  方向的梯度矩阵。

$$I_x = GZ_x ./ G_x, I_y = GZ_y ./ G_y \quad (7)$$

其中,“./”表示矩阵对应位置元素相除,在  $I_x$  和  $I_y$  中元素值  $i_x$  和  $i_y$  分别用来逼近  $\partial z/\partial x$  和  $\partial z/\partial y$ 。最终全局

梯度矩阵  $I_g$ 。

$$I_g = abs(I_x) + abs(I_y) \tag{8}$$

其中,  $I_g$  中的元素值  $i_g$  对应于每个点沿  $x$  和  $y$  方向的梯度绝对值之和,传统方法是通过设置阈值来区分边缘点和非边缘点<sup>[19]</sup>,但阈值难以适用不同场景。本文提出采用基于 K-means<sup>[20]</sup> 的边缘点自动提取方法。

这里仅考虑边缘点和非边缘点的二分类问题。先计算各梯度值 4 次方,使具有较小梯度值的非边缘点和具有较大梯度值的边缘点有更明显的聚类效果。对全局梯度矩阵  $I_g$  中每个元素计算,

$$i_{proc} = i_g^4 \tag{9}$$

式中:  $i_{proc}$  为处理后的矩阵  $I_{proc}$  的元素值(大于 255 的元素值,设置为 255)。对  $I_{proc}$  采用 K-means 聚类,过程如下:

- 1) 从  $I_{proc}$  中初始化  $K$  (本文  $K=2$ ) 个样本点(元素值)作为初始聚类中心  $C$ 。
- 2) 计算剩余各个样本点到每个聚类中心  $C$  的距离,并将距离聚类中心  $C$  较近的点归到此类中。
- 3) 重新计算每个聚类中心  $C$ ,更新聚类中心的位置。
- 4) 重复步骤 2),3),直至  $K$  个聚类中心满足设定条件。

采用图 3 中的场景(2)进行实验,其中第一行是梯度分布散点图,第二行是梯度分布直方图,由图 5 可以看出,通过式(9)处理后,  $I_{proc}$  的聚类效果明显提高。

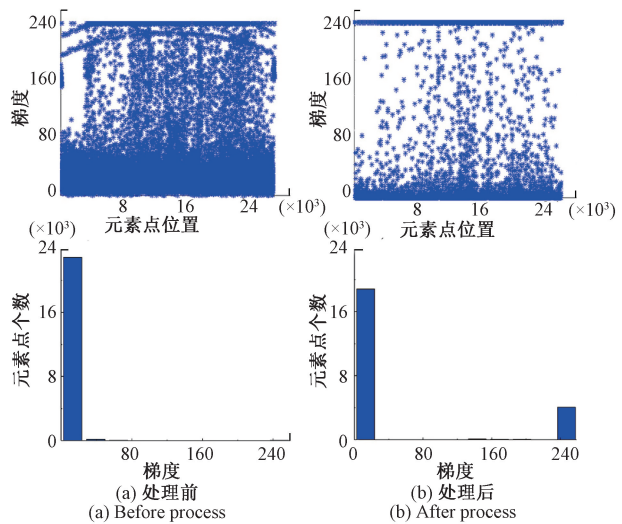


图 5 梯度分布  
Fig. 5 Gradient distribution

将  $[0, 255]$  的元素值,归一化为  $[0, 1]$ ,然后分别计算区间  $[0, 0.2]$ ,  $(0.2, 0.8)$  和  $[0.8, 1]$  中的元素个数占总元素数的比值,结果列在表 1 中。经过上述式(9)处理后,梯度值较小的边缘点从低梯度区间  $[0, 0.2]$  移动到高梯度区间  $[0.8, 1]$ ,即占比由 0.19% 提高到 13.63%,这使得边缘点/非边缘点有效分离,于是可以获得初步的粗边缘图。

表 1 梯度的归一化区间分布

Table 1 Gradient normalized interval distribution (%)

| 分布区间 | 0→0.2 | 0.8→0.8 | 0.8→1.0 |
|------|-------|---------|---------|
| 预处理前 | 99.39 | 0.42    | 0.19    |
| 预处理后 | 85.73 | 0.64    | 13.63   |

1.3 边缘优化与去除孤立点

通过梯度聚类获得了粗边缘图  $R$ ,为了获得精确的细边缘,本文提出采用快速平行边缘细化算法进一步优化  $R$ <sup>[21]</sup>。首先,进行如图 6 所示的模板和“0-1 模式”定义,

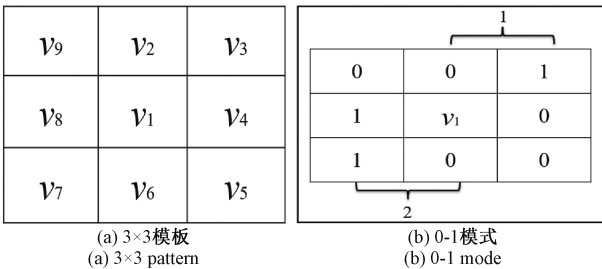


图 6 3×3 模板和 0-1 模式示意图  
Fig. 6 3×3 patterns and 01 mode

图 6(a)为 3×3 模板中像素的排列位置,图 6(b)为针对像素点  $v_1$  的“0-1 模式”示意图,其中 0-1 模式为  $v_2, v_3, \dots, v_9$  的有序集合中相邻点存在“0-1”排列的情况。如果  $R$  的像素点  $v_1$  满足式(10),那么  $v_1$  将会被删除,否则点  $v_1$  将保留。

$$\begin{cases} 2 \leq B(v_1) \leq 6, & A(v_1) = 1 \\ v_2 * v_4 * v_6 = 0, & v_4 * v_6 * v_8 = 0 \end{cases} \tag{10}$$

其中,  $A(v_1)$  表示每个 3×3 模板中 01 模式出现的次数,如图 6(b)所示,  $(v_2, v_3) = (0, 1)$ ,  $(v_6, v_7) = (0, 1)$ ,则  $A(v_1) = 2$ 。  $B(v_1)$  为  $v_1$  的 8 个相邻点像素中非 0 点个数。

$$B(v_1) = v_2 + v_3 + v_4 + \dots + v_8 + v_9 \tag{11}$$

于是粗边缘图  $R$  经过上述处理后,获得细边缘图  $T$ 。图 7 为快速平行边缘细化算法的结果。



图 7 粗细边缘对比图  
Fig. 7 Thick and thin edge comparison

最后,采用掩膜滤波去除细边缘图  $T$  中依旧存在的孤立点噪声。如图 8 所示,如果掩膜中心像素为 1,8 邻



域像素为0,那么中心像素将被设置为0。经这个掩膜过滤后,孤立点噪声有效去除,获得精确边缘。

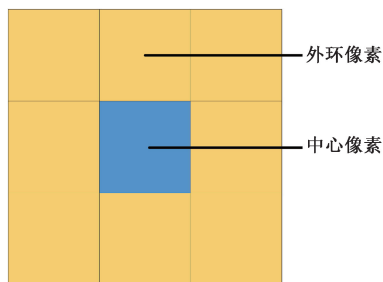


图8 掩膜滤波

Fig. 8 Mask filter

## 2 实验部分

采用 TLESS 数据集以及 TOF 相机实测数据进行实验,验证本文提出算法的有效性,同时与3种已有方法进行性能比较,包括途径1中 PiDiNet 网络<sup>[22]</sup>与 DLBP<sup>[14]</sup>算法,途径2中的 8N-DD<sup>[16]</sup>算法。

### 2.1 TLESS 数据集验证测数据验证

TLESS 数据集是用于无纹理对象检测和 6D 位姿估计的 RGB-D 数据集,包含 30 个行业相关的常见物体。物体没有明显的颜色区别,纹理简单,形状类似。数据集中的深度图分辨率为  $618 \times 634$ 。采集的场景包括单物体、物体遮挡以及堆叠等多种场景<sup>[23]</sup>。

图9中展示了不同方法的边缘检测结果,由图9第3行可知,PiDiNet 网络的边缘提取结果忽略了“箱体”底面凹槽和侧面凸起部分的边缘(场景1和3),主要由于凹槽部分的深度差异较小,在低分辨率深度图中,降低了 PiDiNet 网络的提取效果;DLBP 方法的边缘提取结果(图9第4行)中依旧存在未完全去除的噪声点,并且“箱体”侧面凸起部分和底面的“凹槽”边缘不连续(场景2和3),主要因为 DLBP 方法单一的阈值设定,对深度差异不明显的边缘提取能力差;8N-DD 方法的边缘提取结果(图9第5行)中没有对“箱体”侧面凸起和“箱体”底面凹槽的边缘细节有效提取,主要是由于算法的栅格划分策略,导致距离较近的边缘点会被划分到同一栅格,使得较窄且深度差异不明显的“箱体”凹槽和“箱体”侧面凸起会被深度差异大的“箱体”外部轮廓掩盖。

本文提出的梯度聚类方法利用到场景中的 3D 信息,能较好地提取深度差异不明显的边缘(图9最后一行),如 PiDiNet 网络、8N-DD 未检测到的“箱体”侧面凸起和底面凹槽的边缘;而与 DLBP 相比,通过飞行像素的滤

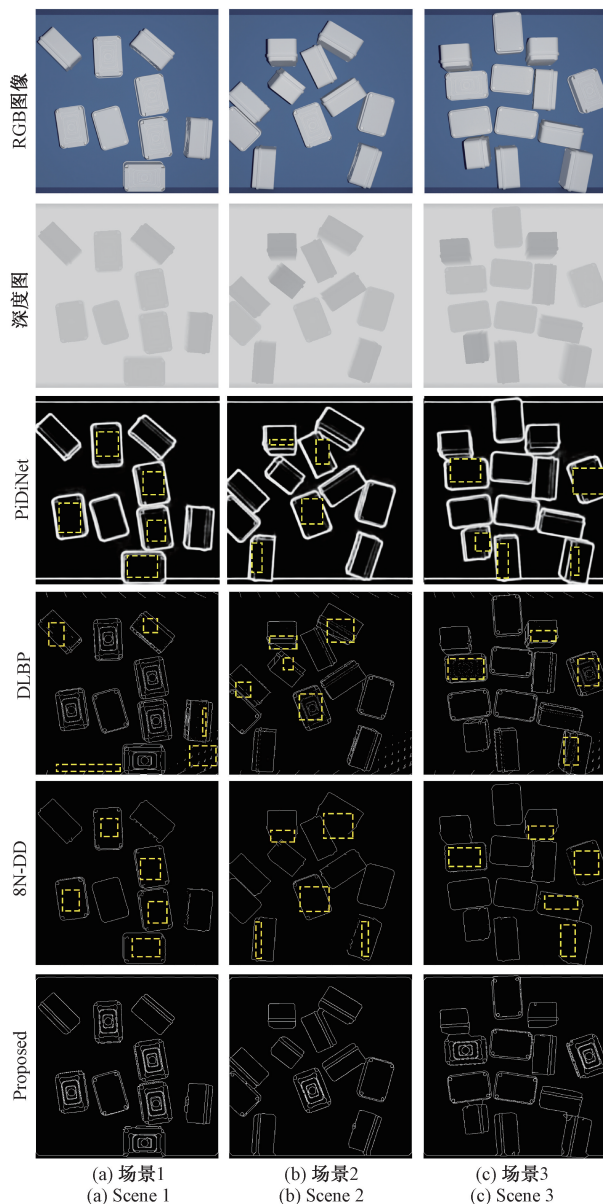


图9 TLESS 数据集的边缘提取结果

Fig. 9 The edge extraction results with TLESS dataset

除,边缘检测效果更加稳定,同时经过梯度自适应聚类方法,检测到的“箱体”侧面凸起与底面“凹槽”边缘连续;经以上结果可以看出,本文方法在 TLESS 数据集的不同场景中均取得较好的边缘提取结果。

### 2.2 TOF 实测数据验证

实验采用 TOF 相机(O3D303, ifm, 德国)用于捕获 3D 数据,TOF 深度相机的参数列在表2中。为了验证提出算法在真实数据上的性能,我们使用不同形状,纹理特征较少的几类物体(“纸箱”、“垫板”、“工具箱”、“线圈”等)布置了3种不同的采集场景,如图10所示。

表 2 O3D303 深度相机参数

Table 2 O3D303 depth camera parameters

| 分辨率/pixel | 采样率/Hz | 测量距离/m | 孔径角/(°) |
|-----------|--------|--------|---------|
| 352×264   | 25     | < 30   | 64×45   |

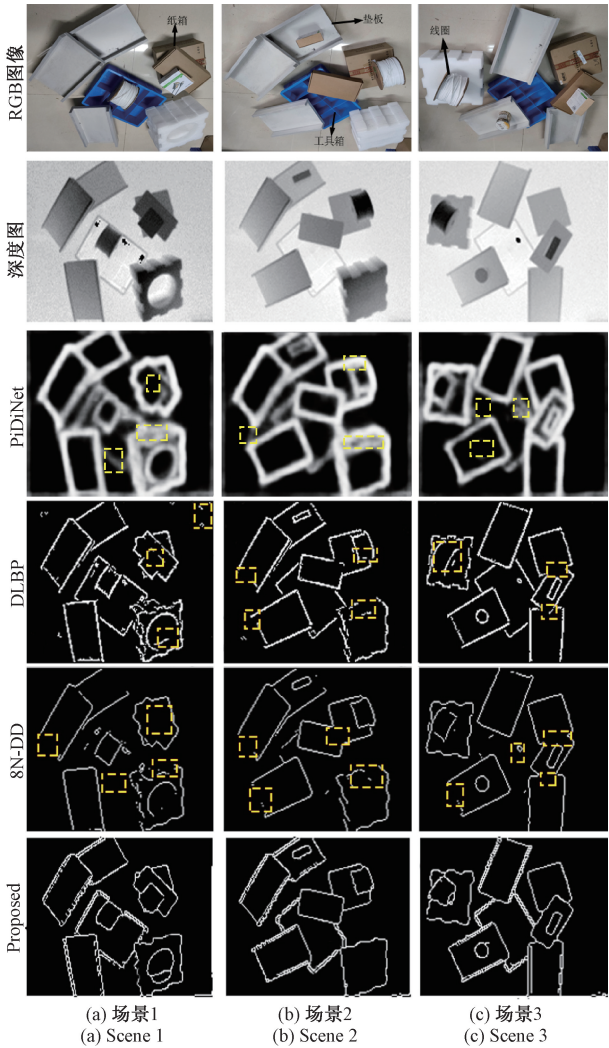


图 10 TOF 点云数据边缘检测结果

Fig. 10 TOF point cloud data edge detection results

由图 10 第 3 行可见, PiDiNet 网络未提取到较窄的“垫板”棱边(场景 2), 物体堆叠遮挡部分的边缘也被忽略(场景 1 和 3), 部分无连接区域被误检出边缘(场景 2), 主要由于 PiDiNet 网络对深度差异不明显且分辨率低的深度图识别能力差; 图 10 第 4 行显示, DLBP 方法提取的结果忽略了“线圈”的侧面边缘(场景 2 和 3), 并且未提取到物体堆叠的边缘(场景 2), 主要是因为单一阈值约束和飞行像素噪声降低了 DLBP 方法的边缘提取能力; 图 10 第 5 行显示, 8N-DD 边缘检测结果中因栅格划分问题, 使得较窄的

“垫板”棱边被忽略(场景 1 和 2), 并且“垫板”与地面之间的交界处, 由于深度值变化不明显, 也未提取到边缘(场景 2 和 3)。

图 10 最后一行显示, 本文提出的梯度聚类方法在 TOF 相机实测数据中能提取到精确的边缘, 与 PiDiNet 网络相比, 未出现边缘误检; 与 8N-DD 算法相比, 在“垫板”棱边能提取到明确边缘; 相对于 DLBP, 对“线圈”的侧面棱边有更好的提取效果, 并且通过滤除飞行像素, 有效抑制飞行像素干扰。进一步, 由图 11 点云边缘对比图可见, PiDiNet 网络和 DLBP 方法的提取的边缘结果都含有大量的飞行像素, 而 8N-DD 方法提取的边缘缺失严重; 本文方法不仅消除飞行像素干扰, 在物体堆叠遮挡的情况下也实现了边缘的稳定检测和准确定位, 结果显示本文方法的可靠性和稳定性。

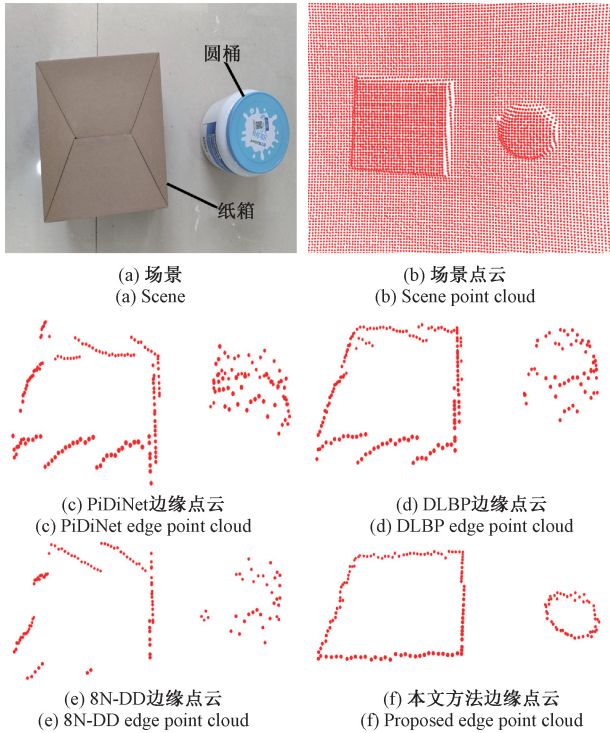


图 11 点云边缘对比图

Fig. 11 Point cloud edge comparison

由图 11 可知, “圆桶”或“纸箱”边缘应处于同一平面, 利用提取的边缘拟合平面, 表 3 对比了几种方法提取边缘的拟合误差, 可见, 本文方法提取边缘的平面拟合误差仅为 2.2 mm, 相比于 PiDiNet 网络, 拟合误差降低 14 倍, 对比 DLBP 和 8N-DD 方法, 降低了约 6 倍。结果显示本文方法可以从 TOF 相机实测的场景数据中稳定地提取精确边缘提取。

表 3 平面拟合误差  
Table 3 Plane fitting error m

| 方法<br>类别 | PiDiNet | DLBP    | 8N-DD   | Proposed       |
|----------|---------|---------|---------|----------------|
| 纸箱       | 0.025 2 | 0.010 9 | 0.109 7 | <b>0.001 8</b> |
| 圆桶       | 0.032 9 | 0.017 2 | 0.027 2 | <b>0.002 2</b> |

2.3 量化评估

用  $F_1$  和  $ACC$  指标对方法进行量化评估,  $F_1$  是准确率和召回率的调和平均值,  $ACC$  为检测精度, 定义如下:

$$F_1 = \frac{2 \times P \times R}{P + R}, ACC = \frac{TP + TN}{TP + TN + FP + FN}$$
 (12)

其中,  $P, R$  如下所示:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN}$$
 (13)

其中,  $P$  为精确率,  $R$  为召回率。  $F_1$  和  $ACC$  的值越大, 表明边缘提取效果越好。 另外, 每秒检测图像帧数  $FPS$  (frame per second) 用于评估算法实时性。 程序在 Intel(R) Core(TM) i5-4210M CPU@ 2.6 GHz 电脑执行。

表 4 给出了量化评估结果。 本文使用的 TOF 相机深度图分辨率低于 TLESS 数据集, PiDiNet 网络在 TLESS 数据集上  $F_1$  为 0.60, 但是在真实的 TOF 数据中仅有 0.52, 说明 PiDiNet 网络用于检测受飞行像素和孤立点干扰的真实场景中物体边缘时, 性能明显降低, 且存在着因分辨率降低导致的检测精度下降问题。 其次, 分辨率变化对 DLBP 方法的影响较小, 但在 TOF 数据集中, 由于其对飞行像素等噪声的滤除能力不足, 导致其召回率偏低,  $F_1$  仅有 0.56, 并且 DLBP 使用了更为复杂的滤波操作, 导致其实时性远低于本文方法。 进一步, 8N-DD 方法会由于栅格划分策略忽略纹理复杂场景中的局部边缘, 因此在纹理特征更少的 TOF 数据中, 其边缘检测精度由 TLESS 数据中的 75% 上升到 79%, 但依然受飞行像素与孤立点等噪声干扰。 另外, 8N-DD 尽管采用栅格划分代替了耗时的单点法向量计算, 但需要进行栅格之间的平均深度比较和栅格内部点的深度比较, 以及较复杂的边缘点选取, 其  $FPS$  仅为 4。 相对于以上方法, 本文方法在边缘检测稳定性及实时性都有明显优势, 由表 4 结果可见场景变化对本文算法的边缘提取精度影响较小, 在 TLESS 数据集和真实数据中的边缘检测的  $F_1$  大于等于 0.62, 同时相比于 8N-DD, 在 TOF 数据集上, 其  $FPS$  提升了 7 倍, 每秒可处理 28 帧的点云数据。 以上结果均体现本文方法的高精度、高可靠性和实时性能。

表 4 TLESS 数据集和 TOF 数据实验结果  
Table 4 TLESS dataset and TOF data experiment results

| 指标<br>算法 | TLESS       |           |           | TOF         |           |           |
|----------|-------------|-----------|-----------|-------------|-----------|-----------|
|          | $F_1$       | ACC/%     | FPS       | $F_1$       | ACC/%     | FPS       |
| PiDiNet  | 0.60        | 85        | 10        | 0.52        | 80        | 25        |
| DLBP     | 0.57        | 84        | 8         | 0.56        | 83        | 18        |
| 8N-DD    | 0.48        | 75        | 0.5       | 0.51        | 79        | 4         |
| 本文方法     | <b>0.62</b> | <b>87</b> | <b>12</b> | <b>0.63</b> | <b>89</b> | <b>28</b> |

3 算法应用及系统分析

3.1 应用系统分析

由于 TOF 深度相机在获取 3D 信息方面具有高帧率、视角广、体积小等优点, 在计算机 CAD 领域有广泛的应用。 基于本文提出的边缘提取算法, 利用 OpenCV<sup>[24]</sup> 和 PCL (point cloud library) 点云库<sup>[25]</sup>, 搭建一个面向自动化装卸应用的物体外观重建和定位系统。 该系统利用 TOF 深度相机捕获的点云数据, 进行实时 3D 模型重建与空间定位。

物体外观重建和定位系统界面如图 12 所示, 系统功能包括连接 TOF 深度相机, 显示提取的边缘点云, 重建 3D 模型, 并且输出物体的尺寸、中心点等信息。 其次可以保存和加载点云文件或者模型文件。 “工具” 栏可以修改点云和模型的颜色。

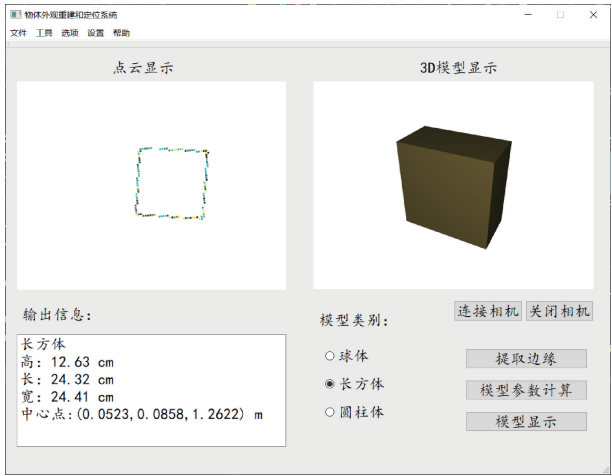


图 12 物体外观重建和定位系统界面  
Fig. 12 Object appearance reconstruction and positioning system

3.2 系统测试

利用物体外观重建和定位系统, 对“篮球”(球体)、“纸箱”(长方体)、“圆桶”(圆柱体)等进行模型重建, 如图 13 所示。将目标物体放于 TOF 深度相机视场中,



利用本文方法从采集的点云数据中检测物体边缘,通过 RANSAC<sup>[26]</sup>拟合边缘所在平面,计算其与地面之间距离;结合拟合直线之间的距离获得模型尺寸和中心点位置,估算物体的尺寸,实现物体的模型重建与定位。

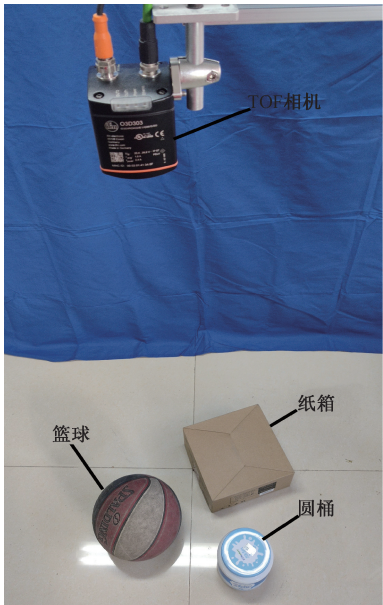


图 13 采集的目标物体  
Fig. 13 The collected target object

表 5 列出系统对目标尺寸的计算结果,并和实际测量值对比,可见此系统能较精确地估算物体尺寸。

表 5 三类模型重建误差结果

| Table 5 Three types of model reconstruction error results |      |       |       |       |       |
|-----------------------------------------------------------|------|-------|-------|-------|-------|
| cm                                                        |      |       |       |       |       |
| 类别\参数                                                     | 长    | 宽     | 高     | 半径    |       |
| 篮球                                                        | 测量   | —     | —     | —     | 11.63 |
|                                                           | 实际   | —     | —     | —     | 11.80 |
|                                                           | 误差/% | —     | —     | —     | 1.44  |
| 纸盒                                                        | 测量   | 24.32 | 24.41 | 12.63 | —     |
|                                                           | 实际   | 24.70 | 24.70 | 12.50 | —     |
|                                                           | 误差/% | 1.53  | 1.17  | 1.04  | —     |
| 圆桶                                                        | 测量   | —     | —     | 30.79 | 13.12 |
|                                                           | 实际   | —     | —     | 30.50 | 13.25 |
|                                                           | 误差/% | —     | —     | 0.95  | 0.98  |

4 结 论

本文提出了一种快速、稳定的 3D 边缘检测方法,以解决在深度相机空间中难以准确定位物体边缘的问题。通过有序点云中的邻域点分析过滤飞行像素噪声;其次,利用梯度聚类快速分离边缘/非边缘点;最后,通过细化边缘及滤除孤立点,获得精确的物体 3D 边缘。在公共数据集和 TOF 相机实测数据上进行了实验验证,结果表明,本文提出方法可以快速地检测到有序点云中的精确边缘,与前期方法相比,在检测精度、召回率及实时性等方面都有显著提升,且本文方法适用场景更广。利用本文提出的边缘检测方法,搭建一个面向自动化装卸应用的物体外观重建和定位系统,实现了物体尺寸计算和位置定位。

参考文献

[ 1 ] ZEINELDIN R A, EL-FISHAWY N A. Fast and accurate ground plane detection for the visually impaired from 3D organized point clouds [ C ]. 2016 SAI Computing Conference (SAI), IEEE, 2016: 373-379.

[ 2 ] 王溪波, 曹士彭, 赵怀慈, 等. 双边特征聚合与注意力机制点云语义分割[ J]. 仪器仪表学报, 2021, 42(12): 175-183.

WANG X B, CAO SH P, ZHAO H C, et al. Semantic segmentation of point cloud via bilateral feature aggregation and attention mechanism [ J ]. Chinese Journal of Scientific Instrument, 2021, 42 ( 12 ): 175-183.

[ 3 ] WU J, HU D, XIANG F, et al. 3D human pose estimation by depth map [ J ]. The Visual Computer, 2020, 36(7): 1401-1410.

[ 4 ] 姚敏, 赵增浩, 赵敏, 等. 复杂结构件内部缺陷检测[ J ]. 仪器仪表学报, 2020, 41(10): 213-220.

YAO M, ZHAO ZH H, ZHAO M, et al. Internal defect detection of complex structural parts[ J ]. Chinese Journal of Scientific Instrument, 2020, 41(10): 213-220.

[ 5 ] HUANG Z, YANG S, ZHOU M C, et al. Making



- accurate object detection at the edge: Review and new approach [J]. *Artificial Intelligence Review*, 2021: 1-30.
- [6] XIE J, FERIS R S, SUN M T. Edge-guided single depth image super resolution[J]. *IEEE Transactions on Image Processing*, 2015, 25(1): 428-438.
- [7] MITTAL M, VERMA A, KAUR I, et al. An efficient edge detection approach to provide better edge connectivity for image analysis[J]. *IEEE Access*, 2019, 7: 33240-33255.
- [8] GONG M, ZHANG Z, ZENG D. A new simplification algorithm for scattered point clouds with feature preservation[J]. *Symmetry*, 2021, 13(3): 399.
- [9] DEY E K, TARSHA K F, AWRANGJEB M, et al. Effective selection of variable point neighbourhood for feature point extraction from aerial building point cloud data[J]. *Remote Sensing*, 2021, 13(8): 1520.
- [10] WANG G, XIANG J. Railway sleeper crack recognition based on edge detection and CNN[J]. *Smart Structures and Systems*, 2021, 28(6): 779-789.
- [11] JING W, JIN T, XIANG D. SAR image edge detection with recurrent guidance filter[J]. *IEEE Geoscience and Remote Sensing Letters*, 2020, 18(6): 1064-1068.
- [12] LEJEUNE A, PIÉRARD S, VAN D M, et al. A new jump edge detection method for 3D cameras[C]. 2011 International Conference on 3D Imaging (IC3D), IEEE, 2011: 1-7.
- [13] AHMADIAN K, REZA-ALIKHANI H. Single image super-resolution with self-organization neural networks and image laplace gradient operator [J]. *Multimedia Tools and Applications*, 2022: 1-24.
- [14] WANG X, CAO J, HAO Q, et al. LBP-Based edge detection method for depth images with low resolutions[J]. *IEEE Photonics Journal*, 2018, 11(1): 1-11.
- [15] 伍阳, 王丽妍, 胡春霞, 等. 可变半径 Alpha Shapes 提取机载 LiDAR 点云建筑物轮廓[J]. *中国图象图形学报*, 2021, 26(4): 910-923.
- WU Y, WANG L Y, HU CH X, et al. Extraction of building contours from airborne LiDAR point cloud using variable radius Alpha Shapes method [J]. *Journal of Image and Graphics*, 2021, 26(4): 910-923.
- [16] 张志佳, 魏信, 周自强, 等. 一种基于八邻域深度差的点云边缘提取算法[J]. *仪器仪表学报*, 2017, 38(8): 2015-2021.
- ZHANG ZH J, WEI X, ZHOU Z Q, et al. 8N-DD-based point cloud edge extraction algorithm [J]. *Chinese Journal of Scientific Instrument*, 2017, 38 ( 8 ): 2015-2021.
- [17] NI H, LIN X, NING X, et al. Edge detection and feature line tracing in 3D-point clouds by analyzing geometric properties of neighborhoods [J]. *Remote Sensing*, 2016, 8(9): 710.
- [18] SABOV A, KRÜGER J. Identification and correction of flying pixels in range camera data[C]. *Proceedings of the 24th Spring Conference on Computer Graphics*, 2008: 135-142.
- [19] ZHANG R, XIAO Q, DU Y, et al. DSPI filtering evaluation method based on sobel operator and image entropy[J]. *IEEE Photonics Journal*, 2021, 13 ( 6 ): 1-10.
- [20] SINAGA K P, YANG M S. Unsupervised K-means clustering algorithm [J]. *IEEE Access*, 2020, 8: 80716-80727.
- [21] ZHANG T Y, SUEN C Y. A fast parallel algorithm for thinning digital patterns [J]. *Communications of the ACM*, 1984, 27(3): 236-239.
- [22] SU Z, LIU W, YU Z, et al. Pixel difference networks for efficient edge detection[C]. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021: 5117-5127.
- [23] HODAN T, HALUZA P, OBDRŽÁLEK Š, et al. T-LESS: An RGB-D dataset for 6D pose estimation of texture-less objects[C]. 2017 IEEE Winter Conference

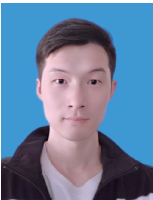
on Applications of Computer Vision (WACV), IEEE, 2017: 880-888.

[24] CULJAK I, ABRAM D, PRIBANIC T, et al. A brief introduction to OpenCV [C]. 2012 Proceedings of the 35th International Convention MIPRO, IEEE, 2012: 1725-1730.

[25] RUSU R B, COUSINS S. 3D is here: Point cloud library (pcl) [C]. 2011 IEEE International Conference on Robotics and Automation, IEEE, 2011: 1-4.

[26] BRACHMANN E, ROTHER C. Neural-guided RANSAC: Learning where to sample model hypotheses[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019: 4322-4331.

作者简介



陈浩,2019 年于安徽工业大学获得学士学位,现为东北大学硕士研究生,主要研究方向为机器视觉。  
E-mail:2000713@ stu. neu. edu. c

**Chen Hao** received his B.Sc. degree in 2019 from Anhui University of Technology, currently is a master student at Northeastern University. His main research interest is machine vision.



丁其川(通信作者)2007 年于山东大学获得学士学位,2015 年于中国科学院沈阳自动化所获得博士学位,现为东北大学机器人科学与工程学院副教授,主要研究方向为机器视觉,智能机器人,人体姿态识别。  
E-mail: dingqichuan@mail. neu. edu. cn

**Ding Qichuan** (Corresponding author), received his B. Sc. degree from Shandong University in 2007, received his Ph. D. degree from Shenyang Institute of Automation, Chinese Academy of Sciences in 2015. Now he is an associate professor in the Faculty of Robot Science and Engineering at Northeastern University. His main research interests include machine vision, robot smart technology, and human gesture identification.



潘磊,2020 年于安徽工业大学获得学士学位,现为东北大学硕士研究生,主要研究方向为计算机视觉,人体姿态识别。  
E-mail:2002012@ stu. neu. edu. cn

**Pan Lei** received his B.Sc. degree in 2020 from Anhui University of Technology, now he is a master student at Northeastern University. His main research interests include computer vision, and human gesture recognition.