基于 Transformer 的语义物体编排

摘要

语义物体编排问题是机器人领域的一个热门课题，准确的物体编排可以大大提高机器人的可用性。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基于 Transformer 的网络架构用于生成满足语义信息的排列，复现代码后发现点云之间经常发生一些碰撞，原因就是网络本身对碰撞并没有先显式约束。为了解决这一问题，引入了碰撞判别器，一旦发生碰撞就重新采样，从而有效地剔除不合理的结果。

关键词： Transformer；语义物体编排；机器人操控

1 引言

物体编排是机器人与日常环境互动中的一项关键任务，涉及许多日常场景，如整理工具、设置餐桌等。这一任务对于实现高效、自动化的家庭和工作环境至关重要。为了降低操作者的负担，提高交互的便捷性，机器人通常需要以用户友好的方式接受指令。因此，利用自然语言作为输入的物体编排成为了机器人研究领域的一个热门课题 [1]。这种方法不仅提高了交互的自然性，还使得非专业用户也能轻松指挥机器人完成复杂任务。在这种语义物体编排的问题设置中，机器人通过处理语言指令和物体的视觉观测信息，输出每个物体的最佳放置位置和姿态即变换矩阵。

2 相关工作

物体编排的相关工作大致可以分为三类，第一类是基于目标状态图像的方法 [3, 6]：这种方法要求用户提供一个整理好的目标图像作为参考。机器人需要识别图像中的物体和布局，并尝试还原这种状态。虽然这种方法直观，但它依赖于精确的目标图像，并且可能对场景中的小差异非常敏感，更重要的是这种算法的假设性太强，对于用户来说提供一张图像是一件繁琐的事情。第二类则是基于生成式方法 [2, 4]：这种方法利用语言指令和对初始场景的观察来生成一个目标状态。然后，机器人将这个目标状态与初始场景进行匹配，以计算出物体的变换矩阵。这种方法的优势在于它能够处理更抽象的指令，但它可能需要复杂的模型来理解语言和视觉信息，并生成合理的目标状态且生成式模型有很大的随机性，可能需要对生成的目标状态做多次采样和筛选。第三类则是让模型去学习语义和物体的信息，以一种端到端的方式计算出物体的变换矩阵 [5, 7]，。这种方法的优势在于它可以直接从输入数据中学习到必要的变换，而不需要显式地定义目标状态。而本次复现论文题目《StructFormer: Learning Spatial Structure for Language-Guided Semantic Rearrangement of Novel Objects》 [5] 这篇论文属于

第三类方法，是 2022ICRA 的一篇工作，介绍了一种名为 StructFormer 的基于 Transformer 的神经网络，使机器人能够根据高级语言命令识别和重新排列对象，形成复杂的结构。

3 本文方法

StructFormer 将物体编排看成选择指令相关的物体和生成物体位姿两个阶段，可以说整个框架分为 Object Selection Network 和 Language Conditioned Pose Generator 两个模块。如前面所说，StructFormer 属于一种端到端的方法，这要求模型能够接受多模态的输入，且能够理解语义关系和物体的空间关系。

3.1 Object and Sentence Encoders

多模态的 transformer 往往是将各个模态的信息透射到相同的隐空间中，因此会对物体点云和语言指令进行编码。在论文实现中对物体的编码器是一个 point cloud transformer(PCT) 模型，将 Transformer 架构作为置换不变函数来处理无序点云数据，语言指令则使用一个学习到的 mapping，将单词转化为向量的形式，如上图所示 $\tilde{c}_i$ 是文本的隐变量，$\tilde{e}_i$ 则是物体点云的隐变量，另外还有 postion embedding 是 Transformer 本来就有的位置编码向量，需要注意的是 Type embedding 用于表示不同类型的数据。最后将各个隐变量 contact 起来得到物体 $e_i = [\tilde{e}_i; p_i; r_i]$ 和文本 $c_i = [\tilde{c}_i; p_i; r_i]$ embedding。

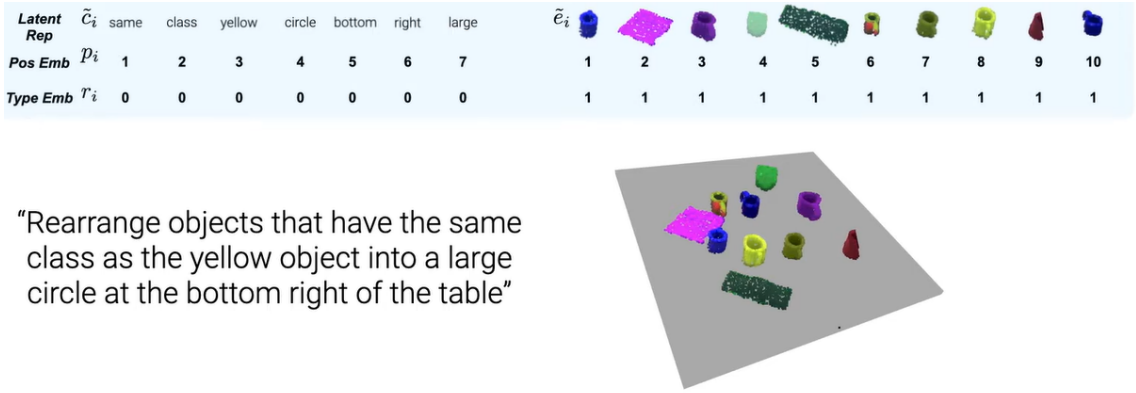

图 1. Object and Sentence Encoders

3.2 Object Selection Network

物体选择网络 k_Φ 可以看成是一个 Transfomer Encoder 去编码物体和文本的隐向量，并将编码器的输出到全连接层直接预测物体选中的概率 k_i，其损失函数是 binary cross entropy loss。这里简单说明一下上图所示中的 Pose Generator Encoder π_Ω 和 k_Φ 是一样的模型，所以写在了一起，详见 3.3 节。

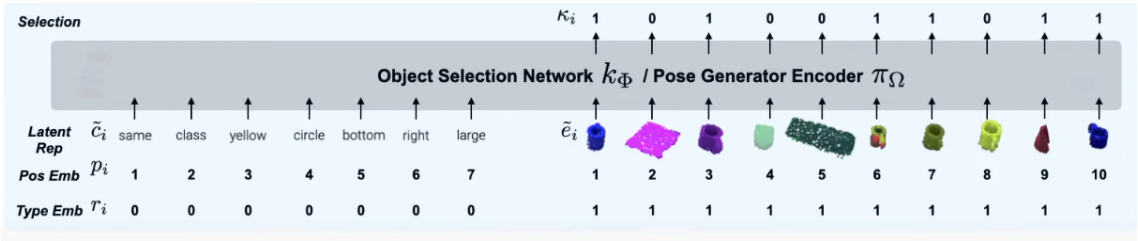

图 2. Object Selection Network

3.3 Language Conditioned Pose

Language Conditioned Pose Generator 则是和常规的 Transformer 类似，是一个 Encoder 和 Decoder 的结构。上一小节提到，Pose Generator Encoder π_Ω 和 Object Selection Network k_Φ 结构是完全一样的。Decoder 则是以 Object Selection Network 选择的物体的隐变量 e_i 以及上一个物体预测的位姿 δ_i 作为输入，自回归的输出下一个物体的位姿，即：

$$(e_0, [\delta_0; e_1], [\delta_1, e2], ..., [\delta_{N_q-1}, e_{N_q}]) \rightarrow (\delta_0, \delta_1 ..., \delta_{N_q})$$

其损失函数则是预测的位姿和 Ground-Truth 的位姿做 L2 损失。

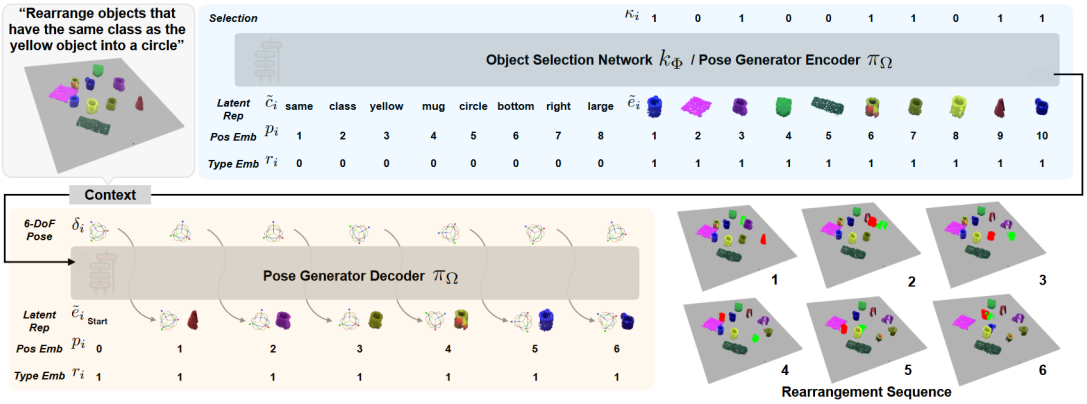

图 3. Language Conditioned Pose

4 复现细节

4.1 与已有开源代码对比

本论文已经开源，其项目地址为：https://github.com/wliu88/StructFormer。在对原论文算法的测试过程中，发现的物体间偶尔会出现一些小碰撞，原因就是网络本身对碰撞并没有先显式约束。类似与生成器-判别器的思想，对原论文做出一点改进，在原有的位姿生成器基础上加入碰撞判别器用于检测物体之间是否发生碰撞。为了利用好物体的分割信息，碰撞判别器以物体两两作为输入，而不是直接对整个场景的点云作为输入。即

$$d_c(p_i, p_j) \in [0, 1]$$

判别器网络选择 `Point Transformer`，损失函数为 focal loss，

$$L_{fl} = \begin{cases} -(1-\hat{p})^{\gamma} log(\hat{p}) & if\ y = 1 \\ -\hat{p} log(1-\hat{p}) & if\ y = 0 \end{cases}$$

数据集则只需收集一些正样本和负样本即可。

4.2 实验环境搭建

从 github 上克隆源代码，按照 github 的 readme 环境配置 python 环境，这里配置了比作者更高版本的 python、pytorch 和 cuda，由于向下兼容也能运行成功，环境安装完成后，按如下步骤训练网络：

- 下载在 github 网址上的数据集，并解压至 $STRUCTFORMER/data_new_objects
- 下载vocabulary list `type_vocabs_coarse.json`,并解压至 $STRUCTFORMER/data_new_objects
- 训练网络分为 Pose Generation Networks 和 Object Selection Network

4.3 界面分析与使用说明

可视化过程分为两步，第一步是 Object Selection，选出和 Instruction 相符的物体，绿色代表选中物体，红色则是与任务无关的物体。

图 4. Object Selection

按 ESC 后，进行 Pose Generation，生成选中物体的位姿，并按照物体的位姿重新可视化物体的点云。

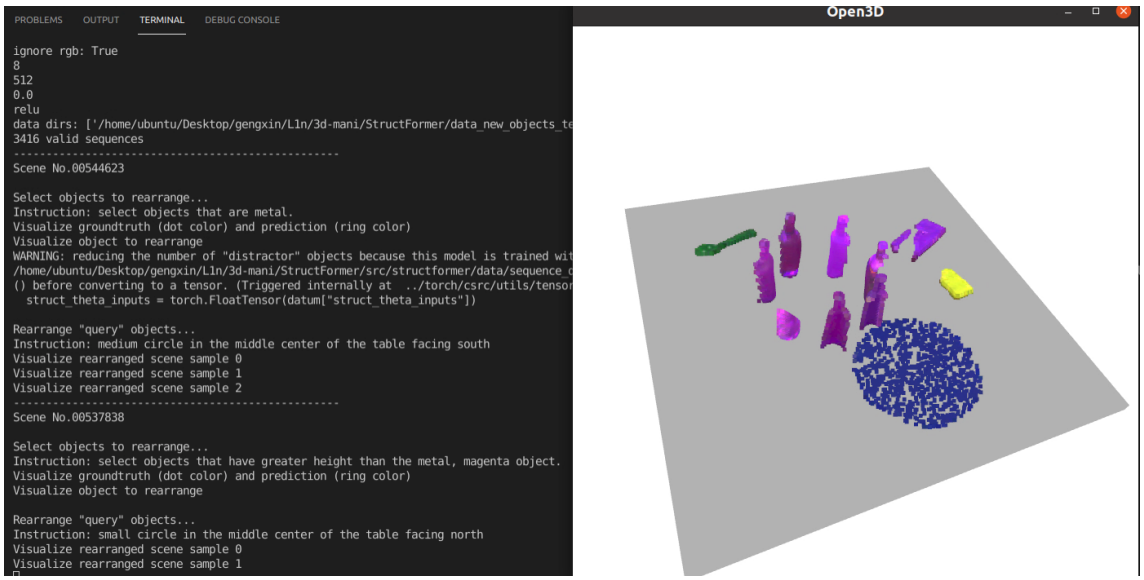

图 5. Pose Generation

5 实验结果分析

下图为原文效果的可视化：

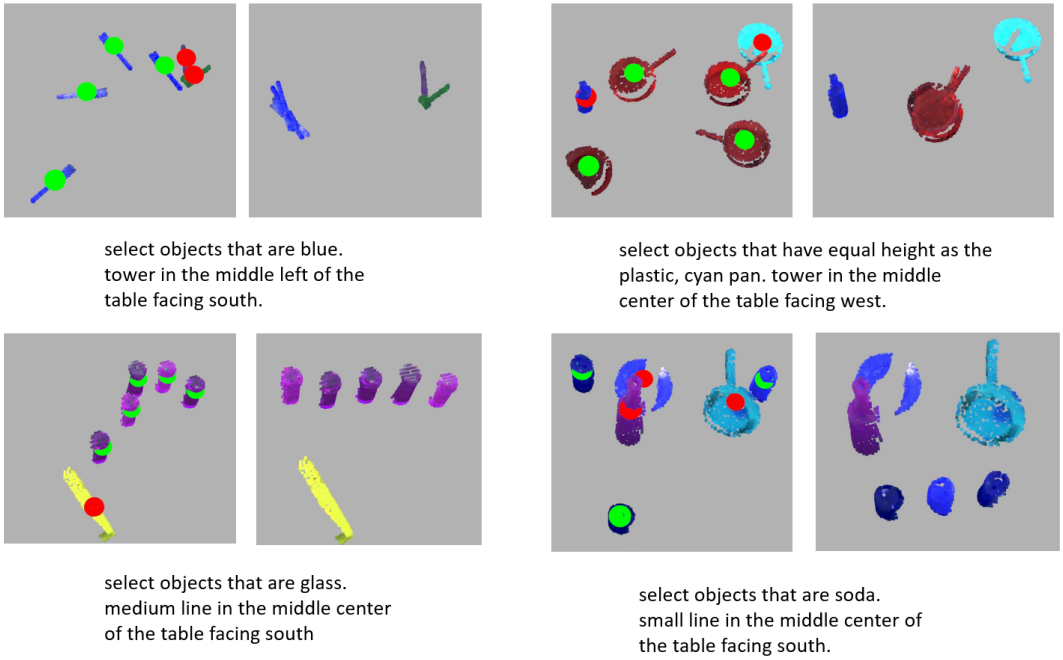

图 6. tower and line

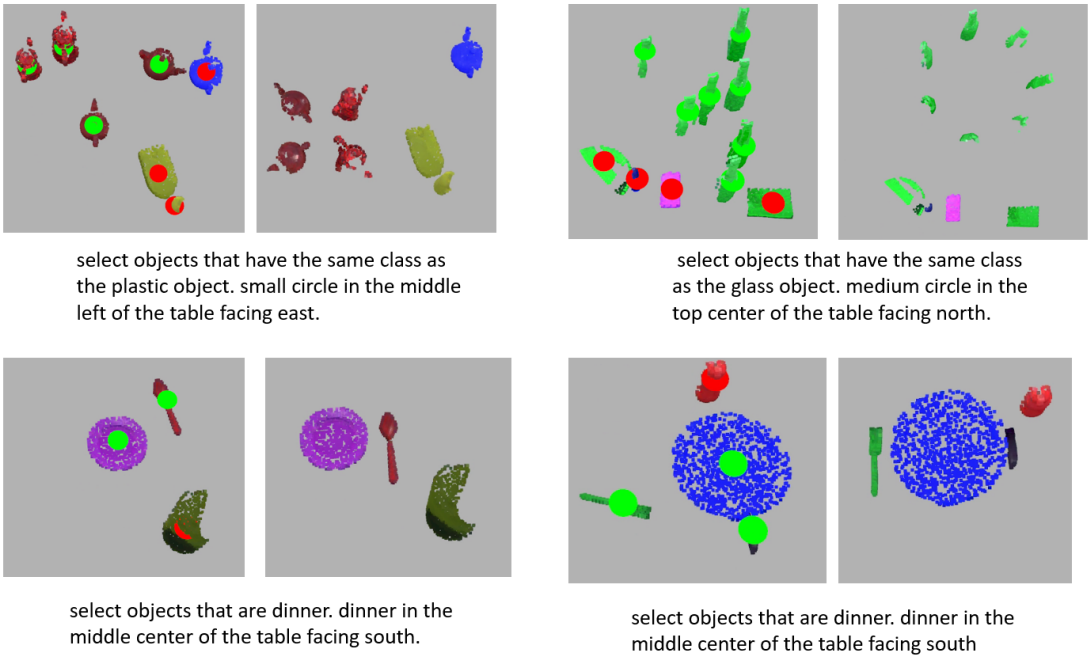

图 7. circle and line

下图为有无碰撞判别器的可视化，对 circles, lines, 和 table 三种 setting 进行测试，至于之前提到的 tower 由于本身就是叠在一起难免会发生碰撞，所以测试了三种结构。

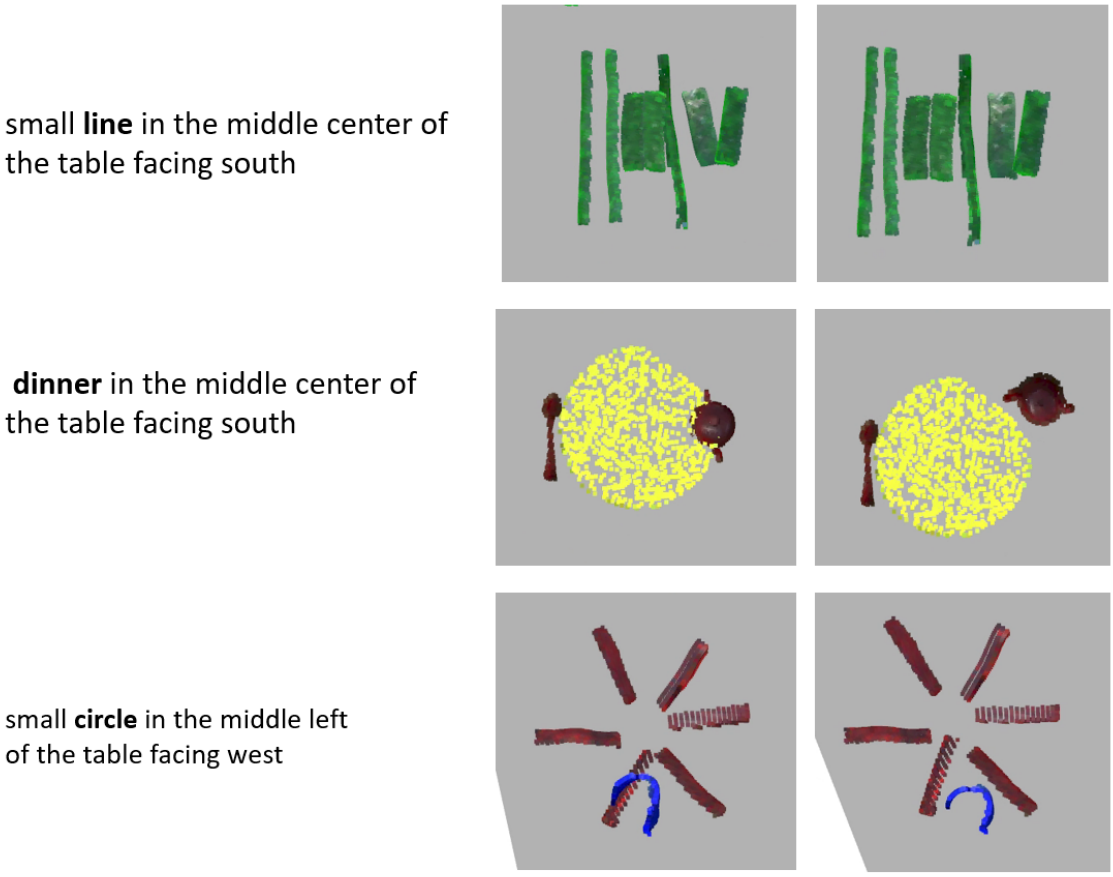

图 8. 加入无碰撞判别器后的可视化

6 总结与展望

StructFormer 这篇论文提出了一种基于学习的方法，用于机器人多物体语义编排的操控，利用了多模态的 Transformer 架构，从输入场景点云和语言指令生成物体的变换位姿。由于算法本身对物体之间的碰撞没有显示的约束，所以加入了一个碰撞判别器去对多个生成结果评分，取得一定的效果。但是 StructFormer 输入的语言指令仍然是一种结构化的形式，未来可以加入 Chatgpt 这类大语言模型用于改善对语言指令的限制。

参考文献

[1] Dhruv Batra, Angel X Chang, Sonia Chernova, Andrew J Davison, Jia Deng, Vladlen Koltun, Sergey Levine, Jitendra Malik, Igor Mordatch, Roozbeh Mottaghi, et al. Rearrangement: A challenge for embodied ai. *arXiv preprint arXiv:2011.01975*, 2020.

[2] Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*, 2023.

[3] Walter Goodwin, Sagar Vaze, Ioannis Havoutis, and Ingmar Posner. Semantically grounded object matching for robust robotic scene rearrangement. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 11138–11144. IEEE, 2022.

[4] Ivan Kapelyukh, Vitalis Vosylius, and Edward Johns. Dall-e-bot: Introducing web-scale diffusion models to robotics. *IEEE Robotics and Automation Letters*, 2023.

[5] Weiyu Liu, Chris Paxton, Tucker Hermans, and Dieter Fox. Structformer: Learning spatial structure for language-guided semantic rearrangement of novel objects. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6322–6329. IEEE, 2022.

[6] Yixuan Wang, Zhuoran Li, Mingtong Zhang, Katherine Driggs-Campbell, Jiajun Wu, Li Fei-Fei, and Yunzhu Li. D ^3 fields: Dynamic 3d descriptor fields for zero-shot generalizable robotic manipulation. *arXiv preprint arXiv:2309.16118*, 2023.

[7] Mingdong Wu, Fangwei Zhong, Yulong Xia, and Hao Dong. Targf: Learning target gradient field to rearrange objects without explicit goal specification. *Advances in Neural Information Processing Systems*, 35:31986–31999, 2022.