基于 The ThreeDWorld 的家庭机器人认知导航、认知规划、认知操作仿真

张瑞程 (Ruicheng Zhang)
中山大学（深圳）智能工程学院

摘 要: 基于期末课程设计要求，我们使用麻省理工学院开源的物理仿真引擎 The ThreeDWorld，设计了一个单智能体认知导航、认知规划、认知操作的多任务挑战。在实验中，我们利用模拟引擎在房子里随机生成一个具有两个 9 DOF 铰链机械臂的可移动机器人代理，代理需要找到散布在不同房间中的多个目标物体，并将它们捡起来运送到所需位置。我们还在房子周围放置了若干容器，这些容器可以帮助机器人一次运输多个物体。为了完成挑战，智能代理必须完成认知导航、认知规划、认知操作等多项子任务，并与物理世界交互。该实验有助于帮助人们更深入地理解智能体在复杂现实任务中的认知过程，为设计更加智能的机器人提供帮助。

关键词: 3D 仿真，具身智能，认知导航，认知规划，认知操作;

图 1: 任务概述。在实验任务中，代理必须运输分散在多个房间的物体并将它们放置在卧室的床上（标有绿色边界框）。代理可以首先拿起一个容器，将物体放入其中，然后将它们运送到目标位置，也可以逐个运送。

1 引言

开发能够在真实物理世界中感知和行动的家用机器人是计算机视觉和机器人领域的一个重要目标，也是认知科学的致力的方向。由于直接用真实机器人训练模型成本高昂且涉及安全风险，因此出现了结合模拟器来训练和评估人工智能算法的趋势。近年来，能够模拟逼真场景的 3D 虚拟环境 [2,3,5] 的快速发展，已成为基于视觉的机器人认知导航进步的主要驱动力 [9,7,1]。迄今为止在这些虚拟环境中定义的任务主要集中在高质量合成场景 [8] 和现实世界 RGB-D 扫描 [6,3] 的视觉导航，很少具有物理交互。但由于嵌入式人工智能的最终目标是开发能够在物理环境中感知和行动的系统，因此物理交互的能力是必需的。交互式物理模拟平台 ThreeDWorld (TDW) [15] 凭借高真实感的渲染和高保真的物理模拟，为设计真实物理环境下的机器人认知导航、认知规划、认知操作实验提供了可能。

在实验中，我们基于认知科学相关任务设计了一个智能体综合性挑战：我们首先基于 TDW 创建一个多房间房屋数据集，并在物理真实的虚拟房屋中随机放置一个配备两个 9 自由度铰接臂的实体代理。智能体需要探索房子，寻找散布在不同房间中的多个物体，并将它们运送到最终位置，如图 1 所示。我们还在房子周围放置了若干容器；代理可以找到这些容器并将物体放入其中。如果不使用容器作为工具，代理一次最多只能传输两个对象。然而，使用容器，代理可以收集多个对象并将它们一起运输。

物理现实环境中的这种综合性任务给具体代理带来的挑战包括：

• 认知规划。由于目标物品和目的地分别位于不同的房间，如何规划正确且最短的路线是智能体要考虑的问题。

• 认知导航。智能体需要具有强大的搜索能力，并精准导航至目标物品所在位置。如果目标物品不在机器人第一人称视角中，或者到达该目标物品的直接路径被阻挡（例如，被桌子阻挡），则智能体无法移动以抓住该对象。

• 认知操作。机器人需要具备对工具（容器）使用的推理能力，虽然容器可以帮助智能体运输两件以上的物品，但找到它们也需要一些时间。因此，智能体必须根据具体情况制定最佳计划。

2 背景

2.1 家庭智能机器人

当前，人工智能和机器人技术已经成为推动现代社会进步的关键力量。在众多机器人应用领域中，家用机器人的开发正逐渐成为科研和工业界的焦点。随着全球性的人口老龄化现象加剧，老年人的护理需求不断增长，家用机器人在提供日常护理、监护和陪伴方面展现出巨大潜力，有助于缓解社会养老压力，提升老年人的生活质量 [19]。同时，在家政服务等行业，劳动力短缺问题日益凸显。家用机器人能够承担清洁、烹饪等家务劳动，为家庭提供实际帮助，提升家庭生活的便利性和舒适度。家庭房间这种高度非结构化、高度复杂化的环境和日常生活中各种综合性的任务，对家用机器人的感知、控制都提出了极高的要求。同时，家用机器人需要大量与人的交互，因此需要强大的认知能力和智能水平。

2.2 认知科学与具身智能

认知科学的快速发展为开发能够在真实物理世界中感知和行动的家用机器人提供了巨大帮助。通过将现实世界中复杂任务分解为多个认知科学领域的内的子任务（认知规划、认知导航、认知控制等），极大地简化了认知机器人地设计思路。近些年来大语言模型（LLMs）的突破为机器人认知能力的构建提供新的思路，将 LLMs 作为机器人的 “大脑 “，利用 LLMs 丰富的常识知识、强大的推理能力、准确的语言理解和文本生成能力，来构建能够进行有效规划、准确认知和通信合作具身智能代理成为当前一大前沿研究方向。

3 相关工作

我们设计的 jia 机器人认知导航、认知规划、认知操作仿真实验建立在 3D 交互环境和具体智能的先前工作基础上。

3.1 3D 交互环境

当前机器人领域中有几种广泛使用的物理引擎，包括 PyBullet [12]、MuJuCo [10] 和 V -REP[14]。许多机器人操作和物理推理挑战也建立在这些引擎之上（例如 RLBench-mark [18]、Meta-World [13] 和 CLEVRER [11]）。然而，这些平台无法渲染逼真的图像，限制了机器人在真实世界中的感知能力。最近，出现了具有真实图像和物理规律的模拟引擎 [15]，旨在减少泛化过程中“Sim to Real“的差距。其中，由麻省理工学院开发的 ThreeDWorld (TDW) [15] 凭借逼真的图像渲染、真实的物理规律模拟和丰富的 API 命令集，成为许多研究者青睐的使用对象。本次实验即基于 TDW 实现。

图 2: ThreeDWorld (TDW) 物理模拟引擎。

3.2 具身智能

具身智能（Embodied AI）是由 “本体” 和 “智能体” 耦合而成且能够在复杂环境中执行任务的智能系统。有别于非具身智能（Internet AI），从互联网收集到的图像、视频或文本数据集中学习，具身智能通过与环境的互动进行学习、进化。这种类似于人类的自我中心感知学习方式，能够更好地帮助机器人建立对真实世界的认知，从而更好地解决真实问题。

当前具身智能针对具体任务的最先进方法主要分为三大类：使用强化学习 [17] 进行策略网络的端到端训练，和使用分层强化学习 [4] 进行感知、导航和分层规划，以及使用大语言模型指导智能体决策 [20]。

图 3: 具身智能原理。

4 模型算法

在本次实验中，我们选择使用了《Building Cooperative Embodied Agents Modularly with Large Language Models》[16] 论文中的 CoELA 模型和预训练参数，文章作者包括来自 MIT 阿默斯特分校、清华大学、上海交通大学等的研究人员，文章发表于 2024 年国际学习表示会议（ICLR）。

CoELA 利用大型语言模型（LLMs）的常识知识、推理能力、语言理解和文本生成能力来构建能够进行有效规划、通信和与合作具身智能代理。。其主体结构包括感知模块、执行模块、记忆模块、通信模块和规划模块五部分：

- **感知模块**（Perception Module）负责处理从环境中接收到的原始感官观察数据（详见附录）。具体来说，使用预训练的 Mask-R CNN 模型来从 RGB 图像中获取实例分割掩码，接着结合深度图像和代理的位置信息，将每个像素投影到 3D 世界坐标系中，创建一个 3D 体素语义图。然后通过累积高度维度来构建一个自上而下的 2D 语义图，该图存储反映了物体在二维平面上的位置和占用空间。
- **记忆模块**（Memory Module）记忆模块模仿人类的长期记忆，由三部分组成：语义记忆、情景记忆和程序记忆。语义记忆存储代理对世界的认知，包括语义图、任务进度、自身状态和他人状态。情景记忆存储代理过去的经验，包括行动历史和对话历史。程序记忆包含执行特定高级计划的具体环境实现代码和神经模型参数。记忆模块可通过大语言模型的上下文记忆能力轻松实现。
- **通信模块**（Communication Module）利用 LLMs 的强大自由形式语言生成能力作为消息生成器。通过设计提示（prompts），结合任务指令、目标描述、状态描述、行动历史和对话历史，来生成要发送的消息。
- **规划模块**（Planning Module）使用 LLMs 作为核心，通过从记忆模块检索相关信息并将其转换为下一步行为的文本描述，行为来自高级计划的行为空间（详见附录）。
- **执行模块**（Execution Module）负责将高级计划转换为在特定环境中可执行的基本动作。使用 A* 算法来找到从当前位置到目标位置的最短路径，并执行与对象交互所需的交互动作（如抓取、放下）。

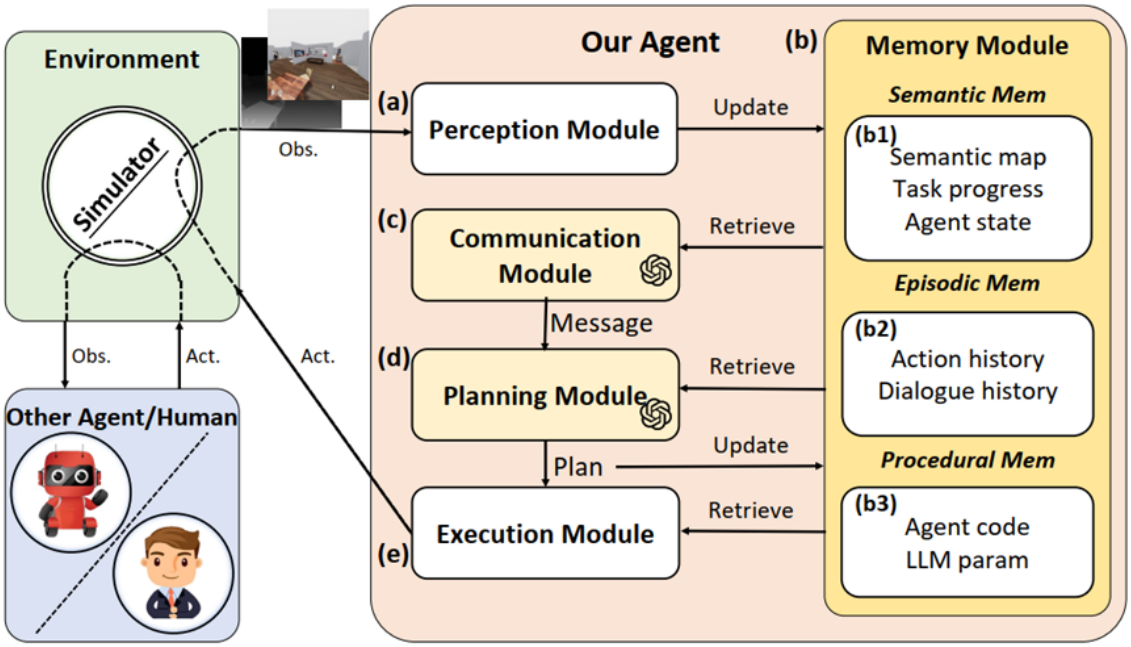

图 4: CoELA 概述。CoELA 框架由感知、记忆、通信、规划、执行五大模块组成，其中通信和规划模块依托大语言模型构建，可完成与物理世界的交互和与其他智能代理的通信、合作。

为更好地完成长视距和稀疏奖励的任务（与我们设计的实验特点一致），CoELA 使用多达 50 帧的视觉图像作为一个强化学习的 step，并可配备深度相机等传感器和语义分割算法，增强模型的视觉认知能力。原作者精心设计了奖励函数，使用 PPO 算法对模型在多个数据集上进行训练。由于数据与算力资源的限制，本次仿真使用原作者提供的预训练模型实现。

5 实验

仿真实验使用 GPT-4（智能体）和 Magnebot 机器人（本体）构建了一个智能代理，以完成家庭环境下多物体运输任务。任务描述如下：

- 任务目标：将房间内的所有水果运输至卧室。
- 传感器：RGB 相机，深度相机
- 观察空间：每一时刻下的 RGB 彩色图像、深度图、语义分割掩码图（由语义分割网络 Mask R-CNN 生成），机器人本体位置坐标。
- 动作空间：由一系列离散化动作构成：向前移动（0.5m），左转（30°），右转（30°），抓取（吸附式），丢弃，放入（容器内），倒出（容器内物体）。

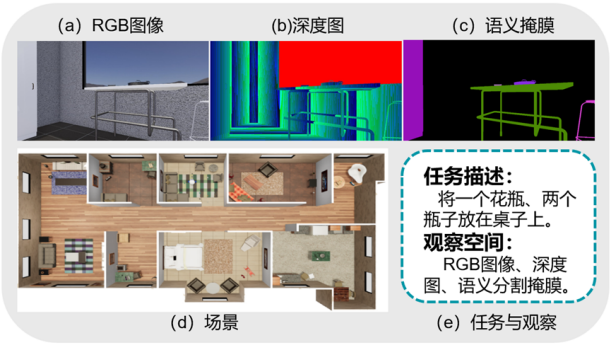

图 5: 仿真实验设计。

5.1 代码布局

本此实验的代码全部打包在 tdw-my-simulation 文件夹中，代码布局与功能与如下：

Listing 1: 项目代码布局

```
|__ tdw-gym          主体函数
    |__ challenge.py          评估代码
    |__ tdw_gym.py     仿真环境代码
    |__ h_agent.py     RHP智能代理
    |__ lm_agent.py       CoELA智能代理
|__ scene_generator  场景生成
|__ dataset          数据集
|__ transport_challenge_multi_agent
    底层环境控制代码
|__ scripts    实验脚本
|__ LLM        CoELA模型
```

5.2 环境配置

在进行仿真实验之前，首先需要配置 python 环境和安装必要依赖。依次运行以下命令：

Listing 2: Conda 环境配置命令

```
conda create -n tdw_mat python=3.9
conda activate tdw_mat
pip install -e .
```

模型部署所需的环境已全部写入 reqirement.txt 文件中，在执行 pip install -e . 命令时，python 编译器会查找项目目录下的 setup.py 文件，并根据里面的配置来安装依赖。

图 6: setup 文件（部分）与 request 文件（部分）。

对于个别 python 包会出现下载失败的情况（如 mmcv、Cython 等），可以通过切换安装源的方式进行重装。安装完成后运行原作者提供的演示场景进行检验：

```
python demo/demo_scene.py
```

测试场景的正常加载标志着 TWD 安装基本成功。

图 7: 环境测试。

5.3 下载数据集

原作者将实验所需的 TDW 环境模型存放在了 Google 云盘中，需使用 VPN 进行下载。下载完成后，将环境数据文件夹 ‘transport-challenge-asset-bundles‘放置在项目的 ‘./scripts‘路径下。

图 8: TWD 环境数据集。

5.4 编写实验脚本

原作者在为代码仓库中提供的示例脚本并不完全适用于我们的实验，需要重新进行脚本编写。在本次实验中，我们将多智能体的协作任务简化为单智能体的独立任务，并将编写的脚本命名为 test-LM.sh 存放在 scripts 文件夹下。

Listing 3: 实验脚本配置与执行

```
lm_id=gpt-4
port=10010
pkill -f -9 "port $port"

python3 tdw-gym/challenge.py \
--output_dir results \
--lm_id $lm_id \
--experiment_name LM-$lm_id \
--run_id run_1 \
--port $port \
--agents lm_agent \
--prompt_template_path
    LLM/prompt_single.csv \
--max_tokens 256 \
--cot \
--data_prefix dataset/dataset_test/ \
--eval_episodes 0 11 17 18 1 2 3 21 22
    23 4 5 6 7 8 9 10 12 13 14 15 16 19
    20 \
--screen_size 256 \
--no_save_img \
--debug

pkill -f -9 "port $port"
```

5.5 运行实验

随后开启终端，运行以下命令启动仿真：

```
./scripts/test_LM.sh
```

程序开始运行！此前如果没有在在环境变量中导入 OpenAi 密钥程序会报以下错误：

```
AttributeError : No API key provided.
    You can set your API key in code
    using 'openai.api_key = <API-KEY>',
    or you can set the environment
    variable OPENAI_API_KEY=<API-KEY>).
    If your API key is stored in a
    file, you can point the openai
    module at it with
    'openai.api_key_path = <PATH>'.
```

经检查报错点代码，发现程序会自动在系统变量中寻找 OpenAi 密钥：

```
openai.api_key =
    os.getenv('OPENAI_API_KEY')
```

如果我们事先没有设定，系统变量中将不包含 OpenAi 密钥，程序自然无法找到。解决方法为通过 “系统” -> “高级系统设置” -> “环境变量”（以 Windows 11 为例，不同操作系统有所差别）来添加一个新的环境变量 `OPENAI_API_KEY`。

图 9: 在系统变量中添加 OpenAI 密钥。

再次运行，仿真顺利启动！

5.6 结果与分析

在仿真实验中，要求 Magnebot 机器人拾取所在房间中的所有散落的水果，并将它们运输到卧室，运输过程鼓励使用容器一次性实现。下图我们展示了 Magnebot 完成任务的关键阶段，同时，录制了一个视频展示仿真全过程，请详见文件加中 “仿真展示.mp4 “。

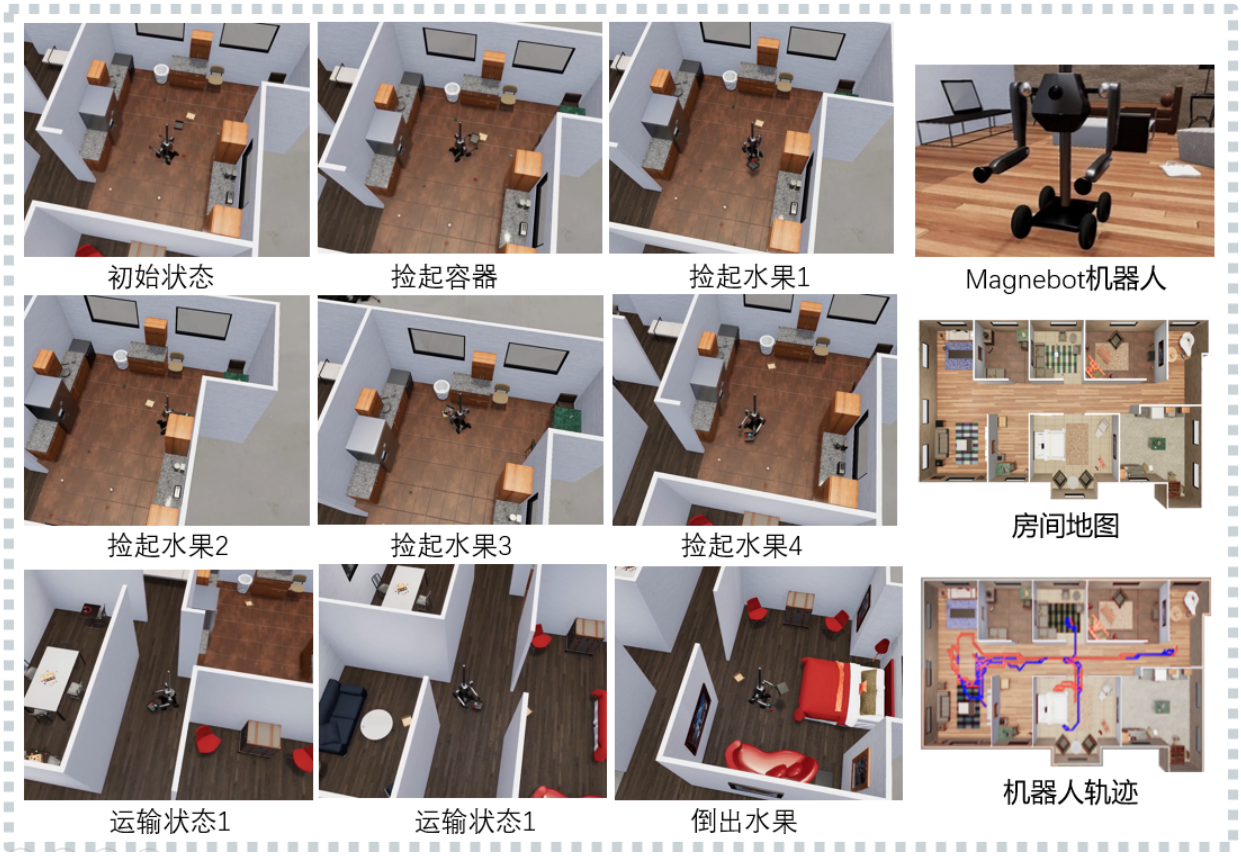

图 10: 实验展示结果。

值得注意的是，在 Magnebot 机器人于房间内启动后，立刻可以开展目标搜索（这与人的行为高度一致），并不需要像普通机器人一样环顾房间一周，使用 SLAM 算法构建地图和对环境的认知。这归功于其智能体基于大语言模型构建，由于 GPT-4 已拥有对视觉信息的强大认知能力，从单视角画面即可推断出场景的大量信息，故可以实现类似人类的自然的认知行为。同时，Magnebot 首先选择拾取辅助工具容器，而不是直接捡起水果，这实现总体运输成本的最小化，体现了其对任务规划的较高认知水平。在操作认知方面，Magnebot 在所有抓取操作中均未出现掉落或与环境碰撞，并且实现了准确高效的工具利用，体现了智能体强大操作认知。

尽管智能体在仿真任务中取得了出色的表现，但仍存在一些问题和不足。比如，在抓取体积较大的物体前（比如容器），机器人可能会对物体造成撞击或碾压，在现实世界中此现象可能会导致机器人或物品的损坏。这种现象是由于认知导航和认知操作之间的协同不一致造成的，即智能体无法准确和适应性地判断，在前行至目标物品多远时停下抓取物品。在物品较小时并不会发生问题，当物品较大时机器人就可能与物品发生相撞。后续，可以通过在强化学习阶段加大对目标物品碰撞的惩罚力度，帮助智能体实现认知导航和认知操作间的协调。

6 总结展望

本研究基于 The ThreeDWorld（TDW）仿真环境，成功设计并实施了一个家用机器人认知导航、认知规划和认知操作的多任务挑战实验。实验结果表明，利用 CoELA 模型和大型语言模型（GPT-4）的智能代理能够有效地完成复杂环境中的物体运输任务，展现了高度的认知能力和与物理世界的交互能力。我们的设计和实验结果不仅验证了智能代理在模拟复杂家庭环境中执行任务的能力，而且展示了认知科学在机器人领域的应用潜力。

在此次实验中，“认知”带给机器人的强大性能令我印象尤为深刻：

- 1. 认知导航与规划能力：实验中，智能代理能够准确识别目标物体，并规划出最短路径以实现高效导航。
- 2. 认知操作能力：代理展示出了对工具使用的推理能力，能够根据任务需求选择使用容器进行物体运输。

- 3. 具身智能的实现：通过与环境的互动学习，智能代理表现出了类似于人类的自我中心感知学习方式，有效地提升了任务完成的效率。

基于大语言模型构建的智能体表现出了对情景的强大理解能力，但在物理交互、导航和操作协调等方面仍存在不足，这同样是当前人工智能领域研究的短板。未来我们希望通过认知科学的理论与技术，增强智能代理对物理交互和人类“无意识”行为的学习能力。我们坚信，随着认知科学领域的不断深化与创新，它将为机器人智能化的发展注入源源不断的动力，开辟新的可能。认知科学不仅为我们提供了理解人类思维与行为的窗口，更将成为推动机器人技术突破的关键力量！

参考文献

[1] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko S □underhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In CVPR, pages 3674–3683, 2018.

[2] Eric Kolve, Roozbeh Mottaghi, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. Ai2-thor: An interactive 3d environment for visual ai. arXiv preprint arXiv:1712.05474, 2017.

[3] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In CVPR, pages 9068–9079, 2018.

[4] Somil Bansal, Varun Tolani, Saurabh Gupta, Jitendra Malik, and Claire Tomlin. Combining optimal control and learning for visual navigation in novel environments. In Conference on Robot Learning, pages 420–429, 2020.

[5] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. ICCV, 2019..

[6] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. arXiv preprint arXiv:1709.06158, 2017.

[7] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural slam. ICLR, 2020.

[8] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In CVPR, pages 1746–1754, 2017.

[9] Saurabh Gupta, James Davidson, Sergey Levine, Rahul Sukthankar, and Jitendra Malik. Cognitive mapping and planning for visual navigation. In CVPR, 2017.

[10] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 5026–5033, 2012.

[11] Zhenfang Chen, Jiayuan Mao, Jiajun Wu, Kwan-Yee Kenneth Wong, Joshua B Tenenbaum, and Chuang Gan. Grounding physical concepts of objects and events through dynamic visual reasoning. ICLR, 2021.

[12] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. GitHub repository, 2016.

[13] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In Conference on Robot Learning, pages 1094–1100, 2020.

[14] James Traer, Maddie Cusimano, and Josh H. McDermott. A perceptually inspired generative model of rigid-body contact sounds. Digital Audio Effects (DAFx), 2019.

[15] Chuang Gan, Jeremy Schwartz, Seth Alter, Martin Schrimpf, James Traer, Julian De Freitas, Jonas Kubilius, Abhishek Bhandwaldar, Nick Haber, Megumi Sano, et al. Threedworld: A platform for interactive multi-modal physical simulation. arXiv preprint arXiv:2007.04954, 2020.

[16] Zhang H, Du W, Shan J, et al. Building cooperative embodied agents modularly with large language models[J]. arXiv preprint arXiv:2307.02485, 2023.

[17] Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, et al. Learning to navigate in complex environments. In ICLR, 2017.

[18] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J Davison. Rlbench: The robot learning benchmark learning environment. IEEE Robotics and Automation Letters, 5(2):3019–3026, 2020.

[19] Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. Virtualhome: Simulating household activities via programs. In CVPR, pages 8494–8502, 2018.

[20] Santhosh K Ramakrishnan, Ziad Al-Halah, and Kristen Grauman. Occupancy anticipation for efficient exploration and navigation. ECCV, 2020.