HumanVLA: Towards Vision-Language Directed Object Rearrangement by Physical Humanoid

Xinyu Xu[12*] **Yizheng Zhang**[2*] **Yong-Lu Li**[1] **Lei Han**[2†] **Cewu Lu**[1†]
[1]Shanghai Jiao Tong University [2]Tencent Robotics X
{xuxinyu2000, yonglu_li, lucewu}@sjtu.edu.cn {yizhenzhang, lxhan}@tencent.com

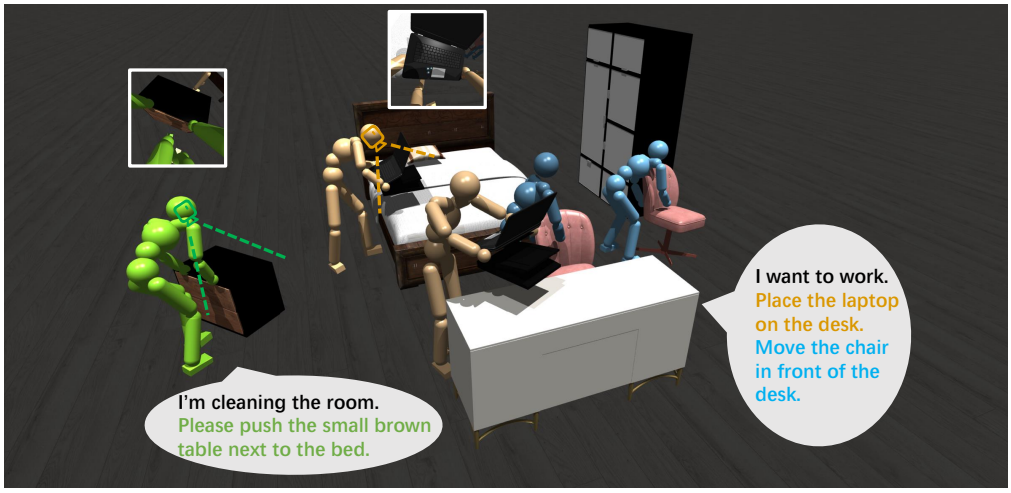

Figure 1: HumanVLA performs various object rearrangement tasks directed by the egocentric vision and natural language instructions.

Abstract

Physical Human-Scene Interaction (HSI) plays a crucial role in numerous applications. However, existing HSI techniques are limited to specific object dynamics and privileged information, which prevents the development of more comprehensive applications. To address this limitation, we introduce HumanVLA for general object rearrangement directed by practical vision and language. A teacher-student framework is utilized to develop HumanVLA. A state-based teacher policy is trained first using goal-conditioned reinforcement learning and adversarial motion prior. Then, it is distilled into a vision-language-action model via behavior cloning. We propose several key insights to facilitate the large -scale learning process. To support general object rearrangement by physical humanoid, we introduce a novel Human-in-the-Room dataset encompassing various rearrangement tasks. Through extensive experiments and analysis, we demonstrate the effectiveness of the proposed approach.

1 Introduction

Learning human-scene interaction (HSI) in realistic physical environments is a vital requirement of many applications, including computer graphics, embodied AI, and robotics. In this field, many

* Equal contribution. † Equal advising.
‡ The code is available at https://github.com/AllenXuuu/HumanVLA.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-0591

previous efforts have been made to promote expressive humanoid control [36, 46, 27, 56], static physical scene interaction [43, 15, 50, 33], and manipulating a specific object [47, 15, 51]. These works have achieved great success in synthesizing plausible HSI controls.

Nonetheless, significant challenges persist in the realm of more extensive HSI applications and two primary issues need to be solved. *Firstly*, the current techniques are limited to static objects, such as sitting on a chair [50, 33], or specific object dynamics, such as carrying a box [15] and throwing a ball [47]. However, in a more complicated real-world environment, humans demonstrate exceptional skills in manipulating a diverse range of objects with different geometries, poses, and weights. It poses a challenging requirement on the varied dynamics of objects in HSI synthesis, i.e., a universal manipulation policy. *Secondly*, ground-truth object and goal states are necessary to direct humanoid controls in previous works. However, without the help of external localization devices, this privileged information is difficult to access in a real-world transfer. It prohibits practical real-world extensions like humanoid robots and necessitates an easily deployable perception method.

Our work takes a step forward in the above two challenges. We investigate the concept of general-purpose object rearrangement performed by a physically interactive humanoid. The whole-body physical humanoid is instructed to carry out daily object loco-manipulations in an indoor room setting. The tasks involve human-like motion controls, interaction with diverse objects, and following desired object dynamics. Moreover, considering the unavailability of privileged information about object and goal states in real-world humanoid applications, we delve into humanoid controls directed by practical vision and language. Compared to privileged states, vision-language modalities are more accessible and offer new potential for practical applications. It also presents an ultimate vision of the research community on humanoids: a human-like agent capable of understanding language instructions, perceiving its environment, and executing daily tasks to assist humans. Fig. 1 provides intuitive examples of our work, where the humanoid agent can push a table, carry a laptop, and pull a chair, all directed by vision and language. Comparisons of our work with previous studies are available in Tab. 1.

Our work starts with learning state-based teacher policy and then distills the policy into a vision-language-action model. In the first stage, we train the policy using goal-conditioned reinforcement learning and adversarial motion priors (AMP) [36], within a generative adversarial imitation learning [16] paradigm. The discrimination reward plus task-conditioned reward encourages humanoids to generate realistic motions and complete the task. However, interacting with diverse objects remains challenging for vanilla AMP. We introduce improved techniques to facilitate general manipulation, in-context navigation, and prioritized task completion. In the second stage, we distill the policy into a student network, named **HumanVLA**, an end-to-end vision-language-action model for physical humanoid. Behavior cloning [2] is used to train the student HumanVLA, i.e., cloning the teacher action at each step. A challenge of learning VLA models is the poor perception quality of the unconstrained camera pose. We propose a novel active rendering technique to improve gaze intention.

To support HumanVLA, we create a novel dataset named **Human-in-the-Room (HITR)**. It consists of four different room layouts: *bedroom*, *livingroom*, *kitchen*, and *warehouse*. Each layout is populated with *separated*, *instantiable*, and *replaceable* objects from HSSD [22] assets to create diverse scenes. The humanoid agent is placed in the scene with an instruction to rearrange the room. Statistically, the HITR dataset consists of 50 static objects and 34 movable objects. In our extensive experiments, we train HumanVLA in IsaacGym [28] with tasks from HITR. Results demonstrate the effectiveness of our method in generalized object rearrangement and vision-language perception.

In summary, our contributions include: (1) We study general object rearrangement by physical humanoids. Several advanced techniques are introduced to interact with diverse objects. (2) We propose HumanVLA, the first vision-language-action model on physical humanoids to complete tasks directed by egocentric vision and natural language instruction. (3) We propose the HITR dataset to facilitate research in this field. Comprehensive experiments are conducted in HITR to validate the effectiveness of our method.

2 Related Works

Motion Synthesis is a long-term research topic in graphics, vision, and robotics. It can be divided into two streams: kinematic motion synthesis [43, 42, 17, 14, 24, 18, 5, 20, 54, 19, 25, 1, 7] and physics-based motion synthesis [34, 36, 35, 21, 50, 27, 46, 56, 41, 51, 33, 47, 6, 26]. Kinematic

Table 1: Comparisons between HumanVLA and past works.

Methods	Physics	Object Interaction	Object Dynamics	Language Instruction	Ego-Vision	# Static Objects	# Movable Objects
NSM [43]		✓	✓			25	2
SAMP [14]		✓				7	-
OMOMO [24]		✓	✓	✓		-	19
PADL [21]	✓			✓		-	-
InterPhys [15]	✓	✓	✓			350	1
InterScene [33]	✓	✓				57	-
UniHSI [50]	✓	✓		✓		40	-
HumanVLA(Ours)	✓	✓	✓	✓	✓	50	34

methods aim at synthesizing visually plausible motions with less penetration, floating, and being semantically faithful. It leverages generative neural networks like VAEs [14, 7], Transformers [1, 19], or Diffusions [24, 18, 20] to predict next state. Our work belongs to the physics-based methods, which have an additional requirement on physical plausibility. It follows a control-then-model paradigm where the control is typically achieved by a learning algorithm, and the model is constrained by a physics simulator. DeepMimic [34] uses reinforcement learning plus imitation learning to track motion references and perform versatile motion controls. NCP [56] advances motion tracking with discrete latent prior. Adversarial Motion Prior (AMP) [36] uses generative adversarial imitation learning to learn natural state transition from unstructured motion data. It is further extended with a reusable controller [35], high-level language [21], expressive control [27], and latent conditions [46]. Recently, there has been an increasing emphasis on the synthesis of interactive motions. InterPhys [15] uses task-conditioned reward plus stylized adversarial reward to perform HSI tasks such as sitting, lying, and box carrying. InterScene [33] extends the paradigm to synthesize long-horizon static interactions. UniHSI [50] leverages the vast knowledge of the language model to provide a unified interface for static HSI. However, previous works are limited to static objects or specific movable objects but fail to interact with various objects. In contrast, our research studies general object rearrangement in a daily room, posed with challenges in diverse object geometries, positions, and weights.

Room Rearrangement is a crucial application of embodied AI, where an instructed agent is placed in a room to search, navigate, and interact with desired objects. Recent efforts [48, 44, 53, 23, 9] have proposed various platforms and benchmarks to facilitate room rearrangement research. Visual room rearrangement [48] takes the agent to transverse both goal and initial scenes to recover object states based on visual observations. OVMM [53] presents open-vocabulary pick-and-place manipulation challenges in pursuit of extreme generalization capability. Recent algorithms [49, 11] leverage commonsense knowledge in large language models to plan rearrangements. However, these works are designed for simple embodiments, such as disc-shaped mobility and gripper manipulation. They are limited to moving on smooth terrain and handling only small-sized objects. In contrast, our work pioneers the design of rearrangement tasks in a complex human-like embodiment. It benefits from bipedal locomotion and stronger interaction motors. For example, our model is capable of carrying $20kg$ objects, which is beyond the capabilities of traditional stretches.

Vision-Language-Action (VLA) Model maps practical vision-language input to generate action controls. It has demonstrated impressive results in the fields of embodied AI and robotics [12, 57, 32, 3, 55]. Thanks to the robust scalability of the vision and language modalities, VLAs also benefit from large-scale training [57, 32], opening up the potential for more general-purpose applications. However, existing VLAs are designed for simple embodiments, such as desktop gripper manipulation. The exploration of VLAs for more complex, high-dimensional humanoids is still in its early stages. Our work is the first to develop humanoid controls directed by practical vision and language.

3 Approach

In this section, we introduce the learning process of HumanVLA. Training HumanVLA directly through large-scale reinforcement learning (RL) presents significant challenges, including a high-dimensional action space, a composite state space, slow rendering speed, and other common issues associated with large-scale RL. To this end, we utilize a teacher-student framework to train HumanVLA, which has been validated in applications like dexterous re-oreintation [8] and grasping [52].

https://doi.org/10.52202/079017-0591

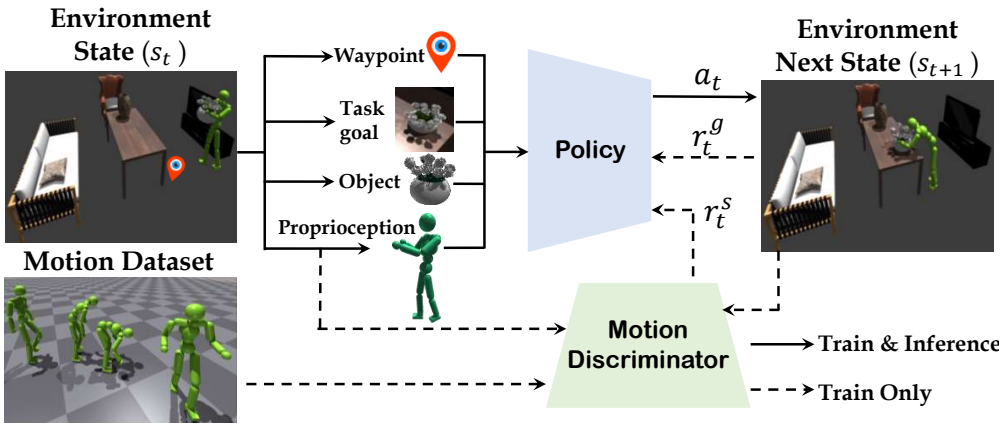

Figure 2: An overview of learning state-based HumanVLA-Teacher policy using goal-conditioned reinforcement learning and adversarial motion prior.

It consists of two phases. In the first phase (Sec. 3.1), we leverage goal-conditioned reinforcement learning and adversarial motion priors [36] to train HumanVLA-Teacher. It is presented with the oracle scene state, including precise object pose, geometry, navigation waypoint, and goal coordinate. In the second phase (Sec. 3.2), we operate in a more practical setting, where the egocentric vision is tasked with perceiving the scene, and natural language instruction is used to specify the goal. In a blueprint of real-world humanoid robots, observations used by HumanVLA are all accessible in a real-world deployment. HumanVLA is trained via behavior cloning [2] from HumanVLA-Teacher, where the pre-trained teacher policy significantly reduces the compute demands of the process.

3.1 State-based Teacher Policy Learning

We train HumanVLA-Teacher with complete knowledge of the scene state to enable a variety of object rearrangement tasks. The rearrangement task is formulated as a reinforcement learning process. To elaborate, at each time step t, given state s_t and goal g, HumanVLA-Teacher π_{tch} predicts an action a_t from policy distribution $\pi_{tch}(a_t|s_t, g)$. The action a_t is processed by a physics simulator $f(s_{t+1}|a_t, s_t)$ to generate the next state s_{t+1}. The learning objective is to maximize the accumulated reward $\mathcal{R}(\pi_{tch}) = \sum_{t=0}^{T-1} \gamma^t r_t$ where γ is the discount factor and r_t is the step reward at time t.

To train robust policies that enable humanoids to interact with objects and achieve various goals g in a life-like manner, it is crucial for the humanoids to learn from authentic human motions and generalize across different tasks. To this end, we use goal-conditioned task reward $r^G(g, s_t, s_{t+1})$ to encourage the agent to complete the task and style reward $r^S(s_{:t+1})$ to imitate human motion prior.

We employ adversarial motion prior (AMP) [36] to model the style reward, which incorporates an adversarial discriminator D to discriminate motions from simulated synthesis or tracked motion dataset. It is trained with the objective:

$$\begin{aligned}\underset{D}{\operatorname{argmin}} -E_{d^M(s_{t:t+t^*})}[\log(D(s_{t:t+t^*}))] - E_{d^\pi(s_{t:t+t^*})}[\log(1 - D(s_{t:t+t^*}))] \\ +w^{gp} E_{d^M(s_{t:t+t^*})}\left[\left|\left|\nabla_\phi D(\phi)\Big|_{\phi=s_{t:t+t^*}}\right|\right|^2\right],\end{aligned} \tag{1}$$

where $d^M(s_{t:t+t^*})$ and $d^\pi(s_{t:t+t^*})$ are distributions of t^*-frame motion clips from dataset M and policy π. The first two items in Eq. 1 are to discriminate motions while the last item with a coefficient w^{gp} regularizes the gradient penalty [30] in adversarial training. The style reward r^S to encourage realistic motion synthesis is then formulated as

$$r^S(s_{:t+1}) = -\log(1 - D(s_{t+1-t^*:t+1})). \tag{2}$$

We uniformly conceptualize goal-conditioned object rearrangement as three processes: locomotion towards the object, contacting the object, and relocating the object to the goal. These steps are accomplished by unified progressively increasing task rewards.

Despite the powers of goal-conditioned reinforcement learning and adversarial motion prior, generalized object rearrangement tasks by humanoids still pose significant challenges. Previous works [15, 33, 50] have been limited to simple tasks such as static sitting, lying, or carrying a specific box. We propose new techniques to overcome challenges in generalized object rearrangement. Generalized object interaction involves geometrically various objects. However, tracking motion data for each individual object is labor-intensive, and infeasible to tackle novel objects. We expect RL to enable automatic object generalization. Thus, we encode object geometry to learn a geometry-aware policy and design a carry curriculum to facilitate the learning. Due to the misalignment of objects in human motion data and the task, we propose style reward clipping to prioritize high-level task execution. Navigating in a complex room requires high-level planning to avoid collisions, we use in-context path planning to enable efficient locomotion. More detailed explanations of our improved techniques are described in the following:

Geometry Encoding. Object state is crucial in HSI synthesis. Previous studies [15, 47] primarily encode object position, rotation, and linear and angular velocities to act on certain objects like boxes or balls. A general policy for interactions with diverse objects should incorporate geometric information. Thus, we augment the teacher policy with geometric object representations via Basis Point Set (BPS) [37] encoding. A shared set of basis points is randomly sampled from a unit sphere and encodes object geometry using delta vectors from each basis point to the nearest object point. In contrast to geometries encoded by a neural net [38], BPS encoding is computationally efficient and accelerates policy learning. Consequently, we use object geometry, position, rotation, and linear and angular velocities to form a comprehensive object observation, thereby facilitating a more expressive policy control.

Carry Curriculum Pre-training. Object rearrangement is conceptualized as a three-step process in the aforementioned paragraph. However, directly learning the entire three-step rearrangement task from scratch is challenging due to the long task horizon. Besides, physics-based object movement presents greater challenges compared to kinematic object movement [24], primarily because the object state is not directly editable. Instead, it requires indirect control of the physical humanoid to interact. To this end, we draw inspiration from the curriculum learning [4] and design an easy carry curriculum to pre-train the policy. The carry curriculum only includes the first two of three steps: locomotion towards the object and carrying up the object for an in-the-air holding. The carry curriculum has a shorter horizon and is empirically easier to converge. Furthermore, the pre-trained in-the-air carry prior significantly benefits the subsequent object relocation. For the carry curriculum, we use objects excluding those on the ground, such as tables and chairs, which are easier to move by pushing and pulling along the ground without a lift. The carry curriculum shares the same learning paradigm with the rearrangement task, except for a different two-stage reward design.

Style Reward Clipping. General object rearrangement involves manipulating novel objects that are not recorded in tracked motion data. This creates a misalignment in optimization directions: strictly imitating reference motion or ensuring high-level task execution. Previous work [15] balanced two items by a weighted sum between task reward and style reward in motion-aligned tasks. However, in our general object rearrangement setting, goal-conditioned task exploration progress can be stagnant and the policy may learn actions devoid of task semantics following the logarithmic gradient in Eq. 2. For instance, when the object is difficult to lift, the policy tends to mimic insignificant hand swings in the motion data, rather than exploring carry-up actions. We insert a style reward clipping to prioritize task execution, formulated as follows:

$$\xi_t = \max(r^G(g, s_t, s_{t+1}), \xi_{min}), \tag{3}$$

$$r_t = w^G r^G(g, s_t, s_{t+1}) + w^S \min(r^S(s_{:t+1}), \xi_t), \tag{4}$$

where w^G, w^S are coefficients, and ξ_t is the upper bound for the style reward. This formulation prioritizes goal-conditioned task execution over motion imitation in reward maximization. In addition, we use a minimum upper bound ξ_{min}, to ensure basic motion imitation during the early stages when the task reward is near zero.

In-context Path Planning. Navigating a populated room requires high-level knowledge since the dense object cluster may collide with the humanoid agent and obstruct natural locomotion. We use in-context path planning to guide the navigation. Point clouds of all objects in the scene make up the spatial occupancy. We perform top-down point projection and grid discretization to derive a 2D obstacle map of $20cm$ x $20cm$ grids. We then plan a navigable path from the starting position to the

https://doi.org/10.52202/079017-0591

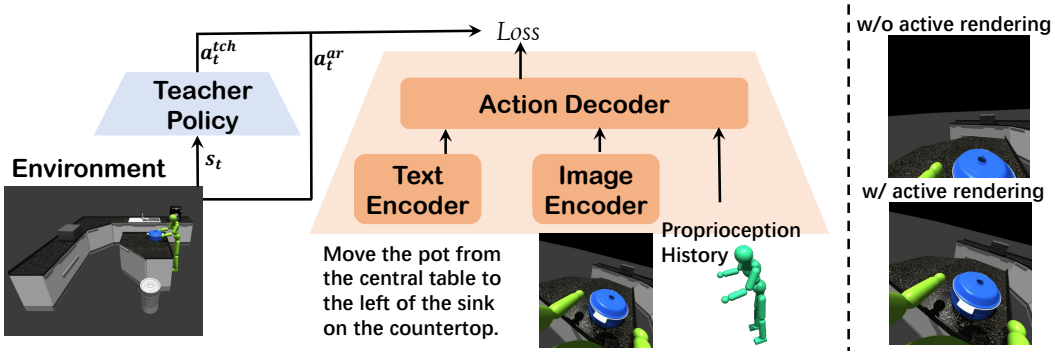

Figure 3: **Left**: An overview of learning HumanVLA by mimicking teacher action and active rendering action. **Right**: Comparison between *w/* and *w/o* active rendering. Active rendering leads to a more informative perception of human-object relationships.

object, and subsequently to the goal using A^* algorithm [13], represented as a series of navigation waypoints to guide locomotion at each step.

Incorporating all the above features, we present an overview of training HumanVLA-Teacher in Fig. 2. Navigation waypoint, task goal, object state, and humanoid proprioception are sent to the policy network to derive an action. The learning process is guided by task reward and motion discrimination reward. Further details about the learning process can be found in the appendix.

3.2 Distilling into Vision-Language-Action Model

While the HumanVLA-Teacher π_{tch} leverages privileged information such as object state, goal state, and waypoint in the global coordinates, our biggest goal is towards practical humanoid control free of privileged information and real-world deployable. To this end, we replace privileged states with flexible egocentric vision and natural language instruction. Notably, the proprioception observation is the only kept item from HumanVLA-Teacher to HumanVLA, which is represented in the local coordinate and can be obtained via forward kinematics and temporal differentiation. The history action is also used in observation. To obtain egocentric vision, we mount a mobile camera on the head of the human. It renders a 256 x 256 image with a field of view spanning 90 degrees at each step. An overview of training HumanVLA is illustrated in Fig. 3.

Behavior Cloning. We train HumanVLA π_{vla} using a teacher-student framework to distill the knowledge from HumanVLA-Teacher via behavior cloning [2]. HumanVLA employs an *EfficientNet-B0* [45] for image encoding and a frozen *bert-base-uncased* [10] for language encoding, whose features, along with proprioception, last action, are sent to the action decoder to derive an action. At each time step t, HumanVLA-Teacher leverages privileged state s_t and g to derive an action $\pi_{tch}(a_t|s_t, g)$ while HumanVLA derives an action $\pi_{vla}(a_t|p_t, a_{t-1}, v_t, l)$ based on proprioception p_t, last action a_{t-1}, egocentric image v_t, and language instruction l. Behavior cloning bridges distributions between $\pi_{vla}(a_t|p_t, a_{t-1}, v_t, l)$ and $\pi_{tch}(a_t|s_t, g)$, which can be directly implemented via supervised learning. In the empirical training process, we observe a severe covariate shift problem in offline behavior cloning. Thus, we opt for a DAgger [39] framework to train HumanVLA which alleviates the problem via online learning.

Active Rendering. Though HumanVLA-Teacher possesses comprehensive knowledge in versatile control, naive policy distillation still suffers from the gap of observation expressiveness. For instance, while egocentric vision is used to perceive objects, the humanoid gaze might not properly focus on the target object but renders a less informative image of the background. Consequently, the perception quality of HumanVLA is significantly affected by the camera pose. However, an optimal camera pose, determined by the head pose, is not guaranteed in the teacher policy, which only imitates a life-like head motion but ignores the vision quality. We propose an active rendering technique to encourage the camera to focus on the object. We analytically calculate the head-to-object direction in the global coordinate and then derive a head orientation. Inverse kinematics is used to obtain active rendering actions a_t^{ar} for the neck joint. It is used to derive a mixed supervision a_t^{vla} in conjunction with the teacher action a_t^{tch}, formulated as

$$a_t^{vla} = (1 - w^{ar})a_t^{tch} + w^{ar}a_t^{ar}, \tag{5}$$

Table 2: Results in box rearrangement. † denotes our implementation.

	Success Rate (%) ↑	Precision (*cm*)↓	Execution Time (*s*) ↓
InterPhys [15]	94.3	8.3	9.1
InterPhys [15] †	97.8	12.6	5.3
HumanVLA-Teacher	**98.1**	**4.2**	**4.6**

where w^{ar} is the coefficient for active rendering. Notably, this is only applied to the neck joint, while other joints only follow the teacher action.

4 Human-in-the-Room Dataset

Existing datasets [48, 53] for object rearrangement are primarily designed for stretches with disc-shaped mobility and gripper manipulation. Human-like embodiment has different physical attributes, such as stronger motors to handle large furniture like chairs and tables. Besides, we follow [36, 15, 33, 50] to use a humanoid model with spherical hands, which can struggle with manipulating small-sized objects, such as picking up a towel. To address these issues, we introduce a novel Human-in-the-Room (HITR) dataset, designed to facilitate vision-language directed object rearrangement tasks on a humanoid. The HITR dataset includes carefully designed objects of various sizes, ranging from 21*cm* to 126*cm*, and provides a variety of rearrangement tasks in various rooms. Each task involves separated, instantiable, and replaceable objects with defined initial and goal states. Additionally, each task is accompanied by a natural language instruction generated by a Large Language Model (LLM).

In constructing the HITR dataset, we reference common objects used in room designs from [23, 44, 53] and utilize object models from HSSD [22] to create basic assets. Object assets are manually resized to ensure the interaction friendliness. We adopt the procedural generation pipeline from [9] to generate diverse scenes. First, we manually design four room layouts: *bedroom*, *livingroom*, *kitchen*, and *warehouse*, then randomly populate replaceable objects within these layout templates to establish the scene, as well as the goal state. Next, we randomly relocate an object in the scene, either to the ground or another receptacle. This relocated object is what the physical humanoid is tasked to rearrange. We concatenate two rendered images of the initial and goal scenes and use the composite image to prompt *gpt-4-vision* [31] to generate an instruction. The LLM is asked to distinguish between two states and provide an instruction to guide the state transition. However, the LLM still struggles with understanding complex spatial relationships, such as left-right errors. To ensure the quality of instructions, we manually review and revise them as necessary. Ultimately, we build the HITR dataset of 615 tasks, with an average of 6.5 objects per task. There are 50 static objects like *bed* and *countertop*, as well as 34 movable objects like *pillow* and *vase*. More details are in the appendix.

5 Experiments

5.1 Settings

Datasets. Our experiments are conducted on the HITR dataset. It is split into *train* and *test* subsets at a ratio of 9:1, containing 552 and 63 tasks respectively. The *test* subset is used to evaluate the generalizability of our method in unseen tasks. For the motion dataset used in training, we utilize OMOMO [24] and a locomotion subset from SAMP [14]. OMOMO provides a variety of short-range motions involving moving different objects, while locomotion motions from SAMP enhance the locomotion aspect of our dataset. We use 30-minute motions in total. The source motion dataset features object rearrangement involving only seven different objects, which is far less than those in the HITR dataset. Despite this, we anticipate that our method can generalize to different objects.

Metrics. We adhere to a 10-second running time limit and follow [15] to evaluate methods with three metrics. **(1) Success Rate:** the proportion of tasks that are successfully rearranged within an error margin of θ. **(2) Precision:** the distance of the final object position to the goal. **(3) Execution Time** the average time taken to complete a run. For the Success Rate, a higher value indicates better performance, but for Precision and Execution Time, the lower the better. All experiments are evaluated using 10 repeat runs. For state-based methods, we follow [15] to set $\theta = 20cm$.

https://doi.org/10.52202/079017-0591

Table 3: Ablation study.

	Privileged State	Success Rate (%) ↑	Precision (*cm*)↓	Execution Time (*s*) ↓
HumanVLA-Teacher	✔	**85.9**	**14.4**	4.5
w/o geometry encoding	✔	64.5	43.4	5.5
w/o carry curriculum	✔	66.3	73.4	5.3
w/o style clipping	✔	79.9	27.5	**4.3**
w/o path planning	✔	67.8	37.2	5.5
HumanVLA	✗	**74.8**	**42.6**	**5.1**
w/o active rendering	✗	67.9	55.6	5.6
w/o online learning	✗	15.3	145.0	8.2

Table 4: Results in unseen tasks.

	Privileged State	Success Rate (%) ↑	Precision (*cm*)↓	Execution Time (*s*) ↓
InterPhys [15]	✔	59.5	52.5	5.9
HumanVLA-Teacher	✔	**79.3**	**19.3**	**4.6**
Offline GC-BC [29]	✗	10.2	152.3	8.5
HumanVLA	✗	**60.2**	**57.0**	**5.8**
w/o active rendering	✗	56.7	65.5	5.9

For vision-language-based methods, where the goal is specified via coarse instructions rather than precision goal coordinates, we set $\theta = 40cm$. This criteria relaxation is also adopted in past works [8] for evaluating policies with different observations.

5.2 Implementation Details

We conduct experiments in parallel environments simulated using IsaacGym [28], with neural networks implemented via PyTorch. Our physical humanoid model, following previous works [15, 33, 50], comprises 15 rigid bodies and 28 joints, each actuated by a PD-Controller. The simulator runs at 60Hz, and the policy is queried at 30Hz. The teacher policy is optimized using Proximal Policy Optimization [40] and takes two days on eight Tesla V100 GPUs to converge. The student policy is trained using DAgger [39] and takes one day on two GPUs. We provide comprehensive details about hyperparameters, neural architectures, observation space, and more in the appendix.

5.3 Comparisons in Box Loco-Manipulation

While our work is pioneering in the exploration of vision-language-directed general object rearrangement, direct comparisons with previous studies are difficult. The work most similar to ours is InterPhys [15], which delved into state-based box loco-manipulation. Due to the unavailability of training data, motions, and codes of InterPhys [15], we instead refer to a box rearrangement subset in HITR to conduct experiments. We train a state-based HumanVLA-Teacher using only box rearrangement tasks, along with an implementation of the InterPhys baseline. Results of box rearrangement are reported in Tab. 2. Given that the box is the simplest object to interact with, both methods achieve high success rates. However, our method exhibits superior precision, with a result of 4.2 *cm*, outperforming both the 8.3 *cm* in the official report [15] and the 12.6 *cm* in our implementation. We use the standard deviation to evaluate the statistical significance of HumanVLA-Teacher in 10 repeated runs. The values are 0.02, 0.004, and 0.04 for Success Rate, Precision, and Execution Time respectively. With high task completion rates and low variance, we demonstrate the effectiveness and robustness of our method in this first trial.

5.4 Ablation Study

We conduct comprehensive ablation studies on the *train* split to validate each design choice, with results presented in Tab. 3. Firstly, we evaluate the impact of improved techniques in HumanVLA-Teacher training, which achieves a success rate of 85.9% and a precision of 14.4cm. However, eliminating any component leads to a decline in performance. The inclusion of geometry encoding and carry curriculum enables the model to manipulate a variety of objects effectively. Without either of these components, the success rate experiences a drop of approximately 20%. Style reward

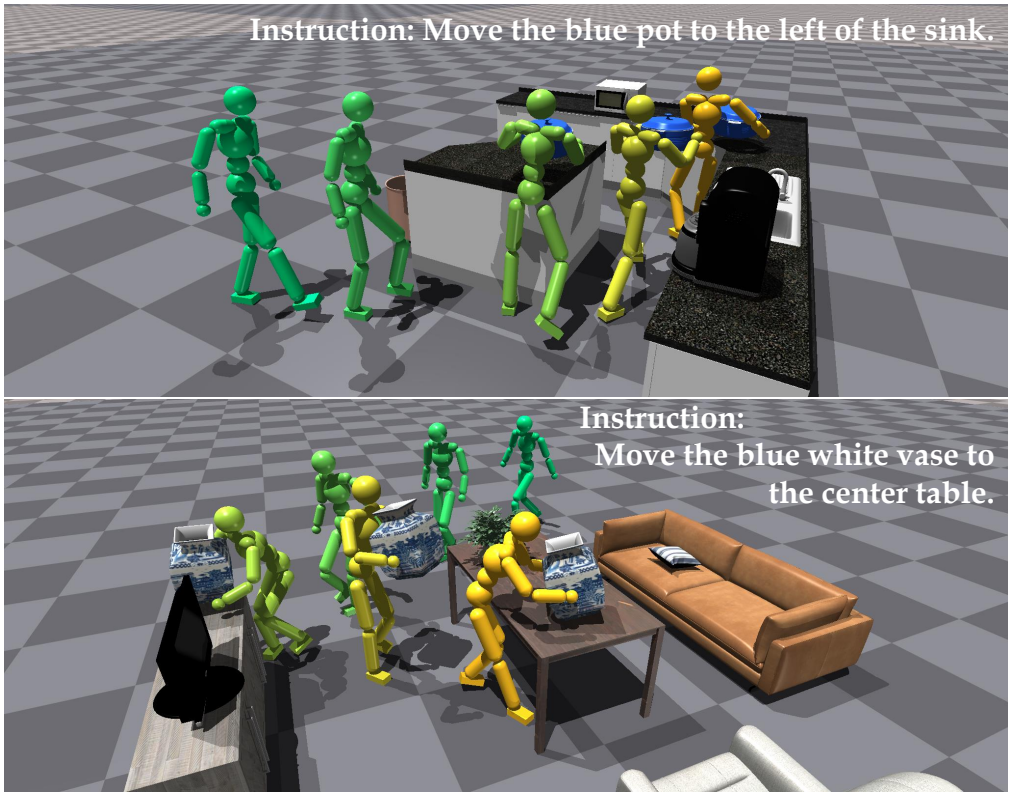

Figure 4: Qualitative results. The color transitions from green to yellow as the task progresses.

clipping prioritizes task execution, whose absence results in a 6% decrease in the success rate. Path planning helps humans navigate complex scenes. Its removal leads to a significant 18.5% decrease in the success rate. Subsequently, we validate design choices in training HumanVLA directed by vision and language. The default HumanVLA achieves a success rate of 74.8% with a precision of 42.6*cm*. However, the absence of active rendering results in a substantial 6.9% success rate drop, emphasizing the importance of perception quality. We implement an offline behavior cloning baseline using ten off-the-shell teacher trajectories per task for training. Without online learning, the system suffers from a severe covariate shift and performs poorly.

5.5 Generalizing to Unseen Tasks

We use the *test* split of HITR to evaluate the generalizability of methods. The unseen data, which includes novel scene compositions, visual appearances, and language instructions, poses a significant challenge to our method. Results are presented in Tab. 4. The state-based HumanVLA-Teacher tends to be more robust in unseen data. Relatively small drops in success rate (6.6%) and precision (4.9*cm*) demonstrate strong generalizability of RL when using privileged information. Moreover, it consistently outperforms the InterPhys [15] baseline on all metrics. Applying HumanVLA to unseen tasks turns out to be more challenging due to the complexity of vision and language modalities. The success rate of HumanVLA decreases to 60.2%, and the precision drops to 57.0*cm*. However, HumanVLA still consistently outperforms baselines without active rendering and the offline goal-conditioned behavior-cloning (Offline GC-BC) [29] method.

5.6 Qualitative Results

We provide qualitative visualizations of how HumanVLA performs object rearrangement tasks in Fig. 4. More results are available in the appendix. We demonstrate that HumanVLA is capable of moving a pot, vase, chair, and box based on language instructions.

https://doi.org/10.52202/079017-0591

6 Conclusion

We investigate vision-language directed object rearrangement by physical humanoids in this work, a fundamental technique for HSI synthesis and real-world humanoid robots. Our system is developed using a teacher-student distillation framework. We propose key insights to facilitate teacher policy learning with privileged states and introduce a novel active perception technique to favor vision-language-action model learning. We present a novel HITR dataset to support our task. In extensive experiments, our HumanVLA model demonstrates superior results in both quantitative and qualitative evaluations. Future works include dexterous manipulation by physical humanoids and long-horizon multi-object interaction.

Acknowledgments

This work was supported by the National Key Research and Development Project of China (No. 2022ZD0160102), National Key Research and Development Project of China (No. 2021ZD0110704), Shanghai Artificial Intelligence Laboratory, and XPLORER PRIZE grants.

References

[1] Joao Pedro Araujo, Jiaman Li, Karthik Vetrivel, Rishi Agarwal, Deepak Gopinath, Jiajun Wu, Alexander Clegg, and C. Karen Liu. Circle: Capture in rich contextual environments, 2023.

[2] Michael Bain and Claude Sammut. A framework for behavioural cloning. In *Machine Intelligence 15*, pages 103–129, 1995.

[3] Suneel Belkhale, Tianli Ding, Ted Xiao, Pierre Sermanet, Quon Vuong, Jonathan Tompson, Yevgen Chebotar, Debidatta Dwibedi, and Dorsa Sadigh. Rt-h: Action hierarchies using language, 2024.

[4] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

[5] Bharat Lal Bhatnagar, Xianghui Xie, Ilya Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Behave: Dataset and method for tracking human object interactions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2022.

[6] Jona Braun, Sammy Christen, Muhammed Kocabas, Emre Aksan, and Otmar Hilliges. Physically plausible full-body hand-object interaction synthesis. *arXiv preprint arXiv:2309.07907*, 2023.

[7] Yujun Cai, Yiwei Wang, Yiheng Zhu, Tat-Jen Cham, Jianfei Cai, Junsong Yuan, Jun Liu, Chuanxia Zheng, Sijie Yan, Henghui Ding, et al. A unified 3d human motion synthesis model via conditional variational auto-encoder. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11645–11655, 2021.

[8] Tao Chen, Jie Xu, and Pulkit Agrawal. A system for general in-hand object re-orientation, 2021.

[9] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Kiana Ehsani, Jordi Salvador, Winson Han, Eric Kolve, Aniruddha Kembhavi, and Roozbeh Mottaghi. Procthor: Large-scale embodied ai using procedural generation. *Advances in Neural Information Processing Systems*, 35:5982–5994, 2022.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[11] Yan Ding, Xiaohan Zhang, Chris Paxton, and Shiqi Zhang. Task and motion planning with large language models for object rearrangement. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2086–2092. IEEE, 2023.

[12] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. In *arXiv preprint arXiv:2303.03378*, 2023.

[13] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[14] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael Black. Stochastic scene-aware motion prediction. In *Proceedings of the International Conference on Computer Vision 2021*, October 2021.

[15] Mohamed Hassan, Yunrong Guo, Tingwu Wang, Michael Black, Sanja Fidler, and Xue Bin Peng. Synthesizing physical character-scene interactions. 2023.

[16] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning, 2016.

[17] Daniel Holden, Taku Komura, and Jun Saito. Phase-functioned neural networks for character control. *ACM Transactions on Graphics (TOG)*, 36(4):1–13, 2017.

[18] Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[19] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems*, 36, 2024.

[20] Nan Jiang, Zhiyuan Zhang, Hongjie Li, Xiaoxuan Ma, Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, and Siyuan Huang. Scaling up dynamic human-scene interaction modeling. *arXiv preprint arXiv:2403.08629*, 2024.

[21] Jordan Juravsky, Yunrong Guo, Sanja Fidler, and Xue Bin Peng. Padl: Language-directed physics-based character control. In *SIGGRAPH Asia 2022 Conference Papers*, SA '22, New York, NY, USA, 2022. Association for Computing Machinery.

[22] Mukul Khanna*, Yongsen Mao*, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X. Chang, and Manolis Savva. Habitat Synthetic Scenes Dataset (HSSD-200): An Analysis of 3D Scene Scale and Realism Tradeoffs for ObjectGoal Navigation. *arXiv preprint*, 2023.

[23] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, et al. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017.

[24] Jiaman Li, Jiajun Wu, and C Karen Liu. Object motion guided human motion synthesis. *ACM Trans. Graph.*, 42(6), 2023.

[25] Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x: A large-scale 3d expressive whole-body human motion dataset. *Advances in Neural Information Processing Systems*, 2023.

[26] Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel Van De Panne. Character controllers using motion vaes. *ACM Transactions on Graphics*, 39(4), August 2020.

[27] Zhengyi Luo, Jinkun Cao, Alexander W. Winkler, Kris Kitani, and Weipeng Xu. Perpetual humanoid control for real-time simulated avatars. In *International Conference on Computer Vision (ICCV)*, 2023.

[28] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. Isaac gym: High performance gpu-based physics simulation for robot learning, 2021.

[29] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation, 2021.

[30] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge?, 2018.

[31] OpenAI. Gpt-4 technical report, 2024.

[32] Abhishek Padalkar, Acorn Pooley, Ajinkya Jain, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anikait Singh, Anthony Brohan, et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023.

[33] Liang Pan, Jingbo Wang, Buzhen Huang, Junyu Zhang, Haofan Wang, Xu Tang, and Yangang Wang. Synthesizing physically plausible human motions in 3d scenes. In *International Conference on 3D Vision (3DV)*, 2024.

 https://doi.org/10.52202/079017-0591

[34] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Trans. Graph.*, 37(4):143:1–143:14, July 2018.

[35] Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. Ase: Large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Trans. Graph.*, 41(4), July 2022.

[36] Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. Amp: Adversarial motion priors for stylized physics-based character control. *ACM Trans. Graph.*, 40(4), July 2021.

[37] Sergey Prokudin, Christoph Lassner, and Javier Romero. Efficient learning on point clouds with basis point sets, 2019.

[38] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation, 2017.

[39] Stephane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning, 2011.

[40] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

[41] Carmelo Sferrazza, Dun-Ming Huang, Xingyu Lin, Youngwoon Lee, and Pieter Abbeel. Humanoidbench: Simulated humanoid benchmark for whole-body locomotion and manipulation. *arXiv Preprint arxiv:2403.10506*, 2024.

[42] Sebastian Starke, Ian Mason, and Taku Komura. Deepphase: Periodic autoencoders for learning motion phase manifolds. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022.

[43] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *ACM Trans. Graph.*, 38(6), nov 2019.

[44] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *Advances in neural information processing systems*, 34:251–266, 2021.

[45] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.

[46] Chen Tessler, Yoni Kasten, Yunrong Guo, Shie Mannor, Gal Chechik, and Xue Bin Peng. Calm: Conditional adversarial latent models for directable virtual characters. In *ACM SIGGRAPH 2023 Conference Proceedings*, SIGGRAPH '23, New York, NY, USA, 2023. Association for Computing Machinery.

[47] Yinhuai Wang, Jing Lin, Ailing Zeng, Zhengyi Luo, Jian Zhang, and Lei Zhang. Physhoi: Physics-based imitation of dynamic human-object interaction, 2023.

[48] Luca Weihs, Matt Deitke, Aniruddha Kembhavi, and Roozbeh Mottaghi. Visual room rearrangement. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021.

[49] Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. Tidybot: Personalized robot assistance with large language models. *Autonomous Robots*, 47(8):1087–1102, 2023.

[50] Zeqi Xiao, Tai Wang, Jingbo Wang, Jinkun Cao, Wenwei Zhang, Bo Dai, Dahua Lin, and Jiangmiao Pang. Unified human-scene interaction via prompted chain-of-contacts. In *The Twelfth International Conference on Learning Representations*, 2024.

[51] Zhaoming Xie, Jonathan Tseng, Sebastian Starke, Michiel van de Panne, and C Karen Liu. Hierarchical planning and control for box loco-manipulation. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 6(3):1–18, 2023.

[52] Yinzhen Xu, Weikang Wan, Jialiang Zhang, Haoran Liu, Zikang Shan, Hao Shen, Ruicheng Wang, Haoran Geng, Yijia Weng, Jiayi Chen, Tengyu Liu, Li Yi, and He Wang. Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy, 2023.

[53] Sriram Yenamandra, Arun Ramachandran, Karmesh Yadav, Austin Wang, Mukul Khanna, Theophile Gervet, Tsung-Yen Yang, Vidhi Jain, Alexander William Clegg, John Turner, et al. Homerobot: Open-vocabulary mobile manipulation. *arXiv preprint arXiv:2306.11565*, 2023.

[54] Kaifeng Zhao, Yan Zhang, Shaofei Wang, Thabo Beeler, , and Siyu Tang. Synthesizing diverse human motions in 3d indoor scenes. In *International conference on computer vision (ICCV)*, 2023.

[55] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024.

[56] Qingxu Zhu, He Zhang, Mengting Lan, and Lei Han. Neural categorical priors for physics-based character control. *ACM Trans. Graph.*, 42(6), dec 2023.

[57] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

 https://doi.org/10.52202/079017-0591

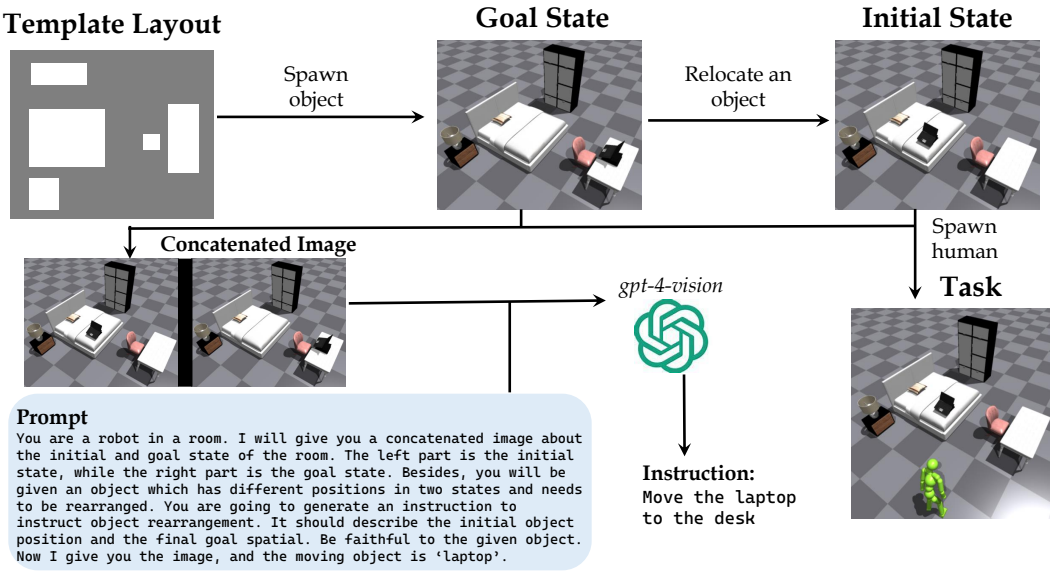

Figure 5: The task generation process of HITR dataset.

A Limitations

This work inherits the humanoid model from [36, 35, 15, 33, 50] with spherical hands. It is hard to manipulate small-sized objects. Dexterous hands can be equipped to facilitate object manipulation in future works.

As the first work on general object rearrangement, our task settings only include one object movement at each time. Long-horizon object rearrangement is left for future work.

The current version of our system does not contain explicit memorizing, planning, navigation, and multi-agent collaboration modules. We leave more ad-hoc designs to future work.

B Broader Impacts

We study simulated physical humans in the work, whose technique holds the potential for extension to real-world humanoid robots. This could have a significant positive societal impact, as humanoid robots have the potential to assist humanity in various ways. However, it is crucial to carefully consider safety concerns associated with the use of humanoid robots.

C Licenses

We use assets from ASE [36], HSSD [22], OMOMO [24], and SAMP [14] in this work. ASE is released under the NVIDIA license. HSSD is released under the CC BY-NC 4.0 license. OMOMO does not have a specified license. SAMP is released with its license on its GitHub repository.

D Dataset

D.1 Generation Process

Figure 5 provides an illustration of how we construct the Human-in-the-Room (HITR) dataset, following the procedural generation pipeline [9]. First, we manually design four distinct room layouts: *bedroom*, *livingroom*, *kitchen*, and *warehouse*. These template layouts are subsequently populated with various object models to create a set of diverse scenes, which serve as the goal states

https://github.com/mohamedhassanmus/SAMP?tab=readme-ov-file#license

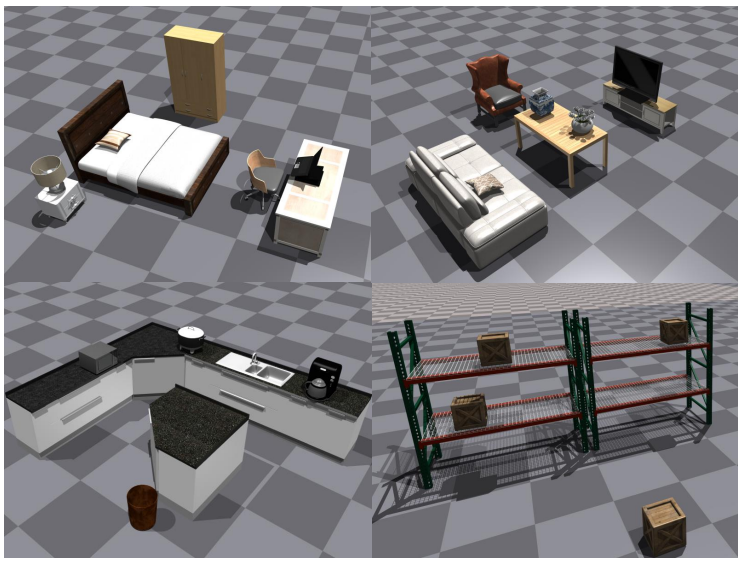

Figure 6: Different rooms in HITR dataset.

Figure 7: Movable objects in HITR dataset.

for the rearrangement tasks. To generate the initial state of each task, we randomly relocate an object to a different receptacle or to the ground. The initial position of the humanoid is randomly sampled from the navigable areas. As for the initial orientation, the humanoid heads to the object. Our dataset guarantees the object visibility in the first view and also covers the target position visibility in 89% of tasks. Notably, our dataset does not specify the initial humanoid pose; instead, it is sampled from the training motion dataset. We then concatenate images of the goal and initial states to create a self-contained information carrier for each task. It is sent to a large language model, specifically *gpt-4-vision*, along with language prompts, to generate the corresponding instruction.

D.2 Statistics

The HITR dataset contains 615 unique tasks in various rooms, examples of which are depicted in Fig. 6. The dataset includes 34 movable objects and 50 static objects, all of which are sourced from HSSD [22]. All movable objects are shown in Fig. 7 and span a wide range of categories such as *chair*, *pillow*, *plant*, *coffeemaker*, among others. To facilitate successful interaction with our humanoid

https://doi.org/10.52202/079017-0591

model, we manually adjust the scale of object models and assign suitable weights. The sizes of the objects vary from 21*cm* to 126*cm*, and their weights range from 5*kg* to 20*kg*. On average, there are 6.5 objects present in each scene.

E Details of the Approach

We describe the complete details of our approach in this section.

E.1 Training HumanVLA-Teacher

E.1.1 Observation Space

The observation space of HumanVLA-Teacher includes proprioception, object, goal, and waypoint. The 223-dimensional proprioception includes:

- root height $\in \mathcal{R}^1$
- root rotation $\in \mathcal{R}^6$
- root linear velocity $\in \mathcal{R}^3$
- root angular velocity $\in \mathcal{R}^3$
- link position $\in \mathcal{R}^{14\times3}$
- link rotation $\in \mathcal{R}^{14\times6}$
- link linear velocity $\in \mathcal{R}^{14\times3}$
- link angular velocity $\in \mathcal{R}^{14\times3}$

The object state includes object position ($\mathcal{R}^3$), rotation ($\mathcal{R}^6$), linear velocity ($\mathcal{R}^3$), angular velocity ($\mathcal{R}^3$) and BPS [37] geometry ($\mathcal{R}^{200\times3}$) encoded by delta vectors of 200 basis points.

The goal state includes the goal position ($\mathcal{R}^3$) and rotation ($\mathcal{R}^6$).

The waypoint is denoted by $x_t^{wp} \in \mathcal{R}^3$.

We follow the default AMP [36] to use a projected observation space for the discriminator. They include:

- root height $\in \mathcal{R}^1$
- root rotation $\in \mathcal{R}^6$
- root linear velocity $\in \mathcal{R}^3$
- root angular velocity $\in \mathcal{R}^3$
- joint rotation $\in \mathcal{R}^{12\times6}$
- joint velocity $\in \mathcal{R}^{28\times1}$
- end-effector positions of left/right hand/foot, and head $\in \mathcal{R}^{5\times3}$
- object position $\in \mathcal{R}^3$

We send 10 consecutive frames to the discriminator; thus the total dimension is $\mathcal{R}^{10\times131}$.

All these features are represented in the local coordinate of the humanoid model. Rotations are encoded using a 6-D normal-tangent representation.

E.1.2 Action Space

The action space ($\mathcal{R}^{28}$) of HumanVLA-Teacher consists of the target positions for 28 Proportional-Derivative controllers. Predicted actions are then utilized by the controllers to generate joint torques for effective control.

E.1.3 Network Architecture

We adopt MLPs as the basic networks for HumanVLA-Teacher. Each linear layer is followed by ReLU activation. The 600-dimensional BPS feature is compressed using an MLP of [512,512,128] layers to generate a low-dimensional representation. This representation is then concatenated with all other observations to derive the action and value. The actor, critic, and discriminator networks are all separate MLPs with hidden layers of [1024, 1024, 512]. The actor and critic networks are trained using default PPO [40] losses. The discriminator is trained using a cross-entropy loss via adversarial learning.

E.1.4 In-context Path Planning

Navigating through intricate scenes is challenging due to the potential for unexpected object collisions. To mitigate this, we employ in-context path planning to facilitate collision-free locomotion, as depicted in Fig. 8. This process involves two substreams: planning a path from the humanoid to the object, and then from the object to the goal. Initially, we sample point clouds from all objects in the scene. These points are then projected top-down onto the ground and divided into 50*cm* x 50*cm* grids to construct

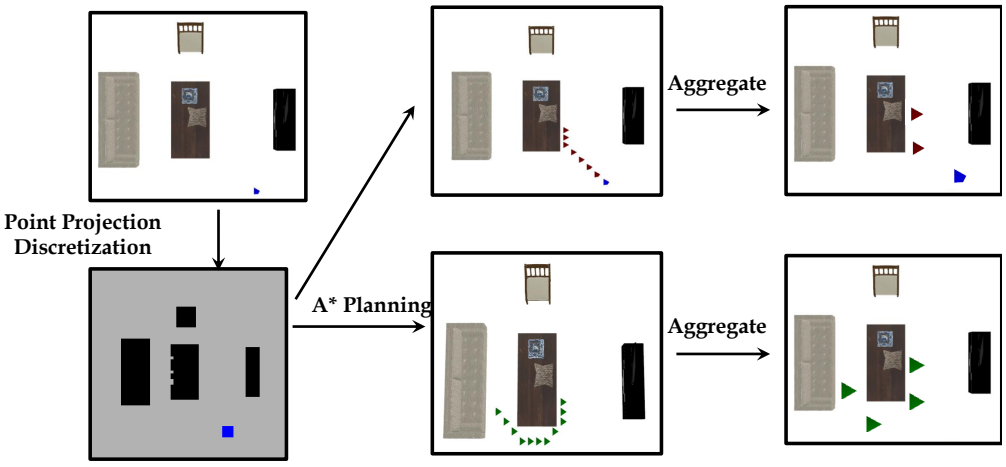

Figure 8: An overview of the path planning process. The blue mark denotes the initial position. Red marks denote the path from the initial position to the object. Green marks denote the path from the object to the goal.

a navigation map. We utilize the A^* algorithm [13] to plan two paths: one from the humanoid to the object, and another from the object to the goal. Each path generated by the A^* algorithm is represented by a series of densely packed waypoints. We consolidate waypoints that share a consistent moving direction to create a sparse waypoint set. During task execution, the humanoid model is guided by a sequence of these waypoints and is encouraged to move toward the waypoint at each step. Once the humanoid reaches the waypoint within a 50*cm* distance, the waypoint proceeds to the next. The last waypoint is the goal position. The waypoint is used in reward computation to guide the movement, described in the following sections.

E.1.5 Carry Curriculum Pre-training

We uniformly conceptualize object rearrangement as a three-step process: locomotion towards the object, making contact with the object, and then relocating the object to achieve the goal. However, directly training these three processes from scratch can be exceptionally challenging. The main difficulty lies in the fact that physical interactions with objects necessitate contacting prior, such as robust object holding, to enable subsequent object dynamics. Through empirical experiments, we have found that carrying objects to other receptacles is significantly more difficult than pushing and pulling objects on the ground. This underscores the need to learn a carry prior to boosting the subsequent object relocation.

Inspired by the curriculum learning paradigm [4], we design a carry curriculum pre-training scheme to learn the carry prior. It encompasses the initial two steps, namely locomotion and contacting, which form an easier curriculum compared to the difficult three-step rearrangement task. We exclude on-ground objects in carry curriculum pre-training, whose initial height and goal height are both smaller than $0.1m$. These objects, such as *small table*, *chair*, can be relocated by direct on-ground pushing and pulling without a lift. The goal of the carry curriculum is to enable the agent to walk towards the object and establish a robust contact, allowing it to securely hold the object in the air; thus, we define reward as a combination of walking reward and contacting reward:

$$r_t = r_t^{walk} + r_t^{contact} \tag{6}$$

where r_t^{walk} encourages the humanoid to walk to the object. Specifically, it encourages a closer distance between root position x_t^{root} and object position x_t^{root}, as well as walking to the waypoint direction x_t^{wp} at a target speed v^*=1.5*m/s*:

$$r_t^{walk} = \begin{cases} 0.1\exp(-0.5||x_t^{obj} - x_t^{root}||) + 0.2\exp(-2||v^* - v_t^{root} \cdot x_t^{wp}||^2) & ||x_t^{obj} - x_t^{root}|| > 0.5 \\ 0.3 & \text{otherwise} \end{cases} \tag{7}$$

https://doi.org/10.52202/079017-0591

Table 5: Hyperparameters for HumanVLA-Teacher training.

Hyperparameter	Value	Hyperparameter	Value	Hyperparameter	Value
Num. envs	16,384	Max episode length	300	Num. epochs	30,000
Discount factor	0.99	GAE parameter	0.95	Observation clip	5
Action clip	1	Optimizer	Adam	Learning rate	3e-5
Actor loss weight	1	Critic loss weight	5	Discriminator loss weight	5
Num. rollouts per PPO update	32	PPO clip	0.2	PPO miniepochs	5
Num. batches per miniepoch	8	AMP consecutive frames t^*	10	Gradient penalty w^{gp}	5
Task reward weight w^G	1	Style reward weight w^S	1	Min style clipping bound ξ_{min}	0.4

$r_t^{contact}$ encourages the humanoid hands x_t^{hand} to contact with the object x_t^{obj}, followed by lifting the object up for $\Delta h = 0.3m$ from an initial height h_{init}^{obj}. It is formulated as:

$$r_t^{contact} = 0.2\exp(-10||x_t^{obj} - x_t^{hand}||) + 0.5\,\text{clip}(h_t^{obj} - h_{init}^{obj}, 0, \Delta h)/\Delta h \tag{8}$$

E.1.6 Rearrangement Learning

The complete rearrangement task consists of the whole three-step process. The reward is formulated as a combination of walking reward, contacting reward, and relocation reward:

$$r_t = r_t^{walk} + r_t^{contact} + r_t^{relocation} \tag{9}$$

where r_t^{walk}, $r_t^{contact}$ have similar formulation but different weights and conditions compared to the carry curriculum pre-training, specifically:

$$r_t^{walk} = \begin{cases} 0.1\exp(-0.5||x_t^{obj} - x_t^{root}||) + 0.1\exp(-2||v^* - v_t^{root}\cdot x_t^{wp}||^2) & ||x_t^{obj} - x_t^{root}|| > 0.5 \\ 0.2 & \text{otherwise} \end{cases} \tag{10}$$

$$r_t^{contact} = \begin{cases} 0.1\exp(-10||x_t^{obj} - x_t^{hand}||) + 0.1\,\text{clip}(h_t^{obj} - h_{init}^{obj}, 0, \Delta h)/\Delta h & ||x_t^{obj} - x_{goal}^{obj}|| > 0.5 \\ 0.2 & \text{otherwise} \end{cases} \tag{11}$$

In the context of the rearrangement task, we do not set an explicit lifting condition, but instead require the contact prior to enhance subsequent dynamics. As a result, the height change, Δh is reduced to $0.1m$ for objects that are going to be carried, and to zero for objects that are consistently on the ground.

$r_t^{relocation}$ comprises four components, namely r_t^{vel}, r_t^{far}, r_t^{near}, and r_t^{rot}. r_t^{vel} encourages moving the object to the next waypoint x_t^{wp}. r_t^{far} encourages a close distance to the next waypoint. r_t^{near} encourages to meet the goal position x_{goal}. r_t^{rot} encourages to meet the rotation position q_{goal}. They are formulated as:

$$r_t^{relocation} = 0.1r_t^{vel} + 0.2r_t^{far} + 0.2r_t^{near} + 0.1r_t^{rot} \tag{12}$$

$$r_t^{vel} = \begin{cases} \exp(-2||v^* - v_t^{obj}\cdot x_t^{wp}||^2) & ||x_t^{obj} - x_{goal}^{obj}|| > 0.5 \\ 1 & \text{otherwise} \end{cases} \tag{13}$$

$$r_t^{far} = \exp(-||x_t^{obj} - x_t^{wp}||) \tag{14}$$

$$r_t^{near} = \exp(-5||x_t^{obj} - x_{goal}||) \tag{15}$$

$$r_t^{rot} = \exp(-2||q_t^{obj} - q_{goal}||) \tag{16}$$

E.1.7 Hyperparameter Setting

We provide a hyperparameter table for HumanVLA-Teacher training in Tab. 5. It is shared for both the carry curriculum pre-training and rearrangement learning.

E.2 Training HumanVLA

E.2.1 Observation Space

The observation space of HumanVLA includes proprioception, last action, egocentric image, and language instruction.

Table 6: Hyperparameters for HumanVLA training.

Hyperparameter	Value	Hyperparameter	Value	Hyperparameter	Value
Num. envs	585	Max episode length	300	Num. epochs	20,000
Observation clip	5	Action clip	1	Optimizer	Adam
Learning rate	5e-4	Num. rollouts per epoch	1	Num. train steps per epoch	5
Batch size	600	DAgger β_0	1	DAgger λ	0.998
Camera resolution	256,256	Camera FoV	90	Active rendering w^{AR}	0.2

The proprioception adheres to the 223-dimensional feature defined in the HumanVLA-Teacher (Sec. E.1.1).

The last action is denoted by $a_{t-1} \in \mathcal{R}^{28}$. An all-zero feature is used for it at the first step.

The egocentric image has 256 x 256 pixels with a field of view spanning 90 degrees. We mount a camera on the head of the humanoid model, with the camera position offset from the head being [0.103, 0, 0.175]. The camera direction aligns with the forward direction of the head.

The natural language instruction specifies the task goal.

E.2.2 Action Space

The action space for HumanVLA aligns with the 28-dimensional HumanVLA-Teacher action space defined in Sec. E.1.2.

E.2.3 Network Architecture

The HumanVLA network consists of an image encoder, a text encoder, and an action decoder. The image encoder is an *EfficientNet-B0* [45] while the text encoder is a frozen *bert-base-uncased* [10]. Our primary design choices for these two models are their fast inference speed. The 1280-dimensional *EfficientNet-B0* feature is passed through a linear layer to yield a compressed 128-dimensional feature. An MLP with [512,128] hidden layers is used to compress the 768-dimensional *bert-base-uncased* feature down 128 dimensions. The compressed image and text features are then concatenated with the proprioception and last action, and this combined feature is sent into the action decoder. The action decoder is a 6-layer MLP, with each hidden layer being 1024-dimensional. Each linear layer is followed by BatchNorm and ReLU. Skip connections are used between the first and third layers, as well as between the third and fifth layers.

E.2.4 Active Rendering Action

We propose the active rendering technique to enhance the quality of visual perception and facilitate object rearrangement. In this section, we provide more details on how to compute the active rendering action, specifically applied to the neck to adjust the camera pose. The primary regulation focuses on the camera's target view, which is the forward direction. To determine this, we compute the center of the object point cloud and use it as the target viewpoint. The direction from the camera position to the point cloud center represents the expected forward direction of both the camera and the head. Since the neck joint is a 3-DoF spherical joint, a single regulation can lead to ambiguous actions. To address this, we introduce the second regulation that controls the side direction of the head. It is perpendicular to the plane formed by the upward torso and the camera view. These two regulations result in a unique head rotation. Finally, inverse kinematics is used to solve neck actions.

E.2.5 Learning Process

We train HumanVLA by cloning actions from HumanVLA-Teacher. At each step, the HumanVLA-Teacher predicts an action using privileged information. It is mixed with an active rendering action to yield the label for supervision. HumanVLA is optimized to minimize the mean square error.

We adopt a DAgger [39] framework to manage the online learning process. DAgger iteratively schedules a mixed policy $\pi = \beta_t \pi^{tch} + (1 - \beta_t)\pi^{stu}$ at epoch t to explore the environment and expand a training dataset. It can alleviate the covariate shift problem between different policies. We use an exponential function to schedule the mixing factor $\beta_t = \beta_0 * \lambda^t$.

https://doi.org/10.52202/079017-0591

Figure 9: Learning curve comparison w/ and w/o style reward clipping.

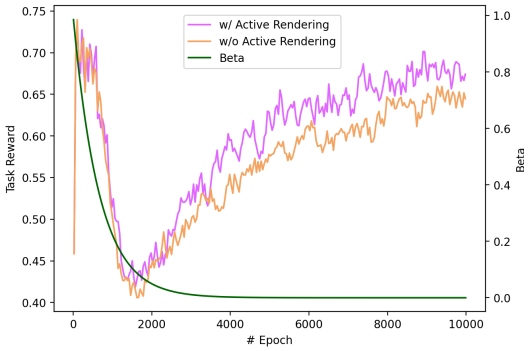

Figure 10: Learning curve comparison w/ and w/o active rendering. The process is dominated by the teacher policy in the early stage with high β and demonstrates the reward upper bound.

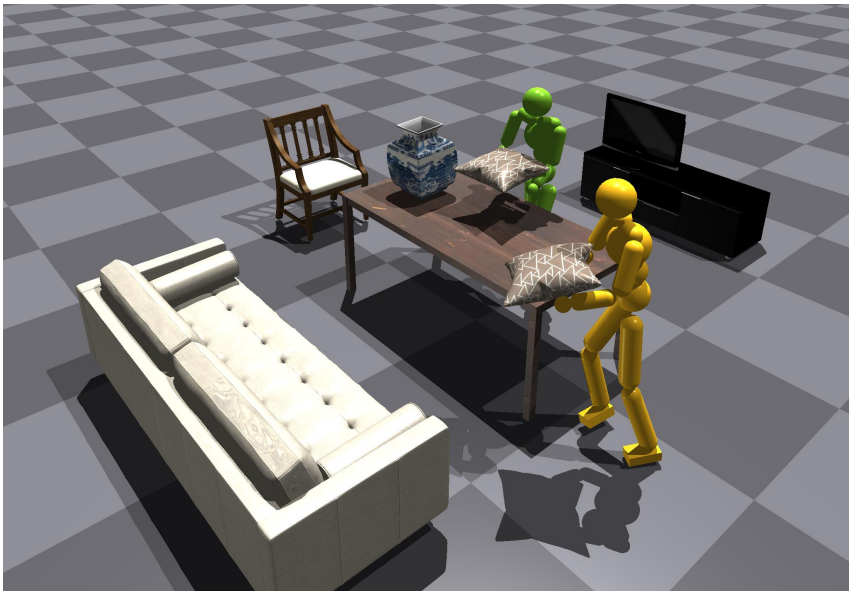

Figure 11: Comparison w/ and w/o path planning. The green humanoid without path guidance fails to get close to the sofa, while the yellow humanoid with path guidance learns to go around the central table. Instruction: Move the pillow to the sofa.

E.2.6 Hyperparameter Setting

We provide a hyperparameter table for HumanVLA training in Tab. 6.

F Additional Results

F.1 Generalization

The primary test set of HITR contains task-level unseen tasks including new compositions of objects in the scene, new placement of objects, and regenerated new text instructions from LLM describing new compositions and new spatial relations. We have proven the generalization ability of HumanVLA in unseen tasks.

However, generalization in any unseen without any similar pattern in the seen data is super challenging, which is also an ultimate goal of embodied AI research. We build extra tiny unseen data to make additional analysis and further disclose our method. We make additional testing data: (1) Unseen texts

Table 7: Unseen data analysis.

	Success Rate (%) ↑	Precision (cm)↓	Execution Time (s) ↓
Useen Text	65	50.4	5.4
Unseen object (visual)	50	72.3	6.2
Unseen object (geometry)	20	118.8	7.9
Unseen scene layout	35	88.5	6.8

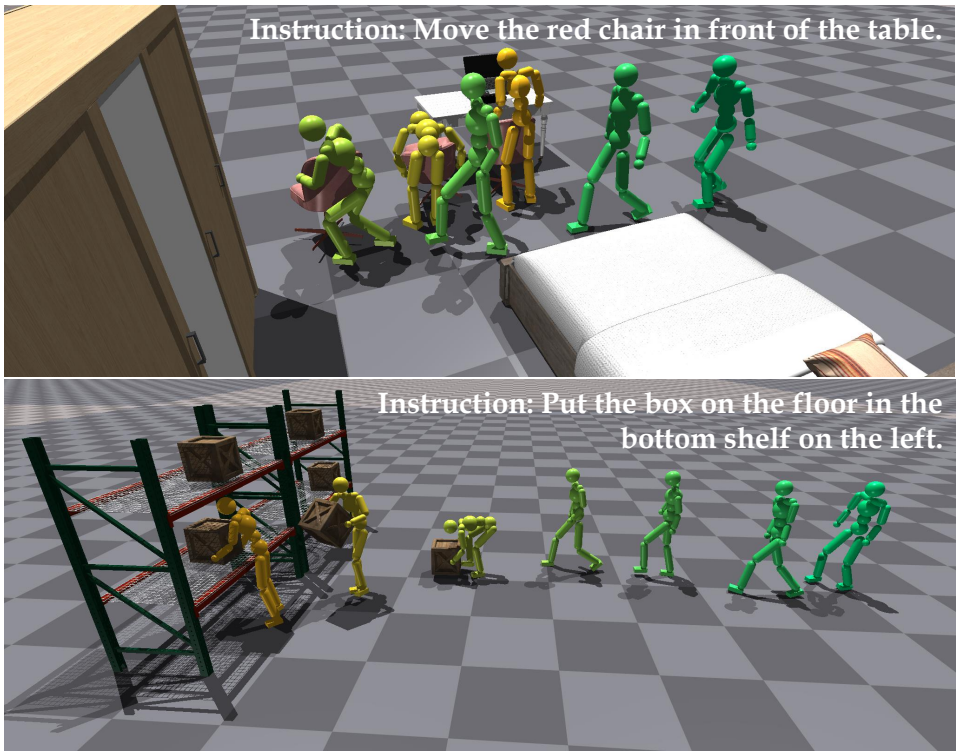

Figure 12: Additional qualitative results.

generated for training tasks, manually reviewed to be distinct from training data. (2) Unseen objects by changing visual appearance in training tasks. (3) Unseen object category (cup) with different geometry. (4) Unseen scene layouts by repositioning static large objects.

Results are reported in Tab. 7. We find that our work suffers less from unseen texts and unseen visual appearance. But generalizing to unseen object categories and execution in the unseen scenes remains a main challenge.

F.2 Learning Curves

We plot the learning curves in Fig. 9 and Fig. 10 to demonstrate the efficacy of our method. In Fig. 9, our method has been proven to converge faster in task completion with style reward clipping. In Fig. 10, active rendering improves perception quality and facilitates the learning of the student policy.

F.3 Qualitative Ablation

A qualitative ablation about the path planning module is represented in Fig. 11. Specifically, the green humanoid fails to navigate close to the goal receptacle, i.e., the sofa. However, the yellow humanoid is guided by planned waypoints to go around the center table and place the pillow on the sofa.

F.4 Additional Qualitative Results

We provide additional qualitative visualizations in Fig. 12 to disclose our results.

https://doi.org/10.52202/079017-0591

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: This work studies vision-language directed object rearrangement. It have been described in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations are in the appendix.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: No theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide complete details in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

https://doi.org/10.52202/079017-0591

Answer: [Yes]

Justification: Data and code are publicly available.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide complete details in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Statistical significance is reported in Sec. 5.3. We report the standard deviation in 10 repeated runs to validate the robustness of our method. For a consistent result format with InterPhys, they are not included in the main table.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Compute costs are in Sec. 5.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This paper conforms NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Broader impacts are in the appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

https://doi.org/10.52202/079017-0591

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited works of related assets. License are mentioned in the appendix.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new assets introduced.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No crowdsourcing experiment.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No crowdsourcing experiment.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-0591