

地铁站场景下的可行驶区域检测^①

王思功, 朱 明

(中国科学技术大学 信息科学技术学院, 合肥 230026)
通信作者: 王思功, E-mail: wangsigong@mail.ustc.edu.cn



摘 要: 在室内环境下的机器人视觉导航任务中, 可行驶区域检测是不可或缺的一部分, 这是保证自动驾驶任务实现的基础。目前较多的解决方法是对数据集中出现过的障碍物进行识别来检测可行驶区域, 缺乏灵活性, 因此本文提出了一种针对地铁站等室内平坦地面的可行驶区域检测方法, 提高实用性。本文采用经典的 MobileNetV3 网络对采集到的前方图像进行分类, 判断是否为地面区域。由于室内地面的地标、箭头等贴纸的影响, 因此需要对非地面区域进一步判断, 与常规的立体障碍物进行区分。本文利用连续帧之间的特征点匹配获得相机移动距离, 并利用直线拟合计算斜率的方法达到区分立体障碍物与平面地标的目的。实验表明, 本文提出的方法能较好地检测机器人前方可行驶区域, 具有较高的实用价值。

关键词: 可行驶区域检测; 特征点匹配; 分类网络

引用格式: 王思功, 朱明. 地铁站场景下的可行驶区域检测. 计算机系统应用, 2023, 32(7): 211-218. <http://www.c-s-a.org.cn/1003-3254/9173.html>

Drivable Area Detection in Subway Station Scenario

WANG Si-Gong, ZHU Ming

(School of Information Science and Technology, University of Science and Technology of China, Hefei 230026, China)

Abstract: In the robot visual navigation task of the indoor environment, the detection of the drivable area is an indispensable part, which is the basis for ensuring the realization of the autonomous driving task. At present, many solutions are to detect the drivable area by identifying obstacles in the dataset, which lacks flexibility. Therefore, a drivable area detection method for indoor flat ground such as subway stations is proposed in this study to improve practicability. The classic MobileNetV3 network is applied to classify the collected front images and determine whether they are ground areas. Due to the influences of stickers such as landmarks and arrows on the indoor floor, it is necessary to further judge the non-ground area and distinguish it from conventional three-dimensional obstacles. In this study, the feature point matching between successive frames is adopted to obtain the camera moving distance, and the method of calculating the slope by straight line fitting is used to distinguish between three-dimensional obstacles and plane landmarks. Experiments show that the proposed method can better detect the drivable area in front of the robot and has high practical value.

Key words: drivable area detection; feature point matching; classification network

1 概述

1.1 研究背景

随着我国移动机器人技术的不断发展, 以及劳动

力成本的不断上升, 利用人工智能和移动机器人技术解决繁琐工作的呼声越来越高, 自动拖地机器人正是这一形势下的产物。目前家用拖地机器人已经成熟, 但

^① 基金项目: 科技创新特区计划 (20-163-14-LZ-001-004-01)

收稿时间: 2023-01-04; 修改时间: 2023-02-03, 2023-02-13; 采用时间: 2023-02-23; csa 在线出版时间: 2023-05-12

CNKI 网络首发时间: 2023-05-15

适用于如地铁站场景的大型平坦室内环境的商用服务型清洁机器人还存在较多问题, 该研究也成为越来越热门的课题^[1]. 为实现拖地机器人的智能化功能, 需要完成可行驶区域的检测, 来判断前方是否可以通行, 这也是导航最基础的任务之一.

目前虽然已有科沃斯^[2]等产品面世并解决了可行驶区域检测问题, 但传感器的堆叠以及较好的算力平台的使用造成了高昂的成本, 这是限制其全面推广的最大因素.

可行驶区域检测的目的即为在当前状态下, 检测出机器人的可通行区域. 可行驶区域检测方法主要分为以下3类: 第1类是基于激光雷达传感器或双目相机的检测方法. Zhao等^[3]利用激光雷达生成的点云图做聚类操作, 将地面与障碍物进行区分. 激光雷达准确率高, 但价格高昂, 不利于大规模普及. Tsai等^[4]利用双目相机产生视差图, 以图像的垂直方向为基准进行查找, 当具有相同视差时认为存在障碍物. 第2类是传统图像处理的方法. 早期的可行驶区域检测更多的是基于图像分割的方法, 主要采用阈值法和区域分割法^[5], 这两种方法比较简单, 适合结构化道路. 曹江华^[6]利用纹理特征检测道路消失点, 通过形成的三角区域完成可行驶区域检测. Hsu等^[7]利用环视系统生成的鸟瞰图完成可行驶区域检测. 传统的可行驶区域检测鲁棒性较差, 速度较慢. 第3类是基于深度学习的可行驶区域检测方法. Frickenstein等^[8]提出了名为DADNet的二值化语义分割网络, 在多个室外道路数据集中取得较好实验效果的同时降低参数量. 王琪琪^[9]提出基于注意力和上下文信息的可行驶区域检测模型MANET, 对结构化和非结构化路面都可取得较好效果. YOLO^[10]、TTFNet^[11]等目标检测网络可以识别数据集中已有的障碍物进而判定可行驶区域, 基于语义分割和目标检测的方法都不能检测不在数据集中的障碍物, 实用性不强. Godard等^[12]提出的单目深度估计网络可以得到类似双目相机的深度图, 但不适用于大型无纹理室内场景.

目前的可行驶区域检测方法并不完全适用于大型无纹理的地铁站场景. 对于体型较小, 速度较慢的拖地机器人, 只需要判断前方区域是否可以通行即可. 影响该区域是否可通行的主要因素在于障碍物的存在, 因此本文将从障碍物的检测与区分入手, 结合深度学习与传统图像处理技术将可行驶区域检测任务分为两个

阶段: (1) 地面类与非地面类的区分. (2) 非地面类中立体障碍物与平面地标贴纸的区分.

1.2 相关工作

目前基于深度学习的图像分类算法已十分成熟, 如ConvNet^[13]等已在公开数据中取得极高的准确率, 但此类算法参数量大, 并不适用于移动端机器人. 基于成本、续航以及算法运行速度的考虑, 本文采取轻量化的MobileNetV3^[14]模型做第1阶段地面与非地面的分类任务, 相较于其他的轻量化卷积神经网络如ShuffleNetV2^[15]、EfficientNet^[16]等, MobileNetV3具备更高的准确率和运行速度.

何少佳等^[17]利用目标场景匹配和目标投影匹配算法相结合区分障碍物和平面地标, 但对平面地标的识别效果不佳. Lee等^[18]提出了一种基于改进逆投影变换的障碍物检测方法, 该方法可以对区分立体障碍物和毛毯等平面物体, 但需要极低的检测视角. Wang等^[19]利用逆投影变换和多尺度积分图像的方法进行障碍物和平面地标的区分, 但存在逆投影变换对相机运动参数敏感的缺点. 刘威等^[20]在Yang等^[21]基础上, 根据相机运动参数和特征点匹配的结果计算特征点的归属, 将图像划分为背景路面和前景障碍物. 然而这种对相机运动参数的估计依赖于大量的地面特征点, 并不适用于地铁站场景下的无纹理瓷砖地面. 本文将这种特征点匹配方法与数据回归方法相结合, 完成第2阶段平面地标贴纸与立体障碍物的区分, 适用于无纹理场景的同时, 对于传感器误差具有更强的鲁棒性.

1.3 本文工作

本文主要贡献如下: (1) 本文采用轻量化MobileNetV3模型做迁移学习对地面类与非地面类进行区分. (2) 针对非地面类图像的细分: 1) 利用特征点匹配技术得到前后两帧对应点的坐标. 2) 对坐标进行旋转补偿. 3) 利用旋转后的坐标点计算机人行走距离做直线拟合, 根据斜率区分立体障碍物和平面坐标. (3) 采用RGB摄像头拍摄并标注地面数据集.

2 理论基础

2.1 MobileNetV3 网络

MobileNetV3^[14]在MobileNetV2^[22]倒残差结构的基础上, 引入逐点卷积提升维度, 在高维空间中利用深度可分离卷积提取特征, 再利用逐点卷积降维, 引入了网络架构搜索技术, 通过搜索最佳模块达到优化整个

网络结构的目的. 为了进一步提高模型的准确率, 引入了一种新的激活函数 *hard-swish* 作为激活函数:

$$\text{hard-swish}(x) = x \cdot \frac{\text{ReLU6}(x+3)}{6} \quad (1)$$

其中,

$$\text{ReLU6}(x) = \min(6, \max(x, 0)) \quad (2)$$

该激活函数可以在移动端浮点数精度较低的情况下, 限制数值范围的无限增长, 保证依然保留较好的数值分辨率. 同时引入了通道注意力机制模块 SE Block 进一步提高精度. 如图 1 所示, MobileNetV3 主要结构为 bneck, 根据 bneck 层数不同, 分为 small 版本和 large 版本.

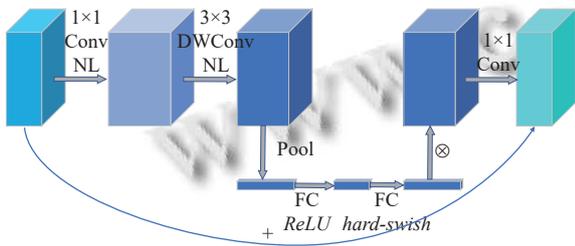


图 1 bneck 结构图

2.2 特征点匹配

图像匹配广泛应用于计算机视觉领域中, 检测图像中的代表性区域如边缘、角点等特征是图像匹配的第 1 步. SIFT 特征点, SURF 特征点, ORB 特征点是最为流行的特征点提取方法. 特征点由关键点和描述子两个部分组成, 关键点是指特征点在图像中的位置, 具有方向和尺度两个信息, 而描述子是一个向量, 具备周围其他像素的信息, 当两个特征点的描述子在向量空间上相近, 就认为是相同的特征点.

SIFT 算法的本质是在不同的尺度空间中寻找特征点, 在原始图像中利用高斯滤波函数构建差分金字塔. 当像素点为同一尺度空间及相邻尺度空间中的极值点时, 认为该点为关键点. 为了保证关键点具备旋转不变性, 以关键点邻域内各像素点的梯度方向为横轴, 梯度幅值的累加值为纵轴做直方图, 峰值处的梯度方向即为关键点主方向. 为了提取特征描述子, 以关键点为中心, 使用 16 个种子点来描述, 每个种子点有 8 个方向的梯度强度信息, 得到 128 维特征向量后, 将特征向量方向与关键点主方向重合, 将特征向量长度做归一化处理, 即可获得特征描述子. SIFT 算法具有旋转不变性

和尺度不变性, 稳定性较高, 但算法复杂度较高, 实时性较差.

SURF 算法在 SIFT 算法的基础上作出修改, 使用卷积而非差分算子构建尺度空间, 引入 Hessian 矩阵改进关键点检测方法并利用积分图加速运算, 引入小波变化对描述子提取算法做出改进, 加快了计算速度.

ORB 算法在 FAST 特征点检测的基础上增加了方向向量, 并采用 BRIEF 描述子加快图像特征的提取速度. ORB 算法在保持较好性能的情况下, 计算时间只有 SURF 算法的 10%, SIFT 算法的 1%.

特征点提取后, 通过特征匹配建立两图之间的关系. 常见的特征匹配有暴力匹配和快速近似最近邻 (FLANN) 算法. 暴力匹配是指求解某一个特征描述子与其他所有描述子之间的距离并排序, 取最小距离的特征点作为匹配点, 然后利用汉明距离或 K 近邻匹配法对误匹配点过滤. FLANN 算法利用 K-D 树或 K-means 对特征进行聚类建模, 多用于特征点极多的情况. 由于本文涉及的图像较小, 特征点较少, 因此采用暴力匹配并用 K 近邻匹配法进行过滤.

3 算法

3.1 算法框架

地铁站清洁车具有体积小、速度慢且恒定的特点, 因此只需要判断正前方区域是否可通行即可. 本文将清洁车前方区域划分为地面和非地面, 而由于地铁站场景下指示地标的存在, 非地面区域又可划分为平面地标和立体障碍物. 如图 2 所示, 清洁车前方区域被划分为 3 种情况: 地面、平面地标、立体障碍物. 在场景判断为地面或平面地标的情况下, 清洁车可以通行, 否则不可通行. 如何对上述 3 种情况进行区分是本文的核心.



图 2 前方区域类型图

算法流程如图 3 所示, 将采集到的 1920×1080 图像按照高度三等分并取最下方的 $1/3$, 然后取中间 $1/2$ 作为检测区域, 将截取后的图片送入训练好的 MobileNetV3 中进行地面与非地面类的判断, 当判定结

果为非地面类时,保存本张图片并采集下一帧作为特征点匹配,根据特征点对计算移动距离并以实际距离

为依据进行数据过滤,最后根据计算距离拟合结果区分立体障碍物和平面地标.

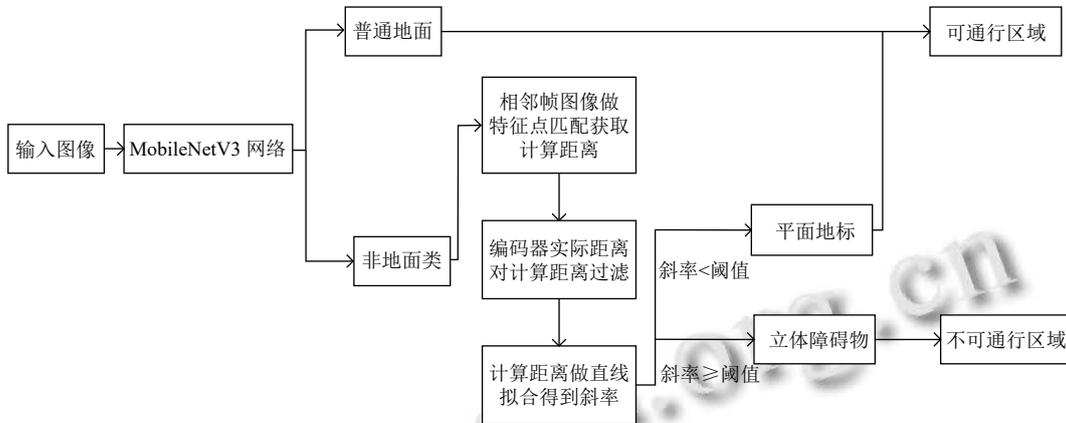


图3 本文算法流程图

3.2 基于 MobileNetV3 网络的迁移学习

基于模型的迁移学习是指利用以往任务中学出的模型参数,来辅助新领域的学习,得到自己的模型.

本文采集了地面、平面地标、立体障碍物3种情况下的数据集,由于自制数据集比已有公开数据集数据量少,因此采用迁移学习对网络进行更好的训练.将输入图像输入 MobileNetV3 经过若干的 bneck 层、卷积层、池化层以及最后两个输出卷积层,最终输出 1000 种分类.由于分类数目不同,本任务需要对 MobileNetV3 的最后一个输出卷积层进行修改,如图4所示,将最后一个卷积层所用的 1000 个卷积核删掉保留 2 个,形成一个新的只输出 2 种分类的卷积层.

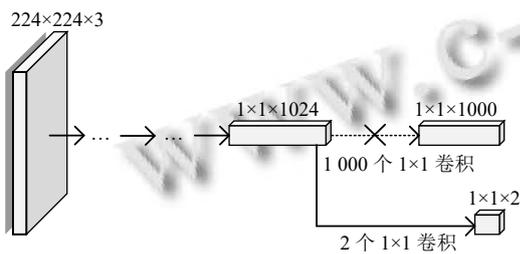


图4 输出层修改示意

利用 ImageNet 分类数据集上的预训练结果对模型进行初始化,继承大数据集上的先验知识,实现整体的模型参数迁移.最后利用自制数据集对模型进行训练.使用迁移学习,模型参数并非从头开始,因此加快了模型训练速度.训练结束后获得模型的权重文件,利用该权重文件即可对输入图片完成检测,做出地面与

非地面类的区分.

3.3 基于特征点匹配的相机移动距离的计算

相机跟随清洁车移动,在地铁站路面水平且无坡起的前提下,建立如图5所示的相机模型.为了简化运算将相机坐标系和世界坐标系重合,使用 $O-XYZ$ 坐标系表示.其中 Z 轴与相机光轴重合, Y 轴垂直于地面, X 轴平行于地面. $O'-UV$ 为像素坐标系,其中 $O'U$ 轴与 X 轴平行, $O'V$ 轴与 Y 轴平行.点 $P(X, Y, Z)$ 是空间中的任意一点, $P_S(u, v)$ 是该点在图像上的成像点, $P_G(X_G, Y_G, Z_G)$ 是该点对地面的投影点,当 P 点为地面上一点时, P 点和 P_G 点重合.

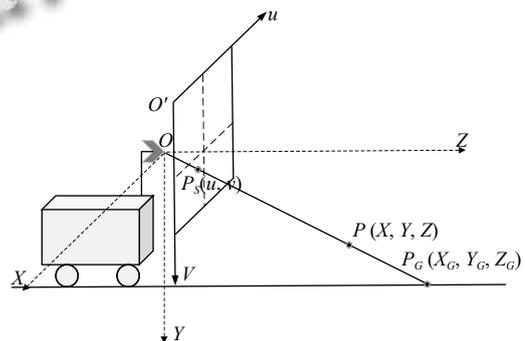


图5 相机模型

对于空间中任意一点 P 与它对应的像素点 P_S 有如下关系:

$$u = f_x \frac{X}{Z} + c_x \tag{3}$$

$$v = f_y \frac{Y}{Z} + c_y \quad (4)$$

其中, f_x, f_y, c_x, c_y 为相机内参, 可以通过张氏标定法测得.

根据成像原理, 直线 PP_G 上所有的空间点在图像中的映射点都是点 P_S , 无法根据图像中的点定位到唯一的空间点. 在清洁车运动过程中, 相机的离地高度保持一致, 即任意一点的投影点 P_G 坐标的 Y_G 值已知. 对于图像中的任意像素, 在已知坐标的情况下, 可以唯一地得到对应地面投影点 P_G 的坐标值.

$$Z_G = \frac{f_y Y}{v - c_y} \quad (5)$$

$$X_G = \frac{u - c_x}{f_x} Z_G \quad (6)$$

清洁车在采集第 1 帧图像的 T_1 时刻到采集第 2 帧图像的 T_2 时刻, 除平移运动外, 还可能会发生绕 Y 轴的旋转运动. 为了避免自身旋转对投影点坐标计算的影响, 需要对 T_2 时刻的相机坐标系进行旋转补偿, 使得两个坐标系的各个轴之间互相平行. 由于相机绕 Y 轴旋转, 因此只需要关注 X 轴和 Z 轴. 如图 6 所示, T_1 时刻的坐标系以 $O-XZ$ 表示, T_2 时刻的坐标系以 $O'-X'Z'$ 表示, 前后时刻的偏角即 β . 对于坐标系 $O'-X'Z'$ 中的任意一点 (X', Z') 可通过如下公式转换到 $O-XZ$ 坐标系中:

$$\begin{pmatrix} X \\ Z \end{pmatrix} = \begin{pmatrix} \cos\beta & -\sin\beta \\ \sin\beta & \cos\beta \end{pmatrix} \begin{pmatrix} X' \\ Z' \end{pmatrix} \quad (7)$$

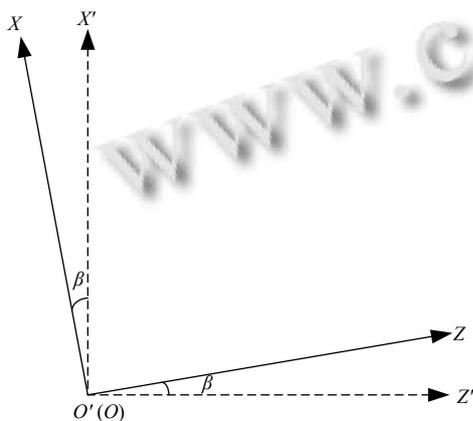


图 6 旋转补偿示意图

由于先后时刻相差较短, T_1 和 T_2 时刻偏移角度较小, 因此式 (7) 可优化为:

$$\begin{pmatrix} X \\ Z \end{pmatrix} = \begin{pmatrix} 1 & -\beta \\ \beta & 1 \end{pmatrix} \begin{pmatrix} X' \\ Z' \end{pmatrix} \quad (8)$$

经过旋转补偿后的前后 T_1, T_2 两个时刻的相机位置如图 7 所示, P 为地面上一点, 在 T_1 时刻的成像点为 P_1 , 地面投影点为 P_{G1} , T_2 时刻的成像点为 P_2 , 地面投影点为 P_{G2} , 其中点 P, P_{G1}, P_{G2} 三点重合. Q 为空间中一点, 成像点分别为 Q_1, Q_2 , 地面投影点分别为 Q_{G1}, Q_{G2} .

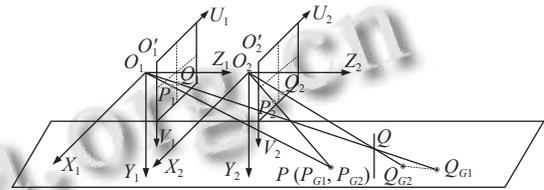


图 7 前后两个时刻的图像采集示意图

对于地面上一点 P 及其投影点, 得到如下关系:

$$\overrightarrow{O_1 O_2} = \overrightarrow{O_1 P_{G1}} + \overrightarrow{P_{G1} O_2} \quad (9)$$

由于点 P, P_{G1}, P_{G2} 三点重合, 可得:

$$\overrightarrow{O_1 O_2} = \overrightarrow{O_1 P_G} + \overrightarrow{P_G O_2} \quad (10)$$

即:

$$\overrightarrow{O_1 O_2} = -(\overrightarrow{O_2 P_G} - \overrightarrow{O_1 P_G}) \quad (11)$$

所以:

$$\|\overrightarrow{O_1 O_2}\| = \|\overrightarrow{O_2 P_G} - \overrightarrow{O_1 P_G}\| \quad (12)$$

即:

$$\|\overrightarrow{O_1 O_2}\| = \sqrt{(Z_{G2} - Z_{G1})^2 + (X_{G2} - X_{G1})^2} \quad (13)$$

其中, $\|\overrightarrow{O_1 O_2}\|$ 为相机位移, 可以根据投影点前后时刻的坐标求出.

对于障碍物上一点 Q 及其投影点, 可以得到如下关系:

$$\overrightarrow{O_1 O_2} = \overrightarrow{O_1 Q_{G1}} + \overrightarrow{Q_{G1} Q_{G2}} - \overrightarrow{O_2 Q_{G2}} \quad (14)$$

即:

$$\overrightarrow{O_1 O_2} + \overrightarrow{Q_{G2} Q_{G1}} = -(\overrightarrow{O_2 Q_{G2}} - \overrightarrow{O_1 Q_{G1}}) \quad (15)$$

根据成像关系可得 $\overrightarrow{O_1 O_2}$ 与 $\overrightarrow{Q_{G2} Q_{G1}}$ 之间夹角为 0, 所以:

$$\|\overrightarrow{O_1 O_2}\| < \|\overrightarrow{O_2 Q_{G2}} - \overrightarrow{O_1 Q_{G1}}\| \quad (16)$$

即:

$$\|\vec{O_1O_2}\| < \sqrt{(Z_{G2} - Z_{G1})^2 + (X_{G2} - X_{G1})^2} \quad (17)$$

本节展示了如何利用相机参数及匹配的特征点求解相机运动距离. 将通过相匹配的特征点计算得到的相机移动距离称为计算距离, 通过车轮编码器得到的称为实际距离, 利用地面特征点求解的计算距离与实际距离相等, 而利用障碍物特征点求解的计算距离大于实际距离. 这是平面地标与障碍物区分的依据之一.

3.4 平面地标与立体障碍物的区分方法

第 3.3 节指出, 对于地面上的一点, 理论上计算距离与实际距离相等. 对于障碍物上的一点, 计算距离大于实际距离, 根据式 (15), 由于 $\vec{O_1O_2}$ 与 $\vec{Q_{G2}Q_{G1}}$ 之间夹角为 0, 为便于讨论可忽略 X 轴, 将图 7 简化成为图 8 所示. Q 和 Q' 为同一障碍物不同高度的两个特征点, 随着高度的增加 $\vec{Q_{G2}Q_{G1}}$ 的距离值也在增加, 所以相机的计算距离随着特征点高度的增加而增加.

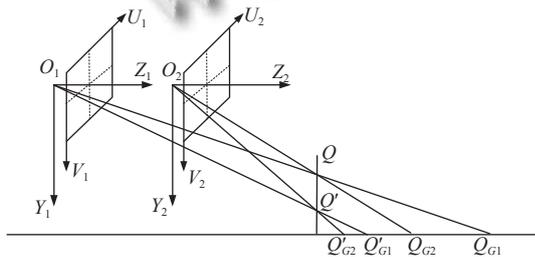


图 8 不同高度特征点采集

由于传感器精度和相机内参测量精度的影响, 本文不是简单的对实际距离与计算距离相比较, 而是利用多组数据以特征点像素高度为自变量, 特征点对的计算距离为因变量做函数拟合, 当斜率大于阈值 λ 时认为前方为障碍物, 否则为可行驶区域.

本节展示了障碍物和平面地标区分的第 2 步, 计算距离随障碍物上点的高度而增加, 通过函数拟合求斜率, 根据斜率大小区分障碍物和平面地标.

4 实验分析

4.1 数据集

本文中笔者自制数据集. 数据采集平台如图 9 所示, 相机安装在清洁车前端, 光轴与地面平行且相机高度为 30 cm, 陀螺仪水平放置在清洁车顶部.

小车保持以 15 cm/s 的速度匀速行驶, 以 3 帧/s 的频率采集 1920×1080 大小的图片, 同时保存每一次图

片采集时刻的陀螺仪角度值和车轮编码器数值. 在多个场景中对不同障碍物和平面地标采集了时长 45 min 的视频. 从中选取 2 419 张图片并进行图片分割, 将其中 80% 图片作为训练集, 10% 用作验证集, 10% 用作测试集.

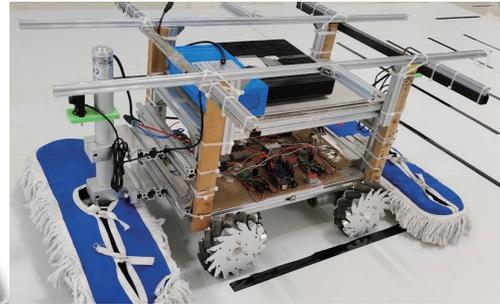


图 9 数据集采集装置图

4.2 实验过程及结果

修改 MobileNetV3 网络使之输出地面类和非地面类两类, 利用 ImageNet 的预训练模型进行迁移学习, 学习率设为 0.000 1, 当验证集的 Loss 不再下降时停止训练. MobileNetV3 训练使用 11 GB 显存的 NVIDIA GeForce GTX1080Ti GPU. 使用训练好的模型对测试集数据进行检测, 所用配置如表 1 所示. 不同版本的 MobileNetV3 均进行实验, 同时引入其他轻量化网络进行对比试验, 实验结果如表 2 所示.

表 1 检测所用机器配置表

类型	型号	参数
系统	Windows	Windows 10
CPU	Intel(R) Core(TM) i7-10510U	8核
GPU	—	—
内存	DDR4	8 GB

表 2 地面与非地面类实验结果

模型	Accuracy (%)	FPS
MobileNetV3-small	97.9	46.3
MobileNetV3-large	98.4	26.5
ShuffleNetV2	93.9	22.6
Xception	90.5	18.9
EfficientNet	96.7	27.2

由检测结果可知, MobileNetV3-small 可以较好地对待地面类与非地面做出区分, 相较于其他分类网络具备较高准确率的同时检测速度最快, 更适合运用在 CPU 场景下.

对于非地面类, 计算每组特征点对应的相机移动

距离,由于误匹配的存在会造成计算距离产生误差,因此根据相机的实际移动距离进行数据剔除,经实验证明,在移动速度恒定的情况下,将大于实际距离2倍以及小于实际距离1/2的计算距离进行舍弃.特征点匹配效果如图10所示.

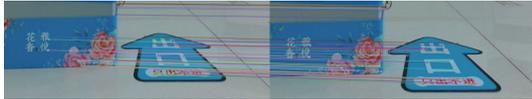


图10 特征点匹配示意图

室内瓷砖纹理条件不同,特征点可能分布在整张图像而非障碍物或平面地标中,因此对已截取的图片按照宽度进行8等分,对每一组数据做以计算距离为因变量像素高度为自变量的直线拟合,得到8组不同的斜率值,当存在一组斜率大于阈值 λ 时,认为当前为立体障碍物.经实验验证, λ 取值为0.01.本文采用表1的机器配置分别测试SIFT、SURF、ORB三种特征点提取结合暴力匹配的方法,并选取文献[17]和文献[19]中提及的方法作为对比实验,结果如表3所示.

表3 平面地标与障碍物区分实验结果

方法	Accuracy (%)	FPS
SIFT	89.3	24.5
SURF	86.7	38.4
ORB	81.3	135.1
何少佳等 ^[17]	65.8	17.2
Wang等 ^[19]	74.7	21.5

综合清洁车移动速度以及识别准确率,本文采用MobileNetV3-small做地面类与非地面类的分类任务,采用SIFT特征点匹配解决非地面类中立体障碍物与平面地标的区分,最终整体准确率达到95.1%.

5 结论与展望

本文提出了一种基于地铁站清洁车的可行区域检测方法.将可行区域检测转化为对普通地面、平面地标、立体障碍物3种路况的区分,采用轻量级网络MobileNetV3区分普通地面,利用匹配特征点求斜率的方法区分立体障碍物和平面地标.本文提出的方法效果较好,速度较快,只利用CPU即可满足实时性要求,解决了可行区域的判定问题,同时降低了成本,下一步计划将本方法用于地铁站清洁车中做实际的应用.

参考文献

- 1 亿欧智库. 2021年中国商用服务机器人市场研究报告. 机器人产业, 2022, (2): 76–90. [doi: 10.3969/j.issn.2096-0182.2022.02.015]
- 2 苏亮. 科沃斯2022秋季战略发布会: 迈向更广阔的机器人市场. 家用电器, 2022, (11): 84–85.
- 3 Zhao YM, Zhang X, Huang XM. A divide-and-merge point cloud clustering algorithm for LiDAR panoptic segmentation. Proceedings of the 2022 International Conference on Robotics and Automation. Philadelphia: IEEE, 2022. 7029–7035. [doi: 10.1109/ICRA46639.2022.9812058]
- 4 Tsai YC, Chen KH, Chen Y, *et al.* Accurate and fast obstacle detection method for automotive applications based on stereo vision. Proceedings of the 2018 International Symposium on VLSI Design, Automation and Test. Hsinchu: IEEE, 2018. 1–4. [doi: 10.1109/VLSI-DAT.2018.8373249]
- 5 王龙飞, 严春满. 道路场景语义分割综述. 激光与光电子学进展, 2021, 58(12): 1200002.
- 6 曹江华. 复杂背景下非结构化道路可行区域检测研究 [硕士学位论文]. 杭州: 浙江科技学院, 2020.
- 7 Hsu CM, Chen JY. Around view monitoring-based vacant parking space detection and analysis. Applied Sciences, 2019, 9(16): 3403. [doi: 10.3390/app9163403]
- 8 Frickenstein A, Vemparala MR, Mayr J, *et al.* Binary DAD-net: Binarized driveable area detection network for autonomous driving. Proceedings of the 2020 IEEE International Conference on Robotics and Automation. Paris: IEEE, 2020. 2295–2301. [doi: 10.1109/ICRA40945.2020.9197119]
- 9 王琪琪. 基于深度学习的道路可行区域检测方法 [硕士学位论文]. 北京: 北京交通大学, 2021. [doi: 10.26944/d.cnki.gbfju.2021.001038]
- 10 Redmon J, Divvala S, Girshick R, *et al.* You only look once: Unified, real-time object detection. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 779–788.
- 11 Liu ZL, Zheng T, Xu GD, *et al.* Training-time-friendly network for real-time object detection. Proceedings of the 34th AAAI Conference on Artificial Intelligence. New York: AAAI Press, 2020. 11685–11692.
- 12 Godard C, Mac Aodha O, Firman M, *et al.* Digging into self-supervised monocular depth estimation. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 3827–3837.
- 13 Liu Z, Mao HZ, Wu CY, *et al.* A ConvNet for the 2020s.

- Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022. 11966–11976. [doi: [10.1109/CVPR52688.2022.01167](https://doi.org/10.1109/CVPR52688.2022.01167)]
- 14 Howard A, Sandler M, Chu G, *et al.* Searching for MobileNetV3. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 1314–1324. [doi: [10.1109/ICCV.2019.00140](https://doi.org/10.1109/ICCV.2019.00140)]
- 15 Ma NN, Zhang XY, Zheng HT, *et al.* ShuffleNet V2: Practical guidelines for efficient CNN architecture design. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2018. 122–138.
- 16 Tan MX, Le Q. EfficientNet: Rethinking model scaling for convolutional neural networks. Proceedings of the 36th International Conference on Machine Learning. Long Beach: PMLR, 2019. 6105–6114.
- 17 何少佳, 刘子扬, 史剑清. 基于单目视觉的室内机器人障碍检测方案. 计算机应用, 2012, 32(9): 2556–2559.
- 18 Lee TJ, Yi DH, Cho DI. A monocular vision sensor-based obstacle detection algorithm for autonomous robots. Sensors, 2016, 16(3): 311. [doi: [10.3390/s16030311](https://doi.org/10.3390/s16030311)]
- 19 Wang SH, Li XX. A real-time monocular vision-based obstacle detection. Proceedings of the 6th International Conference on Control, Automation and Robotics. Singapore: IEEE, 2020: 695–699.
- 20 刘威, 于红绯, 杨恒, 等. 一种新的基于单目视觉的广义障碍物检测方法. 电子学报, 2011, 39(8): 1793–1799.
- 21 Yang CH, Hongo H, Tanimoto S. A new approach for in-vehicle camera obstacle detection by ground movement compensation. Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems. Beijing: IEEE, 2008. 151–156. [doi: [10.1109/ITSC.2008.4732533](https://doi.org/10.1109/ITSC.2008.4732533)]
- 22 Sandler M, Howard A, Zhu ML, *et al.* MobileNetV2: Inverted residuals and linear bottlenecks. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 4510–4520. [doi: [10.1109/CVPR.2018.00474](https://doi.org/10.1109/CVPR.2018.00474)]

(校对责编: 孙君艳)