

DOI:10.3969/j.issn.1673-4785.201212038

网络出版地址: <http://www.cnki.net/kcms/detail/23.1538.TP.20130326.0906.002.html>

基于 Kinect 的机械臂目标抓取

韩峰^{1,2}, 刘华平², 黄文炳², 孙富春², 高蒙¹

(1. 石家庄铁道大学 电气与电子工程学院, 河北 石家庄 050043; 2. 清华大学 智能技术与系统国家重点实验室, 北京 100084)

摘要:为完成机械臂在非特定环境下的自主抓取, 系统采用微软公司研发的 Kinect 对场景内的信息进行实时检测. 通过对 Kinect 采集的深度信息进行背景相减法及帧差法处理可以获得目标抓取点信息. 利用基于工作空间的 RRT 算法对机械臂末端进行路径规划, 并利用梯度投影法进行逆运动学轨迹优化, 求解关节轨迹. 机械臂按照关节角运动时, 可完成目标的抓取. 通过设计一套实时桌面清理实验系统, 验证了该方法的有效性.

关键词:Kinect; 目标检测; RRT; 逆运动学轨迹优化; 机械臂

中图分类号:TP242 **文献标志码:**A **文章编号:**1673-4785(2013)02-0149-07

Kinect-based object grasping by manipulator

HAN Zheng^{1,2}, LIU Huaping², HUANG Wenbing², SUN Fuchun², GAO Meng¹

(1. College of Electrical and Electronic Engineering, Shijiazhuang Railway University, Shijiazhuang 050043, China; 2. Key Laboratory of Intelligent Technology and Systems, Tsinghua University, Beijing 100084, China)

Abstract: Kinect from Microsoft is often used in the robot system for target and real-time detection in unstructured environments. Background subtraction and frame difference algorithms based on the depth information collected by Kinect are used to obtain the information of the grasp point. A Rapidly-exploring Random Tree (RRT) algorithm based on the working space was used to carry out path plan of the end of manipulator, and gradient projection algorithm was used to do inverse kinematics trajectory optimization. The manipulator was able to accomplish object grasping tasks according to the joint angles. An experiment system was developed to verify the effectiveness of the proposed methods.

Keywords: Kinect; target detection; RRT; inverse kinematics trajectory optimization; manipulator

机器人的自主抓取大多是基于视觉的控制, 例如视觉定位和视觉伺服^[1-2]. 如今, 自主抓取研究所面临的一个重要问题是所处的非结构化环境以及自身模型的不精确性所带来的不确定性^[3]. 解决这种问题的关键就是发展机器人所需的传感和感知系统, 使其能在未知的复杂环境下动作. 微软公司研制的 Kinect 视觉传感器能够同时提供二维图像及深度信息, 并且价格低廉, 故被广泛应用到机器人系统的研究中. 目前, 国内外在这方面的研究都取得了一些进展, 如机器人的导航与定位^[4]、目标的跟踪^[5]、

障碍物的检测^[6]等. 本文借鉴了 Kinect 在机器人系统中的应用, 利用其能够同时提供二维图像及深度信息的特性, 简化了目标检测的内容, 并通过基于工作空间的 RRT 法和梯度投影法相结合, 实现了机械臂的轨迹优化, 避免了机械臂末端执行器运动与关节运动分离的情况, 从而最终实现了机械臂系统自主完成目标抓取的功能.

1 机器人系统的构建

本文的机器人系统是由视觉传感器、机械臂系统及主控计算机组成, 其中机械臂系统又包括模块化机械臂和灵巧手 2 部分. 整个系统的构建模型如图 1 所示.

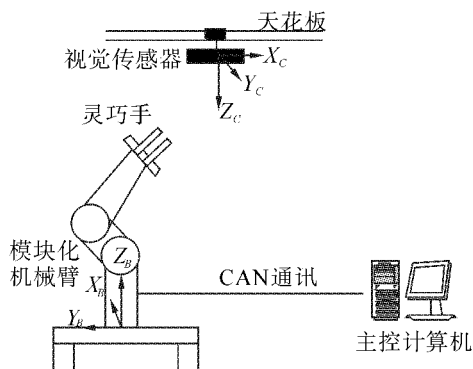


图1 系统模型

Fig.1 Model of the system

1.1 Kinect

系统中的视觉传感器为微软公司研发的 Kinect 深度摄像机。Kinect 主要由一个普通的低分辨率 RGB 摄像机、一个红外发射器、一个红外摄像头以及一组麦克风阵列组成,如图 2 所示。普通 RGB 摄像机能够以 30 帧/s 的速率传送 640×480 分辨率的 RGB 图像,而红外发射器和摄像头则是用来采集相应的深度信息,通过内部芯片转换并传送 320×240 分辨率的深度图像^[7]。

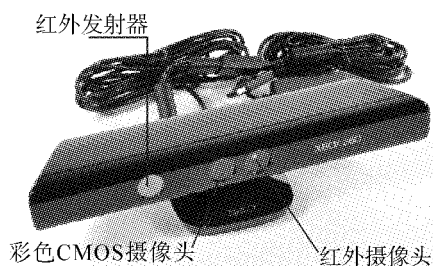


图2 Kinect 深度摄像机

Fig.2 Kinect depth camera

Kinect 在系统中的位置如图 1 所示,位于天花板顶部,机械臂的斜上方。其位置的选择是通过比对多次实验的结果确定的。实验表明 Kinect 置于此地位置视野范围最广,且可以有效地避免遮挡。作为本套服务机器人的“眼睛”,Kinect 的主要作用是进行目标检测。

1.2 模块化机械臂

系统中的模块化机械臂,是由德国 Schunk 公司开发并生产的 Power Cube 模块组合而成,共 7 个关节,有一个冗余自由度,主要用于实施清理操作。采用冗余机械臂的目的是为了克服非冗余自由度机械臂运动不灵活的缺点,即在规定了末端位姿轨迹后,不能避免关节空间存在的奇异点、不能避障、不能避免关节极限、在某些情况下运动学性能差等^[8]。

1.3 灵巧手

灵巧手是本套机器人系统的末端执行器,主要

用于完成对目标的抓取工作。本系统选用的灵巧手为美国 Barrett Technology 公司生产的 BH8.280 灵巧手。该灵巧手是一个可编程的多手指抓持器,功能强大,结构紧凑,具有非常高的灵活性,可以抓取不同形状、大小及姿态的目标物体。

1.4 主控计算机

主控计算机为一台普通 PC 机,其主要功能是对 Kinect 采集的数据进行处理,从而获得目标抓取点的三维信息,之后对机械臂进行轨迹优化。由于主控机还负责与机械臂和灵巧手进行通信,故轨迹优化完成后,主控机又将解算所得的关节角赋给机械臂,并在机械臂达到相应位置后发指令给灵巧手,使其实施抓取动作。

2 目标检测

2.1 Kinect 标定

摄像机标定是目标检测过程中的一部分,主要是指确定摄像机参数的过程^[9]。目前已经有许多摄像机标定的方法,本文所用的是张正友标定法^[10]。

本文对 Kinect 进行标定时摄像机的模型采用的是针孔模型,其定义如下^[11-12]:

$$sm = A[R | t]M,$$

或者

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \end{bmatrix} \cdot \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}.$$

式中: (u, v) 是点投影在图像平面的坐标, A 为摄像机的内参矩阵, $[R | t]$ 为外参矩阵, (X_c, Y_c, Z_c) 是点在摄像机坐标系下的坐标。摄像机坐标系中任意一点投影到图像坐标系的公式如式(1)、(2)所示。

$$X_c = \frac{(x - c_x)Z_c}{f_x}, \quad (1)$$

$$Y_c = \frac{(y - c_y)Z_c}{f_y}. \quad (2)$$

式中: X_c, Y_c 为彩色图像中某点的坐标, Z_c 为深度图像中相同像素点的深度值。然而 Kinect 生成的彩色图像和深度图像并不是完全重合的,故在计算之前需要对彩色图像和深度图像进行匹配。实验表明, OpenNI 自带的函数已经可以较好地解决深度图像的校正问题,故本文通过 OpenNI 自带的函数完成 Kinect 彩色图像和深度图像的配准。

综上所述可以得到场景内任意点在摄像机坐标系下的三维坐标。为了方便后文中机械臂轨迹规划的内容,需要将空间内各点的三维信息都转换在机械臂基坐标系下,因此,首先需要知道摄像机坐标系

与机械臂基坐标系的关系^[13],如式(3)所示。

$$P_C = [R | t] P_B = \begin{matrix} \text{com} \\ \text{end-off} \end{matrix} H \begin{matrix} \text{end-off} \\ \text{base} \end{matrix} H P_B. \quad (3)$$

式中: $[R | t]$ 表示整个旋转平移矩阵, $\begin{matrix} \text{com} \\ \text{end-off} \end{matrix} H$ 表示张正友标定法标出的 Kinect 外参矩阵, $\begin{matrix} \text{end-off} \\ \text{base} \end{matrix} H$ 表示机械臂末端坐标系到基坐标系的旋转平移矩阵。设空间内任意一点 P 在机械臂基坐标系下的齐次坐标表示为 $P_B = [X_B \ Y_B \ Z_B \ 1]$, 在摄像机坐标系下的齐次坐标表示为 $P_C = [X_C \ Y_C \ Z_C \ 1]$, 则点 P 在机械臂基坐标系下的坐标表示为

$$\begin{pmatrix} X_B \\ Y_B \\ Z_B \\ 1 \end{pmatrix} = [R | t]^{-1} \cdot \begin{pmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{pmatrix}$$

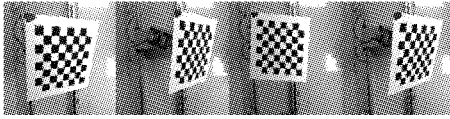


图3 部分标定模版图片

Fig. 3 Parts of the calibration template

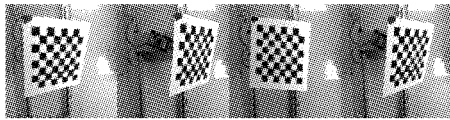


图4 标定过程代表性结果

Fig. 4 Representative results of the calibration process

2.2 目标提取

目前,常用来进行目标检测的方法主要有3种,包括背景相减法、光流计算法和帧差法。本文采用的是基于深度信息的背景相减法。

与传统的背景相减法相同,基于深度信息的背景相减法也是首先在没有目标出现的场景中提取背景模型,只不过提取的背景模型是图像中各个点的深度信息。计算公式为

$$S_k(z) = |N_k(z) - B_k(z)|, \\ G(k) = \begin{cases} 1, S_k(z) \geq \delta; \\ 0, S_k(z) < \delta. \end{cases} \quad (4)$$

式中: $B_k(z)$ 为背景图像上第 k 个点的深度值, $N_k(z)$ 为当前帧上第 k 个点的深度值, δ 为设定的阈值,当 $G(k) = 1$ 时,表明该点为目标物体上的点。

值得注意的是,由于 Kinect 位于系统的上方,故算法得到的结果会包含目标的顶部信息及部分侧面信息。因此,还需对所得结果进行处理,以获得较为精确的目标轮廓。因为物体的顶部各点具有大致相同的深度值,且其深度值必定大于其侧面的深度。由此,可通过提取符合式(4)条件的所有点中最大的深度值,并将其余点的深度值与深度最大值做差值计算,当差的绝对值在某一阈值范围内时,表明此点为目标顶部的点。

此时,对新的数据做处理即可获得更为精确的目标轮廓,如图5所示。

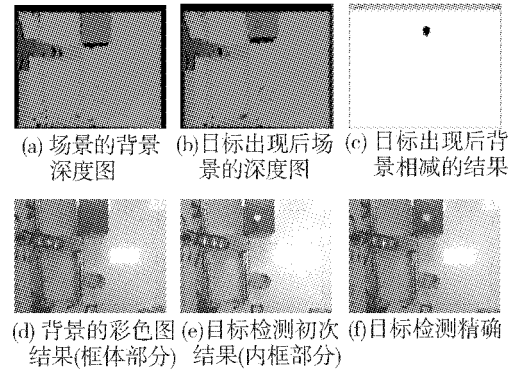


图5 目标轮廓提取

Fig. 5 Object contour extraction

2.3 抓取点确定

每当场景中出现新的目标时都会被 Kinect 检测到。然而,要最终确定目标抓取点的信息还有一项重要的工作,即消除物体移动过程中的干扰和放置物体时人手的影响。经过简单的分析和实验验证,本文决定采用帧差法消除这些因素对目标的干扰。

本系统中定时器的采样频率是 100 帧/ms,为了消除物体移动和人手的影响,每隔 2 s 取一帧图像,即每 20 幅图像中取 1 幅图像与前一帧提取的图像进行减法运算,如果 2 幅图像中点的深度值变化小于某一阈值,则说明目标物体已经被放置到场景中的某处,由此便消除了其他因素的影响。计算公式为:

$$\Delta D_k(z) = |D_k(z) - D'_k(z)|, \\ H(k) = \begin{cases} 1, \Delta D_k(z) \leq \delta; \\ 0, \Delta D_k(z) > \delta. \end{cases}$$

3 轨迹优化

轨迹优化问题分为连续路径问题和点到点问题。连续路径问题又被称为逆运动学轨迹优化问题。对于连续路径问题,传统的方法有梯度投影法和加权最小范数法。对于点到点问题,传统的基于关节空间的 RRT 法形式简单、规划效果好,然而却忽略了末端执行器的运动轨迹。本文采用基于工作空间的 RRT 法,在末端执行器的自由空间构建 RRT 树,并对 RRT 的每一步扩张运用梯度投影进行逆运动学轨迹优化,获得相应的关节运动。

3.1 梯度投影法

冗余机械臂逆运动学轨迹优化最关键的一步是求解速度水平运动方程:

$$\dot{X} = J(q)\dot{q}.$$

式中:

$$\dot{q} = \dot{q}_s + \dot{q}_h = J^+ \dot{X} + (I - J^+ J) v. \quad (5)$$

梯度投影法是由 Liegeoi^[14] 首先提出的一种基于广义逆的算法. 将目标函数的梯度向量乘以系数 k 作为 v , 这样式(5)就变成:

$$\dot{q} = \dot{q}_s + \dot{q}_h = J^+X + k(I - J^+J) \nabla H(q).$$

由于机械臂每个关节都有一定的活动范围, 故取避关节极限作为算法的目标函数^[15]:

$$H(q) = \sum_{i=1}^7 \frac{1}{7} \frac{(q_{i \max} - q_{i \min})^2}{(q_{i \max} - q_i)(q_i - q_{i \min})}.$$

3.2 基于工作空间的 RRT 算法

RRT (rapidly-exploring random trees) 是首先由 Lavalley 和 Kuffner^[16-17] 提出的一种随机规划算法. RRT 方法具有形式简单、计算快速、概率完全等优点. 传统的 RRT 算法是基于关节空间的, 但基于关节空间的 RRT 法无法预知机械臂末端执行器运动轨迹, 造成末端执行器运动与关节运动分离, 不利于任务的完成. 故采用基于工作空间的 RRT 算法, 更合理地将轨迹规划和逆运动学解算结合起来.

基于工作空间 RRT 算法的基本原理如下: 初始位姿 x_{init} 为树的第 1 个节点. 在每一迭代步中, 随机生成状态点 x_{rand} , 参照 ρ 找到离 x_{rand} 最近的一点 x_{near} ; 然后从 x_{near} 和 x_{rand} 中找到节点 x_{new} , 原则是 x_{new} 到 x_{near} 的距离比 x_{rand} 到 x_{near} 的距离更近. 之后, 判断 x_{new} 是否在障碍区域内, 若不在, 则把它纳入 T 中, 形成新的树; 若在, 则继续寻找新的节点. 重复迭代直到 T 中有一点 x 达到目标区域 Q_{goal} . RRT 的节点扩展图如图 6 所示.

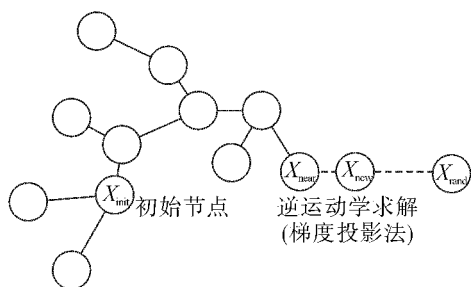


图6 RRT 节点扩展

Fig.6 Node expansion of RRT

由于算法是在工作空间下进行的, 因此 RRT 每次迭代得到的都是末端执行器的路径. 而模型需要求解关节轨迹, 因此还要进行逆运动轨迹优化, 方法采用 3.1 介绍的梯度投影法. 值得说明的是, 这样的求解虽然提高了算法的复杂性, 但实验表明机械臂在进行操作时仍具有较好的实时性.

机械臂在实际的任务操作过程中, 其末端执行器不仅要能达到指定的目标区域, 还要以“恰当”的姿态到达. 所谓恰当, 就是要机械臂的末端执行器的姿态能够较好地完成任务. 比如完成抓取水杯的任务, 如何确定适合的机械臂终端状态是机器人

能否完成工作任务的一个关键.

假设机械臂的手爪模型及目标物体如图 7 所示, 借鉴 Bertram 和 Kuffner^[18] 提出的方法, 建立确定机械臂运动轨迹终端状态函数, 从而选择适合的终端状态.

假设最适合机械臂的终端状态是函数

$$\varphi(q_{\text{goal}}) = \alpha_1 |vH_1 - 1| + \alpha_2 |vH_2 - 1| + \alpha_3 |H_1|.$$

的极小值, 即 $\min \varphi(q_{\text{goal}})$.

式中: $\alpha_1, \alpha_2, \alpha_3$ 为权重因子, 向量 v 代表最佳手爪方向, 图 7 中特征点 O 一般是指目标的重心点, H_1, H_2 是机械臂末端执行器上的点.

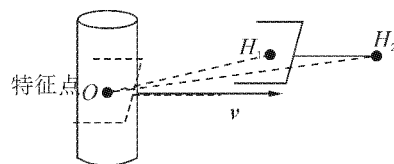


图7 终端状态函数

Fig.7 Terminal state function

极小函数 $\varphi(q_{\text{goal}})$ 意味着尽量使向量 H_1 和 H_2 与 v 平行, 同时还要充分靠近点 O 的最佳状态, 即 $\varphi(q_{\text{goal}})$, 如图 7 虚线所示.

事实上, 要求 $\varphi(q_{\text{goal}}) = 0$ 是很苛刻的, 所以一般只要求 $\varphi(q_{\text{goal}}) < \varepsilon$, ε 为某大于零的参数. 由 $\varphi(q_{\text{goal}})$ 所确定的区域称为目标区域. 在 RRT 的节点扩展时, 只要有节点落在目标区域内即认为机械臂找到了路径. 若对寻找到的路径求解关节轨迹无解, 主控机将对机械臂重新进行轨迹优化, 如果仍未找到合理的关节轨迹, 系统将提示目标点不可达.

4 实验

为了验证前文的研究内容, 本文设计了一套实验系统, 场景如图 8 所示.

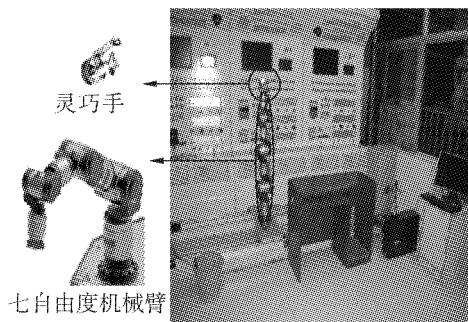


图8 实验场景

Fig.8 Experimental scene

4.1 实验相关说明

1) 实验任务. 以桌面为机械臂的操作空间, 对

出现在其上的目标进行自主抓取,并将其投掷到垃圾桶内,以完成桌面清理的任务。

2) 目标物的选取. 对于机械臂的目标抓取,不同形状的物体,争取不同的抓捕策略,然而由于这项技术目前尚未取得很好的研究成果,故本实验系统选取的目标为形态均匀、大小不同的物体。

3) 抓取点的确定. 本文中的抓取点是指机械臂实施抓取时,其末端执行器的中心点应达到的目标物上的某点. 目标的信息是通过 Kinect 进行目标检测确定的. 如图 1 所示, Kinect 被倒置于天花板顶部,故其获得的目标信息为目标顶面信息. 然而,受到机械臂末端姿态的限制以及出于对抓取稳定性的考虑,将目标的抓取点设在目标物左侧面的中心. 由于目标顶面的轮廓信息可由目标检测获得且目标为形状均匀的物体,故抓取点的三维坐标很容易得出。

4.2 实验流程

系统的工作流程如图 9 所示. 其中轨迹优化包括 2 部分:一是以机械臂零位或垃圾清理完成后的位置为初始位置,抓取点为目标点进行轨迹优化;二是以抓取点为初始位置,投掷点为目标点进行轨迹

点的实际坐标为 $(-0.592, -0.254, 0.649)$, 通过 Kinect 进行目标检测的误差在 1 cm 内。

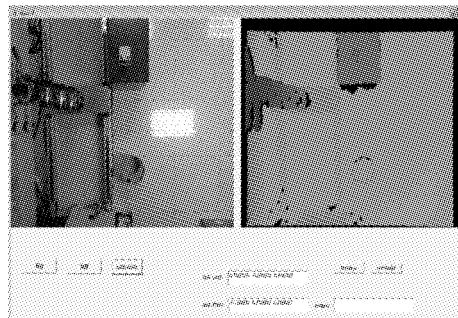


图 10 系统主界面

Fig. 10 Main interface

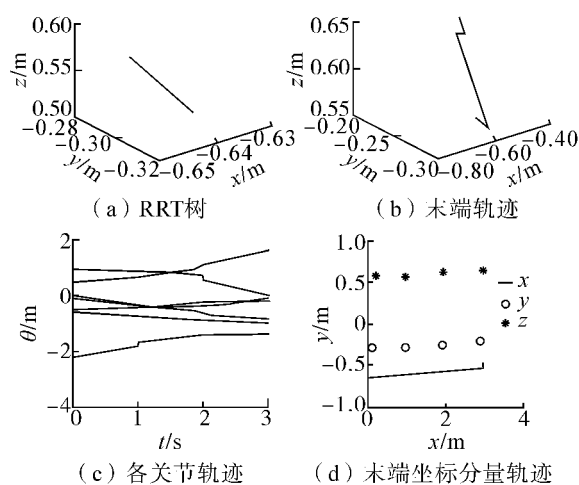


图 11 抓取过程轨迹优化结果

Fig. 11 The trajectory optimization results of the grasp process

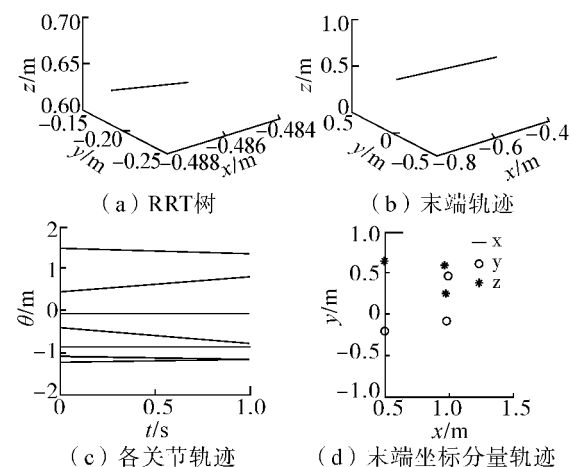


图 12 投掷过程轨迹优化结果

Fig. 12 The trajectory optimization results of the throw process

优化。

4.3 实验结果

图 10 ~ 12 为本实验系统进行实验验证的相关过程. 如图 10 所示, 本次实验的目标为一饼干盒, 检测所得抓取点坐标为 $(-0.587, -0.260, 0.642)$, 投掷点坐标为 $(-0.700, 0.550, 0.250)$. 经测量目标抓取

图 11、12 为对该目标实施清理任务所进行的 2 次轨迹优化的仿真结果. 仿真结果表明, 末端执行器

2次终节点的位置坐标分别为 $(-0.568, -0.231, 0.663)$ 、 $(-0.732, 0.533, 0.281)$,与抓取点的误差在5 cm内。

本次实验共清理了3个目标,包括薯片盒1个、饼干盒1个和罐头瓶1个,如图13所示。图14中,(a)、(b)、(c)分别为清理3个不同目标时的图片。以图14(a)为例,图片分别为机械臂初始状态、在实验场景中放入第1个目标、机械臂向目标移动、机械臂达到抓取点、机械臂对目标进行抓取、机械臂向投掷点(垃圾桶上方某位置)移动,同时第2个目标被放入场景中、机械臂到达投掷点、机械臂进行第1次投掷的情景。至此,一次清理任务完成。之后,机械臂从当前位置向新的目标点移动,继续执行桌面清理任务。

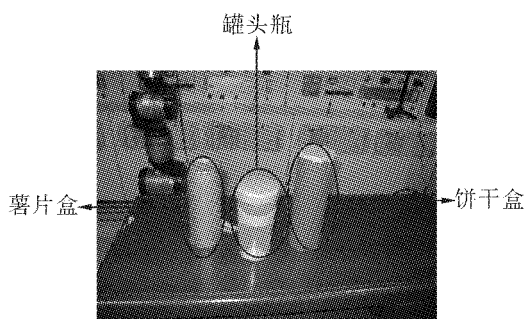
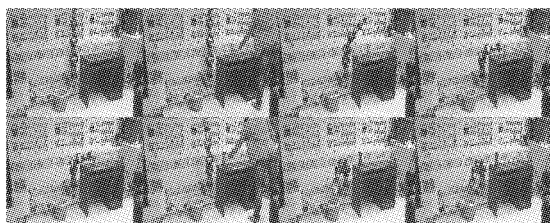


图13 清理任务的目标物
Fig.13 The goal of cleaning tasks



(a)清理第1个目标



(b)清理第2个目标



(c)清理第3个目标

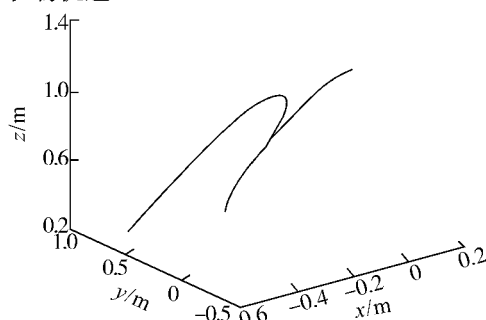
图14 连续清理任务

Fig.14 Continuous cleaning tasks

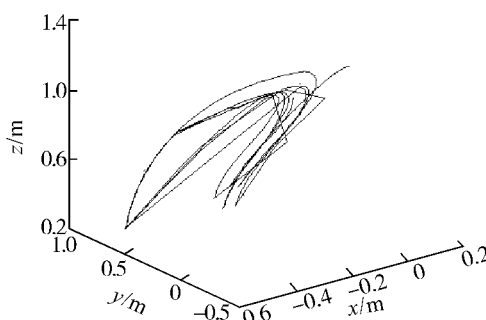
4.4 实验分析

为了验证该系统的可行性与稳定性,本文进行

了多次实验。实验结果表明,当有新的目标进入场景时,如果机械臂处于准备状态(零位),就会立即对目标垃圾实施清理工作;如果机械臂处于清理过程中,就会在完成当前任务后,第一时间实施新的清理任务;如果若清理工作完成后没有新目标的出现,机械臂会归零,处于准备状态,直到目标出现为止。由于RRT为随机规划算法,故不能保证目标在同一位置时机械臂的运行轨迹完全相同,但由于本套系统主要功能在于目标抓取,实验表明本文所用方法已经可以很好地满足要求,且机械臂在运行过程中轨迹较为流畅、平滑,图15即为机械臂在工作过程中其末端轨迹。



(a) 完成一次清理



(b) 完成多次清理

图15 机械臂的末端轨迹

Fig.15 End point trajectory of the manipulator

5 结束语

本文主要通过对Kinect深度摄像机获取的信息进行处理来确定目标抓取点信息,并利用基于工作空间的RRT算法对机械臂进行路径规划,与此同时用梯度投影法对RRT规划出的每一段路径进行逆运动学求解。通过构建一套基于Kinect的桌面清理系统对本文的研究内容进行了验证,实验结果表明基于Kinect的机械臂可以较理想地完成自主的目标抓取任务。

参考文献:

[1]方勇纯. 机器人视觉伺服研究综述[J]. 智能系统学报,

- 2008, 3(2): 109-114.
- FANG Yongchun. Summary of robot visual servo research [J]. CAAI Transactions on Intelligent Systems, 2008, 3(2): 109-114.
- [2] 王麟琨, 徐德, 谭民. 机器人视觉伺服研究进展[J]. 机器人, 2004, 26(3): 277-282.
- WANG Linkun, XU De, TAN Min. Progress of robot visual servo research[J]. Robot, 2004, 26(3): 277-282.
- [3] 李亮. 机械手对目标物体的抓取[D]. 成都: 西华大学, 2008: 11.
- LI Liang. Manipulator crawl on the target object [D]. Sichuan: Xihua University, 2008: 11.
- [4] GANGANATH N, LEUNG H. Mobile robot localization using odometry and Kinect sensor[C]//2010 IEEE International Conference on Emerging Signal Processing Applications (ESPA). NJ, USA, 2010: 91-94.
- [5] 庄佳园, 徐玉如, 万磊, 等. 基于雷达图像的水面无人艇目标检测技术[J]. 哈尔滨工程大学学报, 2012, 33(2): 129-135.
- ZHUANG Jiayuan, XU Yuru, WAN Lei, et al. Target detection of an unmanned surface vehicle based on a radar image[J]. Journal of Harbin Engineering University, 2012, 33(2): 129-135.
- [6] RAKPRAYOON P, RUCHANURUCKS M, ADACOUNDOU L. Kinect-based obstacle detection for manipulator[C]//2011 IEEE/SICE International Symposium on System Integration (SII). NJ, USA: 2011: 68-73.
- [7] 李红波, 丁林建, 冉光勇. 基于 Kinect 深度图像的人体识别分析[J]. 数字通信, 2012(4): 21-26.
- LI Hongbo, DING Linjian, RAN Guangyong. Human identification analysis of the depth of the image based on Kinect [J]. Digital Communications, 2012(4): 21-26.
- [8] 陆振. 冗余自由度机器人原理及应用[M]. 北京: 机械工业出版社, 2006: 1.
- [9] 李跃, 汪亚明, 黄文清, 等. 基于 OpenCV 的摄像机标定方法研究[J]. 浙江大学学报, 2010, 27(3): 417-421.
- LI Yue, WANG Yaming, HUANG Wenqing, et al. Camera calibration technique based on OpenCV[J]. Journal of Zhejiang University, 2010, 27(03): 417-421.
- [10] ZHANG Zhengyou. Flexible camera calibration viewing a plane from unknown orientations[C]//Proc International Conference on Computer Vision (ICCV'99). Corfu, Greece, 1999, 1: 666-673.
- [11] 高文娟, 李健. 基于 OpenCV 的摄像机标定问题研究[J]. 计算机与数字工程, 2008, 36(12): 128-130.
- GAO Wenjuan, LI Jian. Research on camera calibration technique based on OpenCV[J]. Computer and Digital Engineering, 2008, 36(12): 128-130.
- [12] 刘睿祯, 于仕琪. OpenCV 教程[M]. 北京: 北京航空航天大学出版社, 2009: 394-400.
- [13] 尚倩, 阮秋琦, 李小利. 双目立体视觉的目标识别与定位[J]. 智能系统学报, 2011, 6(4): 303-311.
- SHANG Qian, RUAN Qiuqi, LI Xiaoli. Target identification and location of binocular stereo vision[J]. CAAI Transactions on Intelligent Systems, 2011, 6(4): 303-311.
- [14] LIEGEOIS A. Automatic supervisory control of configuration and behavior of multibody mechanisms[C]//IEEE Trans Sys Man Cybernet. [S. l.], 1977: 868-871.
- [15] ZGHAL H, DUBEY R V, EULER J A. Efficient gradient projection optimization for manipulators with multiple degrees of redundancy[C]//Proc IEEE Int Con Robor and Automat. Cincinnati, USA, 1990, 2: 1006-1011.
- [16] LAVALLE S M, KUFFNER J. Randomized kinodynamic planning[C]//Proceedings of the 1999 IEEE International Conference on Robotics and Automation. Detroit, USA, 1999: 473-479.
- [17] LAVALLE S M, KUFFNER J J. RRT-Connect: an efficient approach to single-query path planning[C]//Proceedings of the 2000 IEEE International Conference on Robotics and Automation. San Francisco, USA, 2000: 998-1001.
- [18] BERTRAM D, KUFFNER J, DILLMANN R, et al. An integrated approach to inverse kinematics and path planning for redundant manipulators[C]//Proc IEEE International Conference on Robotics and Automation (ICRA). Orlando, USA, 2006: 1874-1879.

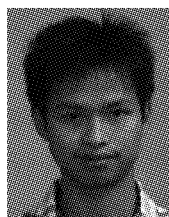
作者简介:



韩峥,女,1986年生,硕士研究生,主要研究方向为智能控制及机器人。



刘华平,男,1976年生,副教授,主要研究方向为智能控制及机器人、计算机视觉等。



黄文炳,男,1989年生,博士研究生,主要研究方向为智能控制及机器人。