Helpful DoggyBot: Open-World Object Fetching using Legged Robots and Vision-Language Models

Qi Wu[1]   Zipeng Fu[1]   Xuxin Cheng[2]   Xiaolong Wang[2]   Chelsea Finn[1]

[1]Stanford University   [2]UC San Diego
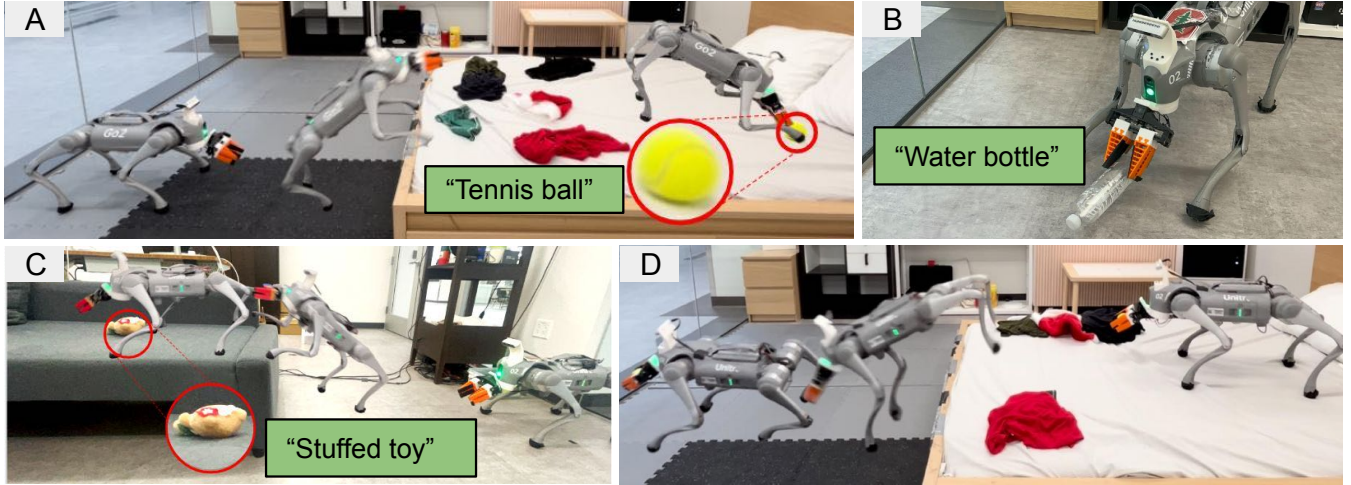
https://helpful-doggybot.github.io/

Fig. 1: **DoggyBot for Open-World Object Fetching**. Using the coordination of commands from VLMs and a low-level whole-body policy, our robot can (A) climb up a bed to fetch a tennis ball, (B) bend down to pick up a water bottle, (C) climb up a couch to pick up a stuffed toy, and (D) climb down the bed after retrieving the tennis bell.

Abstract— **Learning-based methods have achieved strong performance for quadrupedal locomotion. However, several challenges prevent quadrupeds from learning helpful indoor skills that require interaction with environments and humans: lack of end-effectors for manipulation, limited semantic understanding using only simulation data, and low traversability and reachability in indoor environments. We present a system for quadrupedal mobile manipulation in indoor environments. It uses a front-mounted gripper for object manipulation, a low-level controller trained in simulation using egocentric depth for agile skills like climbing and whole-body tilting, and pre-trained vision-language models (VLMs) with a third-person fisheye and an egocentric RGB camera for semantic understanding and command generation. We evaluate our system in two unseen environments without any real-world data collection or training. Our system can zero-shot generalize to these environments and complete tasks, like following user's commands to fetch a randomly placed stuff toy after climbing over a queen-sized bed, with a 60% success rate.**

I. INTRODUCTION

Quadrupedal robots powered by learning-based methods have made significant strides in locomotion capabilities in recent years, demonstrating impressive agility and robustness across diverse terrains [1], [2]. However, their potential for assisting humans in everyday indoor environments remains largely untapped, like the ability to understand and follow language instructions to fetch a bottle of water for you. Several key challenges have hindered progress in this direction. First, equipping quadrupeds with effective manipulation capabilities without compromising agility is difficult, as traditional robotic arms often add significant weight and complexity [3]. While recent advances have demonstrated impressive quadrupedal agility, navigating cluttered indoor spaces and reaching high surfaces like beds or sofas requires a level of body control and environmental reasoning that goes beyond existing approaches. Moreover, bridging the semantic gap between simulation and reality remains a significant hurdle. Learning-based controllers trained in simulation often struggle to generalize to the rich, context-dependent nature of real-world indoor scenes, due to mismatches between simulation rendering and real-world sensing, and complexity in specifying diverse real-world scenarios in simulation. This limits robots' ability to understand and interact with diverse household objects and environments.

In this paper, we present Helpful DoggyBot, a quadrupedal robot system that aims to overcome these limitations and enable helpful mobile manipulation skills that can understand human commands and generalize across different indoor environments. To empower quadrupeds with general manipulation capabilities while still maintaining their agility, we design a simple yet effective 1-DoF gripper that is mounted on the bottom front of the robot. Shown in Figure 1, The gripper, serving as the "mouth" of our robot, allows it to pick up and firmly hold everyday objects through "biting".

To increase the traversability and reachability of quadrupeds compared to prior work [4], [5], we use reinforcement learning and simulation to train a general-purpose low-level controller

using egocentric depth and proprioception. We randomly sample robot commands including linear velocity, angular velocity, and pitch in a task-agnostic fashion in environments full of challenging obstacles during training. During zero-shot deployment in the real world after training, the controller takes in real-time egocentric depth measurements and task-specific commands. This controller equips robots with agile locomotion skills like climbing up a tall 0.5m obstacle and whole-body tilting while moving, powering quadrupeds to reach high workspaces like sofas and beds and to grasp objects located on these places. These agile locomotion skills are also critical for navigating cluttered indoor spaces, where the ability to surmount various obstacles simplifies the otherwise complex maneuvers required for effective navigation.

On the semantic perception and control front for solving useful tasks, instead of relying on collecting human demonstrations that is time-consuming or simulation that has semantic gaps, we leverage off-the-shelf VLMs to achieve zero-shot generalization in objects and configurations. Using VLMs and real-time video streams from a fish-eye top-down RGB camera mounted on the ceiling, our system can parse the open-vocabulary command of an object of interest, identify, localize and track the target object and robot itself within the scene, and generate reactive navigation commands based on the locations of the target object and the robot for the low-level controller. Upon approaching target objects, our system uses an egocentric RGB camera for tracking relative positions of the target object which are converted into velocity, pitch and grasping commands.

Our Helpful DoggyBot integrates simulation training for low-level control, and VLMs for semantic understanding and command generation. We evaluate Helpful DoggyBot in an unseen bedroom and an unseen living room, demonstrating its ability to complete open-vocabulary object fetching tasks like navigating to a randomly placed stuffed toy, climbing a bed and fetching back the toy from atop the bed with a 60% success rate. Notably, our system achieves this generalization without any real-world data collection or training, highlighting the potential of our approach for creating helpful quadrupedal assistants that can adapt to diverse home environments. The key contributions of our system include (1) a simple yet effective 1-DoF gripper design that enables object grasping for quadrupeds, (2) a general-purpose low-level controller trained in simulation that enables real-world parkour-like mobility, (3) an approach leveraging pre-trained VLMs for semantic understanding for quadrupeds through generating reactive velocity, pitch and grasping commands, and (4) experimental validation demonstrating zero-shot generalization to unseen indoor configurations for mobile manipulation tasks.

II. Related Work

Legged Mobile Manipulation. Legged robots have long been of interest for their potential to traverse complex terrains while performing manipulation tasks. Much work in this area focused on bipedal humanoid platforms [6]–[11], demonstrating basic manipulation while maintaining balance [12]–[21]. More recently, quadrupedal robots have gained attention for their inherent stability and agility [1], [4], [5], [22]–[36]. Several approaches have been explored to enable manipulation capabilities on quadrupeds. One common method is to mount a robotic arm on the quadruped's back [3], [37]–[45]. While this provides significant dexterity, it also adds considerable weight and complexity to the system, hence reducing the agility of quadrupeds. An alternative approach is to utilize the quadruped's existing or modified limbs and torso for simple pushing tasks [46]–[51]. Learning-based methods have shown promise in developing legged manipulation skills. For instance, [42] combines imitation learning for target end-effector trajectory generation and reinforcement learning for low-level control. [48] chains multiple polices to complete pushing tasks guided by fiducial markers. However, these approaches often struggle to generalize beyond the specific tasks and environments used during training. Our work builds upon these foundations by introducing a simple yet effective gripper design and a learning approach that enables generalization to unseen environments. Unlike previous work, we focus on enabling helpful indoor tasks that require both agile locomotion and object manipulation.

Robot Learning using Large Pretrained Models. The advent of large pre-trained models, particularly in the domains of computer vision and natural language processing, has opened new avenues for robot learning [52]–[71]. These models, trained on vast amounts of visual data, offer rich semantic representations that can be leveraged for various robotic tasks. In the context of manipulation, [53], [55] use VLMs to generate cost functions for tasks specified in language instructions, while [57], [72]–[74] use VLMs to directly generate executable commands or intermediate presentations. These approaches, however, were primarily focused on static manipulation scenarios. For mobile robots, recent work has explored using large pretrained models for navigation and locomotion [50], [57], [75]–[78]. For example, prior work demonstrates how VLMs can be used to generate navigation commands for wheeled robots [75] and legged robots [58]. However, the integration of these models with mobile manipulation remains relatively unexplored. Our work bridges this gap by leveraging pretrained vision-language models to enable semantic understanding and adaptive behavior generation for a quadrupedal robot performing mobile manipulation tasks. Unlike previous approaches, we demonstrate how large pretrained models can be effectively used in conjunction with learned low-level controllers to enable zero-shot generalization to mobile manipulation tasks.

III. Hardware

Shown in Figure 2, our robot hardware system consists of a 12-DoF Unitree Go2 quadruped robot and a 1-DoF gripper mounted on the bottom from of the robot. Both are powered by the onboard battery of Go2. We 3D-print and custom-build our Finray gripper which is actuated by a Dynamixel XM430-W350-T servo motor through a slider-crank mechanism for fast closing. We use the onboard Jetson to run our learned low-level controller that takes egocentric depth from a RealSense D435 and proprioception as input, and
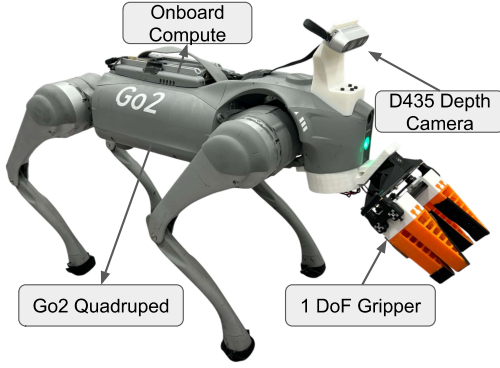
Fig. 2: **Hardware Setup.** We use a Go2 quadruped and a custom-built 3D printed Gripper actuated by Dynamixel XM430-W350-T servo motor. An egocentric RealSense D435 is mounted on the top front of the robot with 30 degrees downwards.

VLMs upon approaching objects that takes egocentric RGB as input. We send high-level commands, that are generated from VLMs running on a separate workstation that takes third-person top-down RGB stream as input, to the Jetsen through Wi-Fi.

IV. LEARNING A GENERAL WHOLE-BODY CONTROLLER

To enable effective mobile manipulation in diverse indoor environments, our robot requires both agile locomotion skills for traversing challenging obstacles and precise whole-body control for expanding its workspace. Previous works have typically addressed these challenges separately [32], [33], but integrating multiple objectives into a single learning framework introduces new complexities. These include increased exploration burden and the potential for sub-optimal behaviors when optimizing for multiple objectives simultaneously [43]. Shown in Figure 3, our approach leverages a two-phase training process focusing on the whole-body control and agility to overcome these challenges.

A. Phase 1: Training with Privileged Information

We develop our agile visual whole-body control policy through a two-phase training process: In the first phase, we train a policy using PPO [79] to optimize both whole-body control and agile locomotion objectives. During this phase, the policy uses privileged information in the form of scandots, capturing heights of terrain near the robot, as observations, allowing for efficient learning in simulation.

Whole-body objective: This objective enables the robot to track a randomly sampled pitch command, expanding the workspace of the 1-DoF gripper. We define the reward as:

$$r_{\text{wb}} = \exp(-3 \cdot |p_{\text{cmd}} - p|) \tag{1}$$

where p_{cmd} is the commanded pitch uniformly sampled from the range $[-30°, 30°]$ and p is the actual pitch angle of the robot's body. We remove this objective only when the robot is encourtering obstacles to avoid conflicting objectives.

Agile locomotion objective: This objective encourages the robot to traverse challenging obstacles such as high steps. To mitigate the exploration burden, we adopt a velocity tracking

reward inspired by [32]:

$$r_{\text{tracking}} = \min(\langle v, \hat{d}_{\text{wp}}\rangle, v_{\text{cmd}})/v_{\text{cmd}} \tag{2}$$

where $v \in \mathbb{R}^2$ is the robot's current velocity in the world frame, $v_{\text{cmd}} \in \mathbb{R}$ is the linear velocity command sampled from the range [0, 1m/s], and $\hat{d}_{\text{wp}}$ is the unit vector pointing towards the next waypoint. We convert $\hat{d}_{\text{wp}}$ into angular velocity command ω_{cmd} as a policy input, which calculates the angular difference between robot's current direction and $\hat{d}_{\text{wp}}$, removing dependency of the policy on global information. We track velocity in the world frame to prevent the robot from learning unintended behaviors like circumventing obstacles. We compute the direction using waypoints placed on the terrain:

$$\hat{d}_{\text{wp}} = \frac{x_{\text{wp}} - x}{|x_{\text{wp}} - x|} \tag{3}$$

where x_w is the location of next waypoint and x is the robot's current position in the world frame.

B. Phase 2: Policy Distillation using Egocentric Depth

To enable real-world deployment, we distill the learned policy during Phase 1 into a deployable policy that operates on depth images from a front-facing camera instead of privileged scandots information. We use Regularized Online Adaptation (ROA) [43] to train an online estimator to recover environmental information from the history of onboard observations. Our online estimator architecture consists of a convolutional neural network (CNN) followed by a gated recurrent unit (GRU) to process the temporal sequence of depth images. This design allows the policy to capture both spatial and temporal information from the visual input. The output of this estimator replaces the scandots input to the base policy learned in Phase 1. A key difference from previous work [32] is that we do not perform dual distillation of both the heading command and the exteroception simultaneously. Instead, we leverage a more powerful VLM to specify the robot's intended heading direction. This approach helps us avoid potential out-of-distribution problems that can arise in dual-distillation processes.

C. Simulation Environments and Training Curricula

To ensure robust performance across diverse scenarios, we train our policy in a variety of simulated environments featuring challenging obstacles such as stairs and uneven terrain. We randomly generate these environments for each training episode, varying parameters like stair height, number of stairs, and terrain friction to promote generalization, as shown in Table V and Table IV

To further improve learning efficiency and policy performance, we employ reward shaping techniques and a curriculum learning approach. We introduce auxiliary rewards for maintaining balance, minimizing energy consumption, and smooth transitions between different locomotion modes (e.g., walking, climbing, and tilting). Rewards are shown in Table III

The curriculum progressively increases the difficulty of the training environments, starting with simple flat terrains and gradually introducing more complex obstacles as the policy improves. By combining these techniques with our two-phase
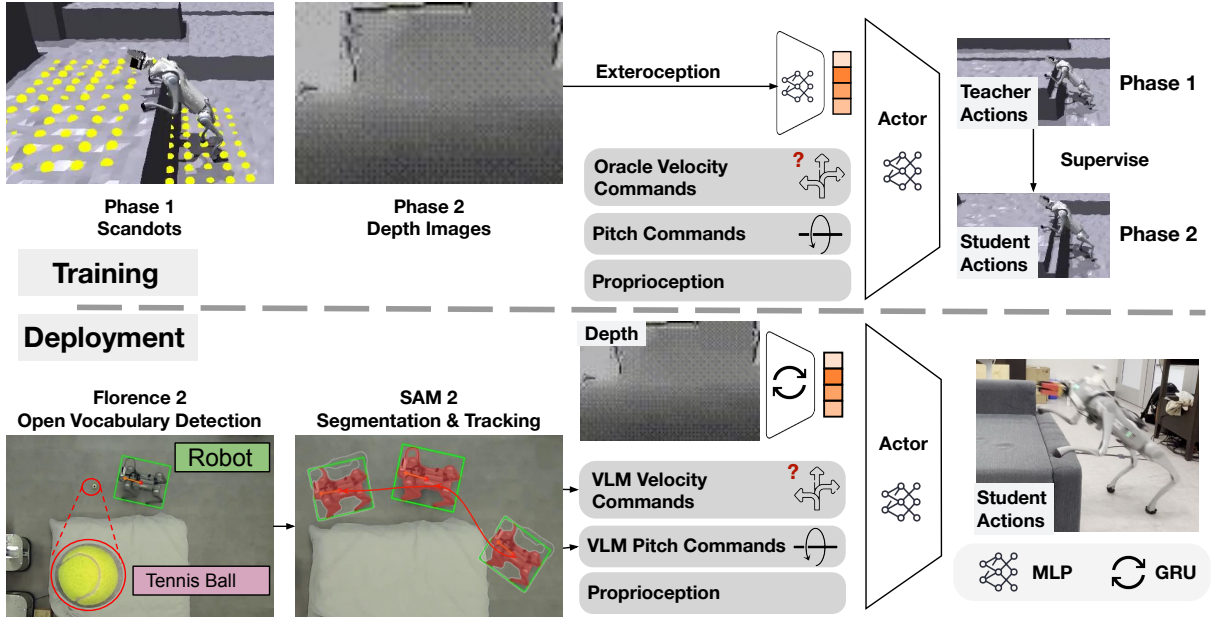
Fig. 3: **System Overview.** We use a two-phase framework to train a depth-based policy as the low-evel whole-body controller. During deployment, we use VLMs for open-vocabulary detection, segmentation and tracking models to provide velocity commands and pitch commands for the controller.

training process, we develop a general whole-body controller capable of agile locomotion and precise manipulation in diverse indoor environments. This approach enables our quadrupedal robot to navigate challenging obstacles and perform complex mobile manipulation tasks without requiring extensive real-world data collection or environment-specific training.

V. Zero-Shot Deployment using VLMs

To enable zero-shot generalization to unseen environments and objects, we leverage pre-trained VLMs for semantic understanding and adaptive behavior generation. Our system integrates open-vocabulary object detection, efficient navigation, and precise grasping, all without requiring task-specific training data or fine-tuning.

A. Open-Vocabulary Detection, Segmentation and Tracking

Our system employs a combination of state-of-the-art vision models to achieve robust open-vocabulary object detection, segmentation and tracking.

Initial Detection: We utilize Florence-2 [80] to perform open-vocabulary object detection. This allows our system to identify and localize both the robot itself and the target objects based on natural language descriptions, enabling flexibility in task specification.

Segmentation: Following initial detection, we apply SAM2 (Segment Anything Model 2) [81] to generate precise object masks. The integration of Florence-2 and SAM2 enables our system to handle a wide range of objects without prior training on specific categories.

Tracking: To maintain real-time performance, we employ SAM2 for object tracking at 10 Hz. This approach allows for continuous updating of the object's position in the environment, crucial for navigation and manipulation tasks.

B. Navigation

Our navigation system leverages a top-down fisheye camera mounted on the ceiling to provide a global view of the environment. This perspective enables simultaneous tracking of both the robot and target object positions, simplifying the planning process. We use the detected object position as a single waypoint for navigation, generating commands that guide the robot efficiently towards its goal. The system maintains a constant linear velocity of 0.8 m/s towards the waypoint, while angular velocity is computed using a proportional controller with $K_p = 0.5$ based on the difference between the robot's current heading and the vector pointing to the waypoint. During this phase, the pitch command is set to 0. To ensure smooth integration of locomotion and manipulation, the system transitions from navigation to grasping mode when the robot is approximately 1 meter away from the target object. We assume that our low-level controller can traverse most indoor obstacles like beds and sofas, thus alleviating the need for obstacle avoidance.

C. Grasping Objects

As the robot approaches the target object, it switches to a precise grasping strategy using its front-mounted gripper, transitioning from global to egocentric perception. The system now relies on egocentric depth and RGB cameras mounted on the robot for fine-grained control. Since SAM2 is compute-intensive and hence unsuitable for onboard inference, we employ an on-device multi-stage perception pipeline for accurate object localization, combining GroundingDINO [82] for object detection at 0.2 Hz, MobileSAM [83] for generating precise object masks on the RGBD input at 0.2 Hz, and Cutie [84] for high-frequency tracking at 10 Hz. This approach maintains accurate object position information between slower

	Success Rate (%) ↑					Average Distance (%) ↑				
	Climb Up	Climb Down	Walk 30° pitch	Walk −30° pitch	Walk	Climb Up	Climb Down	Walk 30° pitch	Walk −30° pitch	Walk
Blind	0	0	100	100	100	11	10	100	100	100
No GRU	0	0	100	100	100	12	13	100	100	100
No Distill	0	0	0	0	0	0	0	0	0	0
No Waypoint	14	12	90	100	100	10	13	100	100	100
Ours	**96**	**90**	100	100	100	**92**	**84**	100	100	100
Oracle (Phase 1)	98	96	100	100	100	95	92	100	100	100

TABLE I: **Simulation Results.** We compare our learned controller with five baselines in simulation. Our performs the best in all tasks in both success rate and average distance reached. We find only small degradation in performance from the oracle policy using priviledged information in Phase 1. All metrics are averaged across 200 trials.

detection updates. From the tracked mask, we extract the (x, y, z) coordinates of the object's center in the robot's local frame. The grasping commands are then generated using proportional controllers: the linear velocity command is controlled based on the x-coordinate with K_p = 0.5, the angular velocity command is adjusted using the y-coordinate with K_p = 0.5, and the pitch command is computed from the z-coordinate with K_p = 1. The system triggers the grasping action when all coordinates are within a small threshold, indicating optimal positioning relative to the target object.

By integrating these components, our system achieves zero-shot generalization to new configurations and objects, enabling the quadrupedal robot to perform complex mobile manipulation tasks without environment-specific training or data collection. The use of pre-trained VLMs and efficient perception pipelines allows for robust performance across a wide range of scenarios, making our approach suitable for diverse indoor applications.

VI. EXPERIMENTS

A. Simulation Experiments

Baselines in Simulation. We compare our controller with several baselines including *Blind*, *No GRU*, *No Distill* and *No Waypoint*. We also include *Oracle (Phase 1)*, the policy trained in phase 1 using privileged information.

- **Blind**: a blind policy using only proprioception and no depth images as observations.
- **No GRU**: a MLP policy baseline. Instead of using a GRU, it uses only the depth image and proprioception at the current time step without any memory to predict actions.
- **No Distill**: an ablation training a deployable policy with GRU directly using PPO with our two-phase training process, so skipping the distillation stage.
- **No Waypoint**: removing the agile locomotion objective guided by waypoints. Directly train the policy in Phase 1 with a reward encouraging tracking sampled linear and angular velocity commands.
- **Oracle (Phase 1)**: The policy from the first training phase, which has access to privileged information only available in simulation, such as terrain scandots.

These baselines allow us to assess the impact of various components in our approach, including the importance of visual input, temporal memory, the two-phase training process, and the use of waypoints in guiding robot forward. Additionally, comparing against the Oracle provides insight into the performance gap between our deployable policy and one with access to privileged environmental information.

Simulation Results. Shown in Table I, the Blind and No GRU baselines exhibit poor performance, failing in most tasks except for the simple Walk task where they achieve 100% success. These baselines lack the necessary spatial awareness or memory mechanisms required for complex sequential navigation tasks involving climbing. Learning from vision directly increases the complexity of the training process, where the network can't learn from scratch properly. The No Waypoint baseline shows moderate success in Climb Up and Climb Down, but still struggles with the more challenging climbing tasks, highlighting the importance of on-the-fly velocity command generation for climbing. Without waypoints as guidance, the robot easily learns to walk pass the obstacle or turn around instead of trying to climb as a result of rewarding local velocity only. In contrast, our approach achieves consistently higher performance, with near-perfect scores in most tasks, especially Climb Up and Climb Down, and outperforms all baselines. We find only small degradation in performance from the oracle policy using priviledged information in Phase 1. Our approach shows that distilled policy can perform as well as the Oracle policy, suggesting the effectiveness of the two-phase training process. The overall results demonstrate the importance of integrating components such as depth information, memory, waypoint guidance and distillation.

B. Real-World Experiments

Baselines and Tasks in Real World. We compare our system deployed in the real world with several baselines. The baselines include *Go2 Default*, *Teleop*, and *No Tracking*:

- **Go2 Default**: the default controller built in with Go2. This controller does not use exteroception.
- **Teleop**: the commands are generated by an expert human operator through a remote controller, replacing VLMs.
- **No Tracking**: the commands are generated open-loop using the initial pose detection of the robot itself and the object of interest.

| | First-Attempt Success Rate (%) ↑ | | | | | | | | | | | Average Time (s) ↓ | | |
| | Bed + Toy | | | | Sofa + Bottle | | | | Ground + Ball | | | Bed + Toy | Sofa + Bottle | Ground + Ball |
	navigate + climb up	pick up	climb down	*Total*	navigate + climb up	pick up	climb down	*Total*	navigate	pick up	*Total*			
Go2 Default	0	0	0	0	0	0	0	0	80	0	0	-	-	-
No Tracking	60	0	0	0	50	0	0	0	40	0	0	–	-	-
Ours	90	78	86	**60**	80	88	86	**60**	100	70	**70**	**62**	**50**	**23**
Teleop	90	89	100	80	90	89	88	70	100	80	80	75	58	38

TABLE II: **Real-World Results.** We compare our system with three baselines including Go2 default controller instead of our learned controller, teleoperation using a remote controller instead of using VLMs, and ours without reactive tracking powered by SAM2. Ours outperform all baselines in average time to completion, and close to teleoperation in success rates. We measure the success rates and average time to completion across 10 trials per setting.

Illustrated in Figure 1, we select three objects and three environments that represent realistic real-world scenarios:
- **Bed + Toy**: The robot needs to fetch a stuffed toy on a bed. The task requires the robot to climb up a queen-sized bed with 40cm in height, pick up the stuffed toy on the bed, and climb down the bed. The stuffed toy is placed uniformly randomly on a 1m by 1m region on the bed. The robot is initially randomly placed in the bedroom.
- **Sofa + Bottle**: The robot needs to fetch an empty plastic water bottle on a sofa. The task requires the robot to climb up a sofa with a height of 44cm, pick up the bottle on the sofa, and climb down the sofa. The bottle is placed uniformly randomly on a 0.2m by 1m region on the sofa. The robot is initially randomly placed in the room.
- **Ground + Ball**: The robot needs to fetch a ball on the ground. The ball is placed uniformly randomly on a 3m by 3m region on the ground.

We test our system and the three baselines on all four tasks. We measure the success rates and average time to completion across 10 trials per setting. Qualitative results can be found on the project website.

Real-World Results. The real-world experiments, as summarized in Table II, demonstrate the effectiveness of our system compared to the three baselines. In the task involving navigating to a toy on a bed, our system achieved a 60% total first-attempt success rate, significantly outperforming the Go2 default controller and No Tracking baselines, both of which failed to complete the task. Go2 default controller fails to climb up high obstacles like beds and sofas, whereas No Tracking only generates an open-loop trajectory of commands and fails to compensate drifting in navigation and subsequent grasping. Our system's performance was close to that of teleoperation, with only a 20% gap in the first-attempt success rate. We find that though teleoperation can solve tasks perfectly given many attempts, the first-attempt success rates of teleoperation are only around 70-80% given an expert human operator. Similarly, in the task of fetching a bottle from a sofa with soft deformabile, our approach achieves a 60% success rate, close to teleportation. This tasks also demonstrates the robustness of our learned controller in walking on soft deformable surfaces. In the Ground + Ball task, which involves simpler navigation and grasping on flat terrain, our system achieving a 70% success rate, outperforming all baselines than teleoperation. In terms of average time to completion, our system consistently outperformed the baselines, completing tasks faster than both the Go2 Default and No Tracking methods. Notably, our system was also faster than teleoperation, particularly in the Ground + Ball task, where it completed the task in 23 seconds on average compared to teleoperation's 38 seconds. These results highlight the strength of our approach in achieving open-vocabulary object fetching in novel environments under a reasonable amount of time.

VII. CONCLUSION, LIMITATIONS & FUTURE DIRECTIONS

We presented Helpful DoggyBot, a quadrupedal robot system capable of zero-shot mobile manipulation in diverse indoor environments, integrating a 1-DoF gripper, learned whole-body control, and vision-language models. While our approach demonstrates progress, limitations include the gripper's restricted dexterity, reliance on ceiling-mounted cameras for navigation, and potential occlusion to the perception system. In future work, we will focus on enhancing manipulation capabilities without compromising agility, developing navigation strategies using only onboard sensors, and future improving agility to achieve cheerful pet behaviors [85]. Additional places for improvement include integrating multiple tasks into complex sequences, improving robustness in dynamic environments, incorporating online learning and human feedback, and exploring societal implications of advanced quadrupedal robots in domestic settings. By addressing these challenges, this research direction has the potential to revolutionize human-robot interaction and assistance in daily life.

REFERENCES

[1] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "RMA: Rapid Motor Adaptation for Legged Robots," in *RSS*, 2021.

[2] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science Robotics*, Oct. 2020.

[3] C. D. Bellicoso, K. Krämer, M. Stäuble, D. Sako, F. Jenelten, M. Bjelonic, and M. Hutter, "Alma-articulated locomotion and manipulation for a torque-controllable robot," in *2019 International conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 8477–8483.

[4] A. Agarwal, A. Kumar, J. Malik, and D. Pathak, "Legged locomotion in challenging terrains using egocentric vision," in *Conference on Robot Learning (CoRL)*, 2022.

[5] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Science Robotics*, Jan. 2022.

[6] I. Kato, "Development of wabot 1," *Biomechanism*, 1973.

[7] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of honda humanoid robot," in *Proceedings. 1998 IEEE international conference on robotics and automation*, 1998.

[8] M. Chignoli, D. Kim, E. Stanger-Jones, and S. Kim, "The mit humanoid robot: Design, motion planning, and control for acrobatic behaviors," in *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, 2021.

[9] G. Nelson, A. Saunders, N. Neville, B. Swilling, J. Bondaryk, D. Billings, C. Lee, R. Playter, and M. Raibert, "Petman: A humanoid robot for testing chemical protective clothing," *Journal of the Robotics Society of Japan*, 2012.

[10] O. Stasse, T. Flayols, R. Budhiraja, K. Giraud-Esclasse, J. Carpentier, J. Mirabel, A. Del Prete, P. Souères, N. Mansard, F. Lamiraux *et al.*, "Talos: A new humanoid research platform targeted for industrial applications," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017.

[11] N. A. Radford, P. Strawser, K. Hambuchen, J. S. Mehling, W. K. Verdeyen, A. S. Donnan, J. Holley, J. Sanchez, V. Nguyen, L. Bridgwater *et al.*, "Valkyrie: Nasa's first bipedal humanoid robot," *Journal of Field Robotics*, 2015.

[12] K. Harada, S. Kajita, H. Saito, M. Morisawa, F. Kanehiro, K. Fujiwara, K. Kaneko, and H. Hirukawa, "A humanoid robot carrying a heavy object," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005.

[13] H. Arisumi, J.-R. Chardonnet, A. Kheddar, and K. Yokoi, "Dynamic lifting motion of humanoid robots," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007.

[14] A. Settimi, D. Caporale, P. Kryczka, M. Ferrati, and L. Pallottino, "Motion primitive based random planning for loco-manipulation tasks," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016.

[15] P. Ferrari, M. Cognetti, and G. Oriolo, "Humanoid whole-body planning for loco-manipulation tasks," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[16] K. Harada, S. Kajita, F. Kanehiro, K. Fujiwara, K. Kaneko, K. Yokoi, and H. Hirukawa, "Real-time planning of humanoid robot's gait for force-controlled manipulation," *IEEE/ASME Transactions on Mechatronics*, 2007.

[17] S. Sato, Y. Kojio, K. Kojima, F. Sugai, Y. Kakiuchi, K. Okada, and M. Inaba, "Drop prevention control for humanoid robots carrying stacked boxes," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021.

[18] Z. Fu, Q. Zhao, Q. Wu, G. Wetzstein, and C. Finn, "Humanplus: Humanoid shadowing and imitation from humans," in *arXiv*, 2024.

[19] X. Cheng, J. Li, S. Yang, G. Yang, and X. Wang, "Open-television: Teleoperation with immersive active visual feedback," *arXiv preprint arXiv:2407.01512*, 2024.

[20] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. Kitani, C. Liu, and G. Shi, "Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning," *arXiv preprint arXiv:2406.08858*, 2024.

[21] J. Dao, H. Duan, and A. Fern, "Sim-to-real learning for humanoid box loco-manipulation," *arXiv preprint arXiv:2310.03191*, 2023.

[22] H.-W. Park, P. M. Wensing, S. Kim *et al.*, "Online planning for autonomous running jumps over obstacles in high-speed quadrupeds," *RSS*, 2015.

[23] Q. Nguyen, M. J. Powell, B. Katz, J. Di Carlo, and S. Kim, "Optimized jumping on the mit cheetah 3 robot," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019.

[24] C. Nguyen, L. Bao, and Q. Nguyen, "Continuous jumping for legged robots on stepping stones via trajectory optimization and model predictive control," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 93–99.

[25] C. Gehring, S. Coros, M. Hutter, C. D. Bellicoso, H. Heijnen, R. Diethelm, M. Bloesch, P. Fankhauser, J. Hwangbo, M. Hoepflinger *et al.*, "Practice makes perfect: An optimization-based approach to controlling agile motions for a quadruped robot," *IEEE Robotics & Automation Magazine*, 2016.

[26] L. Smith, J. C. Kew, X. B. Peng, S. Ha, J. Tan, and S. Levine, "Legged robots that keep on learning: Fine-tuning locomotion policies in the real world," in *ICRA*, 2022.

[27] Y. J. Ma, W. Liang, H. Wang, S. Wang, Y. Zhu, L. Fan, O. Bastani, and D. Jayaraman, "Dreureka: Language model guided sim-to-real transfer," in *Robotics: Science and Systems (RSS)*, 2024.

[28] G. B. Margolis, G. Yang, K. Paigwar, T. Chen, and P. Agrawal, "Rapid locomotion via reinforcement learning," *RSS*, 2022.

[29] G. Ji, J. Mun, H. Kim, and J. Hwangbo, "Concurrent training of a control policy and a state estimator for dynamic and robust legged locomotion," *IEEE Robotics and Automation Letters*, 2022.

[30] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, 2019.

[31] Y. Yang, X. Meng, W. Yu, T. Zhang, J. Tan, and B. Boots, "Continuous versatile jumping using learned action residuals," *L4DC*, 2023.

[32] X. Cheng, K. Shi, A. Agarwal, and D. Pathak, "Extreme parkour with legged robots," *arXiv preprint arXiv:2309.14341*, 2023.

[33] Z. Zhuang, Z. Fu, J. Wang, C. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao, "Robot parkour learning," in *Conference on Robot Learning (CoRL)*, 2023.

[34] N. Rudin, D. Hoeller, M. Bjelonic, and M. Hutter, "Advanced skills by learning locomotion and local navigation end-to-end," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022.

[35] S. Yang, Z. Zhang, Z. Fu, and Z. Manchester, "Cerberus: Low-drift visual-inertial-leg odometry for agile locomotion," *ICRA*, 2023.

[36] D. Kang, J. Cheng, M. Zamora, F. Zargarbashi, and S. Coros, "Rl+ model-based control: Using on-demand optimal control to learn versatile legged locomotion," *arXiv preprint arXiv:2305.17842*, 2023.

[37] H. Ferrolho, V. Ivan, W. Merkt, I. Havoutis, and S. Vijayakumar, "Roloma: Robust loco-manipulation for quadruped robots with arms," *Autonomous Robots*, 2023.

[38] Y. Ma, F. Farshidian, T. Miki, J. Lee, and M. Hutter, "Combining learning-based locomotion policy with model-based manipulation for legged mobile manipulators," *RA-L*, 2022.

[39] S. Zimmermann, R. Poranne, and S. Coros, "Go fetch!-dynamic grasps using boston dynamics spot with external robotic arm," in *ICRA*, 2021.

[40] M. Liu, Z. Chen, X. Cheng, Y. Ji, R. Yang, and X. Wang, "Visual whole-body control for legged loco-manipulation," *arXiv preprint arXiv:2403.16967*, 2024.

[41] T. Portela, G. B. Margolis, Y. Ji, and P. Agrawal, "Learning force control for legged manipulation," *arXiv*, 2023.

[42] H. Ha, Y. Gao, Z. Fu, J. Tan, and S. Song, "UMI on legs: Making manipulation policies mobile with manipulation-centric whole-body controllers," 2024.

[43] Z. Fu, X. Cheng, and D. Pathak, "Deep whole-body control: learning a unified policy for manipulation and locomotion," in *Conference on Robot Learning*, 2022.

[44] N. Yokoyama, A. W. Clegg, E. Undersander, S. Ha, D. Batra, and A. Rai, "Adaptive skill coordination for robotic mobile manipulation," *arXiv preprint arXiv:2304.00410*, 2023.

[45] G. Pan, Q. Ben, Z. Yuan, G. Jiang, Y. Ji, J. Pang, H. Liu, and H. Xu, "Roboduet: A framework affording mobile-manipulation and cross-embodiment," 2024.

[46] S. Jeon, M. Jung, S. Choi, B. Kim, and J. Hwangbo, "Learning whole-body manipulation for quadrupedal robot," *IEEE Robotics and Automation Letters*, 2023.

[47] C. Lin, X. Liu, Y. Yang, Y. Niu, W. Yu, T. Zhang, J. Tan, B. Boots, and D. Zhao, "Locoman: Advancing versatile quadrupedal dexterity with lightweight loco-manipulators," *arXiv preprint arXiv:2403.18197*, 2024.

[48] X. Cheng, A. Kumar, and D. Pathak, "Legs as manipulator: Pushing quadrupedal agility beyond locomotion," *ICRA*, 2023.

[49] Z. He, K. Lei, Y. Ze, K. Sreenath, Z. Li, and H. Xu, "Learning visual quadrupedal loco-manipulation from demonstrations," 2024.

[50] M. Xu, P. Huang, W. Yu, S. Liu, X. Zhang, Y. Niu, T. Zhang, F. Xia, J. Tan, and D. Zhao, "Creative robot tool use with large language models," 2023.

[51] P. Arm, M. Mittal, H. Kolvenbach, and M. Hutter, "Pedipulate: Enabling manipulation skills using a quadruped robot's leg," in *41st IEEE Conference on Robotics and Automation (ICRA 2024)*, 2024.

[52] O. X.-E. Collaboration, A. Padalkar, A. Pooley, A. Jain, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Singh, A. Brohan, A. Raffin, A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Ichter, C. Lu, C. Xu, C. Finn, C. Xu, C. Chi, C. Huang, C. Chan, C. Pan, C. Fu, C. Devin, D. Driess, D. Pathak, D. Shah, D. Büchler, D. Kalashnikov, D. Sadigh, E. Johns, F. Ceola, F. Xia, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Schiavi, H. Su, H.-S. Fang, H. Shi, H. B. Amor, H. I. Christensen, H. Furuta, H. Walke, H. Fang, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Kim, J. Schneider, J. Hsu, J. Bohg, J. Bingham, J. Wu, J. Wu, J. Luo, J. Gu, J. Tan, J. Oh, J. Malik, J. Tompson, J. Yang, J. J. Lim, J. Silvério, J. Han, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Zhang, K. Majd, K. Rana, K. Srinivasan, L. Y. Chen, L. Pinto, L. Tan, L. Ott, L. Lee, M. Tomizuka, M. Du, M. Ahn, M. Zhang, M. Ding, M. K. Srirama, M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Suenderhauf, N. D. Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, P. R. Sanketi, P. Wohlhart, P. Xu, P. Sermanet, P. Sundaresan, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Martín-Martín, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Moore, S. Bahl, S. Dass, S. Song, S. Xu, S. Haldar, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Dasari, S. Belkhale, T. Osa, T. Harada, T. Matsushima, T. Xiao, T. Yu, T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, V. Jain, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Wang, X. Zhu, X. Li, Y. Lu, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Xu, Y. Wang, Y. Bisk, Y. Cho, Y. Lee, Y. Cui, Y. hua Wu, Y. Tang, Y. Zhu, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Xu, and Z. J. Cui, "Open X-Embodiment: Robotic learning datasets and RT-X models," 2023.

[53] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei, "Voxposer: Composable 3d value maps for robotic manipulation with language models," *arXiv preprint arXiv:2307.05973*, 2023.

[54] D. Driess, F. Xia, M. S. M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke, K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, and P. Florence, "Palm-e: An embodied multimodal language model," in *arXiv preprint arXiv:2303.03378*, 2023.

[55] W. Yu, N. Gileadi, C. Fu, S. Kirmani, K.-H. Lee, M. Gonzalez Arenas, H.-T. Lewis Chiang, T. Erez, L. Hasenclever, J. Humplik, B. Ichter, T. Xiao, P. Xu, A. Zeng, T. Zhang, N. Heess, D. Sadigh, J. Tan, Y. Tassa, and F. Xia, "Language to rewards for robotic skill synthesis," *Arxiv preprint arXiv:2306.08647*, 2023.

[56] W. Yuan, J. Duan, V. Blukis, W. Pumacay, R. Krishna, A. Murali, A. Mousavian, and D. Fox, "Robopoint: A vision-language model for spatial affordance prediction for robotics," *arXiv preprint arXiv:2406.10721*, 2024.

[57] S. Nasiriany, F. Xia, W. Yu, T. Xiao, J. Liang, I. Dasgupta, A. Xie, D. Driess, A. Wahid, Z. Xu *et al.*, "Pivot: Iterative visual prompting elicits actionable knowledge for vlms," *arXiv preprint arXiv:2402.07872*, 2024.

[58] A. S. Chen, A. M. Lessing, A. Tang, G. Chada, L. Smith, S. Levine, and C. Finn, "Commonsense reasoning for legged robot adaptation with vision-language models," *arXiv preprint arXiv:2407.02666*, 2024.

[59] H. Huang, F. Lin, Y. Hu, S. Wang, and Y. Gao, "Copa: General robotic manipulation through spatial constraints of parts with foundation models," *arXiv preprint arXiv:2403.08248*, 2024.

[60] F. Liu, K. Fang, P. Abbeel, and S. Levine, "Moka: Open-vocabulary robotic manipulation through mark-based visual prompting," *arXiv preprint arXiv:2403.03174*, 2024.

[61] Y. Du, M. Yang, P. Florence, F. Xia, A. Wahid, B. Ichter, P. Sermanet, T. Yu, P. Abbeel, J. B. Tenenbaum *et al.*, "Video language planning," *arXiv preprint arXiv:2310.10625*, 2023.

[62] B. Chen, Z. Xu, S. Kirmani, B. Ichter, D. Sadigh, L. Guibas, and F. Xia, "Spatialvlm: Endowing vision-language models with spatial reasoning capabilities," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 455–14 465.

[63] J. Gao, B. Sarkar, F. Xia, T. Xiao, J. Wu, B. Ichter, A. Majumdar, and D. Sadigh, "Physically grounded vision-language models for robotic manipulation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 12 462–12 469.

[64] J. Duan, W. Yuan, W. Pumacay, Y. R. Wang, K. Ehsani, D. Fox, and R. Krishna, "Manipulate-anything: Automating real-world robots using vision-language models," *arXiv preprint arXiv:2406.18915*, 2024.

[65] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *arXiv preprint arXiv:2307.15818*, 2023.

[66] M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," in *Proceedings of the 5th Conference on Robot Learning (CoRL)*, 2021.

[67] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9493–9500.

[68] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, "Openvla: An open-source vision-language-action model," *arXiv preprint arXiv:2406.09246*, 2024.

[69] H. Zhen, X. Qiu, P. Chen, J. Yang, X. Yan, Y. Du, Y. Hong, and C. Gan, "3d-vla: A 3d vision-language-action generative world model," *arXiv preprint arXiv:2403.09631*, 2024.

[70] K. Black, M. Nakamoto, P. Atreya, H. Walke, C. Finn, A. Kumar, and S. Levine, "Zero-shot robotic manipulation with pretrained image-editing diffusion models," *arXiv preprint arXiv:2310.10639*, 2023.

[71] A. Stone, T. Xiao, Y. Lu, K. Gopalakrishnan, K.-H. Lee, Q. Vuong, P. Wohlhart, S. Kirmani, B. Zitkovich, F. Xia, C. Finn, and K. Hausman, "Open-world object manipulation using pre-trained vision-language models," in *arXiv preprint*, 2023.

[72] J. Gu, S. Kirmani, P. Wohlhart, Y. Lu, M. G. Arenas, K. Rao, W. Yu, C. Fu, K. Gopalakrishnan, Z. Xu *et al.*, "Rt-trajectory: Robotic task generalization via hindsight trajectory sketches," *arXiv preprint arXiv:2311.01977*, 2023.

[73] P. Sundaresan, Q. Vuong, J. Gu, P. Xu, T. Xiao, S. Kirmani, T. Yu, M. Stark, A. Jain, K. Hausman *et al.*, "Rt-sketch: Goal-conditioned imitation learning from hand-drawn sketches," *arXiv preprint arXiv:2403.02709*, 2024.

[74] S. Belkhale, T. Ding, T. Xiao, P. Sermanet, Q. Vuong, J. Tompson, Y. Chebotar, D. Dwibedi, and D. Sadigh, "Rt-h: Action hierarchies using language," in *https://arxiv.org/abs/2403.01823*, 2024.

[75] D. Shah, B. Osinski, B. Ichter, and S. Levine, "LM-nav: Robotic navigation with large pre-trained models of language, vision, and action," in *6th Annual Conference on Robot Learning*, 2022.

[76] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser, "Tidybot: Personalized robot assistance with large language models," *Autonomous Robots*, 2023.

[77] Y. Tang, W. Yu, J. Tan, H. Zen, A. Faust, and T. Harada, "Saytap: Language to quadrupedal locomotion," *arXiv preprint arXiv:2306.07580*, 2023.

[78] J. Chen, J. Frey, R. Zhou, T. Miki, G. Martius, and M. Hutter, "Identifying terrain physical parameters from vision–towards physical-parameter-aware locomotion and navigation," *arXiv preprint arXiv:2408.16567*, 2024.

[79] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[80] B. Xiao, H. Wu, W. Xu, X. Dai, H. Hu, Y. Lu, M. Zeng, C. Liu, and L. Yuan, "Florence-2: Advancing a unified representation for a variety of vision tasks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[81] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson *et al.*, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024.

[82] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu *et al.*, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," *arXiv preprint arXiv:2303.05499*, 2023.

[83] C. Zhang, D. Han, Y. Qiao, J. U. Kim, S.-H. Bae, S. Lee, and C. S. Hong, "Faster segment anything: Towards lightweight sam for mobile applications," *arXiv preprint arXiv:2306.14289*, 2023.

[84] H. K. Cheng, S. W. Oh, B. Price, J.-Y. Lee, and A. Schwing, "Putting the object back into video object segmentation," in *arXiv*, 2023.

[85] X. Duan, Z. Zhuang, H. Zhao, and S. Schwertfeger, "Playful doggybot: Learning agile and precise quadrupedal locomotion," in *arXiv*, 2024.

reward	expression	scale
tracking goal vel	$\min\left(\frac{\vec{v}\cdot\hat{\vec{t}}}{v_{cmd}+10^{-5}}, \frac{v_{cmd}}{v_{cmd}+10^{-5}}\right)$, where $\hat{\vec{t}} = \frac{\vec{t}}{||\vec{t}||+10^{-5}}$	1.5
tracking yaw vel	$\exp\left(-|\omega_z - \omega_{cmd}|\right)$	1.
tracking pitch	$\exp\left(-3|p_{cmd} - p|\right)$	1.5
lin vel z walking	v_z^2	-9.0
ang vel xy	$\sum \omega_{xy}^2$	-0.05
dof acc	$\sum \left(\frac{\dot{q}_{t+1} - \dot{q}_t}{\Delta t}\right)^2$	-2.5e-7
collision	$\sum \mathbf{1}\left(||f_{\text{contact}}|| > 0.1\right)$	-5.
action rate	$||\mathbf{a}_{t+1} - \mathbf{a}_t||$	-0.1
delta torques	$\sum (\tau_{t+1} - \tau_t)^2$	-1.0e-7
torques	$\sum \tau^2$	-0.00001
hip pos	$\sum (q_{\text{hip}} - q_{\text{hip, default}})^2$	-1
dof error	$\sum (q - q_{\text{default}})^2$	-0.2
feet stumble	$\mathbf{1}\left(||f_{\text{contact, xy}}|| > 4 \cdot |f_{\text{contact, z}}|\right)$	-5
feet edge	$\sum \mathbf{1}(\text{feet at edge})$	-1
feet drag	$\sum \left(\mathbf{1}(\text{contact}) \cdot ||v_{\text{xy}}^{\text{feet}}||\right)$	-0.1
energy	$||\tau \cdot \dot{q}||$	-1e-3

TABLE III: **Reward terms**

	value
num of envs	6144
num of vision envs	384
num of terrains	400
num of difficulty levels	10
stair height	[0, 0.65]
stair per env	[0, 6]
stair width	[0.8, 3]
stair length	[1.5, 2]
goal y range	[-0.1,0.1]
terrain noise	[0.02, 0.06]
curriculum up threshold	0.8*total length
curriculum down threshold	0.5*total length

TABLE IV: **Environment and terrain setup**

	value
friction	[0.2, 2]
push interval (s)	8
max push vel xy (m/s)	0.5
max push vel z (m/s)	0.5
added mass range (kg)	[0., 3.]
added com range (m)	[-0.2, 0.2]
motor strength range	[0.8, 1.2]
action delay(s)	[0 0.02]
vision delay(s)	0.1
vision position rand(m)	0.005
vision angle rand(degree)	[24,34]

TABLE V: **Domain randomization**