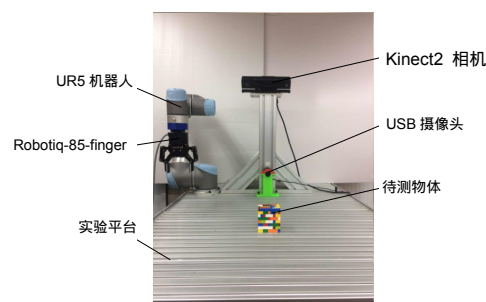


DOI: 10.12086/oe.2018.170650

基于多模态信息的机器人视觉识别与定位研究

魏玉锋, 梁冬泰*, 梁丹, 邢淑敏

宁波大学机械工程与力学学院, 浙江 宁波 315000



摘要: 针对目前物体识别定位算法中, 图像信息来源单一、处理过程复杂与定位误差大等问题, 提出一种基于多模态信息的视觉识别与定位方法, 通过提取二维图像和点云图像的多模态信息, 实现对物体的识别与定位。先利用彩色相机获取目标的二维图像信息, 通过轮廓检测与匹配处理进行轮廓识别, 再提取图像 SIFT 特征进行定位跟踪, 得到物体的位置信息; 同时采用 RGB-D 相机获取目标的三维点云图像信息, 经过预处理、欧式聚类分割、VFH 特征计算、KD 树搜索得到最佳模板, 进行点云图像的识别, 并经点云聚类配准获得物体方向信息。最后, 利用上述二维图像和点云图像处理所得物体信息, 完成对目标的识别与定位。通过机器人抓取实验对本文方法的效果进行了验证, 结果表明, 采用二维图像和点云图像的多模态信息进行处理, 能够有效对不同形状的目标物体进行识别与定位, 与仅采用二维或点云单模态图像信息的方法相比, 定位误差可减小 54.8%, 方向误差减少 50.8%, 具有较好的鲁棒性和准确性。

关键词: 二维图像; 点云图像; 多模态; 特征识别与定位; 机器人

中图分类号: TP391.4

文献标志码: A

引用格式: 魏玉锋, 梁冬泰, 梁丹, 等. 基于多模态信息的机器人视觉识别与定位研究[J]. 光电工程, 2018, 45(2): 170650

Visual identification and location algorithm for robot based on the multimodal information

Wei Yufeng, Liang Dongtai*, Liang Dan, Xing Shumin

School of Mechanical Engineering and Mechanics, Ningbo University, Ningbo, Zhejiang 315000, China

Abstract: To overcome the problem of a single image source, complex processing and inaccurate positioning, a visual identification and location algorithm based on multi-modal information is proposed, and the fusion processing is performed by extracting the multimodal information of the two-dimensional image and the point cloud image to realize object recognition and positioning. Firstly the target 2D image information is obtained by RGB camera. The contour is recognized through the contour detection and matching process. Then the image SIFT feature is extracted for location tracking and the position of the object is obtained. Meanwhile obtaining a point cloud image by RGB-D camera and the best model can be sorted through pre-processing, Euclidean cluster segmentation, computing VFH feature and KD-tree searching, identifying the point cloud image. Then the orientation is obtained by registering the point clouds. Finally, the two-dimensional images and point cloud image are used to process object information, complete the identification and positioning of the target. The effect of the method is verified by the robotic gripping

收稿日期: 2017-11-30; 收到修改稿日期: 2018-01-08

基金项目: 国家自然科学基金青年基金(51305214); 浙江省公益性技术应用研究计划(2017C31094); 宁波市自然科学基金(2017A610124)

作者简介: 魏玉锋(1992-), 男, 硕士, 主要从事机器人抓取的研究。E-mail: 18815282254@163.com

通信作者: 梁冬泰(1981-), 男, 博士, 副教授, 主要从事机器视觉, 图像处理, 机器人感知与操控的研究。

E-mail: liangdongtai@nbu.edu.cn

experiment. The result shows that the multi-modal information of two-dimensional image and point cloud image can be used to identify and locate different target objects. Compared with the processing method using only two-dimensional or point cloud single-mode image information, the positioning error can be reduced to 50%, the robustness and accuracy are better.

Keywords: 2D image; point cloud; multimodal; feature recognition and positioning; robot

Citation: Wei Y F, Liang D T, Liang D, et al. Visual identification and location algorithm for robot based on the multimodal information[J]. *Opto-Electronic Engineering*, 2018, 45(2): 170650

1 引言

21 世纪以来,随着机器人技术快速发展,机器人的应用越来越广泛。工业上,工人往往需要完成大量的搬运装配工作,使用机器人可以提高工厂的工作效率,降低工人的劳动强度。机器人能感知到物体的位置,实现抓取物体是完成这些任务的前提,而机器视觉是解决这个问题的有效方法之一,也是目前机器人领域的研究热点之一。

机器人通过机器视觉实现对物体的识别与跟踪,从而完成抓取工作。传统的检测识别跟踪技术有基于二维图像信息的,基于三维点云图像信息的和基于 QR-Code 等,这些方法已经得到广泛的运用。Collet^[1]等人设计了一种基于视觉的感知系统,它能够在杂乱场景中进行物体识别和姿态估计。同时,它还可以实时检测多个物体已经相对应的六自由度姿态。Munoz^[2]等人针对无纹理的物体识别定位,提出 RAPID-HOG 算法。该方法运用线性预测的方式估计出物体的 3D 姿态,利用方向梯度直方图(histogram of oriented gradient, HOG)特征增加系统的鲁棒性,具有位姿计算速度快、稳定性强等特点,很好地实现了无纹理物体的定位。Zhu^[3]等人提出了一种基于三维物体的图形信息来检测定位三维物体的方法,通过匹配模型边界的轮廓和超像素,对图像中的物体进行分割与定位,在室内室外都具有良好的识别效果。Lai^[4]等人提出一

种多层内核描述子算法,对 RGB-D 数据库中的图像提取出图像特征并匹配识别,得到了较好的识别效果。Rusu^[5]等人提出一种基于随机采样一致性的配准算法,运用 FPFH 特征描述子进行配准,对物体的识别定位具有良好的效果。Braun^[6]等人提出了一个 Pose-RCNN 模型,将 3D 数据用于区域卷积神经网络中,估计出物体的位姿并具有良好的识别定位效果。Pavlakos^[7]等人将语义关键点与具有变形物体模型的卷积神经网络结合,可以估计出物体的位姿,同时该方法对有纹理和无纹理的物体都适用,对变形或者遮挡的物体也具有良好的识别估计物体位姿的效果。

但现有的研究方法中大多数的图像信息来源单一,基本来源于一种类型的图像,随着场景的复杂化,传统的方法存在图像信息不全、识别误差大、准确率低等缺点。为解决这一问题,融合多个信息来对物体进行识别定位势在必行。

物体的位姿 (pose) 由位置 (position) 和方向 (orientation) 组成,位置表达物体距离参考坐标系的偏移量,方向代表物体相对参考坐标系的偏转角度。本文提出一种基于多模态信息的视觉识别与定位算法,结合二维图像定位识别与三维点云图像定位识别的方法,完成了对物体的识别与抓取,从而验证了融合多个信息对物体进行识别定位是可行的。算法的整体设计框架如图 1 所示,主要包括二维图像识别跟踪和三维图像识别跟踪。

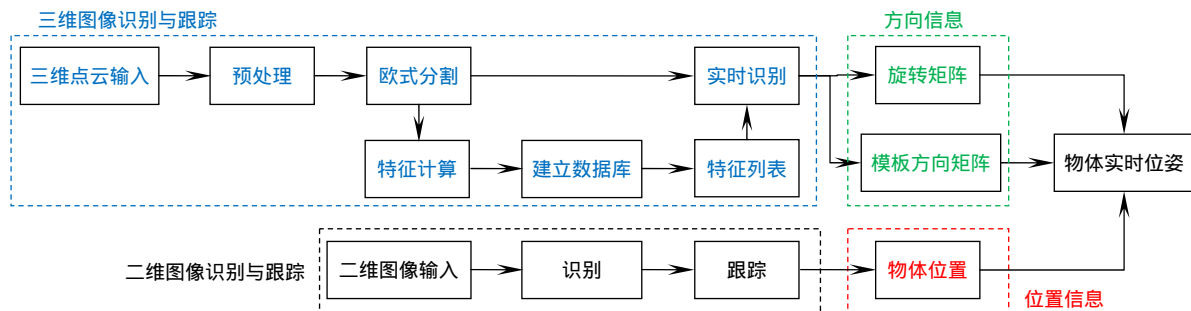


图 1 识别定位算法框架

Fig. 1 The architecture of identification and location algorithm

2 二维与三维图像识别跟踪算法

2.1 二维图像识别跟踪

二维图像识别跟踪是本算法的主要组成部分之一,包括物体识别与物体跟踪两部分,主要完成对物体的位置定位。

2.1.1 物体识别

在二维图像处理中,通过轮廓来识别物体是常用的方法之一。本文采用匹配二维图像中的物体轮廓与3D模型的轮廓,实现物体的识别。

首先使用彩色相机采集二维图像,然后再用Canny算子^[8]进行边缘检测处理。由于处理后的图像中包含许多轮廓线,轮廓线存在直线与弧线,因此本算法使用最短路径搜索法^[9]搜索相距最近的直线与弧线,形成闭合的轮廓线,相邻的闭合轮廓线形成简单的图形^[10]。例如立方体可以认为是三个矩形的组合,圆柱体可以认为是两个圆与两条平行线的组合。图形轮廓形成的流程如图2所示。

在进行Canny算子边缘检测时,假设 $f(x,y)$ 代表输入图像, $G(x,y)$ 代表高斯函数,则:

$$G(x,y) = \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (1)$$

然后对 G 和 f 做卷积运算,得到平滑度较好的图像 f_s :

$$f_s = G(x,y) * f(x,y) \quad (2)$$

使用一阶有限差分法计算偏导数,得到矩阵 P 和 Q :

$$P[x,y] \approx \frac{f_s(x,y+1) - f_s(x,y)}{2} + \frac{f_s(x+1,y+1) - f_s(x+1,y)}{2} \quad (3)$$

$$Q[x,y] \approx \frac{f_s(x,y) - f_s(x+1,y)}{2} + \frac{f_s(x,y+1) - f_s(x,y+1)}{2} \quad (4)$$

计算 P 与 Q 的幅值和方位角:

$$M(x,y) = \sqrt{P[x,y]^2 + Q[x,y]^2} \quad (5)$$

$$\theta(x,y) = \arctan\left(\frac{Q[x,y]}{P[x,y]}\right) \quad (6)$$

通过设定阈值 θ 的大小,获取不同尺度要求下的图像边缘轮廓。

2.1.2 物体跟踪

物体的跟踪主要分为边缘轮廓跟踪与表面纹理跟踪。本文采用边缘轮廓跟踪与表面纹理跟踪相结合的方式,具体实现过程如下:

1) 建立投影关系:首先,按2.1.1小节中所述方法,获取简单图形的轮廓。然后获取简单图形内的图像,称之为正投影。再将简单图形的轮廓与3D模型轮廓做匹配处理,获取与3D模型相对应的简单图形。最终,将对应的简单图形的图像再投影到与之相对应的3D模型面上,建立一种投影关系,其过程如图3所示。在不同的视角下,重复上述方法,直到3D模型每个面都与图像建立投影关系,最终将得到一个含有图像的3D模型^[11]。

2) 边缘轮廓跟踪:先通过彩色相机采集含有待测物体的二维图像,再使用2.1.1中的方法来匹配待测物体的轮廓,进而得到轮廓跟踪的结果。

3) 表面纹理跟踪:首先提取含有待测物体图像与3D模型图像的SIFT特征,再进行匹配以得到表面纹理跟踪的结果。

4) 轮廓与纹理跟踪融合:将轮廓跟踪和纹理的结果相融合,取两者的加权平均值作为最终的跟踪结果。

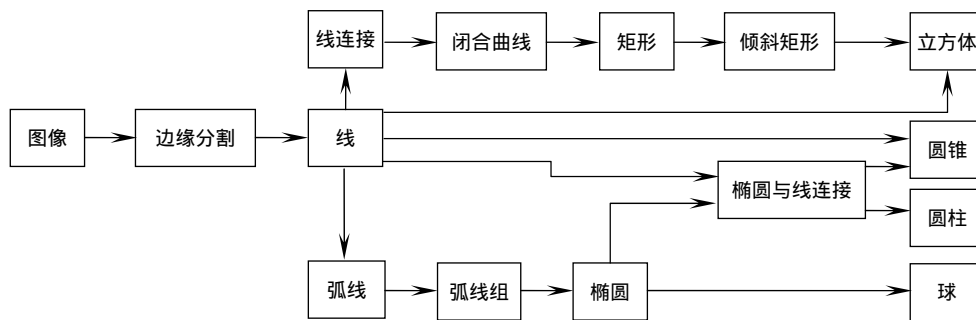


图2 轮廓搜索流程图

Fig. 2 The flow chart of searching contour

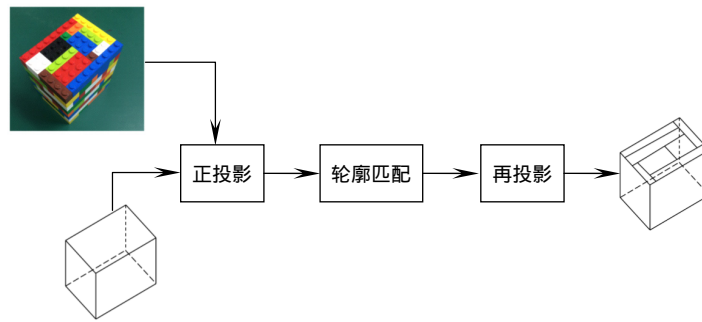


图3 建立投影关系

Fig. 3 Build a projection relationship

在 SIFT 特征点检测时，会检测到错误的特征点，这些点会影响之后的匹配精度。为了排除错误的 SIFT 特征点，提高物体的定位精度，定义了一个阈值 ρ ，如式(7)所示：

$$\rho(O|m, f_t) = \frac{n_{inlier}}{n_{det}} \quad (7)$$

其中： O 代表待测物体， m 代表待测物体的模型， f_t 代表当前帧的图像。在待测物体的图像像素内的所有 SIFT 点用 n_{det} 表示，而符合匹配条件的 SIFT 点数用 n_{inlier} 表示，两者的比值用 ρ 表示。比值 ρ 越大，待测物体的定位可信度越高，反之则可信度越低。实验中，给 ρ 设定一个值，当超过该值时表示定位可信；当低于该值时，略过该帧图像，进入下一帧图像重新定位待测物体。通过轮廓匹配与 SIFT 特征匹配，获取位姿信息。本算法取其中的位置信息用于最终识别定位，从而引导机械臂到达待测物体的相应位置，略去获得的方向信息。

2.2 三维图像识别跟踪

三维图像跟踪识别是算法的另一个主要组成部分，包括点云图像获取、聚类分割、计算 VFH 视点特征直方图、生成特征列表和实时识别，主要完成对物体的旋转方向估计。

2.2.1 获取点云图像

在进行三维图像识别跟踪时，首先需获取包含待识别物体的点云图像，本文采用 RGB-D 相机获取当前实验环境中的点云图像信息。由于是通过三维点云的方式处理图像，所以可以不考虑物体的形状与颜色，但必须将物体置于一个平台上。

一幅点云数据除了包含待测物体的点云聚类，还包括其他物体的点云聚类、背景聚类、噪点等。原始点云数据较大，会影响之后的计算，所以需要点对云数据进行预处理。原始点云中存在着大量的噪点，会

影响待测物体的点云聚类分割，通过统计学滤波方法可以进行降噪处理。此外，通过直通滤波器可以除去原始点云中的背景点云，降低分割待测物体点云的难度。

运用 RGB-D 相机采集到的点云数量在几十万个，一些精度高的相机获取的点云可高达几百万个^[12]。虽然经过前期的去噪滤波处理已经降低了点云的数量，但是剩余的点云数量依然很庞大，会影响后期分割的速度，需要进行下采样处理，减小点云数量，提高后期的分割速度。将点云分为边长为 L 的立方体栅格，该栅格内的点都用该栅格内点集的重心来表示，这样就可以大大减小点云的数量。图 4(a)为 RGB-D 相机获取的原始点云图像，图 4(b)为预处理后的点云图像。

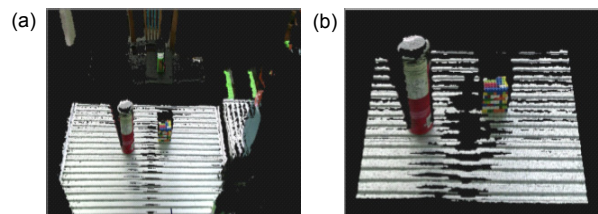


图4 包含目标物体的点云图。(a) 原始；(b) 预处理后
Fig. 4 Point cloud containing the target object. (a) The original image; (b) The image after pretreatment

2.2.2 聚类分割

在 PCL(point cloud library)^[13]中有欧式聚类分割，基于随机采样一致性分割，区域生长分割，最小分割与基于颜色和法线的分割等。由于本实验是在平台上实现抓取，所以采用欧式聚类分割就可以达到很好的效果。本文进行欧式聚类分割算法的基本流程如下：

- 1) 对输入的场景点云 P 建立三维的 KD 树，并建立一个空的聚类 $\{O_i\}$ 和点队列 Q 。
- 2) 对于每个点 $p_i \in P$ ，执行下面的步骤：

- a. 将点 p_i 放到当前查询队列 Q 中。
- b. 对每个点 $p_i \in Q$, 搜索点 p_i 的邻近点集 P_i^k , 使其满足 :

$$\|p_i^k - p_i\| \leq d, p_i^k \in P_i^k. \quad (8)$$

如果每个邻近点 p_i^k 没有被处理过, 就将邻近点 p_i^k 添加到队列 Q 中。循环处理该过程直到队列 Q 中的所有内点被处理过。

- c. 把队列 Q 作为聚类 O_i 推入到聚类列表 $\{O_i\}$ 中, 并清空队列 Q 。

3) 不断重复步骤 2), 直至输入点云 P 中的所有内点被处理。

将预处理后的点云经欧式聚类分割, 可以移除平台平面, 得到单个物体的点云, 结果如图 5 所示。

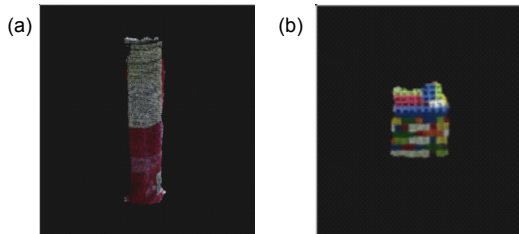


图 5 分割后的点云图。(a) 其他物体; (b) 目标物体
Fig. 5 The image of point cloud after segmentation. (a) The image of other object; (b) The image of target object

2.2.3 计算 VFH 视点特征直方图

在聚类分割处理后, 需要计算视点特征直方图 (viewpoint feature histogram, VFH)。VFH 视点特征直方图是基于 FPFH^[14] 的一种特征描述子。假设有一个点云集 $\{O\}$, P_k 代表点云集中某一点 p 的 k 个邻域内的点集, P_a 和 P_b 为其中两点, n_a 和 n_b 为这两个点的法线, 为了计算他们的偏差, 在其中一个点上定义一个固定的局部坐标系 uvw , 其中:

$$u = n_a, \quad (9)$$

$$v = u \times \frac{P_b - P_a}{\|P_b - P_a\|_2}, \quad (10)$$

那么法线 n_a 和 n_b 之间的偏差则可用一组角度来表示:

$$\alpha = v \cdot n_b, \quad (11)$$

$$\beta = u \cdot \frac{P_b - P_a}{d}, \quad (12)$$

$$\gamma = \arctan(w \cdot n_b \cdot u \cdot n_b), \quad (13)$$

其中: d 表示 P_a 和 P_b 之间的欧式距离:

$$d = \|P_b - P_a\|_2. \quad (14)$$

称计算 α, β, γ 的结果为简化的点特征直方图 SPFH (用 H_{SPFH} 表示), 再重新确定每个点的 k 邻域,

使用邻近的 SPFH 值来计算 P_k 的最终直方图, 称之为 FPFH (用 H_{FPFH} 表示) 特征直方图, 其计算式:

$$H_{FPFH}(P_k) = H_{SPFH}(P_k) + \frac{1}{k} \sum_{i=1}^k \frac{1}{w_k} \cdot H_{SPFH}(P_k), \quad (15)$$

其中: 权重 w_k 表示点 p 与近邻点 p_k 之间的距离。

假设在计算 FPFH 特征直方图时, 考虑到视点特征, 即在原来计算法线 n_a 和 n_b 的一组角度 α, β, γ 的基础上, 再加入计算法线 n_a 与视点 C 的法线 n_c 之间的一组角度 $\alpha_1, \beta_1, \gamma_1$, 则构成了新的特征, 称之为 VFH 视点特征直方图^[15], 整个公式的计算流程如图 6 所示。同一视角下, 物体的 VFH 特征不会随着缩放而改变; 不同视角下, 同一物体的特征会有所改变, 但是变化不大。在 VFH 中有一个视点变量, 可以根据当前视点下的特征, 区别不同的物体, 或者同一物体的不同位姿。

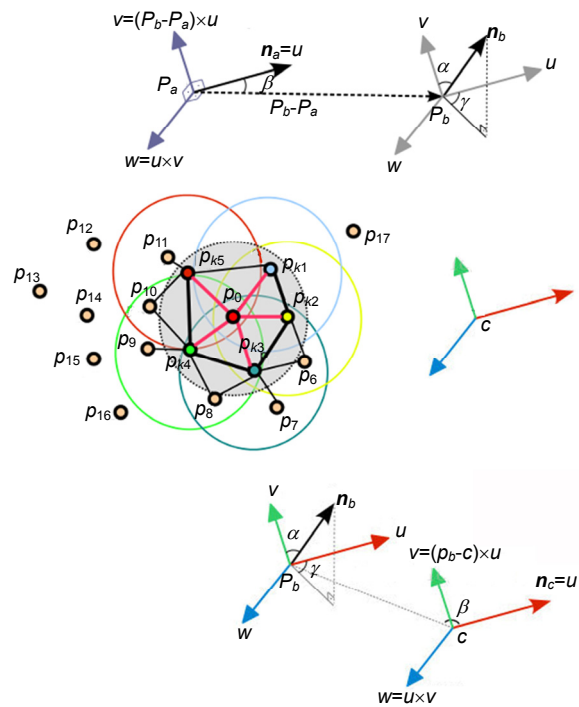


图 6 VFH 特征计算^[16]
Fig. 6 Computing VFH feature^[16]

将待测物体置于水平面上, 用 RGB-D 相机获取不同姿态待测物体的三维点云图像, 经预处理与欧式分割后, 计算待测物体三维点云图像的 VFH 特征信息, 同时记录下待测物体相对于相机的位姿, 建立模板库。

2.2.4 实时识别算法的实现

三维点云的实时识别算法如图 7 所示。首先由

RGB-D 相机获取三维点云图像, 然后经预处理、欧式聚类分割、VFH 特征计算得到每个聚类的特征信息, 其中预处理、欧式聚类分割和 VFH 特征计算等操作与 2.1、2.2 和 2.3 节所述方法一致。在计算得到每个聚类的 VFH 特征信息后, 需要与模板的 VFH 特征相比较, 得到相似度最高的模板。假设聚类集为 $\{O_i\}$, O_i 表示第 i 个聚类, 计算得到它的视点特征直方图 H_i , H_i 是一个由 308 个浮点数组成的阵列。假设 f_n 为落入第 n 个浮点数上的个数, 而落在所有浮点数上的个数之和 $f_1+f_2+\dots+f_n$ 等于 O_i 中点的个数 m , p_n 为落入第 n 个浮点数上的概率, 那么 mp_n 为落在第 n 个浮点数上的理论频数, 聚类 O_i 的卡式分布情况:

$$\chi^2 = \sum_{n=1}^m \frac{(f_n - mp_n)^2}{mp_n} \quad (16)$$

同时计算模板点云 Q_j 的卡式分布 χ_j^2 , 其中 j 表示模板的序号。那么聚类 O_i 与第 j 个模板点云 Q_j 之间的相似度:

$$d(O_i, Q_j) = \sqrt{\chi^2(O_i) + \chi^2(Q_j)} \quad (17)$$

当 d 小于设定的阈值时, 认为聚类 O_i 与模板点云

Q_j 相似; 当 d 大于设定的阈值时, 则认为不相似。如图 8 所示, 从左下角到右上角, 相似度 d 越来越大, 绿色表示 d 的值小于设定的阈值, 红色代表 d 大于设定的阈值。若 d 的取值都大于阈值, 则认为该点云聚类 O_i 不是待测物体点云, 并输入聚类 O_{i+1} 继续判断, 直至所有聚类判断完, 若 d 取值存在小于阈值的情况, 则取 d 最小的模板点云进入下一步。

取当前的点云聚类 O_i 与模板库中最相似的点云聚类 Q_j , 分别计算它们的表面法线和 FPFH 特征, 采用 SAC_IA 贪婪粗配准和 ICP 精配准, 最终得到基于点云 Q_j 的平移旋转矩阵 M_1 。同时, 该算法也会输出模板的位姿矩阵 M_2 , 当前点云的位姿矩阵为

$$M = M_1 \cdot M_2$$

该矩阵 M 包含了待测物体的位置信息和方向信息。

通过上述的配准方法, 获得当前物体的位姿矩阵 M , 本算法取其中的方向信息作为最终识别定位的方向信息, 用于引导机械臂末端执行器旋转到正确的姿态, 略去获得的位置信息。

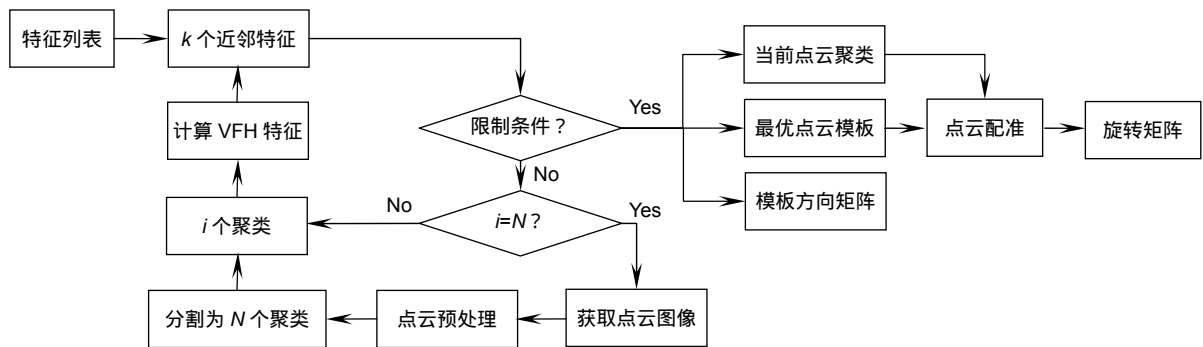


图 7 实时识别算法

Fig. 7 Real-time identification algorithm

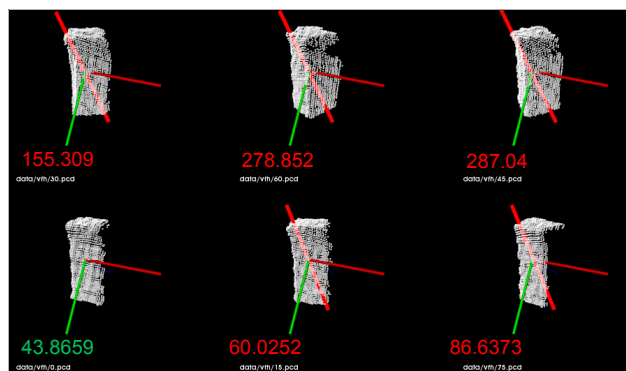


图 8 近邻搜索结果图

Fig. 8 The result of searching nearest neighbors

3 实验结果与分析

3.1 实验配置

为了验证算法的准确性与可靠性，搭建基于 Universal Robot 5 机器臂的实验平台(如图 9 所示)，进行定位抓取实验。平台主要包括 Kinect2 相机，Microsoft USB 摄像头，Universal Robot 5 机器臂和 Robotiq-85-finger 电爪。实验中使用 Kinect2 相机采集工作区的 RGB 图像与 Depth 图像，其中 RGB 图像的分辨率为 1920 pixels×1080 pixels，Depth 图像的分辨率为 512 pixels×424 pixels，根据 RGB 图与 Depth 图像生成三维点云图；使用 Microsoft USB 摄像头采集待测物体的 RGB 图，分辨率为 640 pixels×480 pixels。使用 Ubuntu14.04 下的 ROS indigo 版本为开发环境，使用 QT5.7 作为开发平台，以 Open CV 2.4.9 实现二维图像处理算法，以 PCL 1.7 实现三维图像处理算法。Universal Robot 5 机器臂的工作范围可以覆盖到实验台的任意位置，最大载重为 5 kg；Robotiq-85-finger 电爪实现抓取物体的功能，电爪的最大工作范围为 85 mm；本实验的实验对象为一块规格为 72 mm×56 mm×77 mm 的乐高积木块。

3.2 工具坐标定位

为了使机器人能够准确运动到算法所得物体位姿，需要确定机器人末端执行器坐标与物体坐标间关系^[17]。假设{B}表示机器人基座坐标系，{T}表示工具坐标系，{U}表示 USB 相机坐标系，{K}表示 Kinect2 相机坐标系，{C}表示待测物体的坐标系，{W}表示世界坐标系，那么它们之间的相对变换关系可用 T 表示，例如机器人基座坐标系相对于工具坐标系的位姿可用

T_T^B 表示。通过二维图像的识别与跟踪可以得到待测物体相对 USB 相机的平移矩阵 t ：

$$t = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (18)$$

通过三维图像识别定位可以得到待测物体相对于 Kinect2 相机的旋转矩阵 R ：

$$R = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (19)$$

那么待测物体相对于工具坐标的平移矩阵 t_1 可以用以下公式表达：

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} t_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} T_W^U T_B^W T_T^B \quad (20)$$

待测物体相对工具坐标的旋转矩阵 R_1 可以用以下公式计算：

$$\begin{bmatrix} R_1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} T_W^K T_B^W T_T^B \quad (21)$$

最终待测物体相对工具坐标系的位姿 T_T^C 可以表达为

$$T_T^C = \begin{bmatrix} R_1 & t_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22)$$

在 ROS 中建立待测物体的坐标系(cube)与各个坐标系之间的关系图，如图 10 所示。待测物体的坐标系(cube)与工具坐标系(tool0)之间的关系 T_T^C 已由上述推

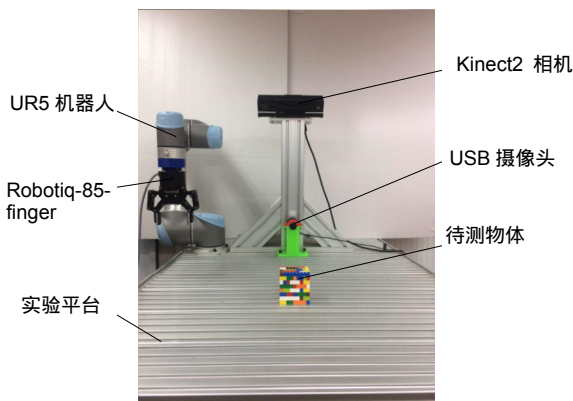


图 9 实验平台
Fig. 9 The experiment platform

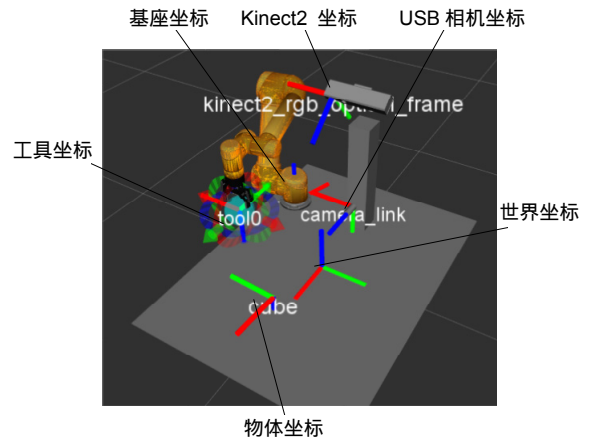


图 10 MoveIt 中的实验平台
Fig. 10 The experiment platform in MoveIt

导式(22)得到, 使用 MoveIt 控制 Universal Robot 5 机器人臂运动, 使得工具坐标系与待测物体的坐标系重合, 到达待测物体的位置, 闭合电爪, 完成对待测物体的抓取。

3.3 检测识别实验与分析

根据以上的实验配置进行物体的抓取实验, 记录下定位的数据并进行分析。为了验证本实验算法的可靠性, 检测算法位移的相对定位精度, 本实验将采用如图 11 所示的方式进行测量, 测量平台为直线导轨, 定位精度为±0.01 mm。实验所测位置坐标均以世界坐标为基坐标, 世界坐标如图 10 所示, 红色轴指向为 X 轴正方向, 绿色轴指向为 Y 轴正方向, 蓝色轴指向为 Z 轴正方向。将直线运动导轨平行于世界坐标的 X 轴放置, 将待测物体放置在运动平台上, 用挡块固定住

待测物体, 每次驱动导轨运动 20 mm, 同时记录下算法所得位置数据并计算两者的误差, 如表 1 所示。

表 1 中实际物体坐标为导轨运动位置的坐标, 计算坐标为算法的计算结果, 定位误差为前后两次运动结果之差。其中 X 轴方向的误差为物体在 X 轴方向计算所得运动距离与实际运动距离之差。分析表 1 中的数据可知, 当物体在 X 轴方向平移时, X 轴上相对误差为 2 mm 以内; Y 轴方向也存在位移但误差不大, 基本在 1 mm 以内; 在 Z 轴上也存在位移, 误差为 2 mm 以内。一般而言, 定位误差相对于物体体积在 5% 以内都可接受, 实验所得结果在 3.5% 以内, 表明相对误差较小, 对成功抓取物体无明显影响。

将直线运动导轨平行于世界坐标 Y 轴, 用同样方法测量 Y 轴方向结果并记录, 如表 2 所示。分析表 2 可知, 在 Y 轴方向运动时, Y 轴相对误差基本在 2 mm 以内, 但是在 X 轴上存在 5 mm 以内的误差, 且误差数值浮动较大; Z 轴方向误差基本在 2 mm 以内。综合表 1 与表 2 可知, 算法的定位结果在前后两次移动时均存在着误差, 但是误差较小, 都在 5% 以内, 不会对成功抓取物体造成影响。

为了验证算法在旋转方向的可靠性, 本文对旋转方向进行误差测量, 测量方式如图 12 所示。待测物体一端点与半圆中心重合, 一边紧靠 0 刻度线, 将物体旋转一定角度后测量并记录下算法计算的旋转角度, 多次随机旋转一定角度, 选择其中 6 组记录, 如表 3 所示。分析表 3 可知, 物体在旋转方向的误差基本在 5° 左右, 且计算结果在真实值附近浮动, 未出现仅偏向真实值一侧的情况。

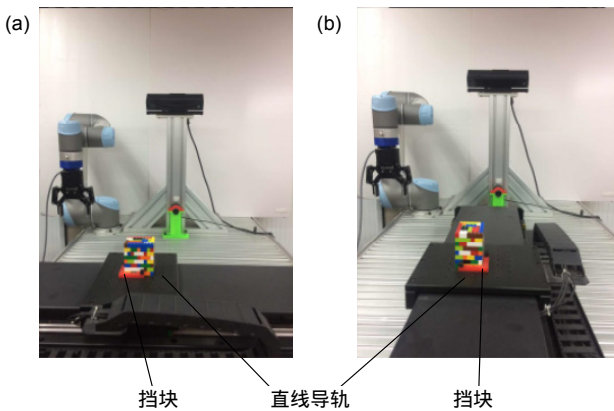


图 11 测量装置

Fig. 11 The measurement equipment

表 1 X 轴方向结果

Table 1 The positioning results of X-axis

| 序号 | 实际物理坐标 | 计算物理坐标 | | | 定位误差 | | |
|----|--------|--------------------|--------------------|--------------------|----------------------------------------------|----------------------------------------|----------------------------------------|
| | X/mm | X _n /mm | Y _n /mm | Z _n /mm | (X _n -X _{n-1})-0.02 /mm | (Y _n -Y _{n-1})/mm | (Z _n -Z _{n-1})/mm |
| 1 | 50.0 | 243.0 | -10.6 | 163.0 | | | |
| 2 | 70.0 | 265.5 | -10.3 | 163.8 | 2.5 | 0.3 | 0.8 |
| 3 | 90.0 | 284.0 | -9.9 | 165.6 | -1.5 | 0.4 | 1.8 |
| 4 | 110.0 | 302.0 | -8.7 | 166.2 | 2.0 | 1.2 | 0.6 |
| 5 | 130.0 | 322.5 | -8.5 | 166.6 | 0.5 | -0.2 | 0.4 |
| 6 | 150.0 | 343.0 | -8.3 | 167.9 | 0.5 | 0.2 | 1.3 |
| 7 | 170.0 | 363.6 | -7.8 | 168.9 | 0.6 | 0.5 | 1.0 |
| 8 | 190.0 | 382.0 | -7.0 | 169.0 | -1.4 | 0.8 | 0.1 |
| 9 | 210.0 | 404.0 | -6.4 | 171.0 | 2.0 | 0.6 | 2.0 |
| 10 | 230.0 | 422.0 | -6.0 | 172.0 | 2.0 | 0.4 | 1.0 |
| 11 | 250.0 | 442.0 | -6.0 | 172.4 | 0.0 | 0.0 | 0.4 |

表 2 Y 轴方向结果

Table 2 The positioning results of Y-axis

| 序号 | 实际物理坐标 | 计算物理坐标 | | | 定位误差 | | |
|----|--------|--------------------|--------------------|--------------------|----------------------------------------|---------------------------------------------|----------------------------------------|
| | X/mm | X _n /mm | Y _n /mm | Z _n /mm | (X _n -X _{n-1})/mm | (Y _n -Y _{n-1})-0.02/mm | (Z _n -Z _{n-1})/mm |
| 1 | 0 | 397.8 | -174.6 | 174.8 | | | |
| 2 | 20.0 | 398.0 | -154.9 | 174.2 | 0.2 | -0.3 | -0.6 |
| 3 | 40.0 | 396.9 | -134.5 | 174.2 | -1.1 | 0.4 | 0.0 |
| 4 | 60.0 | 397.7 | -110.6 | 174.7 | 0.8 | 3.9 | 0.5 |
| 5 | 80.0 | 401.0 | -91.9 | 174.3 | 3.3 | -1.5 | -0.4 |
| 6 | 100.0 | 396.0 | -70.4 | 172.5 | -5.0 | 1.5 | -1.8 |
| 7 | 120.0 | 400.5 | -50.2 | 173.2 | 4.5 | 0.2 | 0.7 |
| 8 | 140.0 | 400.6 | -28.6 | 174.0 | 0.1 | 1.6 | 0.8 |
| 9 | 160.0 | 400.1 | -9.3 | 172.1 | -0.5 | -0.7 | -1.9 |
| 10 | 180.0 | 400.3 | 12.1 | 172.7 | 0.2 | 1.4 | 0.6 |
| 11 | 200.0 | 400.9 | 32.5 | 172.6 | 0.6 | 0.4 | -0.1 |

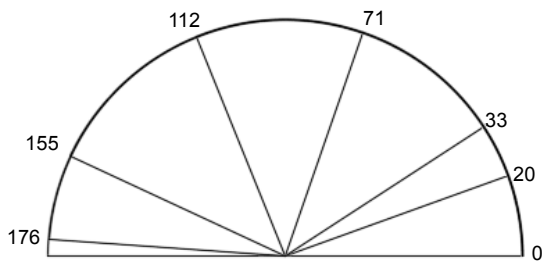


图 12 旋转精度测量图

Fig. 12 The measuring figure of rotation accuracy

表 3 旋转结果

Table 3 The error result of rotation

| 序号 | 真实值/(°) | 计算值/(°) | 误差/(°) |
|----|---------|---------|--------|
| 1 | 20.0 | 16.8 | 3.8 |
| 2 | 33.0 | 38.6 | 5.6 |
| 3 | 71.0 | 66.4 | 4.6 |
| 4 | 112.0 | 116.5 | 4.5 |
| 5 | 155.0 | 160.2 | 5.2 |
| 6 | 176.0 | 171.4 | 4.6 |

本文算法将与传统的单独采用二维图像信息或者三维点云信息的定位方法做对比。将二维图像跟踪定位结果与三维点云图像的识别定位结果，同样按上述方式进行测量并记录误差，如表 4 所示。利用表 4 中的结果，将多模态信息方向误差与二维信息方向误差作差，并与二维信息方向误差相比，可得方向误差减少率；同理，多模态信息位置误差与三维点云信息位置误差比较，可得位置误差减少率。结果表明，与仅采用二维或点云单模态图像信息的处理方法相比，本文所提出的方法定位误差可减小 54.8%，方向误差减少 50.8%。在相同的实验平台与环境下，仅采用二维图像的方法所耗时间为 0.625 s，仅采用三维图像的方法所耗时间为 0.736 s，同时使用两者所耗时间为 0.78 s，与单独采用二维图像相比，运行时间增加了 25%；与单独采用三维图像相比，运行时间增加了 5.9%。同时验证了本算法采用二维图像位置信息和三维图像方

向信息的方法是可行的。

本算法对遮挡物体的识别定位做了实验，实验结果如图 13 所示。图 13(a)为原始的场景图，物体被黑色物体部分遮挡，图 13(b)为识别定位效果图，从效果图中可以验证本算法对部分遮挡的物体也具有识别定位的效果。

为了验证算法对其他物体也具有同样的识别定位能力，选用圆柱体与三棱柱进行实验，并对实验结果进行记录，结果如表 5 所示，识别定位效果如图 14 所示。圆柱体的底面直径为 100.2 mm，高为 21.5 mm；三棱柱底面为边长 84.1 mm 的等边直角三角形，高为 31.1 mm。实验结果显示，本算法能有效识别不同形状的物体，且误差相差很小。

综合上述结果，采用二维图像和点云图像的多模态信息进行处理，能够有效对不同形状的目标物体进行识别与定位，具有较好的鲁棒性和准确性。

表 4 不同算法的误差结果
Table 4 The error results of different algorithms

| 序号 | 二维图像+三维图像 | | 二维图像 | | 三维图像 | |
|----|-----------|--------|-------|--------|-------|--------|
| | 位置/mm | 方向/(°) | 位置/mm | 方向/(°) | 位置/mm | 方向/(°) |
| 1 | 1.2 | 5.2 | 1.3 | 10.1 | 2.1 | 5.3 |
| 2 | 0.8 | 4.5 | 1.5 | 9.8 | 1.9 | 5.1 |
| 3 | 1.1 | 4.4 | 1.2 | 8.8 | 2.2 | 5.0 |
| 4 | 1.0 | 4.8 | 1.5 | 9.7 | 2.2 | 4.8 |
| 5 | 0.6 | 5.1 | 1.0 | 10.2 | 2.0 | 4.9 |
| 均值 | 0.94 | 4.8 | 1.3 | 9.72 | 2.08 | 5.02 |

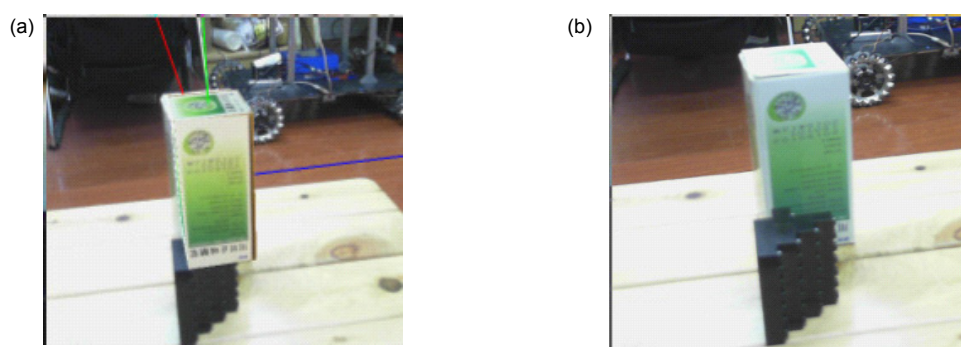


图 13 物体遮挡识别效果。(a) 原始图；(b) 识别后
Fig. 13 The image of object occlusion recognition effect. (a) The original image; (b) After recognizing

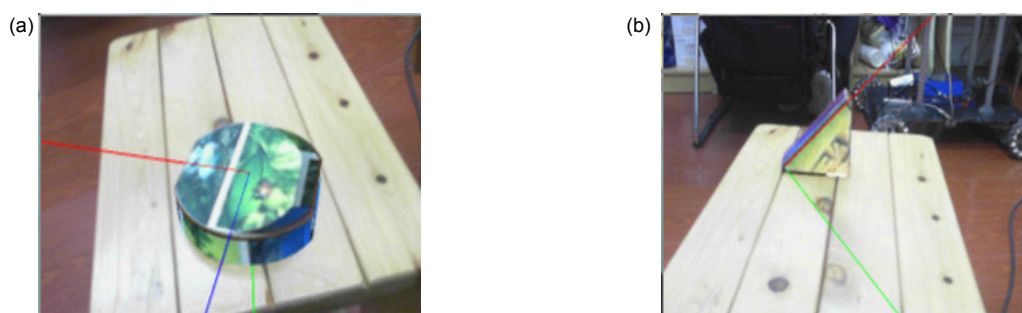


图 14 不同物体的识别效果。(a) 圆柱体；(b) 三棱柱
Fig. 14 The recognition effect image of different objects. (a) Cylinder; (b) Triangular prism

表 5 不同形状物体的误差结果
Table 5 The error results of different objects

| | 位置/mm | 方向/(°) |
|-------|-------|--------|
| 乐高积木块 | 0.94 | 4.8 |
| 圆柱体 | 0.86 | 4.9 |
| 三棱柱 | 1.11 | 5.1 |

4 结 论

机器人识别定位方法涉及机器视觉、机器人操控技术、运动轨迹规划和传感器感知技术等,在机器人应用领域有着十分重要的作用。本文采用二维图像信息处理,进行位置定位,同时处理三维点云信息进行方向定位,最终利用二维图像和三维点云图像处理所得多模态物体信息,完成对目标的识别与定位。实验结果表明,采用二维图像和点云图像的多模态信息进行处理,能够有效对目标物体进行识别定位,与仅采用二维或点云图像的单模态信息处理方法相比,在运行时间增加很小的情况下,定位误差更小,具有良好的准确性与鲁棒性。本文采用二维图像与三维点云图像的多模态信息实现识别定位,该方法同样适用于其他不同形状的目标物体。

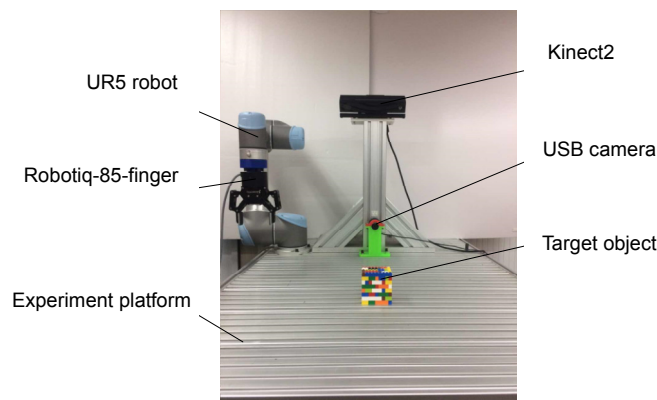
参考文献

- [1] Collet A, Srinivasa S S. Efficient multi-view object recognition and full pose estimation[C]//*Proceeding of 2010 IEEE International Conference on Robotics and Automation*, 2010: 2050–2055.
- [2] Munoz E, Konishi Y, Murino V, et al. Fast 6D pose estimation for texture-less objects from a single RGB image[C]//*Proceeding of 2016 IEEE International Conference on Robotics and Automation*, 2016: 5623–5630.
- [3] Zhu M L, Derpanis K G, Yang Y F, et al. Single Image 3D object detection and pose estimation for grasping[C]//*Proceeding of 2014 IEEE International Conference on Robotics and Automation*, 2014: 3936–3943.
- [4] Bo L F, Lai K, Ren X F, et al. Object recognition with hierarchical kernel descriptors[C]//*Proceedings of 2011 IEEE Conference on Computer Vision and Pattern Recognition*, 2011: 1729–1736.
- [5] Rusu R B, Blodow N, Beetz M. Fast point feature histograms (FPFH) for 3D registration[C]//*Proceeding of 2009 IEEE International Conference on Robotics and Automation*, 2009: 3212–3217.
- [6] Braun M, Rao Q, Wang Y K, et al. Pose-RCNN: Joint object detection and pose estimation using 3D object proposals[C]//*Proceeding of the 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016: 1546–1551.
- [7] Pavliakos G, Zhou X W, Chan A, et al. 6-DoF object pose from semantic keypoints[C]//*Proceeding of 2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017: 2011–2018.
- [8] Duan J, Gao X. Adaptive statistical filtering double thresholds based on improved canny operator edge detection algorithm[J]. *Laser Journal*, 2015, **36**(1): 10–12.
段军, 高翔. 基于统计滤波的自适应双阈值改进 canny 算子边缘检测算法[J]. *激光杂志*, 2015, **36**(1): 10–12.
- [9] Sánchez-Torrubia M G, Torres-Blanc C, López-Martínez M A. Pathfinder: A visualization eMathTeacher for actively learning Dijkstra's algorithm[J]. *Electronic Notes in Theoretical Computer Science*, 2009, **224**: 151–158.
- [10] Richtsfeld A, Vincze M. Basic object shape detection and tracking using perceptual organization[C]//*Proceeding of 2009 IEEE International Conference on Advanced Robotics*, 2009: 1–6.
- [11] Mörwald T, Prankl J, Richtsfeld A, et al. BLORT - the blocks world robotic vision toolbox[Z]. 2010: 1–8.
- [12] DAI J L. A research on preprocessing algorithms of mass point cloud[D]. Hangzhou: Zhejiang University, 2006.
戴静兰. 海量点云预处理算法研究[D]. 杭州: 浙江大学, 2006.
- [13] Rusu R B, Cousins S. 3D is here: Point Cloud Library (PCL)[C]//*Proceeding of 2011 IEEE International Conference on Robotics and Automation*, 2011: 1–4.
- [14] Zhao T, Li H, Cai Q, et al. Point Cloud Segmentation Based on FPFH Features[C]// *Proceedings of 2016 Chinese Intelligent Systems Conference*. Singapore, Springer, 2016.
- [15] Rusu R B, Brdski G, Thibaux R, et al. Fast 3D recognition and pose using the viewpoint feature histogram[C]//*Proceeding of 2010 IEEE International Conference on Intelligent Robots and Systems*, 2010: 2155–2162.
- [16] Zhu D H. *Point cloud library PCL*[M]. Beijing: Beijing University of Aeronautics and Astronautics Press, 2012: 338–355.
朱德海. 点云库 PCL 学习教程[M]. 北京: 北京航空航天大学出版社, 2012: 338–355.
- [17] Huang F S, Qin Y M, Ren Y S. The image recognition method on round bales robot labeling system[J]. *Opto-Electronic Engineering*, 2016, **43**(12): 168–174.
黄凤山, 秦亚敏, 任玉松. 成捆圆钢机器人贴标系统图像识别方法[J]. *光电工程*, 2016, **43**(12): 168–174.

Visual identification and location algorithm for robot based on the multimodal information

Wei Yufeng, Liang Dongtai*, Liang Dan, Xing Shumin

School of Mechanical Engineering and Mechanics, Ningbo University, Ningbo, Zhejiang 315000, China



The experiment platform

Overview: In recent years, various kinds of robots have got studied and used with the development of the robotics technology. In industrial production, workers often need to complete a large number of handling and assembly work. Using robots instead of the workers to complete these tasks would help to increase the efficiency of factories and reduce the labor intensity. Sensing the position of objects is one of the key problems for the robots to complete the task of picking objects. Machine vision is an effective methods to solve this problem, and it is also one of the research hotspots in the robot field at present. Traditional methods are based on two-dimensional images. Because two-dimensional images can not meet the demand, researchers extend their vision to three-dimensional images. Recent years, neural networks have been applied comprehensively, and machine vision has developed rapidly. However, for the traditional research methods, most of the sources of image information are single and basically come from one type of image. As the scene is always rather complicated, the traditional methods are usually faced with the problems such as incomplete image information, large recognition error and low accuracy.

To overcome these problems, a visual identification and location algorithm based on multi-modal information is proposed, and the fusion process is performed by extracting the multimodal information of the two-dimensional image and the point cloud image to realize object recognition and positioning. Firstly, the target 2D image information is obtained by RGB camera. The contour is recognized through the contour detection and matching process. Then, the image SIFT feature is extracted for location tracking and the position of object is obtained. Meanwhile, in order to identify the point cloud image, a point cloud image is captured by RGB-D camera and the best model can be sorted through pre-processing, Euclidean cluster segmentation, computing VFH feature and KD-tree searching. Thus, the orientation can be obtained by registering the point clouds. Finally, use the two-dimensional images and point cloud image to process object information, and to complete the identification and positioning of the target. The effect of the method is verified by the robotic gripping experiment. The results show that the multi-modal information of two-dimensional image and point cloud images can be used to identify and locate different target objects. Compared with the processing method using only two-dimensional or point cloud single-mode image information, the positioning error can be reduced to 50%, and the robustness and accuracy can be improved effectively.

Citation: Wei Y F, Liang D T, Liang D, *et al.* Visual identification and location algorithm for robot based on the multimodal information[J]. *Opto-Electronic Engineering*, 2018, 45(2): 170650

Supported by National Natural Science Foundation of China (51305214), the Public Technology Application Project of Zhejiang (2017C31094) and the Natural Science Foundation of Ningbo (2017A610124)

* E-mail: liangdongtai@nbu.edu.cn