**Robotics: Science and Systems 2025**
**Los Angeles, CA, USA, June 21–25, 2025**

# Prompting with the Future: Open-World Model Predictive Control with Interactive Digital Twins

Chuanruo Ning  Kuan Fang†  Wei-Chiu Ma†
Cornell University
https://prompting-with-the-future.github.io/

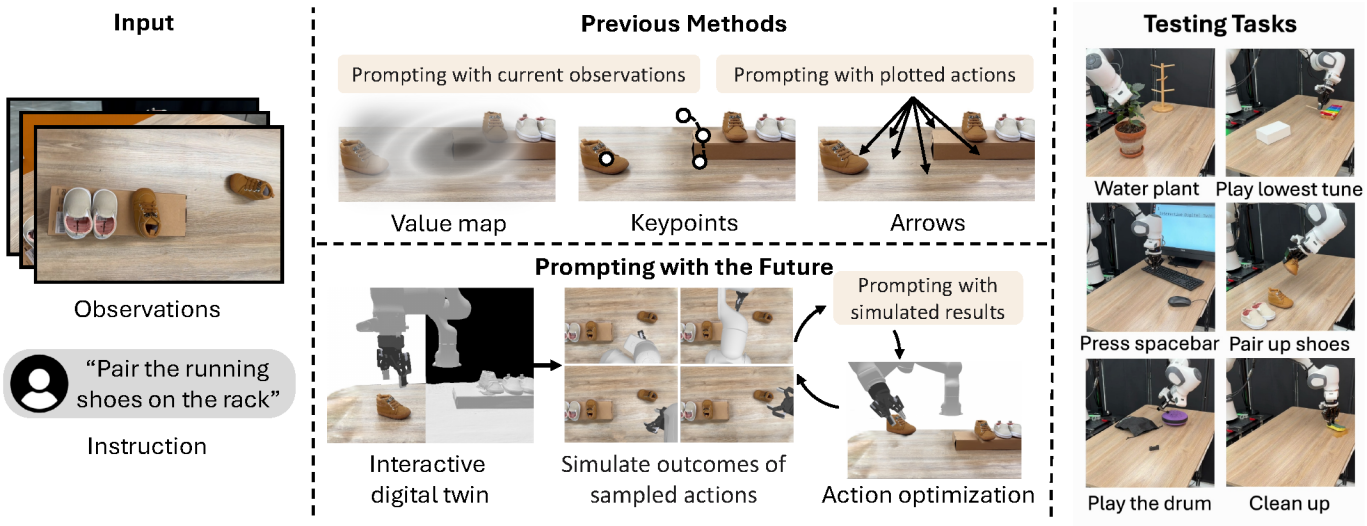


Fig. 1: **Prompting with the Future.** We enable VLM-driven motion planning conditioned on free-form instructions by prompting with simulated future outcomes generated from an interactive digital twin. In contrast, previous methods [20, 13, 31] prompt VLMs with current observations and hand-defined action primitives, demanding implicit physical reasoning that VLMs often struggle to perform reliably. Please use *Adobe Acrobat Reader* to view embedded video demonstrations.

***Abstract*—Open-world robotic manipulation requires robots to perform novel tasks described by free-form language in unstructured settings. While vision–language models (VLMs) offer strong high-level semantic reasoning, they lack the fine-grained physical insight needed for precise low-level control. To address this gap, we introduce Prompting with the Future (PWTF), a model-predictive control framework that augments VLM-based policies with explicit physics modeling. PWTF builds an interactive digital twin of the workspace from a quick handheld video scan, enabling prediction of future states under candidate action sequences. Instead of asking the VLM to predict actions or results by reasoning dynamics, the framework simulates diverse possible outcomes, renders them as visual prompts with adaptively selected camera viewpoints that expose the most informative physical context. A sampling-based planner then selects the action sequence that the VLM rates as best aligned with the task objective. We validate PWTF on eight real-world manipulation tasks involving contact-rich interaction, object reorientation, and clutter removal, demonstrating significantly higher success rates than state-of-the-art VLM-based control methods. Through ablation studies, we further analyze the performance and demonstrate that explicitly modeling physics, while still leveraging VLM semantic strengths, is essential for robust manipulation.**

† Equal advising.

## I. Introduction

General-purpose robots must perform novel, human-specified tasks in unstructured, highly varied settings. Recent vision–language models (VLMs) have made impressive strides in high-level semantic reasoning and zero-shot visual question answering [2, 11, 20]. Yet their understanding of fine-grained physical interactions is typically cannot support he precise, low-level control required in real-world manipulation. Take the seemingly simple job of tidying a cluttered shoe rack (Figure 1). The robot need to, detect and identify each shoe, infer the desired final arrangement, and plan and execute gripper motions that achieve accurate translations and rotations while avoiding collisions. Achieving open-world manipulation thus hinges on seamlessly fusing semantic reasoning with the fine-grained, physics-aware control that real-world tasks demand.

Recent works have investigated various ways to employ VLMs for robotic control. Fine-tuned on massive demonstration trajectories through imitation learning, VLMs can serve as the backbone of policies to ground language instructions in visual observations [7, 25, 34]. However, robotics datasets re-

main orders of magnitude smaller than the question-answering datasets used to train VLMs, limiting the generalization of learned policies to novel environments and tasks. Alternatively, an increasing number of works propose to employz VLMs in zero-shot settings either with the observed images [20] or sampled actions overlaid on the images as visual prompts [13, 31]. While these approaches show promise in simple tasks, they require VLM to infer future interactions, thus struggling with more complex scenarios involving intricate contact, dynamics, and motion due to insufficient physical understanding.

In this work, we propose a fundamentally different approach to address this challenge by decoupling semantic understanding from physical reasoning. In contrast to prior works which require VLM to implicitly reason dynamics, we employ interactive digital twins of the real-world environment to complement the VLM's capabilities. Through physical simulation, the digital twin explicitly models the dynamics of the manipulation scene, providing reliable predictions of result states under diverse actions. By rendering the images of the simulated outcomes, we enable the VLM to interface directly with the predicted future observations before they occur, which allows the VLM to bypass physical reasoning and instead focus on the semantic understanding and evaluation that it excels in.

To this end, we present Prompting with the Future (PWTF), an approach that solves open-world manipulation in a model-predictive control (MPC) manner by integrating a VLM and an interactive digital twin. As shown in Figure 1, given a video scan of the real-world environment, our method first builds an interactive digital twin with a controllable robot and interactable objects to enable physical simulation of possible outcomes of sampled actions. In a sampling-based planning paradigm, our method generates candidate actions, simulates their outcomes within the digital twin, and utilizes the VLM to evaluate the resulting future states. Rather than requiring the VLM to reason about physics directly, we render RGB images from the digital twin to provide observations of predicted outcomes as visual prompts to the VLM. Unlike prior work that relies on fixed camera viewpoints, we adaptively adjust camera poses to optimize the visual inputs for the VLM, facilitating more effective spatial reasoning. We design a visual prompting mechanism that enables the VLM to assess the feasibility and desirability of different action outcomes, enabling planning of the optimal actions to solve the task, Thereby, we enhance physical reasoning in VLM-driven robotic control without requiring additional robot data collection, VLM fine-tuning, or any in-context examples.

We evaluate PWTF in eight real-world manipulation tasks specified by natural language instructions. Our approach demonstrates to be able to effectively solve these tasks with success rates superior to baseline methods that use VLM for robotic control. Additionally, we conduct thorough ablation studies and failure analysis to investigate the key factors contributing to our framework's performance and robustness.

## II. Related Work

**VLM-based robotic control.** Recent advances in VLMs have opened pathways toward open-world robot manipulation without any robotic demonstration by leveraging their strengths in semantic understanding and common-sense reasoning [10, 2, 15, 44, 35].

However, VLMs, primarily trained on visual question-answering tasks, often struggle to reason about the physical effects of interactions necessary for motion planning. To address this limitation, prior works have explored diverse intermediate representations, such as reward functions, 2D keypoints, action vectors, and affordance [20, 13, 31, 21], as outputs for VLMs. While these approaches show promising results, these settings diverge from VLMs' training distribution. Another line of work involves carefully designed chain-of-thought processes [11, 30, 49], which decompose tasks into smaller, predefined steps. Although effective for specific scenarios, this approach struggles to generalize to tasks that cannot be easily divided into such reasoning procedures and is limited by the inherent weaknesses of VLMs.

In contrast, we aim to make open-world manipulation an in-distribution task for VLMs by leveraging digital twins to simulate future results. This allows VLMs to focus on evaluating possible future observations, aligning seamlessly with their strengths in description and evaluation.

**Model Predictive Control.** Model Predictive Control (MPC) is a widely adopted optimization-based control strategy that has proven highly effective in robotics [14, 12, 16, 32, 29]. MPC fundamentally consists of two key components: a dynamic model capable of predicting future system states given a sequence of actions, and an optimization process that determines the optimal action sequence by minimizing a predefined cost function. Recent MPC methods for open-world motion planning usually use images from a video generation model to predict the future results given different actions [50, 19, 51], leveraging the prior from the large-scale pre-training. However, video generation introduces artifacts and fails to generalize to complex scenarios that are far from the training distribution. Besides, generated video cannot guarantee to be precisely conditioned on the low-level actions.

In this paper, we introduce physically grounded dynamic models of by building digital twins of the real-world environments. Besides, compared with the traditional method of designing a cost function for each task, we employ VLMs for flexible and comprehensive evaluation over the results, enabling open-world manipulation on a wide range of tasks.

**3D scene reconstruction for robot control.** 3D reconstruction methods, including neural radiance fields (NeRFs) [3, 4, 28], neural implicit surfaces [45, 46, 33], and Gaussian Splatting [18, 23], have revolutionized 3D scene modeling by enabling photorealistic rendering and dense geometry reconstruction. These methods provide a lot of potential for application in robot motion planning [47, 41, 24, 26, 37]. Previous work leveraging scene reconstruction for motion planning either tries to distill 2D features generated by foundational models

into 3D for spatial perception or enhance the demonstration to enable few-shot learning. However, these works usually assume the scene to be static and do not handle interactions over the reconstructed representation.

Building digital twins from the real-world scenes, as a special branch of 3D scene reconstruction, enables a lot of applications in manipulation (i.e., real-to-sim-to-real method). However, previous methods usually regard the built digital twin as a virtual playground for collecting manipulation data [36, 48, 22] or training robot policy through reinforcement learning [43, 35, 8, 42, 5], which are then transferred to the real world for manipulation.

In this work, we aim to enable dynamic and interactive modeling between the robot and its environment. To achieve this, we combine Gaussian splatting, known for its efficient and realistic rendering, with mesh for accurate physical modeling. Our hybrid representation leverages the strengths of both approaches, introducing interactive capabilities for robot control applications. This enables dynamic, physically grounded interactions in real-world scenarios, bridging the gap between static scene reconstruction and interactive manipulation. We leverage digital twin as a world model that can provide results of actions that have not happen yet in the real world. By combining with VLM as a critic, we enable a zero-shot setting for real-to-sim-to-real manipulation.

## III. Problem Formulation

Our goal is to enable robots to perform manipulation tasks involving unseen objects and diverse goals. We focus on complex scenarios that involve intricate contact, dynamics, and motion. These setups require robots to possess a thorough physical and semantic understanding of the environment. We do not assume access to task-specific training data, in-context examples, or hard-coded motion primitives as used in prior work [20, 27, 13, 25]. We consider a table-top setting with one robotic arm. The framework's input consists of a natural language instruction $l$ specifying the task, and an RGB video scan $v$ of the scene. The output is an action sequence $\{a\}_{t=0}^{T-1}$ for achieving the task goal. Each action $a_t \in \mathbb{R}^7$ is defined as the 6-DoF gripper pose and the finger status (open or closed).

Central to our framework is a pre-trained vision–language model (VLM). The model processes an ordered sequence of interleaved text and RGB images and returns a textual response. We employ the VLM for various purposes with designed prompts.

## IV. Method

We propose to tackle open-world motion planning by complementing VLM's semantic reasoning ability with explicit modeling of dynamics through digital twins. Our framework builds two key components: 1) We introduce a pipeline to automatically build interactive digital twins to support accurate modeling of diverse physical interactions and photorealistic rendering of the simulation outcomes. Section IV-A. 2) We formulate open-world manipulation as a model predictive control problem by prompting the VLM with futures provided by the digital twin, enabling adaptive observation and action optimization. Section IV-C.

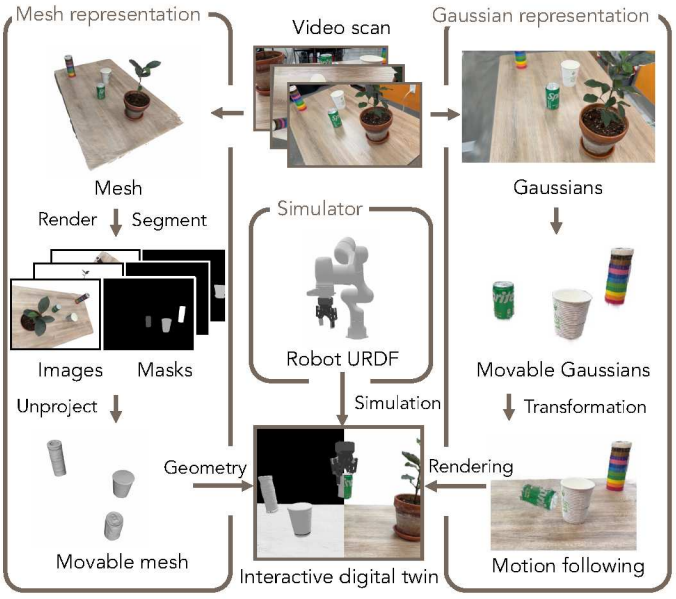


Fig. 2: **Construction of interactive digital twins.** Starting from a video scan of the environment, we construct an interactive digital twin that combines mesh-based simulation and Gaussian-based rendering. The resulting twin enables photorealistic rendering and accurate simulation of object dynamics conditioned on robot actions.

### A. Construction of Interactive Digital Twins

To complement the semantic reasoning of the VLM with grounded physical predictions, we construct interactive digital twins that enable both accurate physical simulation and photorealistic rendering of future outcomes. As shown in Figure 2, our construction pipeline consists of two key stages: (1) reconstructing scenes with accurate geometry and visual appearance, and (2) making the scene interactable to support robot actions and environment dynamics.

Unlike prior work, which often focuses solely on static reconstruction [40, 24], our method produces dynamic, action-conditioned digital twins by combining mesh-based physical modeling with efficient Gaussian splatting for rendering. This hybrid design supports fine-grained simulation of physical interactions while maintaining high-quality visual fidelity.

**Hybrid reconstruction:** Given a video scan of the real-world environment, we first reconstruct a hybrid scene representation that achieves both geometric accuracy and photorealistic appearance (Figure 2). We apply 2D Gaussian Splatting [18] to create a Gaussian representation $G$ supervised by the extracted RGB frames. To recover precise scene geometry, we convert $G$ into a mesh representation $M$ through TSDF volume integration, preserving fine details necessary for physical simulation.

**Interactive objects:** To model dynamics for robotic manipulation, the reconstructed scene must model movable objects. Starting from $M$, we render multi-view images of the scene

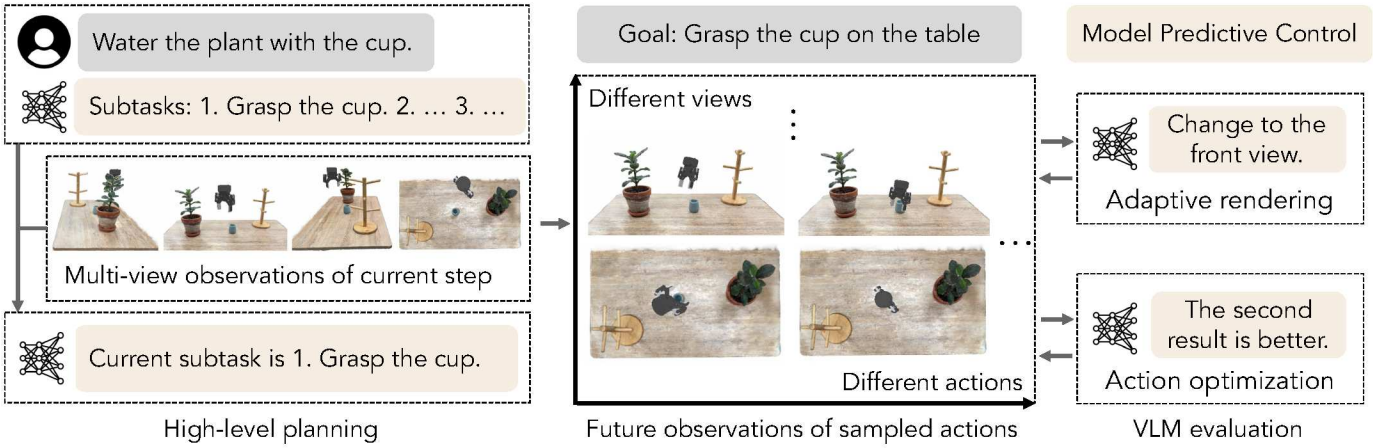


Fig. 3: **Model Predictive Control through Simulation-Informed Prompting.** Given a free-form instruction, our framework first performs high-level planning by generating structured subtasks from multi-view observations. At each step, the interactive digital twin simulates future states for candidate actions and render the outcomes multiple viewpoints. The VLM adaptively selects the most informative view for renderding and evaluates the predicted outcomes for sampling-based motion planning.

and use Molmo [9], a vision-language model (VLM), to identify key movable objects based on task instructions $l$. Given Molmo's 2D keypoints, we employ SAM2 [38] to track segmentation masks across views. These 2D masks are then projected back to 3D, labeling mesh vertices and segmenting out movable object meshes within $M$.

**Realistic rendering:** In addition to movable meshes for geometric information, we also need high visual fidelity and ensure realistic rendering in accordance with object movement. To achieve realistic and responsive rendering, we enable the Gaussians $G$ to dynamically follow the movement of their associated object meshes. Specifically, we first associate each Gaussian point with the nearest movable mesh based on the Euclidean distance between their centers. During simulation, the anchored Gaussians inherit the translation and rotation of their corresponding meshes, ensuring that the rendered appearance remains consistent with object motions.

**Physical simulation:** Finally, we integrate a physics simulator $S$ [17] equipped with the robot's URDF $U$ to model dynamics under interaction. The simulator computes physically plausible state transitions when applying diverse candidate actions, ensuring the digital twin can predict action-conditioned outcomes with high fidelity.

Through this construction pipeline, we obtain an interactive digital twin where the mesh representation provides physical structure, the Gaussian splatting enables efficient and realistic rendering, and the simulator governs the dynamics. Additional implementation details are provided in the supplementary material.

### *B. Simulation and Rendering for Visual Prompting*

We simulate and render future outcomes in the digital twin to generate visual prompts for the vision-language model (VLM). Given a robot action $a \in \mathbb{R}^7$, the simulator $S$ predicts the resulting transformation of the robot and scene objects within the mesh representation $M$, denoted as $\mathcal{T} = S(M, a)$. We apply this transformation $\mathcal{T}$ to the Gaussian scene representation $G$, obtaining an updated state $G(\mathcal{T})$ that encodes the physical consequences of the action.

While VLMs operate purely on 2D images and inherently lack strong spatial reasoning, we facilitate 3D understanding by synthesizing multi-view observations of each predicted future. Given the updated state $G(\mathcal{T})$, we render background scene images from a set of camera configurations $C$, producing $I_g = \text{Render}(G(\mathcal{T}), C)$, which depict the environment without the robot. Separately, we render the robot executing action $a$ using the simulator and its URDF model, yielding $I_r = \text{Render}(S(U, a))$. The background and robot images are composited based on depth information to generate the final multi-view observations $I = I_i \in \mathbb{R}^{W \times H \times 3}$. These synthesized observations allow the VLM to perceive the manipulation scene from diverse viewpoints, enhancing its ability to reason about future states and improving the accuracy of action selection.

### *C. Motion Planning via Simulation-Informed Prompting*

We formulate the control problem as a model predictive control (MPC) framework, where the vision-language model (VLM) serves exclusively as an evaluator of predicted future states. Our approach decomposes tasks into subgoals, adaptively selects viewpoints to facilitate VLM reasoning, and optimizes actions through sampling-based planning.

**Subgoal Decomposition.** We first leverage the semantic reasoning capabilities of the VLM to decompose complex tasks into structured subgoals. Given initial multi-view observations $I_0$ and a task instruction $l$, the VLM generates a set of subtasks $\tau = \tau_1, \tau_2, \ldots$, each specifying an intermediate objective. At each planning step $t$, the VLM is prompted with the current observations $I_t$ and the subtask set $\tau$, and selects the active subgoal $\tau_i$ corresponding to the agent's current progress. This

| Tasks | Instruction | Success Criteria |
|---|---|---|
| Water plant | Water the plant with the cup | The cup is over the plant pot with tilting action |
| Play drum | Hit the drum with the drum stick | The head of the drum stick contacts the drum |
| Clean up | Wipe the tea with the sponge | The sponge makes contact with the spilled tea |
| Press spacebar | Press the space on the keyboard | The spacebar is pressed |
| Cucumber basket | Put the green cucumber into the basket | The cucumber is in the basket |
| Pair up shoes | Pair up the shoes | Corresponding shoes are next to each other in parallel |
| Unplug charger | Unplug the charger | The charger is no longer plugged into the power strip |
| Lowest tune | Play the lowest pitch with the drum stick | The head of the drum stick contacts the lowest key |

TABLE I: **Definition of tasks**. We list the text instructions for prompting VLM and the success criteria of each task.

decomposition localizes the optimization objective, improving sample efficiency and enhancing planning robustness.

**Adaptive Viewpoint Selection.** Since robot actions are defined in $SE(3)$ space but the VLM operates solely on 2D visual inputs, spatial reasoning critically depends on viewpoint selection. At each planning step, we render observations of the updated scene $G(\mathcal{T})$ under a set of candidate camera configurations $C$. Conditioned on $\tau_i$ and $I_t$, the VLM selects the viewpoint $C_t$ that provides the most informative observation for distinguishing between action outcomes. This adaptive rendering procedure strengthens the VLM's ability to reason about geometric and physical variations essential for task success (see Figure 3).

**Sampling-Based Planning.** With the active subgoal $\tau_i$ and selected viewpoint $C_t$ determined, the framework proceeds to low-level action generation. We employ the Cross-Entropy Method (CEM) [39] for structured action optimization, by leveraging the digital twins to explicitly model the dynamics and the VLM to evaluate the predicted outcomes. To select the action, candidate actions $a_k$ are initially sampled from a multivariate Gaussian distribution and simulated within the digital twin to obtain their corresponding future observations $o_{t,k}$. The simulated outcomes $o_{t,k}$ are partitioned into $m$ groups, each individually prompted to the VLM along with the subgoal $\tau_i$, and the VLM selects the outcome that best advances the subgoal. The corresponding elite actions are then used to update the sampling distribution. This sampling and refinement process is repeated for three iterations, after which the mean of the final distribution is taken as the optimized action $a_t$.

## V. Experiments

To validate the effectiveness of our framework, in this section, we design eight real-world manipulation tasks that require 6 DoF control, semantic understanding, and diverse manipulation skills. We compare our approach against prior works on open-world manipulation that leverage VLMs or train on large-scale robotic data. Additionally, we conduct ablation experiments to evaluate the contribution of each component to the overall performance.

### A. Experimental setup

**Tasks.** As shown in Table I, we introduce eight manipulation tasks that require intricate understanding of the physical world and diverse manipulation skills: `water plant`, `play drum`, `press spacebar`, `pair up shoes`, `cucumber basket`, `lowest tune`, `unplug charger`, and `clean up`. For each task, we construct five manipulation scenes, featuring randomized object layouts and different distractors. Please see the supplementary material for more details about task design.

**Baselines.** We compare our model against several state-of-the-art methods. **VoxPoser*** [20] leverages a VLM to predict 3D value map for motion optimization. We enhance it by providing ground-truth segmented object point clouds from our digital twin, significantly improving its perception accuracy. **MOKA** [13] chooses the 2D keypoints as intermediate representations for VLM to predict, which are then converted into actions based on the depth information from a depth camera. **OpenVLA** [25] is a 7B-parameter open-source vision-language-action model fine-tuned from a VLM using 970k real-world robot demonstrations [34]. $\pi_0$ [6], a state-of-the-art vision-language-action model trained on diverse robot demonstrations. For both OpenVLA and $\pi_0$, we report their performances under a zero-shot setting and after task-specific fine-tuning on 20 expert demonstrations for each task.

**Metrics.** We use the success rate as the evaluation metric. A task is considered a failure if the robot causes irreversible results or if the maximum step budget or time limit is reached. The task is successful if the success criteria are met. Please see the supplementary material for more details.

**Implementation details.** We adopt GPT-4o [1] for both our method and the baselines. During inference, we render four camera views and allow VLM to select observations from these perspectives. For the CEM optimization, we use 3 iterations with 90 samples per iteration. The planning policies are rolled out twice per scene to consider the randomness in VLM planning, resulting in 10 trials per task in total. Please see the supplementary material for more details about task and baseline design.

### B. Quantitative results

Table II compares the success rates of our method against those of the baselines. Since Voxposer and MOKA rely on open-vocabulary detectors to detect objects before manipulation, they fail when the perception system cannot recognize specific object parts, such as the spacebar on a keyboard

| Methods | Water plant | Play drum | Press spacebar | Pair up shoes | Cucumber basket | Lowest tune | Unplug charger | Clean up |
|---|---|---|---|---|---|---|---|---|
| Voxposer* [20] | **6/10** | 2/10 | 0/10 | 2/10 | 8/10 | 0/10 | 4/10 | 0/10 |
| MOKA [13] | 4/10 | 5/10 | 0/10 | 2/10 | 5/10 | 0/10 | 0/10 | 5/10 |
| OpenVLA [25] | 1/10 | 0/10 | 1/10 | 0/10 | 4/10 | 0/10 | 0/10 | 0/10 |
| $\pi_0$ [6] | 0/10 | 0/10 | 0/10 | 1/10 | 3/10 | 0/10 | 2/10 | 0/10 |
| OpenVLA-finetuned | 0/10 | 0/10 | 0/10 | 0/10 | 5/10 | 0/10 | 0/10 | 0/10 |
| $\pi_0$-finetuned | 0/10 | 0/10 | 0/10 | 2/10 | 2/10 | 0/10 | 3/10 | 2/10 |
| Ours | 5/10 | **6/10** | **5/10** | **6/10** | **9/10** | **4/10** | **7/10** | **5/10** |

TABLE II: **Comparison against baselines**. PWTF better leverages the reasoning ability of VLM and improves the performance on most of the tasks. Besides, our image-based evaluation on results provides more flexibility than value map or key points, enabling challenging tasks that are hard to be parametrized by previous representations (e.g., play the lowest tune).

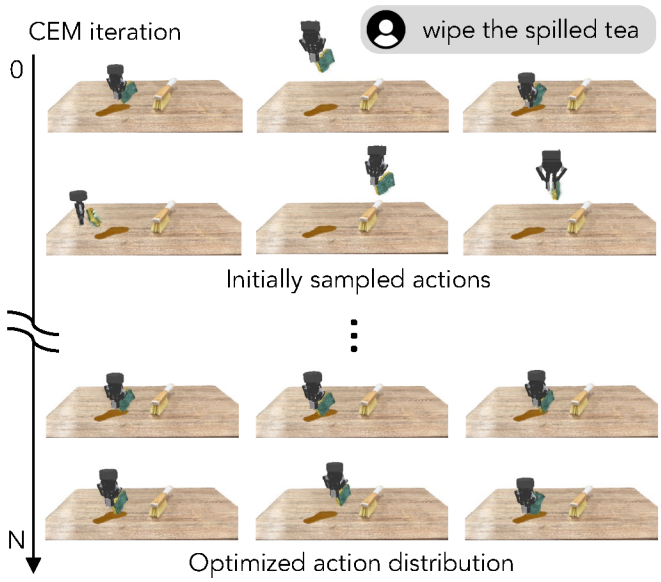


Fig. 4: **Example on action optimization.** We show the action optimization results of one planning step in subtask "wipe the spilled tea". Our digital twin could simulate diverse results with accurate motion and collision of the sponge in initial sampling and VLM could effectively optimize the action distribution to move the sponge towards the tea.

or the lowest key on a xylophone. In contrast, our method directly leverages the VLM to comprehend and reason about simulated future states, eliminating the need for a perception module and resulting in greater robustness. As for OpenVLA and $\pi_0$, while they can perform zero-shot on simple tasks due to their training on large-scale robotic datasets, their generalization is limited by the coverage of the training data, making them less effective for complex tasks. Even with task-specific demonstrations, the trained methods still struggle to generalize to unseen layouts. We hypothesize that more data is needed for trained methods. In contrast, our method benefits from the commonsense reasoning capabilities of the VLM, enabling broader task coverage and adaptability. We particularly excel in tasks requiring precise gripper pose alignment, as our approach allows for simulation-based "rehearsal" before execution.

### C. Qualitative results

We visualize the action optimization process for a single planning step in the "clean up" task in Figure 4. Initially, the digital twin simulates a diverse set of actions with precise dynamics (*e.g.*, object rotating due to grasping, object dropping due to collisions, etc.) and photorealistic rendering. Based on the simulation results and the task instruction, the VLM selects the results that better align with the target objective. An action distribution is fit to the selected elite actions, from which new actions are resampled for further simulation in the digital twin. This iterative process leads to an optimized action distribution that aligns more closely with the goal of wiping the spilled tea using the sponge.

Figure 5 shows some examples of the rollout trajectories planned by our framework in both digital twins and the real world. By mirroring possible interactions in the simulated world, our framework provides a flexible and effective way for VLMs to guide the motion of the robot on diverse tasks in an open-world environment. The digital twin and the real world are aligned based on planning steps. We highlight key planning steps where VLM chooses to change the observation view to better assess the results, showing the benefits of our adaptive rendering design. Please see the supplementary material for more visualization and videos.

### D. Ablation study

To assess the contribution of each component in our framework, we begin with the full system and systematically remove each component in turn. In the "w/o views" setting, we fix the camera to a static top-down perspective instead of allowing the VLM to select a view for each planning step. For "w/o subtasks," we use the user instruction directly as the goal for each planning step rather than first generating subtasks. In the "w/o CEM" setting, we simply take the mean value of the selected actions without optimizing the action distribution or resampling.

As shown in Table III, while performance varies across different tasks due to their diverse requirements, our full method achieves the best results in most of the tasks. Multi-view observations, enabled by our adaptive rendering, enhance spatial perception, particularly in tasks requiring precise control (e.g., Press spacebar, Play the drum). Dividing tasks into subtasks


Fig. 5: **Example trajectories.** Example trajectories planned by our framework in both digital twin and real world (aligned). We highlight some key steps where VLM chooses to adaptively change the rendering view for better result evaluation (e.g., aligning the gripper from different perspectives for grasping, pressing, placing or hitting).

| Methods | Water plant | Play drum | Press spacebar | Pair up shoes | Cucumber basket | Lowest tune | Unplug charger | Clean up |
|---|---|---|---|---|---|---|---|---|
| w/o Views | 1/10 | 0/10 | 0/10 | 3/10 | 8/10 | 0/10 | 3/10 | 1/10 |
| w/o Subtasks | 5/10 | 0/10 | **6/10** | 3/10 | 8/10 | 2/10 | 7/10 | 0/10 |
| w/o CEM | 0/10 | 2/10 | 2/10 | 2/10 | 1/10 | 0/10 | 4/10 | 0/10 |
| Ours Full Method | **5/10** | **6/10** | 5/10 | **6/10** | **9/10** | **4/10** | **7/10** | **5/10** |

TABLE III: **Ablation study.** We validate the effectiveness of our components. Multi-view observations are essential for tasks that are sensitive to perspectives. Subtask division improves the performance on tasks that require multi-stage planning. Most importantly, CEM significantly increases the sampling efficiency, facilitates effective planning.

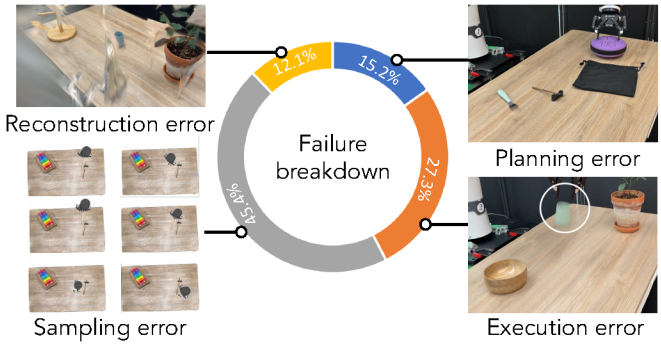


Fig. 6: **Failure analysis.** Our main failure cases can be divided into four categories. We show the percentage and provide an example for each failure type.

allows the VLM to focus on more concrete evaluation criteria for each state, which is beneficial for tasks involving multiple stages, such as playing the drum and cleaning up. Lastly, the CEM process significantly improves sampling efficiency, producing action distributions that better align with the goal, which in general contributes the most to our model predictive control framework.

### *E. Failure breakdown*

The failure cases can be categorized into four groups:

- **Reconstruction error:** The quality of our digital twin depends on the accuracy of camera pose estimation and 3D reconstruction. Noisy in poses or optimization often result in visual artifacts, which can reduce the effectiveness of the VLM.
- **Planning error:** When subtasks are not properly defined or the model fails to recognize the current stage, the robot may execute actions incorrectly. For example, the robot may attempt to move the gripper directly to the drum without first picking up the drumstick.
- **Execution error:** In the real world, the gripper may misalign during grasping or drop objects during movement due to insufficient grip or friction.
- **Sampling error:** This is the primary source of failure in our framework. Due to the inherent randomness of action sampling and errors in VLM reasoning, the system may struggle to sample actions to achieve the goal within a limited step budget.

We show their percentage and some examples in Figure 6. Please see the supplementary material for a more detailed failure breakdown.

## VI. Limitations

This work explores the potential of using vision-language models (VLMs) for open-world robotic manipulation without relying on robotic data or in-context examples. While the approach shows promise, several limitations remain.

A primary limitation lies in computational efficiency. Reconstructing 3D representations from real-world scenes takes about 20 minutes per scene. Inference through prompting VLMs also introduces latency due to token generation. These factors limit the system's applicability to real-time settings. Progress in 3D reconstruction methods and the use of smaller, open-source VLMs may help alleviate these issues. Please see the supplementary material about the discussion and improvement on efficiency.

Second, the performance is bounded by the capability of current VLMs. Inaccuracies in perception can lead to suboptimal evaluations of candidate actions, introducing variance into planning. Nevertheless, the system is designed to benefit directly from advances in VLMs, and we expect robustness and efficiency to improve as models evolve.

## VII. Conclusion

We propose a framework that combines vision-language models (VLMs) with interactive digital twins for open-world robotic manipulation. The system builds hybrid representations from real-world scans and enables physical interactions, supporting explicit modeling of dynamics and photorealistic rendering for action selection with VLMs. By using MPC with adaptive rendering, the method leverages VLMs' high-level semantic understanding while grounding low-level action optimization in physical simulation. Experiments across a range of tasks show that the framework outperforms prior VLM- and vision-language-action model (VLA-based) approaches, demonstrating a viable path toward closing the gap between semantic reasoning and robotic control.

## References


[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv:2303.08774*, 2023.

[2] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv:2204.01691*, 2022.

[3] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *ICCV*, 2021.

[4] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, 2022.

[5] Cristian Camilo Beltran-Hernandez, Nicolas Erbetti, and Masashi Hamaya. Sliceit!: Simulation-based reinforcement learning for compliant robotic food slicing. In *ICRA Workshop*, 2024.

[6] Kevin Black et al. $\pi_0$: A vision-language-action flow model for general robot control. *arXiv:2410.24164*, 2024.

[7] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv:2307.15818*, 2023.

[8] Tianyuan Dai, Josiah Wong, Yunfan Jiang, Chen Wang, Cem Gokmen, Ruohan Zhang, Jiajun Wu, and Li Fei-Fei. Automated creation of digital cousins for robust policy learning. *arXiv:2410.07408*, 2024.

[9] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Jen Dumas, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv:2409.17146*, 2024.

[10] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv:2303.03378*, 2023.

[11] Jiafei Duan, Wentao Yuan, Wilbert Pumacay, Yi Ru Wang, Kiana Ehsani, Dieter Fox, and Ranjay Krishna. Manipulate-anything: Automating real-world robots using vision-language models. *arXiv:2406.18915*, 2024.

[12] Frederik Ebert, Chelsea Finn, Sudeep Dasari, Annie Xie, Alex Lee, and Sergey Levine. Visual foresight: Model-based deep reinforcement learning for vision-based robotic control. *arXiv:1812.00568*, 2018.

[13] Kuan Fang, Fangchen Liu, Pieter Abbeel, and Sergey Levine. Moka: Open-vocabulary robotic manipulation through mark-based visual prompting. In *RSS*, 2024.

[14] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *ICRA*, 2017.

[15] Jensen Gao, Bidipta Sarkar, Fei Xia, Ted Xiao, Jiajun Wu, Brian Ichter, Anirudha Majumdar, and Dorsa Sadigh. Physically grounded vision-language models for robotic manipulation. In *ICRA*, 2024.

[16] Ruben Grandia, Farbod Farshidian, René Ranftl, and Marco Hutter. Feedback mpc for torque-controlled legged robots. In *IROS*, 2019.

[17] Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, Xiaodi Yuan, Pengwei Xie, Zhiao Huang, Rui Chen, and Hao Su. Maniskill2: A unified benchmark for generalizable manipulation skills. In *ICLR*, 2023.

[18] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH*, 2024.

[19] 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 *CVPR*, 2023.

[20] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. In *CoRL*, 2023.

[21] Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. In *CoRL*, 2024.

[22] Zhenyu Jiang, Yuqi Xie, Kevin Lin, Zhenjia Xu, Weikang Wan, Ajay Mandlekar, Linxi Fan, and Yuke Zhu. Dexmimicgen: Automated data generation for bimanual dexterous manipulation via imitation learning. *arXiv:2410.24185*, 2024.

[23] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM TOG*, 2023.

[24] Justin Kerr, Chung Min Kim, Mingxuan Wu, Brent Yi, Qianqian Wang, Ken Goldberg, and Angjoo Kanazawa. Robot see robot do: Imitating articulated object manipulation with monocular 4d reconstruction. In *CoRL*, 2024.

[25] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al.

Openvla: An open-source vision-language-action model. *arXiv:2406.09246*, 2024.

[26] Yunzhu Li, Shuang Li, Vincent Sitzmann, Pulkit Agrawal, and Antonio Torralba. 3d neural scene representations for visuomotor control. In *CoRL*, 2022.

[27] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *ICRA*, 2023.

[28] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[29] Maria Vittoria Minniti, Ruben Grandia, Kevin Fäh, Farbod Farshidian, and Marco Hutter. Model predictive robot-environment interaction control for mobile manipulation tasks. In *ICRA*, 2021.

[30] Yao Mu, Qinglong Zhang, Mengkang Hu, Wenhai Wang, Mingyu Ding, Jun Jin, Bin Wang, Jifeng Dai, Yu Qiao, and Ping Luo. Embodiedgpt: Vision-language pre-training via embodied chain of thought. In *NeurIPS*, 2024.

[31] Soroush Nasiriany, Fei Xia, Wenhao Yu, Ted Xiao, Jacky Liang, Ishita Dasgupta, Annie Xie, Danny Driess, Ayzaan Wahid, Zhuo Xu, et al. Pivot: Iterative visual prompting elicits actionable knowledge for vlms. *arXiv:2402.07872*, 2024.

[32] Julian Nubert, Johannes Köhler, Vincent Berenz, Frank Allgöwer, and Sebastian Trimpe. Safe and fast tracking on a robot manipulator: Robust mpc and neural network control. *RA-L*, 2020.

[33] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *ICCV*, 2021.

[34] Abby O'Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv:2310.08864*, 2023.

[35] Shivansh Patel, Xinchen Yin, Wenlong Huang, Shubham Garg, Hooshang Nayyeri, Li Fei-Fei, Svetlana Lazebnik, and Yunzhu Li. A real-to-sim-to-real approach to robotic manipulation with vlm-generated iterative keypoint rewards. In *CoRL Workshop*, 2024.

[36] Weikun Peng, Jun Lv, Yuwei Zeng, Haonan Chen, Siheng Zhao, Jichen Sun, Cewu Lu, and Lin Shao. Tiebot: Learning to knot a tie from visual demonstration through a real-to-sim-to-real approach. *arXiv:2407.03245*, 2024.

[37] Mohammad Nomaan Qureshi, Sparsh Garg, Francisco Yandun, David Held, George Kantor, and Abhisesh Silwal. Splatsim: Zero-shot sim2real transfer of rgb manipulation policies using gaussian splatting. *arXiv:2409.10161*, 2024.

[38] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv:2408.00714*, 2024.

[39] Reuven Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and computing in applied probability*, 1999.

[40] William Shen, Ge Yang, Alan Yu, Jansen Wong, Leslie Pack Kaelbling, and Phillip Isola. Distilled feature fields enable few-shot language-guided manipulation. In *CoRL*, 2023.

[41] William Shen, Ge Yang, Alan Yu, Jansen Wong, Leslie Pack Kaelbling, and Phillip Isola. Distilled feature fields enable few-shot language-guided manipulation. In *CoRL*, 2023.

[42] Marcel Torne, Arhan Jain, Jiayi Yuan, Vidaaranya Macha, Lars Ankile, Anthony Simeonov, Pulkit Agrawal, and Abhishek Gupta. Robot learning with super-linear scaling. *arXiv:2412.01770*, 2024.

[43] Marcel Torne, Anthony Simeonov, Zechu Li, April Chan, Tao Chen, Abhishek Gupta, and Pulkit Agrawal. Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation. *arXiv:2403.03949*, 2024.

[44] Beichen Wang, Juexiao Zhang, Shuwen Dong, Irving Fang, and Chen Feng. Vlm see, robot do: Human demo video to robot action plan via vision language model. *arXiv:2410.08792*, 2024.

[45] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv:2106.10689*, 2021.

[46] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *ICCV*, 2023.

[47] Yixuan Wang, Mingtong Zhang, Zhuoran Li, Tarik Kelestemur, Katherine Rose Driggs-Campbell, Jiajun Wu, Li Fei-Fei, and Yunzhu Li. D $^3$ fields: Dynamic 3d descriptor fields for zero-shot generalizable rearrangement. In *CoRL*, 2024.

[48] Yuxuan Wu, Lei Pan, Wenhua Wu, Guangming Wang, Yanzi Miao, and Hesheng Wang. Rl-gsbridge: 3d gaussian splatting based real2sim2real method for robotic manipulation learning. *arXiv:2409.20291*, 2024.

[49] Michał Zawalski, William Chen, Karl Pertsch, Oier Mees, Chelsea Finn, and Sergey Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv:2407.08693*, 2024.

[50] Wentao Zhao, Jiaming Chen, Ziyu Meng, Donghui Mao, Ran Song, and Wei Zhang. Vlmpc: Vision-language model predictive control for robotic manipulation. *arXiv:2407.09829*, 2024.

[51] Xinxin Zhao, Wenzhe Cai, Likun Tang, and Teng Wang. Imaginenav: Prompting vision-language models as embodied navigator through scene imagination.

*arXiv:2410.09874*, 2024.