



ELSEVIER

Contents lists available at ScienceDirect

Engineering

journal homepage: www.elsevier.com/locate/eng



Research
Robotics—Article

户外空中双机械手抓取设计和视觉伺服

Pablo Ramon-Soria^{*}, Begoña C. Arrue, Anibal Ollero

Robotics, Vision, and Control Group, School of Engineering, University of Seville, Seville 41092, Spain

ARTICLE INFO

Article history:

Received 30 October 2018

Revised 10 March 2019

Accepted 26 March 2019

Available online 13 November 2019

关键词

空中操纵

抓取设计

视觉伺服

摘要

本文介绍了一种配备有RGB-D摄像机的使用带有双机械手的无人驾驶飞行器 (unmanned aerial vehicle, UAV) 抓取已知物体的系统。空中操纵仍然是一项极具挑战性的任务。本文主要从三个方面对这一任务进行了评价: 目标检测与姿态估计、抓取设计、飞行中的抓取动作。人工神经网络 (artificial neural network, ANN) 首先被用来获得有关物体位置的线索。接下来, 使用对齐算法获取对象的六维 (six-dimensional, 6D) 姿态, 并使用扩展的卡尔曼滤波器进行滤波。然后, 使用物体的三维 (three-dimensional, 3D) 模型来估计空中机械手可实现良好抓取的排列清单。检测算法的结果 (即对象的姿态) 用于更新手臂朝向对象的轨迹。如果由于无人机的振荡而无法达到目标姿态, 则算法将切换到下一个可行的抓取。本文介绍了总体方法, 给出了每个模块的仿真实验结果和实际实验结果, 并提供了视频演示结果。

© 2020 THE AUTHORS. Published by Elsevier LTD on behalf of Chinese Academy of Engineering and Higher Education Press Limited Company This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. 引言

在工业应用中, 无人机的使用正在增加。无人机适用于操作人员无法到达或在通常情况下构成危险的情境下的应用, 如电力线路检查[1,2]、风力涡轮机维护[3]、设施中各种结构的检查[4]和摄影测量[5]。但是, 这些应用仅涉及感知任务。近年来, 已经证明空中机器人也可以有效地执行操纵任务[6]。

由于还存在其他要求, 空中操纵仍然是一项艰巨的任务。平台通常很不稳定, 并在空中保持不了稳定的平衡。为了执行操纵任务, 有必要为机器人配备如感知、设计和一般智能等各种功能, 还需要为机器人配备适当的硬件以执行所需的任务。在这些应用中引入机器人为解决问题提

供了更有效的方案, 减少了时间和金钱成本, 并减少了操作人员由于某些任务单调乏味而产生工作失误所导致的质量问题。另一方面, 可以赋予操作员一些机器人无法充当的更重要的角色。

操纵涉及使用机械手与物理对象进行交互, 这可能包括抓取对象、移动它们或保持与刚性对象的接触。为了执行此类任务, 机器人需要能够检测和定位对象, 以及分析如何与对象建立交互。另外, 为了执行这样的任务, 有必要对机械手的动作和运动进行设计, 并防止它们与环境 and 平台本身产生碰撞。

为能够执行操纵任务, 首先需要找到并跟踪所要操纵的对象。许多计算机视觉算法可用于此任务。这些视觉算法可以根据应用、计算资源和使用的传感器进行分类。

^{*} Corresponding author.

E-mail address: prs@us.es (P. Ramon-Soria).

通常使用用于对象的单目检测区域提议算法。这些算法可以基于简单的颜色算法[7,8]、特征检测[9,10]或机器学习。新型算法使用人工神经网络（artificial neural network, ANN）来检测预先训练的对象。得益于深度学习以及强大的图形处理单元（graphics processing unit, GPU）的使用，神经网络得到了提升，这些图形处理单元可以增强和并行化神经网络。最新的设计（SSD [11]、RCNN [12]、F-RCNN [13]、Mask-RCNN [14]和YOLO [15]）已经取代了大多数传统的目标检测算法。然而，检测彩色图像中的对象的实例不足以执行操纵任务，为了实现操作还必须分析对象的三维（three-dimensional, 3D）信息。为此，对深度神经网络（deep neural network, DNN）的最新研究已经扩展了ANN计算对象的六维（six-dimensional, 6D）位置 [16]。

诸如Kinect或Intel RealSense设备之类的深度传感器可用于直接从环境中提取3D信息[17,18]。但是，这些传感器在室外环境中通常受到较大的限制，因为它们使用的红外结构光在室外通常是难以解析的。相反，无源立体摄像机可以计算出视差图，该视差图可用于执行点的像素级三角剖分，从而形成点云。这种设备通常不如深度相机精确，但几乎在任何条件下都能正常工作。不过，由于它基于可见光谱，因此该设备在弱光条件下的性能很差。

图像特征可用于准确获取物体在空间中的位置[19–21]。这些思路有着良好的结果，并且对遮挡具有鲁棒性。但是，如果对象没有纹理或存在反射，这些算法可能会出现。其他作者使用了投影算法，该算法基于投影到图像上的几何模型，并使用单眼相机找到对象的位置和方向 [22]。

对于操纵本身，目前已经针对各种应用对机器人机械手进行了广泛的研究。机械手的机械和数学模型以及它们与对象的相互作用是机器人操纵分析的基本方面。对象和机械手在配置时的大量组合使这一研究领域具有挑战性。

Miller等[23]提出使用近似对象的图元形状，并以此来产生抓取动作。但是，此方法需要一个简单的对象，不能用于更复杂的形状。在参考文献[24]中，研究人员提出了一种算法，该算法将对象的形状近似为一组用于生成抓取的接触点的平面区域。最近的研究使用机器学习方法来产生更好的抓取动作。在这些方法中，使用合成数据集 [25–27]或强化学习[28]对系统进行训练。

其他方法使用来自传感器的数据来创建对象的概率模型。在参考文献[29–31]中，研究人员提出了一种使用高斯过程重建3D对象的算法，称为高斯过程隐式曲面。

一旦开始抓取，就必须对其进行评估。抓取质量[32]可以根据其几何形状、手指的力量和其他许多方面进行分析。在当前的工作中，我们集中于基础的方面，即强制闭合、最大的最小抵抗扭力（largest minimum resisted wrench）和面向任务的质量指标。

Marturi等[33]提出了一种离线/在线设计器来控制工业机器人的机械臂，以便使机械臂与人类操作者合作。该算法使用预先学习的离线抓取设计器生成一组可能的抓取，之后在线检测对象的姿态，并由第二设计器根据机器人的当前状态和对象的姿态生成一组可行的抓取。

但是，空中操纵比地面操纵更具挑战性。例如，在空中应用中，机器人和目标物体都在移动。此外，手臂的动态会影响空中平台的动态，从而损害其稳定性。一些出版物[34,35]研究了无人机的稳定性、控制性及其机械手。在当前的研究中，我们假设运动不是激进的，并且控制器的速度足够快，以补偿机械臂施加在平台上的半静态外力。

此外，为了实际执行抓取任务，必须运行视觉伺服算法来精确移动机械手。根据数据，视觉伺服或基于视觉的控制可以分为两种类型。如果该算法使用3D位置，则称为基于位置的视觉伺服（position-based visual servoing, PBVS）[36,37]；如果该算法仅使用图像中的2D信息，则将其命名为基于图像的视觉伺服[37,38]。第三种选择涉及信息的组合，即所谓的“混合算法”。本文提出了一种检测目标对象位置的PBVS方法。设计控制律是为了减少末端执行器的位置误差。

高空作业平台需要将自己放置在工作区内，以控制其位置。一种标准技术是使用运动捕捉系统（motion-capture, MOCAP）在研究过程中精确定位无人机。该系统以较高的频率定位机器人，这在测试新技术和算法时非常方便。户外任务还需要机器人的精确位置。GPS为在室外定位机器人提供了大规模而应用广泛的系统。但是，标准设备在操作任务中有令人无法接受的错误。为了解决这个问题，我们研究小组[39]提出了一种使用全站仪的方法，该方法将信息与空中机器人的惯性测量单元融合在一起，以便将机器人定位在户外的任何地方。

空中机械手可以根据手臂的配置进行分类，本工作着重于串行机械手的使用。Laiacker等[40]开发了一种配备有工业机械臂的直升机来执行操纵任务。当前对空中操纵的兴趣日益增加，研究人员将两个（而不是一个）机械手放置在同一平台上，以便其用两只手执行灵巧操作。Korpela等[41]提出在工业环境中使用一对机械手来转动阀门。现有研究已经发现，双机械手配置[42]对于执行操纵任务

有较高的通用性。

本文提出了一个完整的空中操纵系统的设计，其中包括开发用于无人机的机械手，用于检测需要操纵的物体的视觉感知系统以及用于引导手臂的视觉伺服系统。这项工作是在欧盟资助的项目AEROARMS (SI-1439 / 2015) 的框架下进行的。该项目旨在开发一种具有灵巧操纵功能的新型空中平台，以在工业环境中执行检查和维护任务。

这项工作将在AEROARMS项目中开发的履带机器人的模型用作检测算法和掌握实验的例子。同样的方法也可以应用于其他对象。

本文的其余部分安排如下：第2节介绍了为执行操纵任务而开发的空中机械手系统；第3节介绍了用于检测、跟踪和定位目标对象的视觉算法；第4节展示了抓取和重新计划算法，该算法考虑了机器人和目标的可能运动；第5节描述了执行任务的全局过程和为此设计的状态机；第6节展示了通过飞行机械手的实际抓取实验对系统进行的实验分析；最后，第7节讨论了这项研究的结果和未来将采取的行动。

2. 空中机械手

多轴飞行器已经被证明是执行检查和维护任务的实用解决方案[6]。与固定翼飞机不同，多轴飞行器可以在空间中自由移动并悬停在所需位置以执行任务。

该项目选择了共面六旋翼飞机来执行操纵任务。此配置比较小的多轴飞行器具有更高的稳定性和强度，可以满



图1. 装备在操纵器上的机械臂。

足执行任务的有效载荷要求。图1显示了用于实验的平台。平台框架是由DroneTools SL[†]设计和制造的六轴飞行器。

其余的硬件包括双机械手和软件是由来自塞维利亚大学的本文作者设计和开发的。

该平台有一个Pixhawk[‡]自动驾驶仪，负责机器人的飞行控制。底部安装了Intel NUC计算机，以运行从视觉算法到机械手控制在内的所有系统。最后，平台内置了英特尔实感摄像头以感知环境。

为了进行操纵，该多轴飞行器配备有一对机械臂。这些臂具有三自由度 (three-degree-of-freedom, 3DoF) 和集成且可互换的末端执行器，根据应用情况，它们可提供4~6个自由度。图1显示了带有抓爪作为末端执行器的臂的模型。使用两个机械手代替一个机械手在很多角度上是有利的。一方面，两个机械手可以执行更复杂的操作任务，因为每个手臂可以执行不同的操作。另外，具有两个末端执行器 (如两个抓爪) 可以使抓取更加稳定，因为可以从两个不同的侧面抓取物体。仅使用一个抓取器，抓取就更加受限，因为在许多情况下，施加在质心上的重力将不对称。因此，如果可能的话，唯一稳定的抓取位于目标顶部，这会将操纵限制在顶部具有手柄的目标上。

此处介绍的机械臂已在名为hecatonquiros^{††}的开源项目中发布，该项目的目标是通用、便宜且易于使用的机械臂。这些手臂设计轻巧，因此相对较小的无人机就可以携带它们。它们是3D打印的，因为这种材料相对便宜并且不需要后期处理，从而降低了整体生产成本，并可以直接组装。单臂的成本约为150美元 (包括智能串行伺服电机)。该项目还为运动学求解器提供了基于OpenRAVE的包装器，并支持在进行实际实验之前使用机器人操作系统 (robot operating system, ROS) 进行仿真。

图2 (a) 显示了这对手臂的运动学可达性，用于估计机器人执行操作任务的适当位置。图2 (b) 和 (c) 是为机器人定义的坐标系，其被用于将检测到的对象的坐标从相机的坐标转换为手臂的坐标系，并以此移动末端执行器。

如前所述，无人机配备了英特尔RealSense D435设备。选择该设备是因为它能够在户外获得合理准确的深度信息。视觉算法利用深度信息来获取对象的6D姿态以执行操纵过程。该摄像机还可以在近距离 (0.2 m) 处捕获深度，因此在短距离内也具有出色的质量，这使其成为机器人操

[†] <http://www.dronetools.es>.

[‡] <https://pixhawk.org/>.

^{††} <https://github.com/bardo91/hecatonquiros/>.

作任务中手眼协调的理想选择。

作为总结，表1概括了整个空中系统的规格。

3. 目标检测和姿态跟踪

本节介绍用于检测和跟踪目标对象的算法。目标检测是一项艰巨的任务，但是，深度学习技术的使用彻底改变了感知的方式。所提出的方法将卷积神经网络（convolutional neural network, CNN）与随机样本共识算法相结合，以检测目标并为空中机械手计算其姿态。该算法总结在图3中。

该算法基于两个阶段。首先，应用目标检测CNN生成候选对象。然后，使用对齐算法来计算目标对象的确切位置。

CNN如今已被广泛使用，并为许多困难问题提供了解决方案，如本研究中的目标检测。如果目标对象是已知的，则仅需要标记的图像数据库即可训练网络。不过，尽管这些算法高度可并行化，但它们在数学运算中比较烦琐。通常使用图形单元处理器（GPU），因为它们是专门为有效执行并行操作而设计的。为此，最常用的一些框架是TensorFlow [43]和Caffe [44]。这些框架为实施最新网络提供了广泛的“模型动物园”，从而加快了测试和开发过程。但是，必须考虑到最终算法将由无人机中的机载计算机执行，这可能会有一些限制，如计算统一设备体系结构（compute unified device architecture, CUDA）兼容性。在第6节中，比较了三种不同设备（具有GTX1070的笔记本电脑、Nvidia的Jetson TX1和英特尔NUC计算机）上不同算法（F-RCNN、SSD和YOLO）之间的比较。

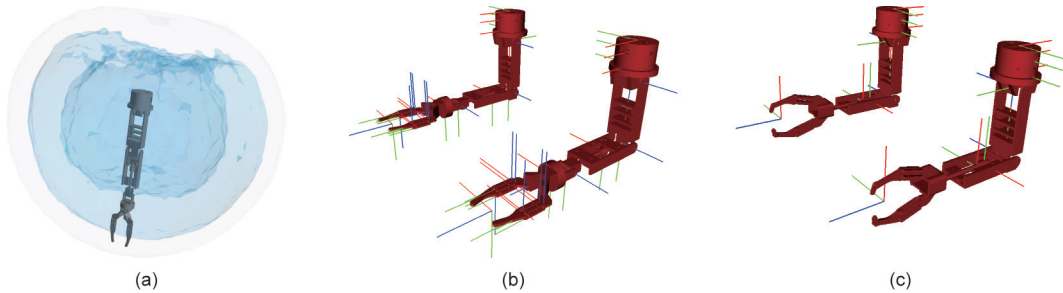


图2. 机械手的运动学可达性及用于在线模拟和设计的计算机辅助设计模型。(a) 空中机械手的运动学可达性；(b) OpenRAVE中的6自由度机械臂模型和坐标系；(c) 在OpenRAVE中建立4自由度机械臂模型和坐标系。

表1 平台特征

Subsystem	Characteristic	Specification	
Arms	DoF	4	6
	Maximum range	~500 mm	~580 mm
	Maximum payload per arm	~500 g	~400 g
	Mass per arm	632 g	663 g
UAV	Mass (including arms)	3.75 kg	
	Maximum payload	1.5 kg	
	Detection range	2.5 m	

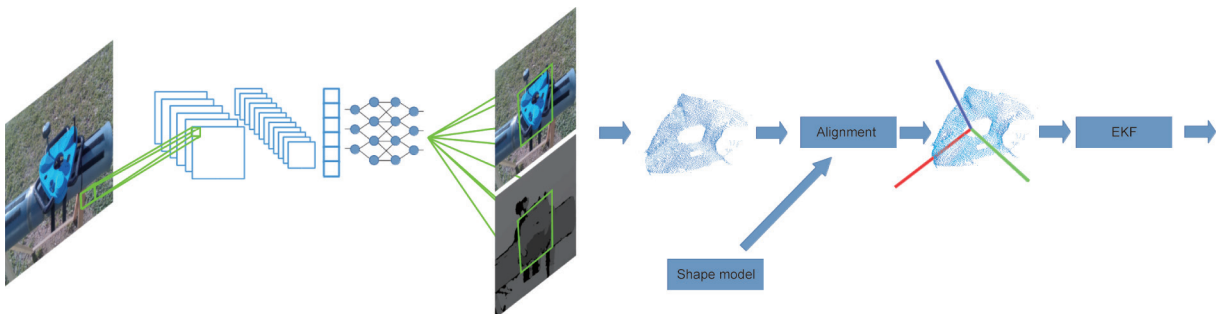


图3. 用于目标物体检测和姿态估计的视觉算法方案。EKF(extended Kalman filter): 扩展卡尔曼滤波器。

摄像机提供的RGB图像被送入CNN。结果，获得了一组候选边界框。具有最高概率的边界框被用于裁剪RGB和深度图像。裁剪的图像被用于计算局部点云，以将图片中的信息与设备的校准矩阵结合起来。在下一步中这些彩色的点云将被使用。

简而言之，使用目标检测算法可减少对齐阶段对对象的搜索，将可搜索区域减少50%，如果搜索器只占用图像的一小部分，则可减少多达80%。

此时，已经获得了包含目标对象的点云片段。为了定位对象，将使用对齐算法。该算法把对象从背景中过滤出来。一个搜索器的点云模型被用于执行基于迭代最近点(iterative closest point, ICP)的算法[45,46]，该算法将会输出目标的姿态。ICP方法使用所有可用信息来增强对齐过程。它包括点的3D位置[式(1)]、从模型和场景获得的法线信息[式(2)]以及颜色数据[式(3)]。在仿射变换的迭代计算之前，在选择点的过程中以剔除离群值的形式使用此信息。

$$D_{\text{dist}} = \sqrt{(x_1 - x_2)^2} \quad (1)$$

式中， D_{dist} 是点云的任意两个点 x_1 和 x_2 之间的几何距离。

$$D_{\text{norm}} = \cos[\angle(A_1, A_2)] \quad (2)$$

式中， D_{norm} 是法线 A_1 和 A_2 之间的夹角的余弦值； A_1 和 A_2 分别是点云的任意两个点 x_1 和 x_2 的法线。

$$D_{\text{color}} = |R_1 - R_2| + |G_1 - G_2| + |B_1 - B_2| \quad (3)$$

式中， D_{color} 是分配给点云的任意两个点 x_1 和 x_2 的颜色之间的绝对距离； R 、 G 、 B 分别是红色、绿色和蓝色通道的值。

值得注意的是，ICP算法对初始条件敏感。因此，在前一时刻($k-1$)中计算出的姿态(k 是前一次计算出的云的对应时刻)被用于为算法提供良好的估计值。这个事实，加上输入云的裁剪和清理，使算法收敛到正确解的可能性最大化。

最后，算法的结果由扩展的卡尔曼滤波器(EKF)过滤[47,48]。该滤波器有助于减少结果中的噪声，并有可能预测对象的相对速度，这可用于改善对对象姿态的预估。研究选择了一个简单的运动学模型来估计物体的状态。

4. 抓取设计

为了能够抓取目标物体，开发了抓取设计算法，该算法包括以下步骤：首先，考虑目标的形状，生成一组可行的抓取；接下来，将这些抓取根据其属性进行排列；最后，根据计算出的每个瞬间的视觉信息选择最佳的抓取方式。在抓取过程中，如果对象的姿态使当前的抓取无法实现，则设计器将选择最接近的最佳抓取方式。

4.1. 抓取动作的生成

首先，在目标对象的表面上生成一组可能的接触点。我们的方法基于参考文献[49]中描述的抓取生成算法。该算法将一组光线从边界框投射到目标对象，每条射线与表面碰撞一次，然后计算表面法线。一个使用该算法的例子如图4所示。

但是，此方法可能无法为形状复杂的对象生成可行的抓取方案。图5展示了用于U形物体的射线追踪算法的示例。可以看出，如果抓取器的孔径小于图5(b)中标为“a”的尺寸，则该物体对于机器人来说是无法抓取的。但是，如果该算法考虑了对象表面的所有可能的内部角点，且如果抓取器的孔径大于图5(b)中标为“b”或“d”的尺寸，

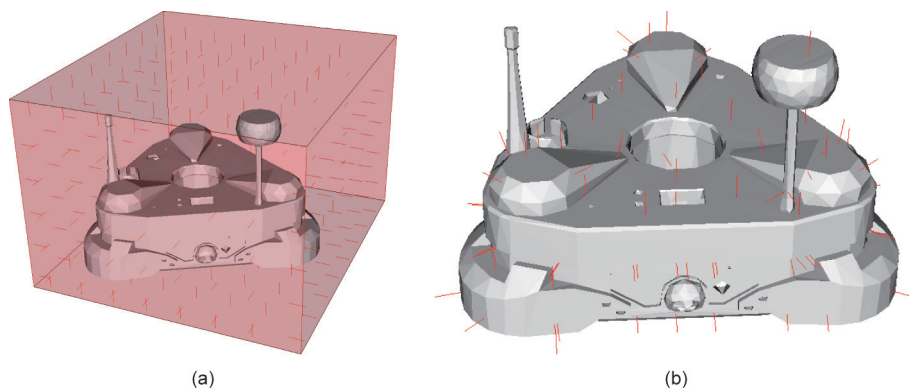


图4. 用于接触点计算的射线追踪算法示例。(a) 用于计算抓取点的初始射线；(b) 使用射线跟踪算法计算得到的抓取点。

那么机器人就可以抓取对象。

一般而言，令 p 为穿过物体的单射线， p 与网格的碰撞次数为 $M_{\text{collisions}} = M_{\text{folds}} \times 2$ ；因此，可能产生抓取的相对接触点的组合数量为：

$$M_{\text{grasps}} = \sum_{j=1}^{M_{\text{folds}}} j \quad (4)$$

式中， M_{grasps} 是可能的抓取次数； M_{folds} 是射线反射的层数； j 是SUM运算符的内部循环变量。

如果对象的网格过于复杂，则生成候选项可能会占用大量时间。为了加快算法的这一步，算法将生成的接触点存储在二进制文件中，这样，如果对象相同，则可以在任何任务中重用它们。一旦执行了该光线跟踪，未转换的结果就会存储在数据库中；如果以后要掌握同一对象或同一对象的不同实例，则将恢复该数据库，从而避免了所有重复的光线跟踪计算。

然后，将接触点集用于生成候选抓取设计。根据对象相对于空中机械手的姿态，在内部模拟器中对抓取进行采样。该算法计算每次抓取的可达性、机械手与目标对象之间可能发生的碰撞以及每次抓取的质量。首先，每个手臂采样所有可能的候选解，以生成每个手臂的可行抓取集。接下来，对所有可行的抓取进行成对测试，以获取双重抓取的列表。这些抓取的质量根据力封闭和最大的最小抵抗扭力进行分析，如参考文献[32]所述。然后将所有这些抓取信息存储起来，以在设计过程中使用。

4.2. 设计、伺服和抓取

布置好抓取列表后，将根据当前的相对目标对象姿态选择最佳抓取。这是双机械手最紧密、最稳定的抓取。在抓取之前，将手臂移到捕捉前位置；这可以防止机械手执行大型危险动作，因为防止机械手与机器人的飞行组件碰

撞非常重要。

假设它是由描述其关节状态的标量变量 $\Theta = \theta_1, \dots, \theta_n$ 指定的多链接机器人的机械手 L 。主体的每个链接的末端都有一个特定的位置 $S = s_1, \dots, s_n$ ，其中一些是所谓的末端执行器。可以使用Denavit-Hartenberg (DH)公式[50]将这些位置描述为机械手关节之间的转换链，并通过一组参数 $\text{DH}_i = \{\alpha_i, a_i, d_i, \varphi_i\}$ 进行计算。 α_i 表示关节的 z 轴之间的角度； a_i 是沿着第一关节的 x 轴方向的关节之间的距离； d_i 是沿着第一关节的 z 轴方向的两个关节之间的距离； φ_i 是沿着第一个关节的 z 轴方向的关节的 x 轴之间的角度。每个 DH_i 元素都是一个矩阵，该矩阵在链接 $i-1$ 到 i 之间转换；也就是说，它执行正向运动学（forward kinematics, FK）。

目的是获得一组适当的接头 Θ ，该接头将末端执行器放置在目标位置 T_r 上，即执行逆向运动学。这个问题已被广泛研究。遗憾的是，在大多数情况下没有简单的解决方案。迭代方法[51]和采样方法[52]非常适合此问题。FK由函数 $\Theta \mapsto S(\Theta)$ 即 $\mathbb{R}^N \mapsto \mathbb{R}^6$ 给出，该函数描述从 N 维空间到6D空间的FK，该FK由机械手末端执行器的位置和方向组成。这些函数可以使用接近当前关节状态 Θ 的雅可比矩阵 $[J(\Theta)]$ 线性近似。速度可以表示为：

$$\dot{S}(\Theta) = J(\Theta)\dot{\Theta} \quad (5)$$

式中， $S(\Theta)$ 代表末端执行器在关节空间 Θ 中的位置。

$$J(\Theta) = \begin{pmatrix} \delta s_m \\ \delta \theta_n \end{pmatrix}_{m,n} \quad (6)$$

式中， δ 是偏导数； θ_n 是关节空间的特定关节； m 和 n 是用来定义完整的雅可比矩阵 $J(\Theta)$ 的变量的索引迭代器（index iterator）。

最终目的是利用来自传感器的视觉信息将手臂引导

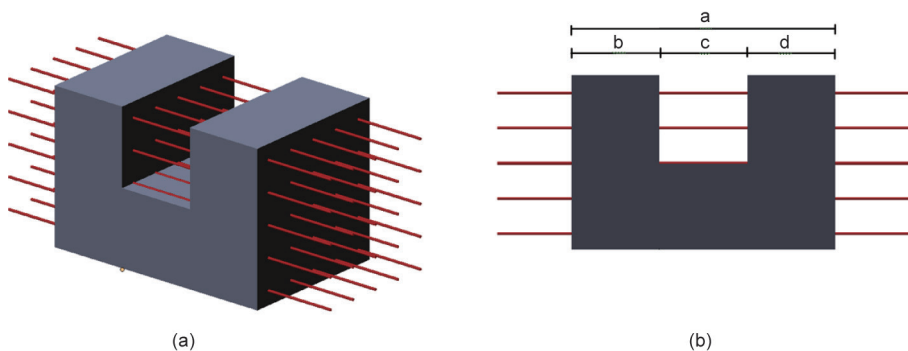


图5. 对于非凸对象使用抓取算法的示例。(a) 穿过物体的光线的等距视图；(b) 前视图，红线代表用于追踪可能的抓点的射线。

向目标抓取。在这项工作中，采用了雅可比阻尼最小二乘（damped-least square, DLS）梯度下降法[53,54]来确保手臂朝目标位置汇聚。事实证明，该方法对反方法和伪逆方法[54]更稳健，接近于雅可比矩阵中的不稳定性 and 奇异性。令 X_k 和 Q_k 为由抓取计划给出并由视觉算法更新的末端执行器的定向目标位置。目的是通过更新其关节来更新机械手的姿态。

DLS进行如下操作：式（7）和式（8）根据机器人的当前状态和末端执行器的目标姿态来计算；接下来，式（9）用于计算一个矢量，该矢量包含机器人关节值的增量，该增量允许末端执行器朝目标姿态移动。

$$\text{error} = \begin{cases} X_{\text{target}} - X_K \\ Q_{\text{target}} - Q_K \end{cases} \quad (7)$$

式中， X_{target} 和 Q_{target} 是末端执行器的目标位置； X_k 和 Q_k 是 K 处瞬间末端执行器的目标位置和方向。

$$J_c = \begin{bmatrix} J_X \\ J_Q \end{bmatrix} \quad (8)$$

式中， J_c 、 J_X 和 J_Q 是当前状态下对应的雅可比位置和方向。

$$\Delta\theta = \left[J_c^T(\theta_K) \cdot J_c(\theta_K) + \lambda^2 \cdot I \right]^{-1} \cdot \text{error} \quad (9)$$

式中， θ_k 代表当前关节的手臂角度； λ 是阻尼系数，用于减少与矩阵求逆有关的问题。 λ 必须足够大，以确保算法的行为足以接近奇点，但又不能太大，以免不能保证良好的收敛速度。 I 表示大小为 J_c 的单位矩阵。这些是根据目标对象的位置计算的，该位置在视觉模块中计算。

此外，使用一对机械手还需要其他约束。主要约束是需要避免与其自身以及与其他对象的碰撞。但是，检查任何非凸对象的碰撞都并非易事。在某些情况下，使用凸包是有利的，因为可以用来检查它们。但是，这种简化在某些应用中可能很困难。现代方法将物体分成一组凸包[55,56]。这样可以得到更准确的解决方案，同时保留使用凸包的优势和简化的数学方法。当前工作中使用了OpenRAVE给出的实现[57]，它集成了这些快速方法。

最后，由于空中机器人的振动，或者如果目标物体是可移动的，都可能无法获得最佳抓取。该算法在单独的线程中对照抓取数据库连续测量当前抓取的质量，以克服此问题。如果抓取变得不可行，算法将切换到新的最佳选择。

5. 状态机和完成任务

本文介绍的工作涵盖了无人机从平台起飞到抓取目标物体后降落的整个过程。如此完整的任务需要复杂的系统设计。该系统由5个模块组成，如图6和图7所示。每个模块负责无人机中的单个任务。为了简化开发过程，模块之间的通信由ROS进行[58]。在本节中，每个模块均以空闲状态初始化，该空闲状态由全局信号Start触发。

图6（a）展示了全局状态机，它负责任务的全局行为。它由三种状态组成：接近、抓取和归位。接近会触发Take_off信号并等待，直到无人机起飞。之后，全局状态机将接近信号并携带一个初始航路点，发送到靠近目标物体的无人机控制器。一旦无人机到达目标位置，它就会一直保持到找到目标物体为止。然后状态机切换到抓取状态，在该状态下，视觉伺服开始起作用以抓取目标。抓住物体后，状态机将切换到归位状态，该状态将在无人机降落时结束。

图6（b）显示了无人机控制器的状态机，它由6个状态组成。它从空闲状态开始。无人机起飞后，它将一直悬停直到下一个信号。如果全局状态机发送了接近信号，则无人飞行器控制状态机将切换到航点状态，并将无人飞行器移至目标位置。如果找到了目标，则它将切换到跟踪状态。跟踪模块负责检查目标物体是否处于要抓取的合理位置。如果是这样，它将发送接近信号。当物体丢失或抓住物体时，此状态结束。

图7（a）显示了视觉状态机，它具有最简单的结构。该状态机连续运行视觉算法，以检测目标对象并计算其姿态。当找到对象并计算出其姿态时，状态机将该信息与Found信号一起发布。如果丢失了对象，则发送丢失的信号以引起适当的效果。

图7（b）显示了抓取设计器模块的状态机，该状态机负责计算对象的适当抓取。它还根据目标抓地力运行伺服算法来计算所需的手臂关节。它从估计抓取的已安排列表开始。之后，如果对象靠近手臂范围内，则状态机将切换到伺服状态，在该状态下，视觉伺服会发布手臂的目标关节。如第4节所述，手臂不会直接被发送到抓取姿态。而是将它们发送到靠近它的位置，称为“接近姿态”。一旦手臂处于接近位置，状态机就会切换到抓取状态，在该状态下，手臂会快速朝着对象移动并抓住它。

最后，图7（c）显示了手臂控制器状态机。机械手控制器负责发布有关机械手的信息，并提供用于移动它们的界面。它还限制了手臂的速度和轨迹，以确保其安全。为

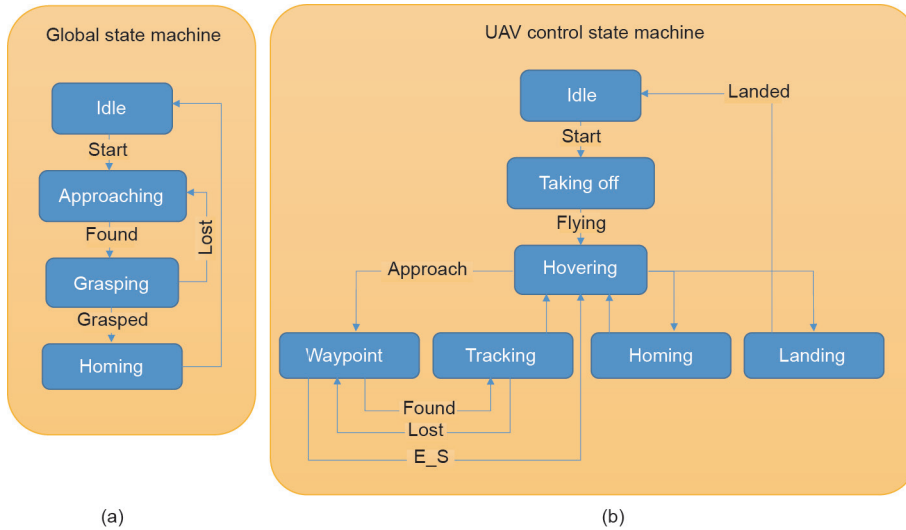


图6. 系统中的前两个子模块。(a) 全局状态机；(b) 无人机控制状态机。E_S: 紧急停止。

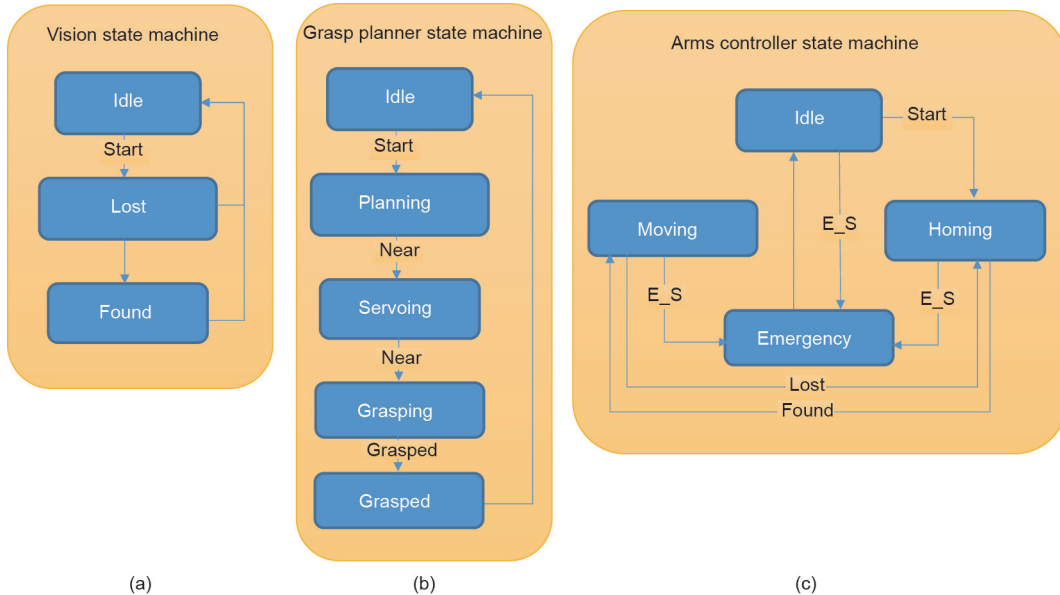


图7. 系统的前三个子模块。(a) 视觉状态机；(b) 抓取设计器状态机；(c) 手臂控制器状态机。

了安全起见，已设置紧急停止（E_S）状态以从外部使手臂停止。紧急停止还会使无人机保持悬停状态，以防止其坠毁。

6. 实验

本节首先描述了为评估和验证系统和算法而进行的实验。接下来，本文将描述和讨论一组实验，以显示系统在执行空中操纵任务中的性能。如前所述，这项工作着重于

抓取在项目AEROARMS期间开发的、用于工厂管道检查的履带机器人。但是，该算法对任何对象都能同样有效地使用，因为视觉算法和计划过程都没有对此特定对象的任何特定限制。可以从网站上找到实验的视频摘要[†]。

6.1. 系统评估

如第3节所述，在视觉算法的第一阶段，使用目标检测CNN来检测对象。这为计算成本更高的对齐算法提供了良好的初始解决方案。

[†] <https://youtu.be/nXYIzqwM8kA>.

为了选择最佳的选项，使其满足操纵任务的系统要求和空中机械手的有效载荷限制，在三个不同的设备中对三个CNN进行了测试。测试的特定CNN是F-RCNN、SSD300和YOLO（微型YOLO v2）。将它们以下三种设备中进行了测试：装有GTX1070的笔记本电脑、杰特逊TX1和具有虹膜GPU的英特尔NUC。前两个设备具有CUDA功能，因为它们具有Nvidia GPU。但是，由于缺少CUDA兼容性，第三种设备无法使用通用框架。表2总结了不同设备中不同检测算法的平均计算时间。由于英特尔NUC不兼容CUDA，因此仅评估了YOLO。具体来说，是YOLO的OpenCL实现[†]。

从表中可以清楚地看到，笔记本电脑的性能超过了其他设备的性能。但是，由于有效载荷的严格限制，在使用无人机时只能考虑其他两个设备。首先，YOLO在Jetson TX1设备中的运行速度要快一些。但是，英特尔NUC计算机的CPU功能大大超过了Jetson。由于整个系统需要使用其余算法的许多流程，因此选择了英特尔NUC平台。

区域提议算法（CNN）的主要好处在于：它可以大大减少对齐过程中要使用的云的大小（在50%~80%之内）。它具有双重改进，因为它减少了此步骤的计算时间，同时还允许算法收敛于一个最优解；由于梯度下降算法通常对异常值和初始条件敏感，因此可以将收敛性降至最低。

一旦检测到物体并估计其姿态，该算法便会执行第4节中所述的抓取计划过程。这个过程可以分为三个阶段。第一阶段是接触点的计算，此过程可以预先计算，因为它仅取决于为算法选择的分辨率和对象的模型。第二阶段是分别计算每个手臂的可行握力。表3总结了各个对象每个阶段的平均计算时间。最后，将成对的抓取按照其质量指标进行排列。第二列评估生成初始接触点集所花费的时间。网格分辨率是在第4.1节中所述的光线跟踪评估期间，使用的初始光线组之间的距离。第三列显示了评估上一步生成的抓取的可行性所花费的时间。最终，第四列显示了评估抓取的质量所花费的时间。

为了防止手臂与高空作业平台的任何部分发生碰撞，须在手臂的肩膀上方留一个安全的空间。该体积对应于机器人的飞行部分，绝不能与之碰撞。出于安全原因，在设计过程和伺服算法中会考虑该体积。

图8显示了在OpenRAVE中运行的在线模拟序列，以进行抓取计划过程。之后，将一组可行的抓取信息存储起来，以供以后在重新计划阶段使用。

6.2. 模拟和试验台实验

为了在实际实验之前验证手臂的性能及其控制，进行了一组模拟。所要抓取的物体的姿态带有一定的噪声。然后对这些对象进行动画处理，以强调手臂伺服算法。显示了三个具有不同形状的对象。但是，在此阶段可以选择任何其他对象，因为该算法能够计算任何网格的抓地力。

图9显示了一系列模拟实验的快照，以演示在不同情况下手臂的可达性。在这些模拟实验中，算法与实际实验一样进行，但是对象的姿态由模拟器提供，并且具有一定的噪声水平。首先，如第4.1节所述，该算法计算出一组

表2 被测设备中不同算法的计算时间

Computer	F-RCNN (s)	SSD (s)	YOLO (s)
Laptop	0.067	0.027	0.0103
Jetson TX1	0.470	0.113	0.0510
Intel NUC	—	—	0.0530

表3 每个网格面用于抓取计划过程的计算时间

Computer	CP generation (grid resolution 0.015 mm) (s)	Feasibility (per grasp) (s)	Quality (per grasp) (s)
Laptop	0.407	0.0072	0.0051
Jetson TX1	2.534	0.0113	0.0100
Intel NUC	0.417	0.0077	0.0506

CP: contact point.



图8. 抓取计划过程的不同样本。

[†] <https://github.com/ganyc717/Darknet-On-OpenCL>.



图9. 在不同时间使用不同对象进行的模拟测试样本。

可行的抓取。接下来，根据对象的位置，会选择最佳的抓取并将手臂朝对象引导，如第4.2节所述。一旦手臂靠近抓取姿态，系统就会发送信号以闭合抓取器。

这些实验的目的是证明不管视觉如何，该算法都能够处理不同的对象和形状。它生成一组可行的抓取，并根据对象姿态的估计动态选择最佳选项。

在实际飞行之前，整个系统（包括视觉系统）都在测试台上进行了测试（图10）：空中机械手被放置在固定的结构中，并移动目标物体。

6.3. 飞行实验

本节描述了在真实环境中演示系统性能的实验。为进行实验打印了一个爬行机器人的模型，准备了两个不同的设置。在第一个室内的设置中，根据MOCAP系统获得的位置，无人机执行了完全自主的操作。第二种设置旨在对室外系统进行测试。两个实验均选择了4DoF夹持器，因为它比6DoF版本更坚固。

第一个关键模块是视觉系统，该视觉系统计算要进行抓取计划的被抓取物体的姿态，将手臂模块送入伺服系统，并为无人机提供参考位置。图11显示了该算法的结果。图11（a）展示了使用CNN在RGB图像中的目标检测结果。图11（b）显示了使用深度图像计算的点云片段，以及根据卡尔曼滤波器过滤后的对齐算法估算出的物体姿态。如图所示，姿态估计通过重叠的红点云和坐标系突出显示。

图12展示了如何使用MOCAP系统从视觉系统对照地面实况对物体的姿态进行估算。估算显示均方根误差（root mean square error, RMSE）的平均值小于0.02 m。该准确性对于确保机器人能够执行抓地任务至关重要。在估计机

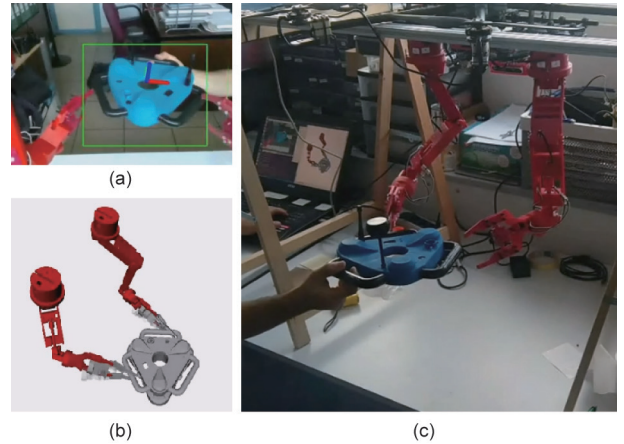


图10. 在室内测试台上测试整个系统。(a) 相机视图及履带模型的二维检测；(b) 机器人的3D虚拟化，包括履带和目标抓取的估计位置；(c) 实验的第三视角。

机器人的位置时，较大的误差会带来问题，从而导致机器人错过抓取，甚至与环境相撞。

图13显示了另一个测试视觉检测模块的实验。在此实验中，实体模型沿管道移动。无人机和履带的位置都通过视觉系统和MoCap系统进行了测量。

该系统的准确性首先通过MoCap系统在室内的受控环境下进行测试。任务是从无人机起飞开始的；然后，无人机执行第5节中说明的管道操纵。图14提供了该实验的快照。图15显示了两次视觉伺服测试期间的关节值。虚线对应于目标关节值，实线对应于实际关节值。该图显示了当先前的抓取变得无法到达时系统如何切换到新的抓取。

最后，图16提供了户外实验测试台的照片。在此实验中，出于安全原因，用安全绳将无人机系在结构上飞行。该照片显示了抓取爬行器模型时机器人的视角。该实验表

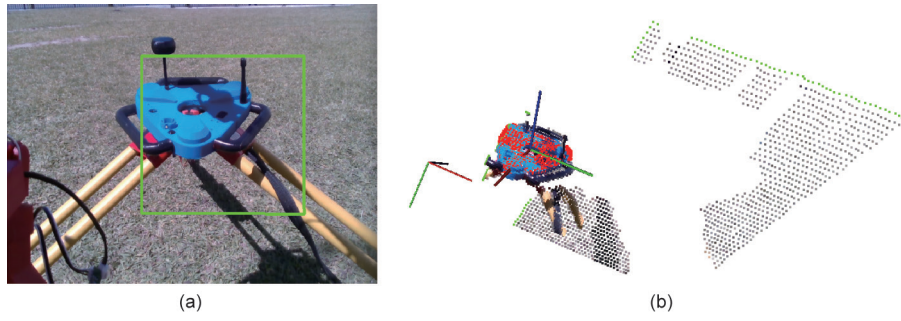


图11. 物体检测和姿态估计结果。(a) YOLO产生的限位框;(b) 基于重叠红点模型和坐标系的云片段及姿态估计。

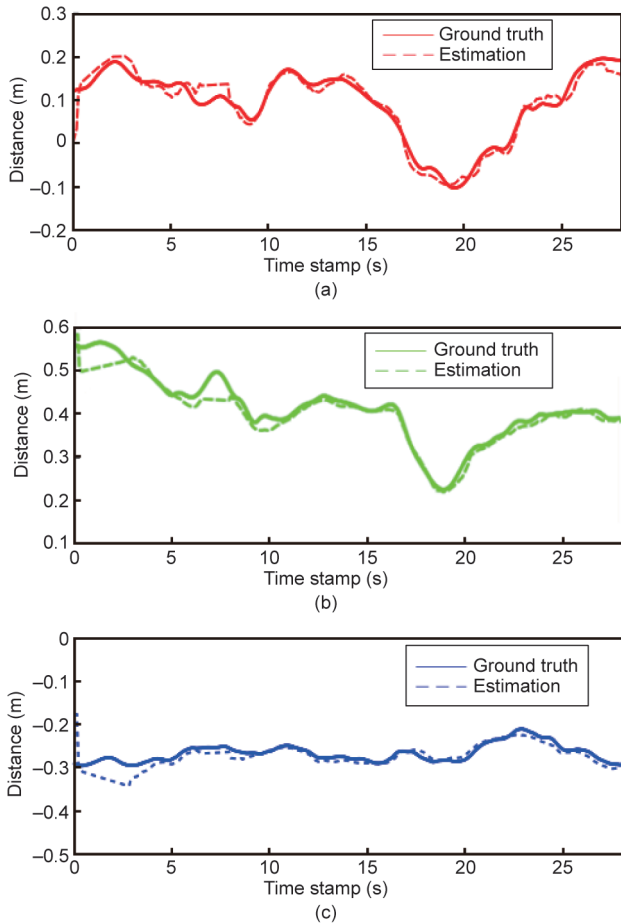


图12. 估计的履带姿态与无人机坐标系中的地面实况。(a) 相机的估计距离与X轴上的地面真实情况;(b) 相机的估计距离与Y轴上的地面真实情况;(c) 相机的估计距离与Z轴上的地面真实情况。

明, 无论光照条件如何, 感知算法都可以使用RealSense深度相机在户外工作。

7. 结论

本文提出了一个用于在空中平台上执行操纵的完整系统。集成两个机械手, 而不只是一个, 就可以抓取更大、

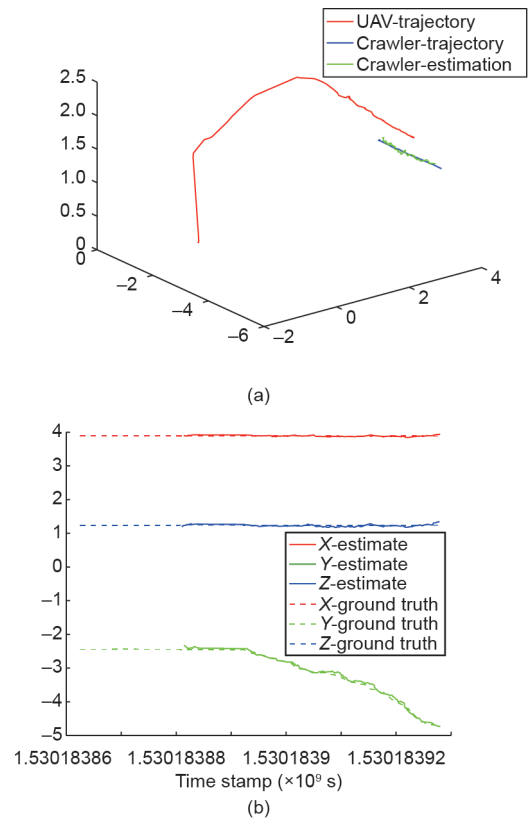


图13. 无人机的轨迹及由视觉系统在整体坐标(实线)和地面实况(虚线)中测量的履带的估计位置。(a) 在移动实验期间, 无人机和履带的3D轨迹;(b) 使用每个轴上的视觉估算值估算履带的位置与实际位置。

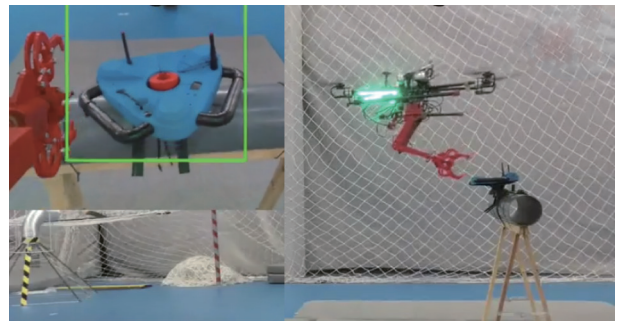


图14. 使用MOCAP系统的室内自主任务。

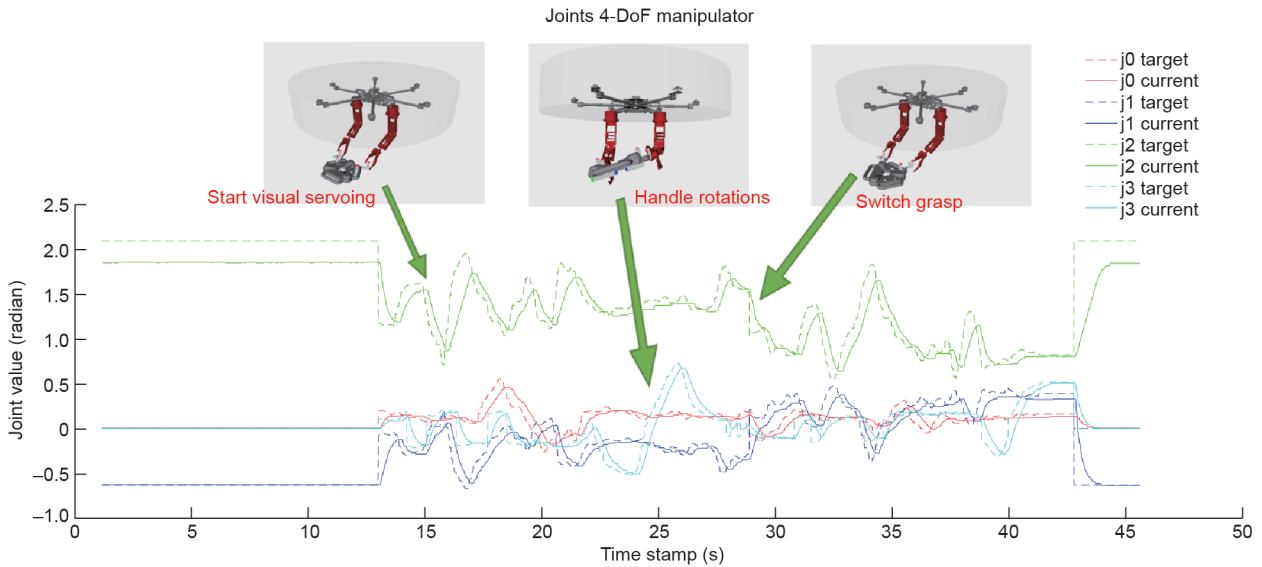


图15. 实验期间的关节值（以弧度为单位）。首先，手臂跟随抓取目标。在某一点后，对象旋转，因此系统切换到另一个可行的抓取。



图16. 附加安全绳的户外自主测试。

更复杂的对象。此外，拥有两个机械手有助于无人机消除由于外部干扰而引起的波动，甚至可以选择更稳定的抓取，因为它们包围了质量中心。提供了定性和定量实验数据，以证明该系统能够执行抓取操作。

视觉系统的平均RMSE小于0.02 m。已经证明，使用物体检测网络对于姿态估计的速度非常有益。主要限制来自可用的机载设备。大多数最先进的CNN都已经过测试和设计，可用于功能强大的计算机，这使得将它们集成到空中平台更具有挑战性。但是，在第6节中进行分析之后，所选算法为任务提供了足够的结果。下一步将包括估计神经网络中的物体姿态。

已经在几种环境中证明了该系统能够操作和抓取目标物体。此外，这些结果已得到模拟对象的支持，以证明抓取计划算法能够计划其他形状的操纵任务。视觉系统已在

弱光条件下以及户外直射阳光下进行了测试。该算法在两种光照条件下都能正确运行。

Acknowledgement

This work was carried out in the framework of the AEROARMS (SI-1439/2015) EU-funded project and the national project ARMEXTENDED (DPI2017-89790-R).

Compliance with ethics guidelines

Pablo Ramon-Soria, Begoña C. Arrue, and Anibal Ollero declare that they have no conflict of interest or financial conflicts to disclose.

References

- [1] Pouliot N, Richard PL, Montambault S. Linescout technology opens the way to robotic inspection and maintenance of high-voltage power lines. *IEEE Power Energy Technol Syst J* 2015;2(1):1–11.
- [2] Matikainen L, Lehtomäki M, Ahokas E, Hyyppä J, Karjalainen M, Jaakkola A, et al. Remote sensing methods for power line corridor surveys. *ISPRS J Photogramm Remote Sens* 2016;119:10–31.
- [3] Wang L, Zhang Z. Automatic detection of wind turbine blade surface cracks based on UAV-taken images. *IEEE Trans Ind Electron* 2017;64(9):7293–303.
- [4] Almadhoun R, Taha T, Seneviratne L, Dias J, Cai G. A survey on inspecting structures using robotic systems. *Int J Adv Robot Syst* 2016;13(6):1729881416663664.
- [5] Reinoso JF, Gonçalves JE, Pereira C, Bleninger T. Cartography for civil engineering projects: photogrammetry supported by unmanned aerial vehicles. *Iran J Sci Technol Trans Civ Eng* 2018;42(1):91–6.
- [6] Ruggiero F, Lippiello V, Ollero A. Aerial manipulation: a literature review. *IEEE Robot Automat Lett* 2018;3(3):1957–64.
- [7] Gevers T, Smeulders AWM. Color-based object recognition. *Pattern Recognit*

- 1999;32(3):453–64.
- [8] Bruce J, Balch T, Veloso M. Fast and inexpensive color image segmentation for interactive robots. In: Proceedings of 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2000 Oct 31–Nov 5; Takamatsu, Japan. New York: IEEE; 2000. p. 2061–6.
- [9] Lowe DG. Object recognition from local scale-invariant features. In: Proceedings of the 7th IEEE International Conference on Computer Vision; 1999 Sep 20–25; Corfu, Greece. New York: IEEE; 1999. p. 1150–7.
- [10] Zhou H, Yuan Y, Shi C. Object tracking using SIFT features and mean shift. *Comput Vis Image Underst* 2009;113(3):345–52.
- [11] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C, et al. SSD: single shot multibox detector. 2015. arXiv:1512.02325.
- [12] Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition; 2014 Jun 23–28; Washington, DC, USA. New York: IEEE; 2014. p. 580–7.
- [13] Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. 2015. arXiv:1506.01497.
- [14] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. 2017. arXiv:1703.068702017.
- [15] Redmon J, Farhadi A. YOLOv3: an incremental improvement. 2018. arXiv:1804.02767.
- [16] Xiang Y, Schmidt T, Narayanan V, Fox D. PoseCNN: a convolutional neural network for 6D object pose estimation in cluttered scenes. 2018. arXiv:1711.00199.
- [17] Rai A, Patchaikani PK, Agarwal M, Gupta R, Behera L. Grasping region identification in novel objects using Microsoft Kinect. In: Proceedings of 19th International Conference on Neural Information Processing; 2012 Nov 12–15; Doha, Qatar. Berlin: Springer; 2012. p. 172–9.
- [18] Lysenkov I, Eruhimov V, Bradski G. Recognition and pose estimation of rigid transparent objects with a Kinect sensor. In: Roy N, Newman P, Srinivasa S, editors. Robotics: science and systems VIII. Cambridge: MIT Press; 2013. p. 273–80.
- [19] Gordon I, Lowe DG. What and where: 3D object recognition with accurate pose. In: Ponce J, Hebert M, Schmid C, Zisserman A, editors. Toward category-level object recognition. Berlin: Springer; 2006. p. 67–82.
- [20] Collet A, Berenson D, Srinivasa SS, Ferguson D. Object recognition and full pose registration from a single image for robotic manipulation. In: Proceedings of the 2009 IEEE International Conference on Robotics and Automation; 2009 May 12–17; Kobe, Japan. New York: IEEE; 2009. p. 48–55.
- [21] Ramon Soria P, Arrue BC, Ollero A. Detection, location and grasping objects using a stereo sensor on UAV in outdoor environments. *Sensors* 2017;17(1):103.
- [22] Marchand E, Bouthemy P, Chaumette F, Moreau V. Robust real-time visual tracking using a 2D-3D model-based approach. In: Proceedings of the 7th IEEE International Conference on Computer Vision; 1999 Sep 20–27; Kerkyra, Greece. New York: IEEE; 1999. p. 262–8.
- [23] Miller AT, Knoop S, Christensen HI, Allen PK. Automatic grasp planning using shape primitives. In: Proceedings of 2003 IEEE International Conference on Robotics and Automation; 2003 Sep 14–19; Taipei, China. New York: IEEE; 2003. p. 1824–9.
- [24] Xue Z, Xia S, Dillmann R. An efficient grasp planning algorithm based on decomposition of grasp regions. In: Proceedings of 12th IEEE-RAS International Conference on Humanoid Robots; 2012 Nov 29–Dec 1; Osaka, Japan. New York: IEEE; 2012. p. 686–91.
- [25] Song D, Huebner K, Kyrki V, Kragic D. Learning task constraints for robot grasping using graphical models. In: Proceedings of 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2010 Oct 18–22; Taipei, China. New York: IEEE; 2010. p. 1579–85.
- [26] Saxena A, Driemeyer J, Ng AY. Robotic grasping of novel objects using vision. *Int J Robot Res* 2008;27(2):157–73.
- [27] Morrison D, Corke P, Leitner J. Closing the loop for robotic grasping: a realtime, generative grasp synthesis approach. 2018. arXiv:1804.05172.
- [28] Baier-Lowenstein T, Zhang J. Learning to grasp everyday objects using reinforcement-learning with automatic value cut-off. In: Proceedings of 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2007 Oct 29–Nov 2; San Diego, CA, USA. New York: IEEE; 2007. p. 1551–6.
- [29] Dragiev S, Toussaint M, Gienger M. Gaussian process implicit surfaces for shape estimation and grasping. In: Proceedings of 2011 IEEE International Conference on Robotics and Automation; 2011 May 9–13; Shanghai, China. New York: IEEE; 2011. p. 2845–50.
- [30] Björkman M, Bekiroglu Y, Högman V, Kragic D. Enhancing visual perception of shape through tactile glances. In: Proceedings of 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2013 Nov 3–7; Tokyo, Japan. New York: IEEE; 2013. p. 3180–6.
- [31] Martens W, Poffet Y, Ramón Soria P, Fitch R, Sukkarieh S. Geometric priors for Gaussian process implicit surfaces. *IEEE Robot Autom Lett* 2017;2(2):373–80.
- [32] Roa MA, Suárez R. Grasp quality measures: review and performance. *Autono Robots* 2015;38(1):65–88.
- [33] Marturi N, Kopicki M, Rastegarpanah A, Rajasekaran V, Adjigble M, Stolkin R, et al. Dynamic grasp and trajectory planning for moving objects. *Autono Robots* 2019;43(5):1241–56.
- [34] Ruggiero F, Trujillo MA, Cano R, Ascorbe H, Viguria A, Peréz C, et al. A multilayer control for multirotor UAVs equipped with a servo robot arm. In: Proceedings of 2015 IEEE International Conference on Robotics and Automation; 2015 May 26–30; Seattle, WA, USA. New York: IEEE; 2015. p. 4014–20.
- [35] Suarez A, Jimenez-Cano AE, Vega VM, Heredia G, Rodríguez-Castaño A, Ollero A. Lightweight and human-size dual arm aerial manipulator. In: Proceedings of 2017 International Conference on Unmanned Aircraft Systems; 2017 Jun 13–16; Miami, FL, USA. New York: IEEE; 2017. p. 1778–84.
- [36] Hutchinson S, Hager GD, Corke PL. A tutorial on visual servo control. *IEEE Trans Robot Autom* 1996;12(5):651–70.
- [37] Chaumette F, Hutchinson S. Visual servo control. I. basic approaches. *IEEE Robot Autom Mag* 2006;13(4):82–90.
- [38] Kim S, Seo H, Choi S, Kim HJ. Vision-guided aerial manipulation using a multirotor with a robotic arm. *IEEE/ASME Trans Mechatron* 2016;21(4):1912–23.
- [39] Prada Delgado J, Ramon Soria P, Arrue BC, Ollero A. Bridge mapping for inspection using an UAV assisted by a total station. In: Ollero A, Sanfeliu A, Montano L, Lau N, Carreira C, editors. Robot 2017: third Iberian robotics conference. Cham: Springer; 2018. p. 309–19.
- [40] Laiacker M, Huber F, Kondak K. High accuracy visual servoing for aerial manipulation using a 7 degrees of freedom industrial manipulator. In: Proceedings of 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2016 Oct 9–14; Daejeon, South Korea. New York: IEEE; 2016. p. 1631–6.
- [41] Korpela C, Orsag M, Oh P. Towards valve turning using a dual-arm aerial manipulator. In: Proceedings of 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2014 Sep 14–18; Chicago, IL, USA. New York: IEEE; 2014. p. 3411–6.
- [42] Suarez A, Heredia G, Ollero A. Design of an anthropomorphic, compliant, and lightweight dual arm for aerial manipulation. *IEEE Access* 2018;6:29173–89.
- [43] Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: large-scale machine learning on heterogeneous distributed systems. 2016. arXiv:1603.04467.
- [44] Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, et al. Caffe: convolutional architecture for fast feature embedding. 2014. arXiv:1408.5093.
- [45] Rusinkiewicz S, Levoy M. Efficient variants of the ICP algorithm. In: Proceedings of Third International Conference on 3-D Digital Imaging and Modeling; 2001 May 28–Jun 1; Quebec City, QC, Canada. New York: IEEE; 2001. p. 145–52.
- [46] Men H, Gebre B, Pochiraju K. Color point cloud registration with 4D ICP algorithm. In: Proceedings of 2011 IEEE International Conference on Robotics and Automation; 2011 May 9–13; Shanghai, China. New York: IEEE; 2011. p. 1511–6.
- [47] Liu W, Hu J, Fang Y, Shao Q, Zheng K, Zhu GN. Real time pose estimation based on extended Kalman filter for binocular camera. In: Proceedings of 2016 Asia-Pacific Conference on Intelligent Robot Systems; 2016 Jul 20–22; Tokyo, Japan. New York: IEEE; 2016. p. 142–6.
- [48] Hajimolahoseini H, Amirfattahi R, Khorshidi S. Real-time pose estimation and tracking of rigid objects in 3D space using extended Kalman filter. In: Proceedings of 2014 22nd Iranian Conference on Electrical Engineering; 2014 May 20–22; Tehran, Iran. New York: IEEE; 2014. p. 1545–9.
- [49] Diankov R. Automated construction of robotic manipulation programs [dissertation]. Pittsburgh: Carnegie Mellon University; 2010.
- [50] Barrientos A, Peñín LF, Balaguer C, Aracil R. Fundamentals in Robotics. New York: McGraw Hill; 1997.
- [51] Goldenberg A, Benhabib B, Fenton R. A complete generalized solution to the inverse kinematics of robots. *IEEE J Robot Autom* 1985;1(1):14–20.
- [52] Courty N, Arnaud E. Inverse kinematics using sequential monte carlo methods. In: Perales FJ, Fisher RB, editors. Articulated motion and deformable objects. Berlin: Springer; 2008. p. 1–10.
- [53] Wampler CW, Leifer LJ. Applications of damped least-squares methods to resolved-rate and resolved-acceleration control of manipulators. *J Dyn Sys Meas Control* 1988;110(1):31–8.
- [54] Buss SR. Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods. *IEEE J Robot Autom* 2004;17(1–19):16.
- [55] Ghosh M, Amato NM, Lu Y, Lien JM. Fast approximate convex decomposition using relative concavity. *Comput Aided Des* 2013;45(2):494–504.
- [56] Mamou K, Ghorbel F. A simple and efficient approach for 3D mesh approximate convex decomposition. In: Proceedings of 2009 16th IEEE International Conference on Image Processing; 2009 Nov 7–10; Cairo, Egypt. New York: IEEE; 2009. p. 3501–4.
- [57] Diankov R, Kuffner J. OpenRAVE: a planning architecture for autonomous robotics. Pittsburgh: Robotics Institute; 2008.
- [58] Quigley M, Gerkey B, Conley K, Faust J, Foote T, Leibs J, et al. ROS: an opensource robot operating system. In: Proceeding of IICRA Workshop on Open Source Software; 2009 May 12–17; Kobe, Japan. New York: IEEE; 2009. p. 1–6.