

基于机器视觉的目标定位与机器人规划系统研究

杨三永, 曾碧

(广东工业大学 计算机学院, 广州 510000)

摘要: 为完成机械臂在非特定复杂背景环境下的自主抓取, 通过设计 RGB-D 相机对场景内的物体进行实时检测, 采用基于深度学习的目标检测定位方法, 并对相机—机械臂—目标物体的三维标定模型进行研究; 将物体的三维坐标信息通过 ROS 话题机制发送给机械臂, 并通过 moveIT 编程规划抓取规划; 通过设计一套基于 ROS 的视觉检测和机械臂抓取系统, 将计算机视觉检测技术以及机械臂运动规划抓取应用在机器人操作系统 ROS 平台上; 实验结果表明, 该系统可以实时高效地操作机器人来完成指定的控制作业, 提高了系统对环境的适应能力, 该系统具有抓取准确、物体识别准确率高的特点, 解决了传统机械臂操控中的不足。

关键词: 目标检测; ROS; 深度学习; 运动规划

Research on Target Positioning and Robot Planning System Based on Machine Vision

Yang Sanyong, Zeng Bi

(School of Computers, Guangdong University of Technology, Guangzhou 510000, China)

Abstract: In order to complete the autonomous capture of the manipulator in a non-specific complex background environment, the object in the scene is detected in real time by designing the RGB-D camera, and the target detection and positioning method based on deep learning is adopted. The three-dimensional calibration model is studied, and the computer vision detection technology and the robot arm motion planning are applied to the robot operating system ROS platform. The experimental results show that the system can operate the robot in real time and efficiently to complete the specified control operations, and improve the system's adaptability to the environment. The system has the characteristics of accurate grabbing and high object recognition accuracy, which solves the traditional robot arm control insufficient.

Keywords: target detection; ROS; deep learning; motion planning

0 引言

近年来, 随着人工智能的发展, 特别是深度学习技术对物体识别准确率的提高, 机器人通过视觉获取外部环境的感知处理各种任务的能力也在逐步提高。机器人得到任务, 如何快速而准确地通过视觉从环境中提取目标、识别物体, 通过“手”和“眼”的协调合作规划机械手的运动是目前机器人研究的热点之一^[1]。而随着机器人功能的日趋完善, 为机器人编写软件的工作却越来越复杂, 机器人操作系统 (Robot Operating System, ROS) 是一种用于机器人的次级操作系统, 它提供了一种分布式处理框架, 针对传统机器人控制系统面对特定对象时, 系统设计工作量大、可移植性差的问题, ROS 通过提供统一的软件平台, 使得代码重用和模块化设计变得更加简便, 可显著提高机

器人应用开发速度^[2]。

本文基于 ROS 系统多模块松耦合的特性作为统一的设计平台, 将视觉检测、深度学习、机械臂运动规划与物体抓取等技术相结合。深度学习进行实时物体检测, 通过手眼标定建立机械臂和相机的空间模型, 计算得到待抓取目标的三维坐标, 最后完成基于视觉的机械臂抓取系统的研究。

1 系统结构及原理

整个系统的构成如图 1 所示, 机器人在接收到任务后, 通过视觉传感器获取规划场景信息, Kinect 相机采集图像, 对目标物体进行视觉检测, 同时通过视觉标定建立的相机—机器人—目标物体三维坐标模型, 将物体三维位置坐标发送给机器人, 机器人在接收到信息后进行轨迹规划处理。

2 系统硬件设计

1) Kinect 相机: 采用微软 (Microsoft) 公司 Kinect V1 深度相机, 相机分辨率最大为 640 * 480, 满足系统设计的要求。

2) 光源: 采用 CCS 公司的 LED 环形光源, 该光源响应时间快, 可以获得高品质、高对比度的图像。

3) 电脑主机: 配有英特尔 (Intel) I7-8700K 处理器

收稿日期: 2019-05-15; 修回日期: 2019-06-10。

基金项目: 广东省应用型科技研发专项项目 (2015B090922012)。

作者简介: 杨三永 (1993-), 男, 河南驻马店人, 硕士, 主要从事机器人视觉抓取, 机器人运动规划方向的研究。

曾碧 (1963-), 女, 广东广州人, 博士, 教授, 主要从事智能信息处理, 智能机器人方向的研究。

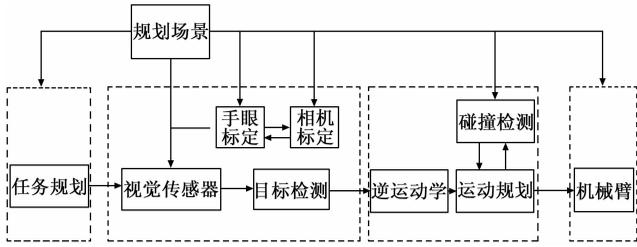


图 1 系统构成图

(CPU)，英伟达 (NVIDIA) GTX-1060 显卡的计算机，负责接收 Kinect 相机采集的图像，通过算法处理完成物体识别与定位之后，将物体的三维坐标信息发送给机械臂并控制机械臂的规划运动。

4) 机器人：采用丹麦 Universal Robots 公司的 UR3 型机器人，该机器人采用 TCP/IP 通讯，具有 6 个关节自由度，延伸半径 500 mm，重复定位精度为 0.03 mm。



图 2 系统硬件结构

3 系统软件设计

相机通过拍摄目标物体的图像，经过目标检测算法处理之后获取目标物体相对于相机的位置信息，机器人无法直接进行物体抓取，因此需要建立机器人—相机—目标物体的三维坐标模型，将目标物体的信息转换为三维空间的位姿。软件系统设计的其中一个难点在于，模型建立后，还需考虑对模型准确性产生影响的因素，因此需要对镜头制作过程产生的径向畸变和在相机组装过程中产生的切向畸变进行相机标定和手眼标定消除其产生的影响^[3]。

3.1 相机标定

如图 3， uov 是相机中的像素坐标系，它反映了相机芯片中像素矩阵的排列情况^[4]。原点 o 位于图像的左下角，坐标轴 u 、 v 轴分别与像平面的两边平行。由像素坐标与图像坐标的关系，可得如下矩阵：

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1/dx & 0 & u_0 \\ 0 & 1/dy & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

在获得像素坐标与相机坐标的关系之后，为了完成相机—目标物体的坐标转换，还需要建立相机坐标与世界坐标的关系。如图 3 所示，由小孔成像模型可知，在摄像机的光心处建立坐标系， Z 轴沿光轴方向， X 轴沿图像坐标沿图像坐标水平增加的方向，在相机坐标系 $O-XYZ$ 中，空间任意一点 M 与其图像点 m ， M 与相机光心 O 的连线为 OM ， OM 与像平面的交点 m 即为空间点 M 在图像平面上

的投影。有如下矩阵表示：

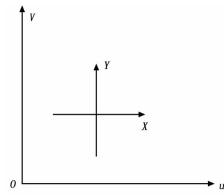


图 3 相机坐标与像素坐标关系

$$s \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} -x \\ y \\ z \\ 1 \end{bmatrix}$$

其中： $Z=f$ ， f 为相机的焦距， s 为比例因子 (s 不为 0)， f 为焦距 (光心到图像平面的距离)， $(x, y, z, 1)^T$ 是空间点 P 在相机坐标系中的其次坐标。因此可以通过相机坐标系可以建立起世界坐标系与图像像素坐标系的关系矩阵：

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1/dx & 0 & u_0 \\ 0 & 1/dy & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

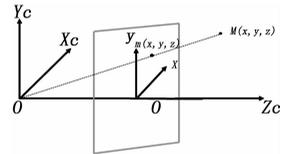


图 4 图像坐标与世界坐标转换

3.2 手眼标定

在基于机器人视觉的物体检测中，相机与机械臂末端手眼标定的准确性将直接影响到机械臂抓取的精度^[5]。手眼标定的目的就是求得相机坐标系到机械臂基坐标系的变换关系矩阵^[6]。

在手眼标定过程中，启动相机内外参标定程序，相机获取视野内标定板的棋盘格图像。同时启动机械臂标定程序，相机记录机械臂末端执行器在棋盘格标定板中移动过的位置，标定程序经过对记录的位置进行处理，通过相机坐标系—标定坐标系、机械臂基坐标系—机械臂末端坐标系—标定坐标系矩阵变换等，从而得到机械臂—相机—标定物体坐标系的转换关系，建立参数模型。

如图 3， B 为机械臂固定基座坐标系， E 为机械臂末端坐标系， C 为相机坐标系， W 是用于标定的棋盘格坐标系，相机坐标系与末端坐标系间无相对运动，棋盘格坐标系 W 固定不动。手眼标定的目标就是要求得相机坐标系到机械臂基坐标系的变换关系矩阵： $P_b = R_{cb} * P_c + T_{cb}$ ，其中 P_c 为相机坐标系下的一个坐标点，通过手眼变换关系可以得到点 P_c 在机械臂基坐标系下的坐标值，标定的目的是要求出两个坐标系的变换矩阵 R_{cb} 和平移向量 T_{cb} ，也可以写成 $P_b = H_{cb} * P_c$ 。

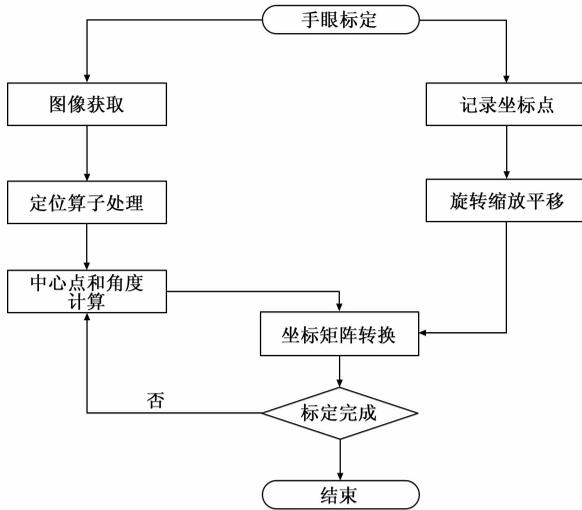


图 5 手眼标定流程

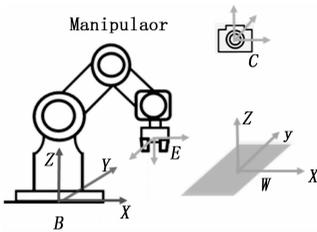


图 6 手眼标定坐标转换

九点标定直接建立相机和机械手之间的坐标变换关系^[7]。控制机械臂末端从标定板上位置 1 依次移动到位置 9, 同时相机识别 9 个点得到像素坐标。对于每一个位置都有:

$$P_b = H_e * P_e \quad (1)$$

$$P_c = H_{wc} * P_w \quad (2)$$

$$P_b = H_{cb} * P_c \quad (3)$$

这样就得到了 9 组对应的坐标。公式 (2) 中 H_{wc} 是相机标定得到的世界坐标系与图像像素坐标系的变换, 即棋盘格坐标系到相机坐标系的变换矩阵。通过 9 组坐标即可求得相机坐标系到机械臂的基坐标系的变换关系矩阵 H_{cb} 。

3.3 基于深度学习的物体定位与检测算法

传统的目标检测大多采用 OpenCV 模板匹配或者如方向梯度直方图 (histogram of oriented gradient, HOG)、尺度不变特征变换 (scale-invariant feature transform, SIFT) 的方法, 模板匹配原理是通过在输入图像 image 上滑动图像块, 对实际的图像块和模板图像 templ 进行匹配^[8-9]。这种方式的缺点是由于模板匹配采用的是像素点的匹配, 并且受环境光照等的限制较大, 所以一旦当识别的种类增加或者外部环境变化时, 模板匹配的效果并不理想。

目前, 基于深度学习的目标检测在性能上的表现明显超出了传统的目标检测算法, 对比于手工设计的特征, 基于深度学习的目标检测能够进行自主学习, 题图不同层级的特征图, 使检测性能大幅度提高。而 SSD (Single Shot Multibox Detector) 是一种基于卷积神经网络的端到端

(end-to-end) 目标检测算法^[10]。SSD 算法满足数据处理的效率、性能、速度和智能化等各个方面要求, 与 Faster RCNN^[11]相比, SSD 不再需要 RPN (Region Proposal Network) 来生成候选区域 ROIs, SSD 采用 CNN 来直接进行检测, 另外 SSD 网络可以提取不同尺度的特征图进行检测, 这使 SSD 在检测小目标物体上具有良好的性能。因此, SSD 算法很好地改善了检测速度, 更能满足目标检测的精确性要求^[12]。

SSD 采用 VGG16 作为基础模型, 然后在 VGG16 网络的基础上新增了卷积层来获得更多的特征图用于检测, SSD 将 VGG16 的 fc6 和 fc7 两个全连接层换成两个卷积层, 并去除 VGG16 的 dropout 层和分类层, 最后通过非极大值抑制 (NMS) 得到最终检测结果^[13]。

本系统采用 SSD 算法进行物体检测, 并通过获得的相机坐标系到机械臂的基坐标系的变换关系矩阵计算物体坐标, 为后续机器人抓取规划提供可靠信息。本文使用自建的数据集对目标检测算法进行训练与评估, 数据集中包含 1 800 张原始图片用于训练与验证, 另外包含 200 张图片用于测试。为了提高训练的准确性, 数据集使用随机改变图片亮度、对比度、灰度化处理以及对原始图片进行剪切的方法进行数据增强, 并且数据集中包含大量目标物重叠等复杂背景。算法的一般步骤为:

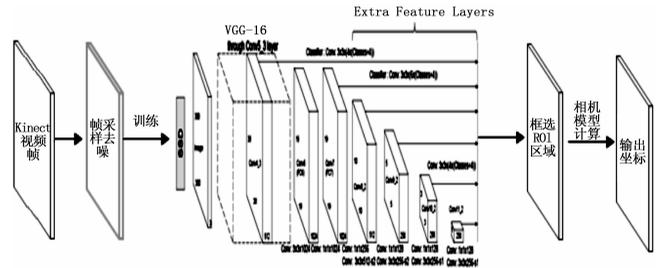


图 7 目标检测处理流程

- 1) 对相机实时获取到的视频帧进行去噪、隔帧采样等处理, 为了提高检测的实时性, 再对图像进行灰度化处理。
- 2) 对预处理后的图像输入神经网络, 经过 SSD 算法处理框选出物体在图像中的像素位置。
- 3) 通过标定建立的三维模型, 将像素转换为相机坐标, 通过与世界坐标转换得到物体的三维坐标。
- 4) 将物体的三维坐标信息通过 ROS 通讯机制发送给机械臂, 机械臂控制算法进行规划运动。

3.4 机器人运动规划

运动规划是指移动机械臂在解空间中通过算法和约束条件等操作避开障碍物, 找到一条具有最优解的机器人运动路径^[14]。

传统的路径规划算法, 如人工势场法、遗传算法、模拟退火算法以及其它智能优化算法, 由于需要在机器人解空间内进行建模, 因此在机器人自由度增多时计算的复杂度成指数上升, 传统的算法并不适合多自由度机器人在复杂环境条件下的规划^[15]。

基于快速扩展随机树 (RRT / rapidly exploring random tree) 的路径规划算法, 是一种在高维空间中效率很高的路径规划算法, RRT 的思想的快速像树一样扩张从而寻找到最优的路径。

RRT 的基本步骤是:

- 1) 起点作为一颗种子, 从它开始进行随机搜索;
- 2) 在机器人的解空间中, 随机生成一个点 X_{rand} ;
- 3) 在树上找到与 X_{rand} 最近的点, 记为 $X_{nearest}$;
- 4) 连接 X_{rand} 和 $X_{nearest}$, 朝着树的方向生长, 如果没有碰到障碍物就把生长后的 $X_{nearest}$ 添加到树上, 返回步骤 2);

随机点 X_{rand} 一般是均匀分布的, 所以没有障碍物时树会近似均匀地向各个方向生长, 这样可以快速探索空间。RRT 算法通过对状态空间中的采样点进行快速的碰撞检测, 避免了对空间的建模有效地解决了高维空间和复杂约束的路径规划问题^[16]。

机械臂的运动轨迹的规划是通过 ROS 中的 moveIT! 包实现的, 本系统使用 RRT 算法, 在 ROS 平台上结合使用 UR3 机械臂对通过相机进行转换后的三维物体坐标进行运动规划控制。

4 实验结果与分析

4.1 实验步骤和方法

通过基于深度学习的视觉检测算法得到物体的在相机中的图像坐标, 利用标定建立的参数化模型计算物体的三维坐标, 然后将物体的世界坐标映射到机器人的基础坐标系中得到物体在机器人基础坐标系中的位置, 系统操控机器人进行轨迹规划与目标物体的抓取。实验采用将机器人对同一目标物体在不同位置进行抓取, 随机测试 3 个不同的位置点, 每个位置点让机器人进行 8 次规划。机器人定位精度为 0.03 mm, 实验表明, 机器人可以准确地抓取目标物体, 本系统具有较高的定位精度, 能够满足一般工业的生产制作要求。抓取实验如图 8 所示。



图 8 抓取实验测试

图 8 (a) 为相机中 SSD 算法检测到目标物体, 通过事先建立好的相机-机器人-目标物体参数模型计算出目标物体的三维坐标, 然后在计算机中通过运动规划算法模拟出机器人的运动轨迹, 图 8 (b) 为机器人按照运动轨迹进行目标物体的抓取。

4.2 传统算法与 SSD 算法比较

图 9 (a) 是采用传统的模板匹配算法进行目标物体检测, 在与基于深度学习的 SSD 目标检测算法对比中, SSD

算法表现出了优秀的目标检测能力。在复杂的背景条件下, SSD 算法依然表现出了很强的识别能力, 而且算法的实时性也满足系统设计的要求。

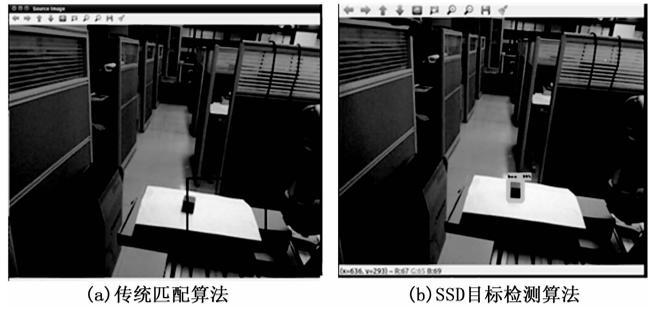


图 9 传统算法与 SSD 算法比较图

4.3 实验结果比较

实验随机选择 3 个位置点, 每个位置点测试 8 组数据, 比较目标物体在图像坐标和世界坐标的坐标值。结果如表 1 和表 2 所示。

表 1 位置 1 实验结果对比

序号	坐标转换	
	图像坐标 $(x, y) px$	世界坐标 $(x, y, z) m$
1	(360.86406, 314.75006)	(-0.2084, 0.3867, 0.0891)
2	(360.60466, 315.16739)	(-0.2094, 0.3862, 0.0891)
3	(360.44715, 315.03530)	(-0.2095, 0.3873, 0.0883)
4	(360.56072, 314.81171)	(-0.2093, 0.3875, 0.0885)
5	(360.58208, 314.79022)	(-0.2090, 0.3868, 0.0888)
6	(360.69236, 315.07510)	(-0.2089, 0.3865, 0.0893)
7	(360.82923, 314.96798)	(-0.2091, 0.3865, 0.0884)
8	(361.00967, 314.94209)	(-0.2094, 0.3857, 0.0893)

表 2 位置 2 实验结果对比

序号	坐标转换	
	图像坐标 $(x, y) px$	世界坐标 $(x, y, z) m$
1	(394.34459, 341.1128)	(-0.1586, 0.3179, 0.0880)
2	(395.13626, 340.9000)	(-0.1586, 0.3199, 0.0873)
3	(393.97480, 341.0317)	(-0.1588, 0.3241, 0.0835)
4	(394.14615, 341.0534)	(-0.1588, 0.3220, 0.0847)
5	(393.65783, 341.6653)	(-0.1595, 0.3211, 0.0849)
6	(393.78746, 341.0398)	(-0.1589, 0.3237, 0.0844)
7	(393.43372, 342.1332)	(-0.1592, 0.3239, 0.0822)
8	(393.65234, 341.7791)	(-0.1589, 0.3246, 0.0824)

实验数据均来自于实验室 ur3 机械臂和 Kinect 双目相机。以上数据, 左侧数据展示的是物体在图像里的坐标, 右侧数据展示的是经过标定模型转换后得到的物体在世界坐标的位置。经过测试, 图像坐标误差在 1.2 px 内, 反映到物体实际坐标满足实验要求。

5 结束语

本系统基于机器视觉的目标物体进行识别定位与抓取, 在机器人操作系统 ROS 中进行开发, 通过对本系统建立的

表 3 位置 3 实验结果对比

序号	坐标转换	
	图像坐标 $(x, y)px$	世界坐标 $(x, y, z)m$
1	(394.34459, 341.1128)	(-0.1586, 0.3179, 0.0880)
2	(395.13626, 340.9000)	(-0.1586, 0.3199, 0.0873)
3	(393.97480, 341.0317)	(-0.1588, 0.3241, 0.0835)
4	(394.14615, 341.0534)	(-0.1588, 0.3220, 0.0847)
5	(393.65783, 341.6653)	(-0.1595, 0.3211, 0.0849)
6	(393.78746, 341.0398)	(-0.1589, 0.3237, 0.0844)
7	(393.43372, 342.1332)	(-0.1592, 0.3239, 0.0822)
8	(393.65234, 341.7791)	(-0.1589, 0.3246, 0.0824)

相机—机器人—目标物体的参数化模型, 能够准确识别并且定位目标物体的位置坐标, 结合机器人的运动规划 RRT 算法, 最后系统控制机器人能够准确抓取目标物体。实验证明, 该系统具有比较高的定位精度, 而且基于 ROS 进行开发, 具有较高的可扩展性, 便于针对不同的需求进行二次开发, 满足工业生产的要求, 对于提高工业自动化水平具有重要的意义。

参考文献:

- [1] 邱文彬, 苏剑波, 席裕庚. 基于在线识别的机器人动态手眼协调 [J]. 机器人, 2001 (3): 222-226.
- [2] 李 凤. 基于 ROS 的机械臂控制系统设计 [J]. 自动化技术与应用, 2018, 37 (11): 72-76.
- [3] 尚忠义, 董明利, 李伟仙, 等. 基于手眼标定方程 $AX=XB$ 的精度影响因素研究 [J]. 传感器与微系统, 2017, 36 (3): 36-39.
- [4] 李艳辉, 吴 玲, 尚胤彤. 机器人立体视觉中摄像机的标定 [J].

(上接第 160 页)

3) 利用 LabVIEW 强大的数据分析工具包和良好的软件兼容能力, 设计了星载机子系统综合测试系统, 并完成了相应的测试工作。验证了 PXI 进行大规模数据处理能力以及扩展性能, 其实时性能也很好满足了星载机的设计要求。

本文中实验和仿真得到的结论反映了 PXI 总线技术的先进性, 为仿真与测试方面的应用提供了理论基础。本文的研究成果具有良好的通用性, 本系统可作为航天器的通用仿真与测试平台。本文的结论利用 PXI 来进行星载机仿真与测试具有重要的实际价值。

参考文献:

- [1] 朱承元, 杨 涤, 杨 旭. 卫星姿轨控实时仿真系统及串口通讯建模 [J]. 哈尔滨工业大学学报, 2005, 37 (1): 26-31.
- [2] 陈光明, 廖明宏, 吴翔虎. 小卫星星载计算机及其外围设备的管理 [J]. 哈尔滨工业大学学报, 2002, 34 (2): 201-204.
- [3] 孟祥迪, 郭静寰, 熊木地. 基于 PXI 总线技术的星载计算机性能测试系统研究 [J]. 计算机测量与控制, 2007, 15 (5): 0571-0573.
- [4] 马秀娟, 徐国栋, 马兴瑞, 等. HITSAT 星载计算机系统 [J]. 微处理机, 1999 (3): 36-39.

- 自动化技术与应用, 2012, 31 (3): 61-63, 81.
- [5] 王金桥, 段发阶, 汪 润. 精确标定关节臂视觉检测系统手眼关系 [J]. 计算机工程与应用, 2015, 51 (21): 225-229.
- [6] 沈爱华, 罗印升, 宋 伟. 一种基于双目视觉的手眼标定方法 [J]. 机械设计与制造工程, 2016, 45 (5): 42-46.
- [7] 王奇志, 王东生, 徐 德. 利用三角形及其九点圆的摄像机标定 [J]. 数学的实践与认识, 2012, 42 (11): 91-98.
- [8] 黄添娣. 基于肤色分割和统计模板匹配的手势识别人机交互系统 [D]. 广东技术师范学院, 2017.
- [9] 黄凯奇, 任伟强, 谭铁牛. 图像物体分类与检测算法综述 [J]. 计算机学报, 2014, 37 (6): 1225-1240.
- [10] Liu W, et al. SSD: Single Shot MultiBox Detector. [EB/OL] [2016-12-29/2017-9-20]. URL: <https://arxiv.org/pdf/1512.02325.pdf>.
- [11] Ren S Q, He K M, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39 (6): 1137-1149.
- [12] 郭叶军, 汪敬华, 吉明明. SSD 算法推理过程的探析 [J]. 现代计算机 (专业版), 2018 (5): 3-7, 11.
- [13] 陈冰曲, 邓 涛. 基于改进型 SSD 算法的目标车辆检测研究 [J]. 重庆理工大学学报 (自然科学), 2019, 33 (1): 58-63, 129.
- [14] 陈 飞, 王振华, 陈国栋, 等. 基于 ROS 的移动操作臂运动规划研究 [J]. 煤矿机械, 2018, 39 (6): 36-38.
- [15] 霍凤财, 迟 金, 黄梓健, 等. 移动机器人路径规划算法综述 [J]. 吉林大学学报 (信息科学版), 2018, 36 (6): 639-647.
- [16] 王 好, 范平清, 王岩松, 等. 基于 ROS 平台的机械臂精确控制研究 [J]. 轻工机械, 2018, 36 (6): 42-47.
- [5] 王宁强, 刘向东. 卫星姿态控制系统硬件在回路仿真研究 [J]. 计算机仿真, 2005, 22 (10): 78-81.
- [6] 王冠营, 陈文通, 刘海涛, 等. 基于虚拟技术的小卫星测试平台 [J]. 测试技术学报, 2004, 18 (1): 21-24.
- [7] 李云端, 黄小虎. FY-1C 卫星地面综合测试软件技术 [J]. 上海航天, 2001 (2): 51-55.
- [8] Janowiak T J. A simulation for combat systems development and acceptance testing [A]. Proceedings of the 1990 Winter Simulation Conference [C]. 1990: 210-213.
- [9] 张世杰, 曹喜滨, 王 峰. 微小卫星姿态控制系统半物理仿真设计及验证 [J]. 中国空间科学技术, 2005, (6): 43-50.
- [10] 杨 涤, 李立涛, 杨旭等. 系统实时仿真开发环境与应用 [M]. 清华大学出版社, 2002.
- [11] 王 峰, 徐国栋, 耿云海, 等. 基于微形核信息电子系统卫星姿态控制半实物仿真 [J]. 系统仿真学报, 2007, 19 (5): 1131-1135.
- [12] 张世杰, 曹喜滨. 基于 MicroSim 仿真平台的航天器交会对接物理仿真系统 [J]. 航天控制, 2006, 24 (2): 63-68.
- [13] 孙兆伟, 徐国栋, 林晓辉, 等. 小卫星设计分析与仿真验证一体化系统 [J]. 系统仿真学报, 2001, 13 (5): 623-626.