机器人重排列任务中基于语义的物体匹配

许聚展

摘要

物体匹配问题是机器人的重排列任务中非常重要的一个环节，准确的物体匹配可以保证后续机器人的操作成功率。本文复现的工作就是这个问题研究上的最新成果。这篇论文提出了一种新的物体匹配方法，该方法使用预训练的大型视觉语言模型，通过利用语义和视觉特征计算物体之间的相似性，我复现的代码最后得到了较好的匹配成绩，并且成果应用到了模拟环境下的机器人重排列任务中。

关键词：机器人重排列；物体匹配；视觉语言模型

1 引言

作为机器人与环境交互的一项关键能力，物体重新排列通常涉及物体检测、识别、抓取和高级规划。该任务的研究工作中，人们经常使用描述所需场景布局的图像作为一种目标的表示方法。这涉及到一个关键挑战：如何准确推断机器人面前的物体与所提供的目标图像中的哪件物体是匹配的。它也被称为物体匹配问题。这个问题是机器人的重排列任务中非常重要的一个环节，准确的物体匹配可以保证后续算法指导机器人的重排列操作成功率。因此我希望能进一步了解现有物体匹配算法的研究情况，以便对机器人重排列任务有更加细致的理解，所以将复现选题定为物体匹配问题。最终本次课题我要复现的论文题目为：*Semantically Grounded Object Matching for Robust Robotic Scene Rearrangement*[1]。它是发表在 ICRA 2022 的最新工作，一定程度上代表了当前该问题上算法能达到的最佳效果。本篇工作复现的论文发现现有工作的一个普遍局限：源图像和目标图像共同出现的物体实例必须相同。这一前提一定程度上提高了训练数据收集的难度，同时也限制了算法实际部署的范围。由此这篇工作提出了一种新的物体匹配方法，该方法使用大型预训练的视觉语言模型，通过利用语义和视觉特征作为一种更稳健、更普遍的相似性衡量标准，在跨实例环境下匹配物体。结果也证明了算法对匹配性能的提升，可用于指导机器人操纵者从一个与机器人场景不共享对象实例的图像中进行多对象重新排列。

2 相关工作

现有的一系列物体重排列任务都在强调视觉信息的重要性，目前常用目标图像作为表示任务目标状态的视觉信息的手段。在机器人平台上进行桌面重排列的流程中会把算法进行模块化[2-4]，作为执行流程的第一个模块就是物体匹配：对当前场景和目标图像场景中的物体进行检测，然后对两个场景中的物体进行匹配。之后再让机器人通过规划抓取和放置动作去实现对应物体的转移变换。但这些方法中的匹配模块只处理相同实例的对应关系，不能依据场景之间的语义进行夸实例的匹配，为此这篇文章尝试在机器人任务中利用对象语义进行指导。

早期为了解决物体匹配问题，研究人员提出了一些人工设计的视觉特征（如颜色直方图）[5]。近些年内，随着深度学习的发展，从 CNN 分类器提取的特征也开始被应用于解决匹配问题[2,4]，这些基于神经网络提取的高维特征已经在大规模数据集上进行了训练，并证明了对干扰因素（如照明和 2D

旋转）的不变性。然而这些特征还没有被训练成对输入物体的实例偏移保持不变，因此当源图像和目标图像包含语义相同对象的不同实例时，这类算法匹配性能会降低。物体匹配任务属于模板匹配这一领域下的研究，近些年也有不少使用深度学习的方法进行基于图像的物体模板匹配[6-7]，但这些实例匹配方法都需要在源图像和目标图像之间物体是相同实例。而本次复现的工作处理了一个更一般的情况，利用视觉语义基础，实现了具有相同语义描述的任意不同实例的匹配。

已经有许多尝试通过将语言指令与视觉观察相结合来实现语言引导机器人操控的工作。虽然和直接使用目标图像的指导模式不同，但这些工作在帮助机器人消除视觉世界歧义方面与本篇文章的目标是相关的。之前已经有对 CLIP 模型[8]提取的文本特征进行模拟学习，以提高通用性的研究[9]。最近的几项工作提出引导机器人从场景中选择特定对象的系统，语言信息就是是为参考表达式进行输入的[10-11]。本次复现的工作希望使用语言作为解决视觉歧义的机制，去处理任意对象类别的场景。

3 本文方法

3.1 本文方法概述

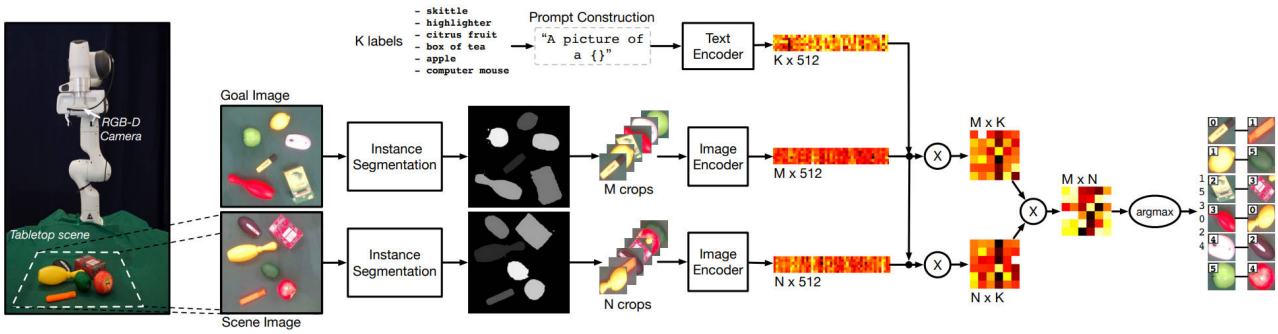


图 1: 方法示意图

本工作的算法流程如图 1所示。对于一个机器人重排列任务中获取到的当前状态场景图片（源图片）以及用户提供的目标图片，该算法首先使用物体分割算法[12]将其中的每个物体区域裁剪出来，假设当前状态的源图片得到 N 个物体的裁剪图片，目标图片得到了 M 个物体裁剪图像，将这些图片集合经过一个图片编码器得到各自的图像特征。场景中可能出现的 K 个物体语义类别则作为文本输入经过一个文本编码器得到文本特征。根据图像特征与文本特征我们可以计算出两个分类矩阵 C_s 和 C_t，分别表示源图像与目标图像中每个物体属于这 K 个物体类别的置信度。借由这两个分类矩阵，我们可以进一步计算出源图像中物体与目标图像中物体之间的相似性矩阵。其中图像编码器与文本编码器均为预训练好的 CLIP 编码器[8]。得到物体之间的相似性矩阵后，我们可以使用匈牙利算法进行物体匹配，得到匹配分数最高的组合。

3.2 对比算法设计

为了体现语义信息对匹配算法带来的效果提升，文章设计了 3 种不同的算法进行对比。第一种称为 CLIP-V，它是只使用图像编码器的匹配网络，根据两组图像特征直接计算相似性矩阵。第一种算法的执行伪代码如下所示：

Procedure 1 CLIP-V 算法执行伪代码

Input: 源图片 *S,* 目标图片 *T*
Output: 物体相似度矩阵 *M*
$\{s_i\}_N$ = 实例分割 (*S*)
$\{t_i\}_M$ = 实例分割 (*T*)
F_s = 图像编码器 ($\{s_i\}_N$)
F_t = 图像编码器 ($\{T_i\}_M$)
$M = F_t F_s^T$

第二种称为 CLIP-K，该算法使用词库中所有的 K 个物体类别作为语义输入，假设场景中出现的物体类别数量是 N，那么 $K > N$。最后一种称为 CLIP-N，这个算法只把任务场景中出现的 N 个物体语义进行语义编码。CLIP-K 与 CLIP-N 的执行代码如下所示：

Procedure 2 CLIP-K 与 CLIP-N 算法执行伪代码

Input: 源图片 *S,* 目标图片 *T,* 物体语义词汇 *L*
Output: 物体相似度矩阵 *M*
$\{s_i\}_N$ = 实例分割 (*S*)
$\{t_i\}_M$ = 实例分割 (*T*)
F_s = 图像编码器 ($\{s_i\}_N$)
F_t = 图像编码器 ($\{T_i\}_M$)
F_l = 文本编码器 (*L*)
$C_s = F_s F_l^T$
$C_t = F_t F_l^T$
$M = C_t C_s^T$

4 复现细节

由于本文并没有开源代码，所以本次课题主要是复现论文算法，并使用相同的数据集进行测试评估复现效果。本次实验使用的 CLIP 语言模型代码来自 openAI 官方开源的 CLIP python 库，我使用其预训练好的文本编码器与图像编码器来实现本论文的算法。进行物体匹配之前对图像进行物体分割的代码来自 NVlabs 的项目：https://github.com/NVlabs/UnseenObjectClustering。

4.1 实验设置

本次实验使用的数据集为：Large Vocabulary Instance Segmentation (LVIS) 数据集[13]。该数据集包含 100170 张图片，涵盖 1203 个物体类别，大概有 127 万份语义标记。该数据集中每个类别都含有一个“出现频率”，是数据集的维护团队统计的一个属性，他们根据物体出现的次数将这些物体类别分为“稀有”、“普通”与“频繁”3 大类。我从出现频率属于“频繁”的物体类别中选出了 400 个类别，每个类别至少有 10 张对应的图片，这个子数据集将作为我测试复现算法的数据集。

测试的任务称为“N-way problem”，每次实验时，会从数据集中随机选择 N 个物体类别，每个类别会随机选出两张包含该类物体的图片，将包含该物体的区域图片裁剪出来；然后每一类的这两张图片分别作为该类物体的源图片与目标图片，我们将 N 个类的源图片集合与目标图片集合作为算法输入，求得相似性矩阵，最后基于相似性矩阵和匈牙利算法求解出两组图片中物体的匹配结果。我使用的是 scipy 算法库实现的匈牙利算法来进行最终的匹配。用匹配的准确率衡量算法的效果。

5 实验结果分析

根据 4.1的设置，我们进行了 8-way 和 20-way 两种实验，我们对比了 3.2种提到的 3 种算法，分别进行 500 次测试，统计匹配的平均准确率，其结果如表 1所示。可以看到，只使用视觉信息进行物体匹配的效果最差，增加了语义信息后算法的匹配效果有显著提升。从 CLIP-N 的结果可以看到，如果我们提供的场景信息越准确，算法的匹配效果也会更好。

表 1: 本次复现算法得到的匹配准确率

Model	8-way	20-way
CLIP-V	54.2%	40.3%
CLIP-K	72.4%	54.1%
CLIP-N	**77.1%**	**57.7%**

表 2展示的结果为文章原文里面的准确率，由于我们无法保证测试数据集和原作者的完全一致，因此测试的数值上会有所偏差，但最终的结论和原文的一致。

表 2: 算法原文得到的匹配准确率，数据来自[1]

Model	8-way	20-way
CLIP-V	40.3%	27.4%
CLIP-K	52.9%	37.8%
CLIP-N	**58.8%**	**40.1%**

6 机器人重排列应用

论文最后还搭建了一个真机的执行环境进行实验。我也参考了文章的设置，在模拟器中搭建了一个机器人重排列的环境，将前面复现的匹配算法进行应用。

6.1 环境设置

我们使用的模拟环境为 NVIDIA Omniverse，它是 NVIDIA 开发的仿真平台。如图 2所示，我模拟器中布置了一个带有真空吸盘的机器臂用于抓取物体，在机器人前面一个平台用于摆放要进行重排列的物体。物体模型来自 graspnet 数据集[14]。我们在白色平台的正上方设置了一个观察相机，用它来获取源图像和目标图像。

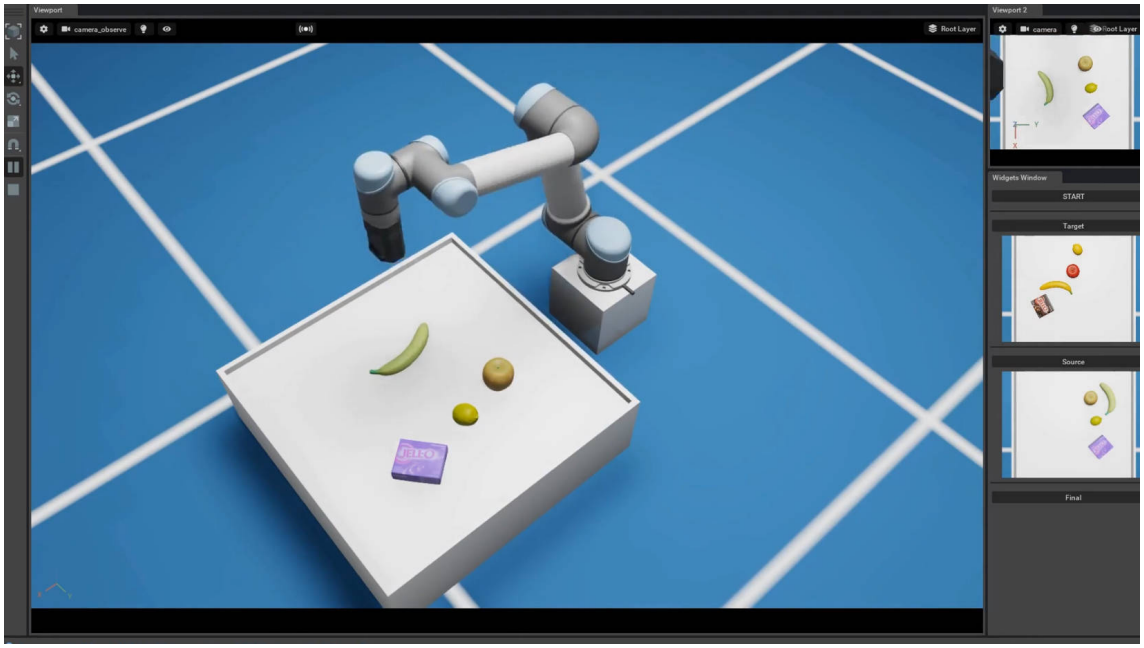
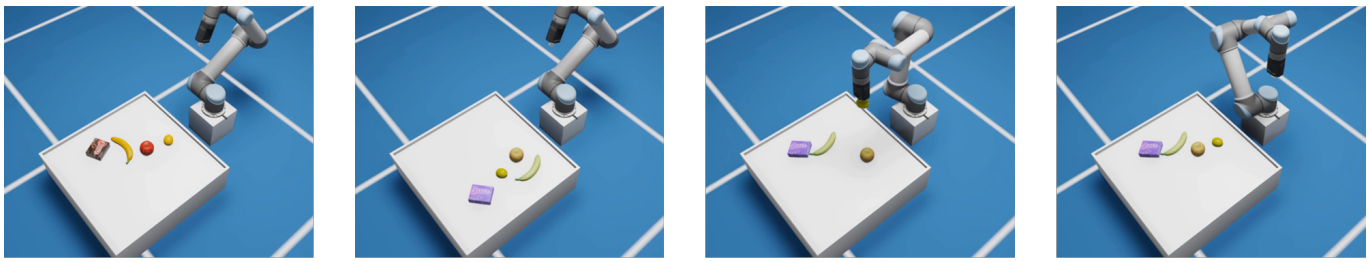


图 2: 机器人重排列场景

6.2 实验流程

图 3展示了机器人重排列的过程。首先我们现在平台上摆放好物体，用相机拍摄得到目标状态的图像，移除这些物体后，用不同材质的同类物体作为要摆放的对象随机放置在平台上，重新用观察相机拍摄得到源图像，使用实例分割算法提取出两张图片各自的物体区域，将其作为匹配算法的输入得到两组物体的对应关系。相机在获取彩色图像的同时我们也会保存其深度图，我们结合深度图和物体分割区域得到每个物体的空间位置，将其作为机器人抓取的位置，让机器臂从物体上方从上往下抓取，之后移动到对应的目标位置上放下。由于本课题重点并不涉及重排列任务中的规划，所以这里的实验不考虑物体转移前后是否发生位置冲突，因此设置初始状态和目标状态的时候并不会让两者的物体区域有重叠。



(a) 目标状态　(b) 起始状态　(c) 物体抓取转移　(d) 重排列结果

图 3: 机器人重排列流程

7 总结与展望

本次课题复现的工作是物体匹配任务上的最新论文，他们引入场景的语义信息进一步提升了匹配算法的效果。我基于论文的算法设计将其复现，并在同一个数据集上进行匹配测试，得到了与原文相同的实验结论。最后我搭建了一个机器人重排列环境，用实现好的匹配算法去指导该任务。不过由于本课题重点并不涉及重排列任务中的规划，所以机器人实验中没有考虑排列顺序，同时也没有考虑目标物体的姿态，当前实现的结果只是简单地把物体放到目标区域内。因此后续我可以从更加复杂的顺序规划以及考虑摆放姿态的抓取这两个方面进行补充完善，让整个重排列系统更加完整鲁棒。

参考文献

[1] GOODWIN W, VAZE S, HAVOUTIS I, et al. Semantically grounded object matching for robust robotic scene rearrangement[C]//2022 International Conference on Robotics and Automation (ICRA). 2022: 11138-11144.

[2] QURESHI A H, MOUSAVIAN A, PAXTON C, et al. Nerp: Neural rearrangement planning for unknown objects[J]. arXiv preprint arXiv:2106.01352, 2021.

[3] DANIELCZUK M, MOUSAVIAN A, EPPNER C, et al. Object rearrangement using learned implicit collision functions[C]//2021 IEEE International Conference on Robotics and Automation (ICRA). 2021: 6010-6017.

[4] LABBÉ Y, ZAGORUYKO S, KALEVATYKH I, et al. Monte-carlo tree search for efficient visually guided rearrangement planning[J]. IEEE Robotics and Automation Letters, 2020, 5(2): 3715-3722.

[5] SWAIN M J, BALLARD D H. Color indexing[J]. International journal of computer vision, 1991, 7(1): 11-32.

[6] MERCIER J P, TROTTIER L, GIGUERE P, et al. Deep object ranking for template matching[C]//2017 ieee winter conference on applications of computer vision (wacv). 2017: 734-742.

[7] MERCIER J P, GARON M, GIGUERE P, et al. Deep template-based object instance detection[C]//Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2021: 1507-1516.

[8] RADFORD A, KIM J W, HALLACY C, et al. Learning transferable visual models from natural language supervision[C]//International Conference on Machine Learning. 2021: 8748-8763.

[9] SHRIDHAR M, MANUELLI L, FOX D. Cliport: What and where pathways for robotic manipulation [C]//Conference on Robot Learning. 2022: 894-906.

[10] ZHANG H, LU Y, YU C, et al. Invigorate: Interactive visual grounding and grasping in clutter[J]. arXiv preprint arXiv:2108.11092, 2021.

[11] MEES O, BURGARD W. Composing pick-and-place tasks by grounding language[C]//International Symposium on Experimental Robotics. 2021: 491-501.

[12] XIANG Y, XIE C, MOUSAVIAN A, et al. Learning rgb-d feature embeddings for unseen object instance segmentation[C]//Conference on Robot Learning. 2021: 461-470.

[13] GUPTA A, DOLLAR P, GIRSHICK R. Lvis: A dataset for large vocabulary instance segmentation[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 5356-5364.

[14] FANG H S, WANG C, GOU M, et al. GraspNet-1Billion: A Large-Scale Benchmark for General Object Grasping[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR). 2020: 11444-11453.