

智能车辆 3-D 点云快速分割方法

王 肖^{1,2}, 王建强¹, 李克强¹, 徐 成¹, 李晓飞¹

(1. 清华大学 汽车安全与节能国家重点实验室, 北京 100084; 2. 军事交通学院 汽车工程系, 天津 300161)

摘 要: 针对于智能车辆环境感知实时性要求, 研究一种基于 3-D 全景式激光雷达的点云快速分割方法。采用带地面状态变化判断的分块直线拟合算法来去除背景地面数据, 生成前景点云图像并利用机器视觉算法进行聚类分割。建立并利用坐标映射模型计算出定向包围体参数。在 3 组基于城市道路环境的测试集下, 该方法同栅格法相比能够较好地降低过分割与欠分割错误率, 其中车辆分割准确率约为 95%, 行人分割准确率约为 85%, 且耗时不受场景复杂度影响, 能够较稳定控制在 55 ms/帧左右。

关键词: 智能车辆; 3-D LIDAR; 点云分割; 映射模型

中图分类号: TP 391.4

文献标志码: A

文章编号: 1000-0054(2014)11-1440-07

Fast segmentation of 3-D point clouds for intelligent vehicles

WANG Xiao^{1,2}, WANG Jianqiang¹, LI Keqiang¹,
XU Cheng¹, LI Xiaofei¹

(1. State Key Laboratory of Automotive Safety and Energy,
Tsinghua University, Beijing 100084, China;

2. Automobile Engineering Department, Military Transportation
University, Tianjin 300161, China)

Abstract: This paper describes a fast segmentation method for 3-D point clouds targeted at real-time performance of intelligent vehicles. Efficiency is achieved in this approach by splitting the segmentation problem into two subproblems to remove the ground data and to segment the foreground data using machine vision algorithms. A coordinate mapping model is then used to calculate the oriented bounding-box representing the target's shape, position and direction. The method's performance is evaluated on real data acquired in complex traffic environments. Test results show that the accuracy of the vehicle segmentation is 95% and that of pedestrians is 85%, with an average of 55 ms needed for each frame with the runtime unrelated to the traffic environment.

Key words: intelligent vehicle; 3-D LIDAR; point cloud segmentation; coordinate mapping model

全景式 3-D 激光雷达(LIDAR)能够捕获目标物基本形状特征和丰富局部细节, 具有可靠性及精

度高等优点, 因此被广泛应用在智能车环境感知中。点云是由 LIDAR 生成的在同一空间参考系下表达目标物空间分布和表面特性的海量点的集合。它的分割是根据整体密度分布和局部聚集特性, 将点云聚类分离成独立的子集, 各子集均对应于具有物理意义的感知目标, 并反映目标物几何及位姿特征。这种分割是保证后续目标分类及跟踪性能的重要基础。但由于点云获取量为每 s 约上百万个点, 普通聚类算法根本无法满足分割实时性, 因此有必要研究一种快速准确的分割方法。

目前, 国内外点云分割研究通常基于非模型投影方法, 根据投影方向又分为基于地面投影及基于虚拟像平面投影两类。

基于地面投影是将点云投影至俯视平面, 再利用栅格地图法进行处理^[1-3]。基于栅格的最大最小值高度图方法, 通过计算投影至每个栅格点云的最大最小高度值及其差值, 再与经验阈值比较来判断栅格为障碍或地面, 最终通过障碍栅格聚类来检测目标物。由于该方法实现简单高效, 同时在栅格基础上有利于实现多传感器融合, 因此在美国 DARPA 城市挑战赛及中国智能车未来挑战赛中得到广泛应用^[1]。但栅格法存在以下问题: 1) 栅格单元参数无法自适应调节, 容易造成欠分割与过分割; 2) 数据存储量巨大, 不利于工程应用。

基于虚拟像平面投影是将点云投影至以 LIDAR 旋转轴为轴的虚拟圆柱体上, 其像素值为对应于相应点的 Euclidean 距离^[4-6]。在深度图像分割研究中, Hoover^[4]对 4 种主要深度图像分割算法(USF, UE, WSU, UB)进行对比实验, 但这些算法

收稿日期: 2013-12-09

基金项目: 国家自然科学基金资助项目(51175290);

戴姆勒-清华联合项目

作者简介: 王肖(1984—), 男(汉), 四川, 博士研究生。

通信作者: 李克强, 教授, E-mail: likq@tsinghua.edu.cn

主要针对计算机图形学,适用于噪声少、几何形状简单的理想点云数据,难以应用在智能车行驶的复杂环境中。

Moosmann 等^[5]提出深度图像与表面几何特征融合的分割方法。首先,建立深度图像并利用插值方法进行增强,将点云转化为结构化图像;再利用 4 邻域的法向量快速求解算法并构造局部凸性特征,最终利用标记算法进行分割。但由于局部凸性过于依赖点云法向量的精度,同时单帧处理耗时需 600 ms,限制了该方法在复杂场景中的应用。

Golovinskiy 等^[7]将图像分割领域中的图割方法引入至 3-D 点云分割中,该算法输入为单帧点云数据及目标物中心点坐标与半径估计值,输出为属于该目标的点集,采用 kd-tree 算法实现近邻点搜索并构建 s-t 网络。其中,能量函数自变量采用各点至中心点距离,数据项采用线性函数设计。该分割方法理论性较强,分割精度较高,但只能针对已知位置及半径的目标,同时计算量较大。

为满足点云分割实时性与准确率要求,本文首先对现有点云前、背景分离方法进行研究改进,然后提出一种前景点云转化至图像域中的分割标记方法及 LIDAR-图像坐标映射模型,并在城市道路环境下验证算法的有效性。

1 算法概述

图 1 为算法基本流程,主要包括数据获取、预处理、背景点云去除、前景点云分割及输出 5 个部分。

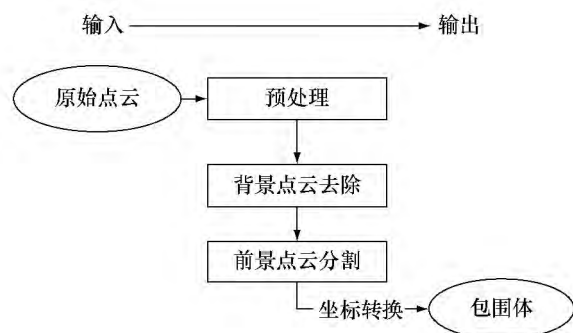


图 1 点云分割算法流程图

1.1 数据获取

算法输入为单帧原始点云,每帧数据约包含 13 万个点。设第 t 帧点云为 $P_t = \{p_1, \dots, p_N\}$, 其中: $p_i = (x_i, y_i, z_i)$, x_i, y_i, z_i 为 LIDAR 坐标系下 3-D 坐标,单位为 m。预处理包括感兴趣区域及坐标映射模型的建立。由于主要针对于城市道路条件,因此感兴趣区域设置为自车行驶的当前道路,不包括边

界线以外部分。坐标映射模型为 LIDAR 与图像坐标系的联系纽带,用于坐标系间参数的转换计算。预处理后点云包括前景和背景两个部分,背景点云主要指地面数据组成,由于它不仅占用大量存储及计算量,并且影响分割精度,因此需首先进行前背景分离。前景点云主要由车辆、行人、自行车等地面运动目标物组成,前景分割即为将各个目标进行聚类标记。坐标转换采用映射模型将图像坐标系下分割目标的位置参数转换至 LIDAR 坐标系下,得出定向包围体。定向包围体为 LIDAR 坐标系下表征目标物形状、位置和方向等的参数向量 B_i 。其中, i 为从检测开始时刻自增的实数编号,每个定向包围体编号唯一。

1.2 数据预处理

1.2.1 感兴趣区域的建立

全景式 3-D LIDAR 最远距离可达上百 m,但在实际应用中会由于目标距离过远而使得点云过于稀疏,因此需要建立感兴趣区域来缩小处理范围。

在智能车纵向行驶方向上,假设目标物为具备理想反射表面的立方体盒子(设反射面宽高比为 $a:b$)。设 LIDAR 水平方向上相邻激光束夹角为 α ,纵向上夹角为 β ,则距离为 L 的目标物理理想反射点数量可由式(1)粗略估计,

$$\text{Num}_{\text{ideal}} = \frac{ab}{4L^2 \sin(\alpha/2) \sin(\beta/2)}. \quad (1)$$

由式(1)可知,反射点数同 L^2 成反比,这说明了目标物点云数量同其与 LIDAR 的距离成反比。已知 $\alpha=0.09$, $\beta=0.4$, 假设前方目标盒子反射面形状参数 $a=1.9$, $b=1.5$, 则距离为 50 m 的目标物理理想反射总数约 100 个。但由于目标颜色、凹凸表面及玻璃等多种因素影响,实际点云数量约 10~30 个。因此,为保证点云分割质量,同时结合智能车行驶所需安全视距,设置车辆纵向行驶方向上 LIDAR 感兴趣区域最大距离为前后各 50 m。

在横向行驶方向,以视觉识别的车道线为矩形边界。设 d_{left} , d_{right} 表示智能车中心至左右车道线距离,则 LIDAR 坐标系下矩形区域 $100 \text{ m} \cdot (d_{\text{left}} + d_{\text{right}})$ 为点云分割感兴趣区域,但由于 d_{left} 及 d_{right} 为动态变化值,与 1.2.2 节中坐标映射模型相矛盾,因此在当前预处理阶段则以 LIDAR 为中心的 $100 \text{ m} \times 60 \text{ m}$ 矩形区域为点云分割感兴趣区域,而在前景点云分割阶段则采用 $100 \text{ m} \cdot (d_{\text{left}} + d_{\text{right}})$ 矩形为感兴趣区域。

1.2.2 坐标系及映射模型

本研究涉及 3 个坐标系: LIDAR 坐标系, 车辆坐标系及图像坐标系。由于车辆坐标系可以通过 LIDAR 坐标系经过简单几何关系进行转化, 因此点云数据统一用 LIDAR 坐标系表示。映射模型为目标物参数值在图像坐标系同 LIDAR 坐标系之间转换的准则。

如图 2 中 $Oxyz$, LIDAR 坐标系是以 LIDAR 中心为原点 O , x 轴指向检测车辆的前进方向, z 轴平行于 LIDAR 纵向切面并指向上方, y 轴垂直于 x 轴和 z 轴所在的平面指向驾驶员左侧。

图像坐标系 $O'uv$ 建立在 2-D 俯视点云图像上, 以图像左上角顶点为原点 O' , u 轴沿原点 O' 水平向右, v 轴为沿原点 O' 垂直向下。点云图范围为 LIDAR 坐标系下矩形感兴趣区域, 像素值利用二值化进行处理, 如图 2 中灰度值为 255 的像素表示有效反射点, 灰度值为 0 表示无效反射点。

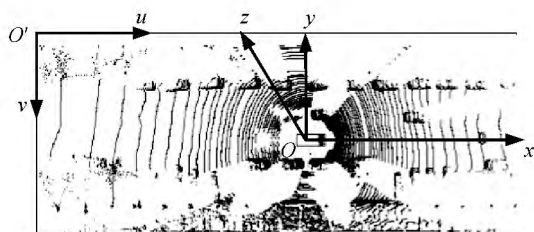


图 2 LIDAR 坐标系与图像坐标系示意图

设点云图分辨率为 $m \times n$, 对应的 LIDAR 坐标系区域为 $d \cdot s$, 则 LIDAR 坐标系及图像坐标系的映射模型为

$$\begin{cases} x = \left(u - \frac{m}{2}\right) \cdot \frac{d}{m}, \\ y = \left(\frac{n}{2} - v\right) \cdot \frac{s}{n}. \end{cases} \quad (2)$$

式(2)中模型精度可通过点云图分辨率参数 m , n 进行调节, 但由于过高像素数将在后期处理中占用较大计算资源, 同时考虑到智能车辆实际行驶中无需追求过高的识别精度, 可取 $m = 2\,000$, $n = 1\,200$, 即精度可控制在 5 cm 左右。

1.3 背景点云去除

背景地面数据去除是减少数据存储、计算量及提高分割精度的有效手段, 主要方法有高程阈值法、地面分块直线拟合、Gauss 过程回归模型等。其中: 高程阈值法以前期标定的地面高程值作为阈值进行地面数据去除, 该方法简单高效, 但是极大地依赖于 LIDAR 外标定, 同时无法处理坡度及起伏路面;

Gauss 过程回归模型可以表示很复杂的地面模型, 如斜坡、起伏的越野路面、平坦路面等, 但是其计算量比较大, 不能够满足实时性的要求; 而地面分块直线拟合是一种简单快速的算法^[8], 充分利用了地面在每个分块中应该是平滑连续的特性, 对 LIDAR 的外标定精度要求不高, 且复杂度与点云个数无关, 因此本研究选用地面分块直线拟合法。

本研究提出假设为: 在极短行驶时间间隔内 (如 0.1 s), 前后两帧路面应具有较大相似性, 如相似性达到一定程度, 则可以直接利用前帧地面参数作为当前帧参数, 无需对每帧地面参数进行估计。

根据以上思路, 本文提出如图 3 所示的基于地面状态变化判断的分块直线拟合算法。图中: K 为地面参数向量函数; t 为帧数, 当 $t=1$ 时, 系统初始进行分块直线拟合算法, 并保存参数, 当 $t>1$ 时, 系统则首先利用相似性准则判断地面状态变化, 而后根据结果选择下一步流程。

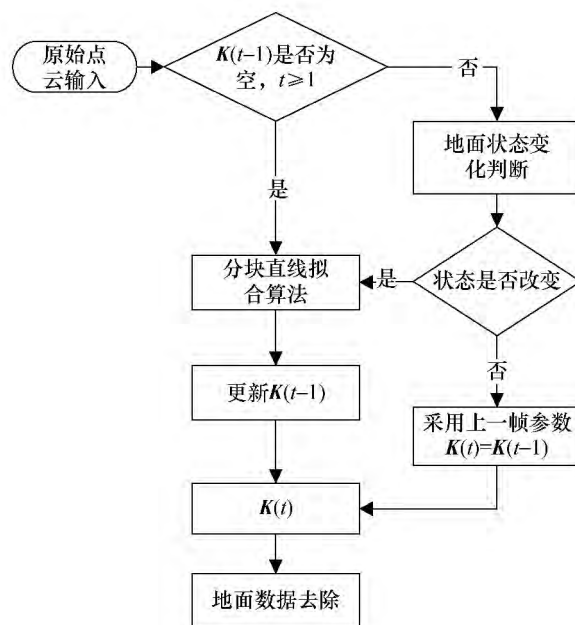


图 3 背景点云数据去除流程图

在地面状态判断中, 以第 t 及 $t-1$ 帧点云的 z 方向的局部高程值分布差异作为相似性判断主要依据, 视局部高程值为独立样本且服从正态分布, 采用 z 检验判断两样本总体差异是否显著, 步骤如下:

1) LIDAR 坐标系下选取 3 个点 $p_{1,2,3}(x_{1,2,3}, y=0)$ 为中心, 以边长 1 m 构建正方形检测区 A_1, A_2, A_3 。统计第 t 及 $t-1$ 帧落入各区域的点云数量 n_i 及其高程值。

2) 建立假设检验, 确定检测水平:

H_0 : 第 t 及 $t-1$ 帧区域 A_i 内点云总体差异不

显著;

显著性水平: $\alpha=0.05$ 。

3) 各区域内点云高程值服从正态分布 $N(\mu_i, \sigma_i^2)$, 其中: i 为检测区域编号; μ_i, σ_i 为高程值的均值及标准差。由 z 检验计算式(3)可计算 t 及 $t-1$ 帧区域 A_1, A_2, A_3 的 z_i 值,

$$z_i = \frac{\mu_i(t) - \mu_i(t-1)}{\frac{\sigma_i(t)}{n_i(t)} + \frac{\sigma_i(t-1)}{n_i(t-1)}}. \quad (3)$$

4) 如 $z_{i=1,2,3} > 2.58$, 则按 $\alpha=0.05$ 水准拒绝 H_0 , 即 3 个检测区域只要存在一个区域前后帧状态不一致, 则认为整体地面状态改变。

由于感兴趣区域为矩形, 因此将扇型分块直线拟合算法^[8]改进为基于矩形子块的地面直线拟合算法。首先, 将感兴趣区域划分为 $M \times N$ 个矩形子块并将各子块点云投影至 x, y 平面, 在其中 $2T$ 个最小点中随机选取 T 个点作为直线拟合样本集, $2T$ 值随着各子块中心点距离 l_c 增大而减小。各子块拟合直线模型由式(4)表示,

$$A_j z + B_j x + C_j = 0. \quad (4)$$

式(4)中: x, z 为 Oxz 坐标系下变量值。直线拟合算法通常采用增量算法^[8-9]或最小二乘法^[10-12], 但由于实际数据包括很多噪声点, 因此采用随机抽样一致性算法(random sample consensus, RANSAC)^[13-14]增强鲁棒性, 其参数选择为: 迭代次数 $\text{Num}_{\text{ite}}=50$, 模型所需局内点数量阈值 $\text{Num}_{\text{in}}=T/2$ 。

通过 RANSAC 算法可以选择出较优局内点集, 再通过最小二乘法即可拟合出最优直线。各分块可利用直线模型对其区域内点云进行判断, 即当 $z_{p_i} < \frac{B_j \cdot x_{p_i}}{A_j} + C_j$ 时, 则认为该点为地面数据并去除。

1.4 前景点云分割

1.4.1 前景点云图像分割

针对前景点云图像, 采用种子填充法进行连通区域分析。文[1]采用对点云图像 8 邻域膨胀处理, 虽然在部分情况下能够获得理想结果, 但是如果两个目标物十分靠近, 则容易导致聚合在一起, 而对于距离较远目标则像素点较稀疏, 则容易将同一物体分割成两个甚至多个子块。

针对这种情况, 本文提出一种先腐蚀后膨胀的策略进行离散像素聚合。腐蚀的作用一方面是将过于离散且非主体组成部分的像素点去除, 另一方面是去除空气中杂质带来的噪声像素。相对于文[1]

中固定结构元素的膨胀方法, 采用动态矩形结构元素的方法, 即在不同的像素范围采用不同的结构元素。采用矩形结构元素的原因是考虑到车辆在横向上像素点分布较稠密, 纵向上较稀疏。

图 4 为采用动态矩形结构元素膨胀策略处理后的结果。由于先采用了腐蚀处理, 部分过于离散的像素和噪声像素均被去除。利用动态矩形结构元素的膨胀能够较好地将远距离离散像素聚合在一起, 同时近距离也不会出现多个目标物聚合在一起的现象, 如图 4 中实线椭圆标注所示。在此基础上可采用种子填充法进行连通区域聚类分割, 分割目标以最小外接多边形 R_i 表示, 并计算图像坐标系下 R_i 的 4 个角点像素坐标值。

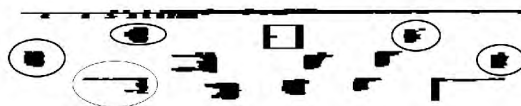


图 4 动态矩形结构元素膨胀处理结果

1.4.2 定向包围体参数计算

定向包围体既是分割算法输出, 也是下一步信息融合或决策部分的输入^[15-17]。它是建立在 LIDAR 坐标系下的参数向量 $B_i = [l, w, f_l, f_w, \theta, x_s, y_s]$, 其值可利用 R_i 参数及式(2)进行计算。其中: l, w, θ 分别为定向包围体长、宽及相对于 x 轴的夹角; f_l, f_w 为长、宽标志位, 当获取准确值后置为 true, 否则为 false; x_s, y_s 为定向包围体参考点坐标, 通常选用定向包围体中心^[1], 但由于在实际环境中目标物受距离或遮挡等因素影响导致观测点云数据不完整, 使得定向包围体大小时刻变化, 因此无法获取其精确中心点坐标。

在目标物前后帧数据关联基础上, 本文提出动态参考点算法, 其基本思想如图 5 所示。在 LIDAR 坐标系下将 $100 \text{ m} \times 60 \text{ m}$ 区域分为 $A_1, A_{12}, A_2, A_{23}, A_3, A_{34}, A_4, A_{14}$ 8 个矩形子区域, 其中 $B_{i=1,2,3,4}$ 分别为 $A_{i=1,2,3,4}$ 区域内定向包围体。由图 5 可知, 在 $A_{i=1,2,3,4}$ 区域定向包围体由于自遮挡导致部分尾部和侧部数据丢失, 称此区域为自遮挡区。在区域 A_{12}, A_{34} 则能够较准确获取车辆长度参数 l , 如 B' 所示, 区域 A_{14}, A_{23} 则能够获取车辆准确宽度参数 w , 如 B'' 所示, 因此称为半自遮挡区。因此, 目标车辆动态参考点选择遵循以下准则:

1) 初始化定向包围体参数。当目标车辆第 t 帧在 $A_{i=1,2,3,4}$ 区时, 以定向包围体 B_i 最近邻角点作为参考点 $P_i(t)$, 计算 $P_i(t)$ 坐标及长宽参数 l, w ,

并置其标志位为 $f_l = f_w = \text{false}$ 。

2) 自遮挡区域内参数更新。当 B_i 当前帧仍处于自遮挡区时,则 $P_i(t)$ 仍为上一帧角点,只需更新其位置参数即可,其余参数无需更新。

3) 半自遮挡区域内 l, w 值的确定。当 B_i 进入半自遮挡区域时,获取其 l 或 w 参数,且置相应标志位为 true,同时根据当前 $P_i(t)$ 坐标值及 l 或 w 值计

算可出 B_i 纵向或横向另一个角点 $P'_i(t)$ 准确坐标值。

4) 参考点更换。当 B_i 进入新的自遮挡区时,则以 $P'_i(t)$ 为新的参考点。例如当定向包围体 B_1 从 A_1 至 A_2 时,参考点从 B_1 右前角点 $P_i(t)$ 更换为右后角点 $P'_i(t)$,当 B_1 从 A_1 至 A_4 时,参考点从 B_1 右前角点 $P_i(t)$ 更换为左前角点 $P'_i(t)$ 。

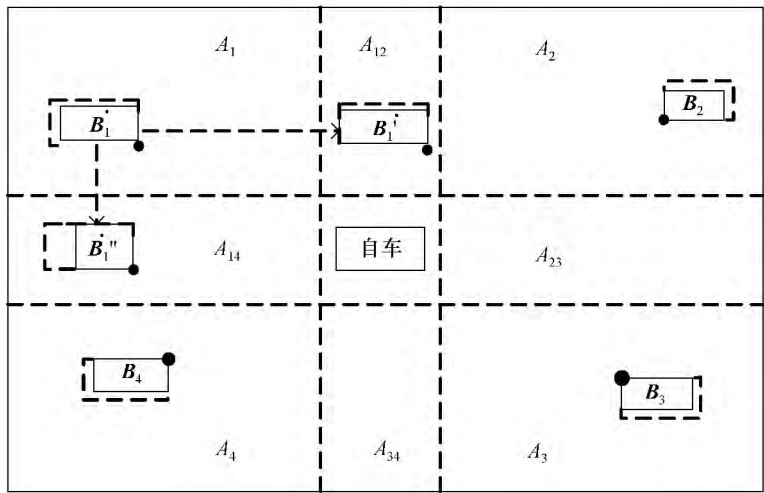


图5 动态参考点的选择示意图

2 实验分析

本实验主要测试点云分割精度及实时性。在事先采集的北京城区多种道路数据中选择其中3组连续200帧数据作为测试样本集,其中包括普通道路及十字路口两种场景。计算所用笔记本电脑配置为:INTEL 双核 2.4 GHz CPU, 4 GB 内存。编程环境为 C++ 及 OpenCV, 并利用 OpenGL 进行 3-D 显示。

2.1 前景点云分割

首先,对每帧数据中感兴趣目标的参考位置及总数进行人工标记,并以此作为正确率计算的参考。实验同时采用栅格法进行对比,其单元栅格参数取文[2]中的 $0.15\text{ m} \times 0.15\text{ m}$ 。表1为两种方法对车辆与行人两种典型目标的正确率统计。

表1 本文方法与栅格法结果对比

方法	目标	正确率	过分割率	欠分割率	丢失率
		%	%	%	%
本文方法	车辆	95	2	3	0
	行人	85	0	15	0
栅格法	车辆	89	6	3	2
	行人	76	0	16	8

从表1中可以看出,对于车辆目标,本文方法优于栅格法,能够较好地降低过分割与欠分割错误率。如图6a所示,虚线椭圆标注处车辆及行人距离十分接近,容易造成欠分割;而实线椭圆标注的车辆则由于距离较远,尾部点云数据稀疏,容易导致过分割。

图6b为本算法分割结果,目标物均以定向包围体进行表示,参考点以黑色角点表示。本算法较好地解决了虚、实椭圆内目标物过分割和欠分割问题。而栅格法由于单元栅格过大,导致本来稀疏目标更加稀疏,从而造成较严重的过分割或目标丢失。

针对行人,虽然本算法优于栅格法,但正确率仍有待提高。如图6b中虚线矩形标注的两个定向包围体所示,其内应各为两个行人并肩而行,但本算法却将其作为一个目标物,这是因为本算法以单一考虑空间距离尺度作为聚类标准,难以克服多目标间距离过近的情况。

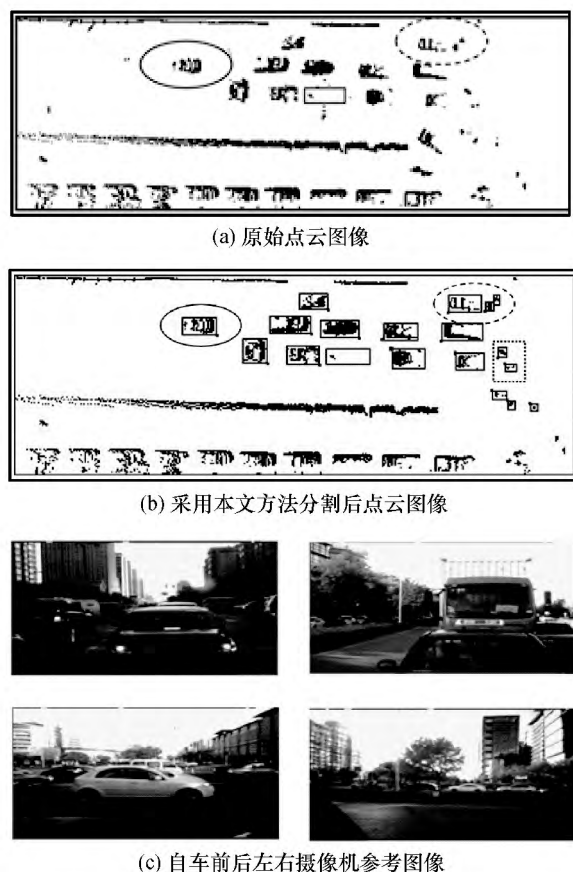


图6 原始、分割处理后点云及四周摄像机图像

2.2 算法实时性

算法耗时测试将同极坐标栅格法及径向边界最近邻聚类法 (radially bounded nearest neighbor, RBNN) 及栅格法进行比较。图 7 所示为样本集中的 1 组数据的 3 种算法耗时对比曲线, 本文算法平均耗时约 55 ms/帧, 栅格法为 80 ms/帧, 而 RBNN 算法则需 300 ms/帧左右。通过记录本算法运行过程中地面状态变化得知, “状态未改变” 输出比例高达 85%, 这意味着只有总帧数的 15% 需要运行地面分块直线拟合算法, 其余则直接利用上一帧参数即可, 因此大大降低了平均耗时。

在耗时稳定性方面, 本算法非常稳定, 基本控制在 50~65 ms/帧范围, 而出现 75 ms/帧左右耗时是因为地面状态发生改变, 之后就迅速下降为正常耗时。栅格法也比较稳定, 但 RBNN 聚类算法最大最小耗时相差约 200 ms/帧, 这是因为其算法受环境复杂度影响较大。

3 组样本总的耗时结果为: 本文方法平均耗时约 55 ms/帧, 栅格法约 78 ms/帧, 而 RBNN 法约 310 ms/帧。可见, 本算法受环境复杂度影响较小, 能够较精确控制、预测运行时长。

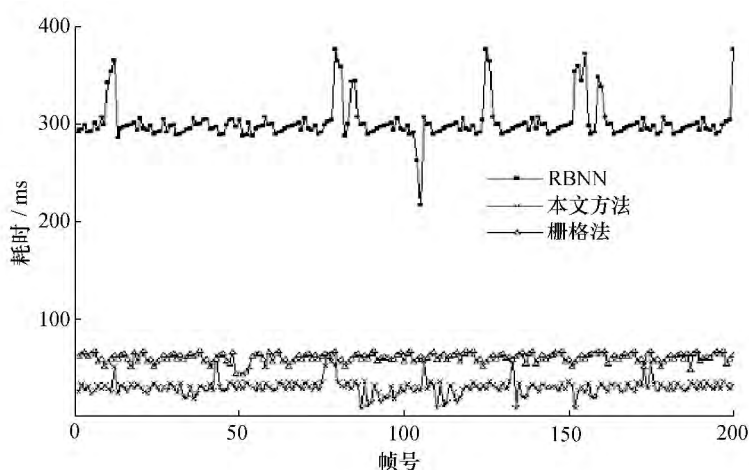


图7 3种算法的耗时曲线对比图

3 结 论

本文提出一种点云数据快速分割方法, 通过地面分块直线拟合去除地面数据, 建立前景点云图像并利用机器视觉算法对点云图像进行分割, 再根据坐标映射模型计算定向包围体参数。在实时性方面, 由于在地面数据去除中利用 χ^2 检验判断前后帧地面状态变化, 优化了算法结构, 可以大大降低平均

耗时, 同时由于分割是直接对图像进行处理的, 因此可以利用当前成熟的视觉算法, 本算法实时性进一步得到保障; 在分割方面, 本算法可灵活、简单地调整图像分辨率像素来达到不同范围的精度, 同时由于采用腐蚀与动态矩形结构元素膨胀结合的策略, 较好地克服了过分割及欠分割问题。在 3 组基于北京道路环境的测试集下, 同栅格法相比, 本文算法能

够较好地降低过分割与欠分割错误率,其中车辆分割准确率约为95%,行人分割准确率约为85%,且耗时不受场景复杂度影响,能够较稳定控制在55 ms/帧左右。

在本文实验中发现,本文算法尚不能较好地处理距离过近的目标物,如行人互相贴近、行人贴近车辆等。下一步将研究采用整体形状无关分割及基于单个扫描线分割相结合的方法来减少过分割与欠分割错误率。

参考文献 (References)

- [1] 杨飞, 朱株, 龚小谨. 基于三维激光雷达的动态障碍实时检测与跟踪 [J]. 浙江大学学报: 工学版, 2012, 46(9): 1565-1571.
YANG Fei, ZHU Zhu, GONG Xiaojin. Real-time dynamic obstacle detection and tracking using 3-D LIDAR [J]. *Journal of Zhejiang University: Engineering Science*, 2012, 46(9): 1565-1571. (in Chinese)
- [2] Himmelsbach M, Müller A, Lüttel T. LIDAR-based 3-D object perception [C]// Proceedings of 1st International Workshop on Cognition for Technical Systems. Munich, Germany, 2008.
- [3] 谌彤童. 三维激光雷达在自主车环境感知中的应用研究[D]. 长沙: 国防科技大学, 2011.
CHEN Tongtong. Research on 3-D LIDAR-Based Environment Perception for ALV [D]. Changsha, China: National University of Defense Technology, 2011. (in Chinese)
- [4] Hoover A, Jean-Baptiste G, Jiang X, et al. An experimental comparison of range image segmentation algorithms [J]. *IEEE Trans Pattern Anal Mach Intell*, 1996, 18(7): 673-689.
- [5] Moosmann F, Pink O, Stiller C. Segmentation of 3-D LIDAR data in non-flat urban environments using a local convexity criterion [C]// 2009 IEEE Intelligent Vehicles Symposium. Xi'an, China, 2009: 215-220.
- [6] Klasing K, Wollherr D, Buss M. A clustering method for efficient segmentation of 3-D laser data [C]// 2008 IEEE International Conference on Robotics and Automation. Pasadena, CA, USA, 2008: 4043-4048.
- [7] Golovinskiy A, Funkhouser T. Min-cut based segmentation of point clouds [C]// 2009 IEEE 12-th International Conference on Computer Vision Workshops. Kyoto, Japan, 2009: 39-46.
- [8] Himmelsbach M, Hundelshausen F, Wuensche H. Fast segmentation of 3-D point clouds for ground vehicles [C]// 2010 IEEE Intelligent Vehicles Symposium. San Diego, CA, USA, 2010: 560-565.
- [9] Wang D Z, Posner I, Newman P. What could move? Finding cars, pedestrians and bicyclists in 3-D laser data [C]// 2012 IEEE International Conference on Robotics and Automation. Saint Paul, MN, USA, 2012: 4038-4044.
- [10] Vosselman G, Dijkman S. 3-D building model reconstruction from point clouds and ground plans [J]. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 2001, 34(3): 37-44.
- [11] Jiang X, Bunke H. Fast segmentation of range images into planar regions by scan line grouping [J]. *Machine Vision and Applications*, 1994, 7(2): 115-122.
- [12] Felzenszwalb P F, Huttenlocher D P. Efficient graph-based image segmentation [J]. *International Journal of Computer Vision*, 2004, 59(2): 167-181.
- [13] Triebel R, Shin J, Siegwart R. Segmentation and unsupervised part-based discovery of repetitive objects [C]// Robotics: Science and Systems. Zaragoza, Spain, 2010: 314-319.
- [14] Woo H, Kang E, Wang S, et al. A new segmentation method for point cloud data [J]. *International Journal of Machine Tools and Manufacture*, 2002, 42(2): 167-178.
- [15] MA Dengwu, YE Wen, LI Ying. Survey of box-based algorithms for collision detection [J]. *Journal of System Simulation*, 2006, 4: 1058-1064.
- [16] Jiménez P, Thomas F, Torras C. 3-D collision detection: A survey [J]. *Computers & Graphics*, 2001, 25(2): 269-285.
- [17] Chang J W, Wang W, Kim M S. Efficient collision detection using a dual OBB-sphere bounding volume hierarchy [J]. *Computer-Aided Design*, 2010, 42(1): 50-57.