

双舵轮 AGV 视觉里程计的设计方法^①



梅瑞冬¹, 张捷¹, 李强³, 刘超³, 王书亭²

¹(华中科技大学 能源与动力工程学院, 武汉 430074)

²(华中科技大学 机械科学与工程学院, 武汉 430074)

³(广东省智能机器人研究院, 东莞 523000)

通讯作者: 张捷, E-mail: Bakerzh@tom.com

摘要: 针对双舵轮 AGV 在地面崎岖不平 and 轮胎打滑的情况下编码器失效的问题. 本文提出一种使用价格低廉的 RGB-D 相机做视觉里程计的方案, 避免了双舵轮 AGV 直接运动学建模导致里程计航迹推算累积误差过大的问题. 本文采用 ORB 算子对图像进行特征提取和匹配, 使用 ICP 的方法进行位姿估计. 然后在 Linux+ROS 平台下搭建视觉里程计, 并且和激光雷达数据融合, 通过粒子滤波算法进行定位. 最后, 分别在不同环境下对比了编码器和视觉里程计的定位效果, 并验证了整个系统的鲁棒性.

关键词: 自动定位和建图; 自动引导小车; 视觉里程计; 双舵轮

引用格式: 梅瑞冬, 张捷, 李强, 刘超, 王书亭. 双舵轮 AGV 视觉里程计的设计方法. 计算机系统应用, 2019, 28(7): 228-233. <http://www.c-s-a.org.cn/1003-3254/6967.html>

Design Method of Visual Odometry on Dual Steering Wheel AGV

MEI Rui-Dong¹, ZHANG Jie¹, LI Qiang³, LIU Chao³, WANG Shu-Ting²

¹(School of Energy and Power Engineering, Huazhong University of Science and Technology, Wuhan 430074, China)

²(School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China)

³(Guangdong Intelligent Robotics Institute, Dongguan 523000, China)

Abstract: Aiming at the problem of encoder failure of dual rudder wheel AGV under rough road conditions, a scheme of using low-cost RGB-D camera as visual odometer is proposed. This method can avoid the direct kinematics modeling of dual rudder wheel AGV and solve the problem of excessive accumulated error in odometer track estimation. In this study, we use pinhole camera model to find the correspondence between spatial points and pixels firstly. Then the ORB operator is used to extract and match the image, after that the method of Iterative Closest Point(ICP) is used for pose estimation. A visual odometer is built on the platform of Linux + ROS, which is fused with lidar for localization using particle filter algorithm. The positioning effects of encoder and visual odometer are compared in different environments, and the robustness of the whole system is verified.

Key words: simultaneous localization and mapping; automated guided vehicle; visual odometry; double rudder

随着电商的崛起和“工业 4.0”的提出, 基于 AGV(Automated Guided Vehicle, 自动巡航小车)的智能物流系统相较于传统的人工分拣具有精度高, 安全可靠, 运行成本低廉的优势. 并且 AGV 小车可以 24 小

时不间断地工作, 大大提高了仓库的分拣效率^[1].

AGV 按照驱动方式大概可以分为三种: 单舵轮, 差动轮, 双舵轮. 其中双舵轮式 AGV 具有非常高的灵活性, 可做出原地旋转, 平移, 侧移等全向的移动. 底盘

① 基金项目: 国家自然科学基金 (51675197)

Foundation item: National Natural Science Foundation of China (51675197)

收稿时间: 2018-12-24; 修改时间: 2019-01-18; 采用时间: 2019-01-31; csa 在线出版时间: 2019-07-01

模型如图 1 所示。

双舵轮的 AGV 小车具有良好的发展前景, 是目前研究的热点, 但是因为使用两个舵轮, 其运动模型复杂且导航定位精度不高. 双舵轮 AGV 小车还不能大规模、多领域地投入使用。

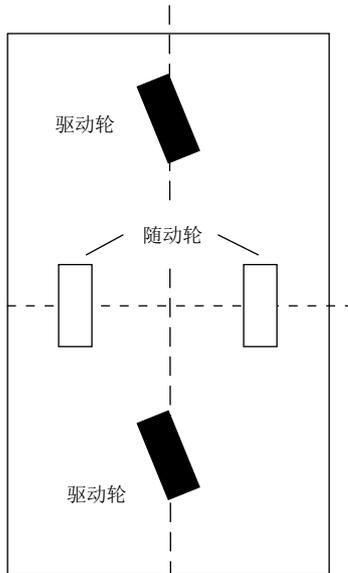


图 1 双舵轮底盘模型

导航定位是 AGV 小车控制的核心部分^[2], 能实时地上传 AGV 小车精确的坐标位置, 是后续路径规划算法和避障算法的保障, 目前的导航方法主要有依赖环境的: 光学导航、磁带导航、二维码标志等, 和依赖于 AGV 自身传感器的: 激光导航、视觉导航等. 基于自身传感器的 AGV 小车使用范围更加广泛, 成为目前研究的热点. 其中激光导航技术相对成熟但是精密的激光雷达成本非常高昂商业运用价值低^[3]. 随着技术的发展, 视觉在各个领域的应用逐渐增多, 比如: 三维重建^[4]、SLAM^[5]等前沿技术, 而且其传感器价格低廉具有很好的前景。

基于自身传感器定位的 AGV 导航系统不管是视觉导航还是激光导航都有一个非常重要的功能模块——里程计. 其作用是记录小车每个时刻和前一微小时刻的运动增量, 然后将这个增量传递给后台来计算出 AGV 小车的运动轨迹, 得到小车的精确定位. 目前工业运用的两轮差速 AGV 大多采用轮式编码器来记录里程, 然后融合激光雷达进行定位. 但是双舵轮 AGV 小车运用编码器来记录里程会导致定位误差大, 在路面崎岖的情况下尤为明显. 本文用视觉里程计

(Visual Odometry, VO) 去计算里程, 避免对地盘的直接运动学建模提高双舵轮 AGV 小车的定位精度. 本文将详细地论述在实际工程中 VO 的原理和搭建过程, 选取的传感器是微软的 R-GBD 摄像头 kinect1, 搭载在双舵轮的移动重载搬运平台上. 如图 2 所示为算法流程图。

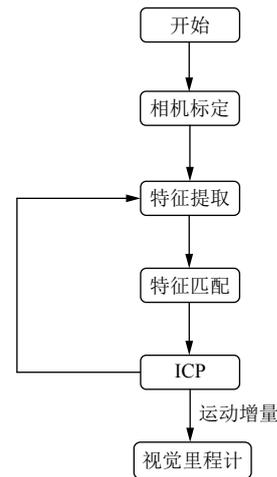


图 2 算法流程图

1 相机模型

1.1 针孔相机模型

相机采集的数据是一帧一帧的数字图像, 它的本质是一个离散的数值矩阵, 这个矩阵记录了各个像素点灰度值 (或者 RGB 值), 但是矩阵和图像都是二维的, 而相机周围的环境是三维的. 因此必须建立相机模型, 找到三维空间点和像素坐标点的对应关系. 如图 3 所示是针孔相机模型。

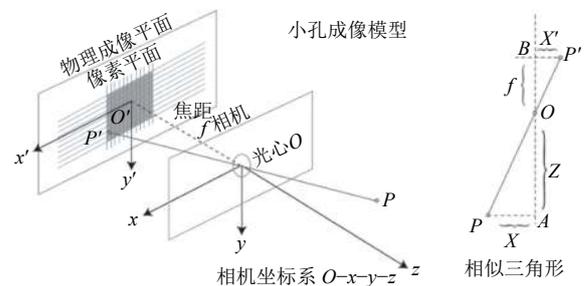


图 3 针孔相机模型

在针孔相机模型中存在着 4 个坐标系, 一个是世界坐标系, 另一个是以相机的光心 O 为原点的相机坐标系, 还有一个是以相机内部成像平面中心为原点的

物理成像平面. 最后一个是像素坐标系, 像素和物理成像平面在一个平面上, 不过其原点在平面的左上角并且度量单位为像素.

现实世界中一个空间点 P 在如图所示的位置, 其通过相机的光心 O 投影在相机的物理成像平面上的 P' 点处, 于是得到了两个相似三角形, 可以得到以下关系:

$$\frac{Z}{f} = -\frac{X}{X'} = -\frac{Y}{Y'} \quad (1)$$

由上式可以看出成的是倒立的像, 相机会将图片自动翻转, 因此上式变为:

$$\frac{Z}{f} = \frac{X}{X'} = \frac{Y}{Y'} \quad (2)$$

这个公式反映了三维空间点和其在相机投影平面上的位置关系. 将点在相机坐标系 (注意是相机坐标系, 和世界坐标系相差一个平移和旋转) 中的坐标映射到物理成像平面, 但是相机操作的是像素, 因此需要将物理成像平面的坐标再转换成像素坐标 (相差一个平移和缩放), 最后得到如下的关系:

$$Z \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \triangleq KP \quad (3)$$

其中, (X, Y, Z) 表示空间点在相机坐标系下的三维坐标, $(u, v, 1)$ 表示投影点在像素坐标系下的齐次坐标. 参数 K 表示的是空间点和相机中投影点的对应关系, 这个参数在相机出厂之后是固定的, 称之为相机的内参 (可由相机标定得到). 由此建立起 3D-2D 的对应关系.

在实际工程运用中空间点的坐标往往是世界坐标 P_w , 和相机坐标系下的坐标 P 相差一个旋转 R 和一个平移 t . 因此有如下等式:

$$P = RP_w + t \quad (4)$$

最后结合上面两式并化为齐次坐标:

$$P_{uv} = KTP_w \quad (5)$$

1.2 相机标定

虽然通过相机模型可解出空间点和像素点的对应关系, 但是摄像头是光学原件, 存在着透视失真如: 径向畸变、切向畸变. 畸变会扭曲图像 (比如现实中的一条直线因为畸变的原因在相机图像中变成了一条曲线) 导致最后实验的结果产生很大的误差^[6].

对相机进行标定, 修正相机的参数是计算精确里程前必须要做的一步, 将像素坐标转换为极坐标并且

引入 5 个畸变纠正参数, 可以得到如下关系.

径向畸变:

$$x_{dis} = x(1 + k_1r^2 + k_2r^4 + k_3r^6) \quad (6)$$

$$y_{dis} = y(1 + k_1r^2 + k_2r^4 + k_3r^6) \quad (7)$$

切向畸变:

$$x_{dis} = x + 2p_1xy + p_2(r^2 + 2x^2) \quad (8)$$

$$y_{dis} = y + 2p_2xy + p_1(r^2 + 2y^2) \quad (9)$$

联立上述方程可以解出这 5 个系数, 进而纠正相机的畸变找到空间点和图像投影点正确的对应关系. 本文采用 OpenCV 结合棋盘格对 kinect1 进行标定, 结果如图 4 所示.

```

mrd@mrd-HP-Pavilion-g4-Notebook-PC: ~
[narrow_stereo]
camera matrix
520.319335 0.000000 319.863969
0.000000 523.744969 240.200597
0.000000 0.000000 1.000000

distortion
0.042950 -0.151297 0.001569 0.000375 0.000000

rectification
1.000000 0.000000 0.000000
0.000000 1.000000 0.000000
0.000000 0.000000 1.000000

projection
517.345520 0.000000 320.093066 0.000000
0.000000 524.978982 240.721951 0.000000
0.000000 0.000000 1.000000 0.000000

('Wrote calibration data to', '/tmp/calibrationdata.tar.gz')
    
```

图 4 相机标定结果

通过棋盘格标定精确地得到了相机的内参数和 5 个纠正系数. 标定后的相机修正了畸变带来的误差, 为后面视觉里程计的搭建提供了条件.

2 特征提取和匹配

2.1 特征提取

相机模型可以得出三维空间点和图像投影点之间的对应关系, 但是一副图像中包含了大量的像素点, 而计算机的计算能力有限, 不足以对每个像素点都进行计算. 因此需要找到一些比较特殊的像素点代表这幅图像, 只需对这些特殊的点进行处理, 并计算对应空间点的相对运动^[7]. 这个过程叫做特征提取, 这些特殊的点叫做“特征点”, 大多数特征点都由两部分构成: 关键点和描述子.

常见的特征提取算子有很多, 比如经典的 Harris 角点检测算子、SIFT 算子、SURF 算子、FAST 算子,

和根据 FAST 改进的 ORB 算子等等^[8]. 每种算子都有自己适合的运用场景, 如 SIFT 算子具有旋转不变性, 尺度不变性, 对光照和噪声不敏感等优点, 适合复杂的环境^[9]. 但是其对计算机性能要求高, 耗时长, 不适合本文的运用场景. 结合本文实际的运用场合选取 ORB 算子来进行特征提取, 其最大的特点是提取速度非常快, ORB 算子由 Oriented FAST 关键点和 BRIEF 描述子构成. 如图 5 所示为 Oriented FAST 关键点的原理图:

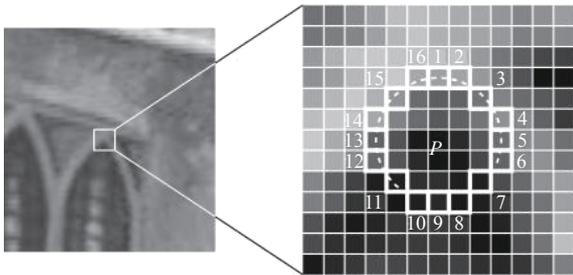


图 5 Oriented FAST 原理图

Oriented FAST 关键点是基于灰度检测的, 如果一个像素点与其周围的绝大多数的像素点的灰度值都差距较大, 则将这个点定义为关键点, 认为这个点是这幅图像的一个特征. 其具体的算法实现步骤如下:

(1) 选取一个点 P , 其灰度值为 I_p , 并以 P 为圆心画一个半径为 3 的圆.

(2) 设定一个阈值 T (一般设为 I_p 的 20%), 然后依次比较圆上 16 个点的灰度值和 I_p 的大小, 如果有连续 12 个点的灰度值大于 I_p+T 或者小于 I_p-T , 则将 P 记为关键点.

(3) 对每一个像素都进行以上两步操作, 得出所有的关键点.

(4) 对得出的所有关键点进行非极大值抑制, 防止关键点扎堆.

本文通过 OpenCV 实现上述算法, 并验证了算法的可行性, 如图 6 所示, 对于 kinect1 采集的图像, 此算法能准确快速地提取出 Oriented FAST 关键点.

2.2 特征匹配

Oriented FAST 可以找到每个关键点在图像中的位置, 视觉里程计需要通过两幅图像计算相机的运动增量. 因此要将两幅图像中的相同的特征点进行匹配, 需要给关键点加上一个独一无二的 id: 描述子. 描述子描述了每个特征点和其周围像素点的一种关系, 因此

不同特征点的描述子是不一样的. 而且在不同的图像中, 相同特征点的描述子差距不大, 计算机可以通过描述子对相同的特征点进行匹配. BRIEF 描述子的基本原理如图 7 所示.

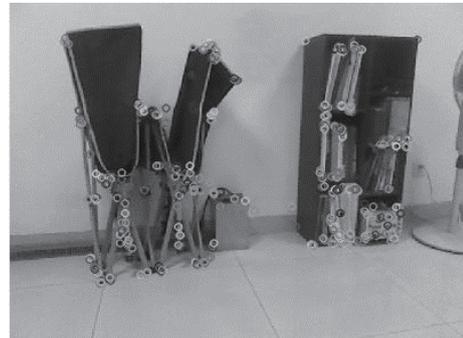


图 6 Oriented FAST 关键点

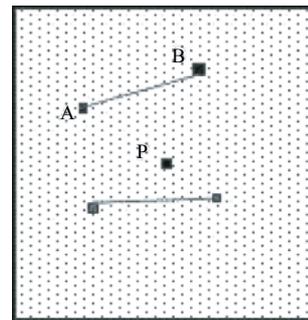


图 7 BRIEF 描述子

以关键点 P 为中心选取一个固定大小的正方形区域, 然后根据一定的概率分布随机地选取两个点 A 和 B 组成一个点对, 比较这两个点灰度值的大小, 如果 A 大于 B 则输出 1, 反之输出 0. 按照此方法重复地选取 n 个点对, 将最后得到的结果记录在一个相应大小的数组中 (采用这样类似于二进制编码的描述方式, 计算机读取描述子的速度非常快). 然后重复以上操作对每一个关键点都求其描述子.

上述方法得到的特征点对旋转, 尺度这两个因素敏感, 当场景发生改变时容易丢失. 通过搭建图像金字塔来改善尺度不变性, 通过求图像的质心来解决旋转不变性. 最后通过计算汉明距离结合快速近似最近邻 (FLANN) 算法对两幅有微小旋转和平移的图像进行特征匹配. 以上算法全在 OpenCV 下实现, 然后在 kinect1 上去检测算法的效果, 结果如图 8 所示.

ORB 算子能有效地提取两幅图像中的特征点, 并

将相同的特征点进行特征匹配。

2.3 视觉里程计

通过前文的步骤,可以准确快速地提取相机图像中的特征点,并对相邻两幅图像中的特征点进行匹配,然后找到与之对应的空间点坐标。最后,可以通过这些空间点的几何关系去求解出相机在这两幅图片中的运动增量。本文采用 ICP(Iterative Closest Point, 迭代最近点)的方法来求解这个运动 R 和 t ^[10,11]。

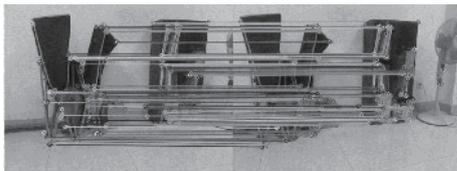


图8 ORB 算子特征匹配图

选取一对已经经过特征匹配的空间点:

$$P = (p_1, \dots, p_n), P' = (p'_1, \dots, p'_n) \quad (10)$$

定义一个误差项 e_i , 可以建立一个最小二乘的问题:

$$p_i = Rp'_i + t \quad (11)$$

$$e_i = p_i - (Rp'_i + t) \quad (12)$$

对于这个最小二乘的问题, 求解 R, t 使误差平方和最小。

$$\min_{R,t} J = \frac{1}{2} \sum_{i=1}^n \| (p_i - (Rp'_i + t)) \|_2^2 \quad (13)$$

至此, 一个基于 RGB-D 摄像头的视觉里程计就搭建完成了。最后在 Linux+ROS 平台下对上述算法进行代码的实现。如图 9 是视觉里程计效果图。

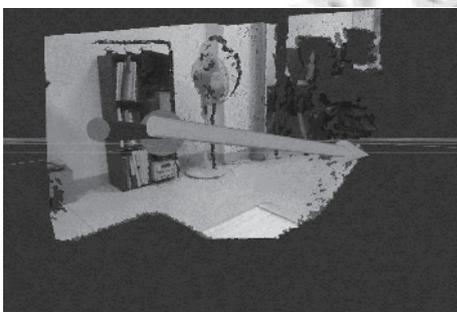


图9 视觉里程计

左边的箭头表示相机当前的位姿, 始终指向相机坐标系的正前方, 右边的箭头实时地接收里程计返回的数据并可视化的表示出来。

3 实验结果与分析

将视觉里程计装载上图 10 所示的双舵轮 AGV 小车台架上和激光雷达数据融合测试并对比定位精度。

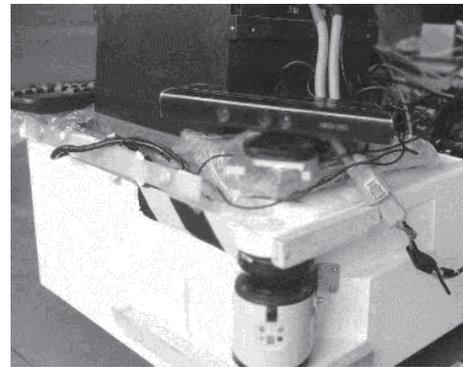


图10 试验台架

本文采用重复定位的方式来测试定位精度, 在系统运行稳定后, 分别重复 20 次定位并绘制出散点图, 和基点比较得出定位精度。图 11 为路面平坦的情况下编码器做里程和视觉做里程的定位精度对比图, 编码器的定位精度为 19.46 mm, 而视觉里程计的定位精度为 16.7 mm, 可以看出视觉里程计的定位精度更高。

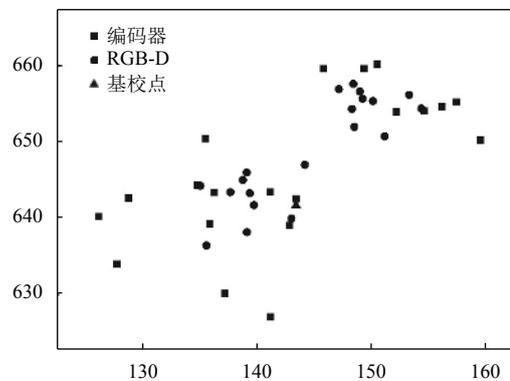


图11 平坦路面定位散点图

但是实际工厂中的环境往往更加复杂, AGV 小车一般会在路面崎岖的环境下工作, 针对崎岖的路面对比两种方案的定位精度, 结果如图 12 所示, 可以看出在崎岖的路面中轮子颠簸和打滑加上运动模型复杂, 编码器不能很好的记录里程信息。定位误差增大为 31.46 mm。视觉里程计避开了直接求解运动模型而且路面对其影响较小, 定位效果有了明显的改善, 重复定位精度为 19.65 mm, 相对于平坦路面的情况下定位精度变化不大。

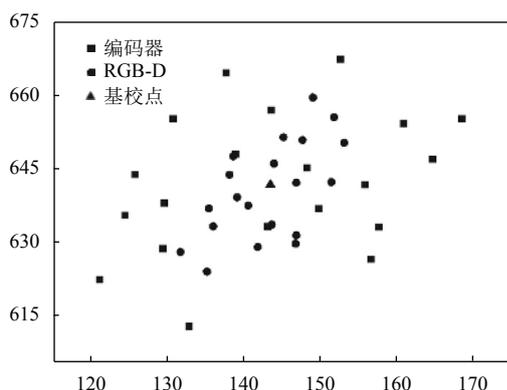


图12 崎岖路面定位散点图

最后为了对比系统的鲁棒性,在崎岖的路面上分别应用两种方案连续运行8小时并绘制时间轴。可以看出视觉做里程的情况下,系统的鲁棒性相对稳定,在连续运行8小时的情况下定位精度几乎保持在一个稳定的范围类,而编码器做里程其对路面非常敏感,定位精度不高累计误差大。

通过图13、图14对比可以看出在双舵轮平台上,运用视觉做里程定位精度较高,在不同路面环境下都能保持在一个稳定的范围内,同时鲁棒性较好,和现在的编码器做里程相比,避免了对底盘的直接运动学建模并且价格低廉具有很好的发展潜力。

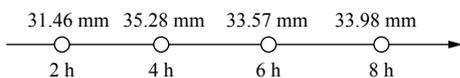


图13 编码器定位精度时间轴

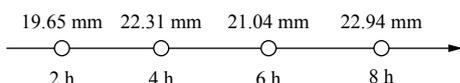


图14 视觉里程计定位精度时间轴

4 结论

针对双舵轮AGV在地面崎岖不平的情况下编码器失效的问题。本文提出一种使用价格低廉的RGB-D相机做视觉里程计的方案,避免对双舵轮AGV的直接运动学建模来解决编码器航迹推算累积误差过大的问题。并在不同路面条件下对视觉里程计和编码器里程计两种方案的定位精度做了对比,视觉里程计的定位精度更高特别是在崎岖路面的条件下,而且系统的鲁棒性相对稳定,在连续运行8小时的情况下定位精度保持在一个稳定的范围类。而编码器做里程对路面

非常敏感,定位精度不高累计误差大。结果表明使用RGB-D相机做视觉里程计运用于双舵轮AGV具有很好的实际运用价值。

参考文献

- 1 Vis IFA. Survey of research in the design and control of automated guided vehicle systems. *European Journal of Operational Research*, 2006, 170(3): 677-709. [doi: 10.1016/j.ejor.2004.09.020]
- 2 Hansen C, Fuerstenberg K. Enabling robust localization for automated guided carts in dynamic environments. In: Zachäus C, Müller B, Meyer G, eds. *Advanced Microsystems for Automotive Applications*. Cham: Springer, 2017. 47-57.
- 3 Ito S, Hiratsuka A, Ohta M, et al. Small imaging depth LIDAR and DCNN-based localization for automated guided vehicle. *Sensors*, 2018, 18(1): E177.
- 4 吴福培, 方信佳, 阳春, 等. 基于单目视觉系统的三维重建方法. *检测技术学报*, 2016, 30(5): 369-374.
- 5 Mur-Artal R, Tardos JD. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Transactions on Robotics*, 2017, 33(5): 1147-1163.
- 6 Luhmann T, Fraser C, Maas HG. Sensor modelling and camera calibration for close-range photogrammetry. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2016, 115: 37-46. [doi: 10.1016/j.isprsjprs.2015.10.006]
- 7 Guyon I, Elisseeff A. An introduction to feature extraction. In: Guyon I, Nikraves M, Gunn S, et al, eds. *Feature Extraction*. Berlin, Heidelberg: Springer, 2006. 1-25.
- 8 Rublee E, Rabaud V, Konolige K, et al. ORB: An efficient alternative to SIFT or SURF. *2011 International Conference on Computer Vision*. Barcelona, Spain. 2011. 2564-2571.
- 9 Lowe D G. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004, 60(2): 91-110. [doi: 10.1023/B:VISI.0000029664.99615.94]
- 10 Pomerleau F, Colas F, Siegwart R. A review of point cloud registration algorithms for mobile robotics. *Foundations and Trends in Robotics*, 2015, 4(1): 1-104. [doi: 10.1561/23000000035]
- 11 Sharp G C, Lee S W, Wehe D K. ICP registration using invariant features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, 24(1): 90-102. [doi: 10.1109/34.982886]