World4Omni: A Zero-Shot Framework from Image Generation World Model to Robotic Manipulation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Improving data efficiency and generalization in robotic manipulation remains a core challenge. We propose a novel framework that leverages a pre-trained multimodal image-generation model as a world model to guide policy learning. Exploiting its rich visual-semantic representations and strong generalization across diverse scenes, the model generates open-ended future state predictions that inform downstream manipulation. Coupled with zero-shot low-level control modules, our approach enables general-purpose robotic manipulation without task-specific training. Experiments in both simulation and real-world environments demonstrate that our method achieves effective performance across a wide range of manipulation tasks with no additional data collection or fine-tuning.

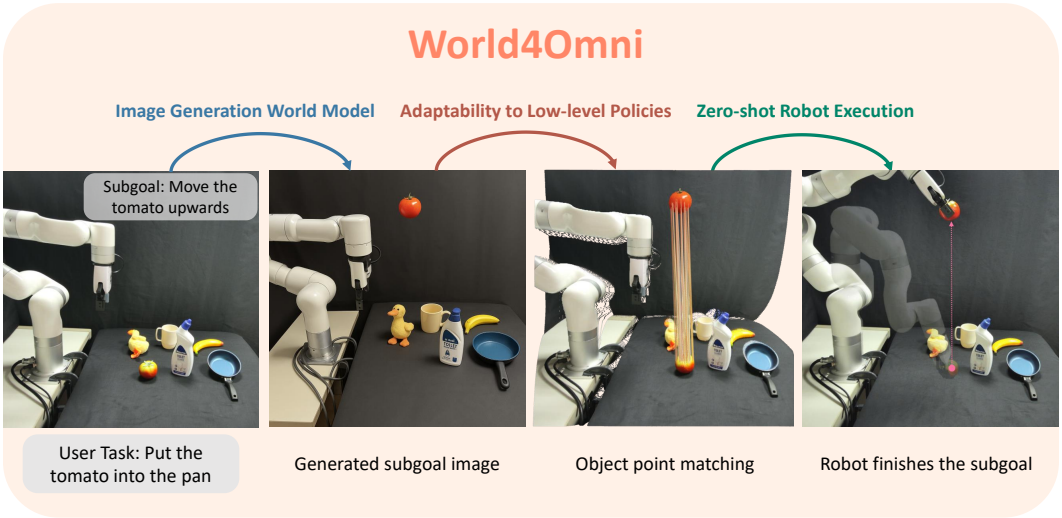


Figure 1: **Overview of the World4Omni framework.** We propose World4Omni, which leverages a pretrained multimodal image-generation model as a world model to guide low-level policy. Task instructions are decomposed into subtasks, each of which is fed into the world model along with the current scene image to generate a subgoal image depicting the scene after completing the current subtask. Predicted future images can be transformed into point clouds, enabling the high-level world model to adapt across different low-level policies. Object point matching validates the plausibility of predicted future images and enables their translation into concrete robot actions. Finally, a low-level policy is used to move the object from its initial position to its target position.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

1 Introduction

General-purpose embodied intelligence has long been a central aspiration in AI and robotics research [1–3], aiming to develop versatile robotic agents capable of handling diverse real-world tasks. Despite recent advancements, generalization remains a critical challenge. To perform manipulation tasks, robots must perceive environments, interpret complex instructions, and execute appropriate actions. However, variability in environmental conditions, task instructions, object properties, and robot embodiments significantly impacts robotic performance and poses considerable difficulties for generalization [4–6]. Many existing manipulation methods exhibit strong performance when operating in scenarios similar to their training environments, yet they frequently fail in unseen contexts [4, 7, 8]. Addressing these limitations typically involves two primary pathways: (1) increasing the volume and diversity of training data, and (2) developing techniques that enhance data efficiency.

Foundation models, pretrained on massive datasets, have significantly enhanced generalization in robotic tasks [9–11]. One paradigm involves training end-to-end models that directly map visual observations and language instructions to low-level actions [12–15]. However, collecting robot action data remains costly and time-consuming [10]. Consequently, even the largest robotics datasets [16] are dwarfed by Internet-scale text and image corpora [17], limiting these methods' capacity to generalize to novel tasks and scenarios [12–15]. An alternative paradigm adopts a hierarchical structure, leveraging Large Language Models (LLMs) and Vision-Language Models (VLMs), pretrained on extensive textual and visual data, to perform high-level planning and prediction before interfacing with low-level action modules [18–21]. Although LLMs and VLMs improve high-level generalization, their text-based outputs restrict flexibility when integrating with low-level action models. Initial works relied on predefined low-level skill libraries, limiting generalization to unseen tasks [22–24]. Subsequent approaches introduced intermediate representations, yet their low-level policies still depend on additional action-labeled data, constraining overall generalizability [18, 19, 25].

The ability of pre-trained foundation models to generate images has recently attracted widespread attention [26–28]. We found multimodal large-scale models trained on extensive web text–image data exhibit strong generalization across diverse scenarios, suggesting their suitability as a world model for robotic manipulation. Prior studies have shown that world models can substantially enhance data efficiency, thereby alleviating the generalization gaps caused by data scarcity in robotics [29–31]. Other methods employ video-generation models as world models [25, 32–35]; yet, generating future videos requires far greater temporal consistency than generating future images, and current large pre-trained video-generation models fail to achieve zero-shot generalization in robotic manipulation tasks. As a result, the vast majority of these approaches still demand additional task-specific training.

In this work, we employ a pre-trained foundation model as a world model to generate images depicting future object states. To mitigate inconsistencies in image outputs, we introduce a Vision-Language Model (VLM) as a Reflection Agent, which evaluates and refines these generated images. Additionally, we propose a Task Planner Agent that decomposes tasks into sequential subtasks, enhancing reasoning ability for long-range tasks. These predicted images can subsequently be transformed into point clouds via single-view depth estimation techniques [36]. As a result, our framework supports diverse input modalities for low-level modules—including current and predicted RGB images, point clouds, and structured representations derived from them (e.g., keypoints or object transformations). We evaluate our approach on representative manipulation tasks, assessing its zero-shot generalization both within hierarchical methods and against alternative paradigms.

Overall, our main contributions are as follows.

- We introduce a novel framework, **World4Omni**, capable of zero-shot, cross-embodiment generalization across diverse robotic manipulation tasks without any additional training.
- We employ a pre-trained large-scale multimodal image-generation model as a world model, incorporating an agent-based collaborative reflection process to iteratively refine imagined future scenes, thereby generating more plausible and consistent subgoal images.
- Our framework supports plug-and-play integration of low-level modules designed for different input modalities, showcasing its versatility and strong adaptability.
- We demonstrate the zero-shot generalization and cross-embodiment capabilities of our framework by applying it to diverse robotic manipulation tasks in simulation and the real world, achieving favorable results across the evaluations.

2 Related Work

2.1 World Models for Robotic Manipulation

Early work on world models for robotic manipulation primarily focused on learning visual dynamics directly from raw pixel observations to predict future frames [37, 38]. Subsequently, latent-space world models were introduced to encode the underlying physical dynamics compactly. For instance, dreamer and its variants [29, 30, 39, 31] learn internal latent-state representations and optimize robot behavior by simulating or "imagining" future trajectories. These latent-space models enhance data efficiency by augmenting limited real-world data with imagined experiences. Recent studies have also explored video-generation models as world models [32, 25, 33, 33, 34]. Although promising, such models require high temporal consistency, and existing pre-trained, large-scale video-generation models often fail to generalize effectively to novel scenarios [25, 32–35], thus limiting their practical application in diverse robotic manipulation tasks [40, 35]. In this work, we use pre-trained, large-scale image-generation models as world models. By predicting only essential subgoal images at key frames, our method avoids the temporal consistency issues faced by video-generation approaches and achieves robust generalization across various robotic manipulation scenarios.

2.2 Reflection in Foundation Models

Reflection mechanisms, which enable generative models to iteratively critique and refine their outputs, have recently attracted growing attention as a promising method for enhancing robotic manipulation capabilities. In generative modeling, recent studies demonstrate that incorporating self-feedback or iterative critiques substantially improves the quality and coherence of generated outputs [41–43]. Notable examples include CritiqueLLM [44] and Idea2Img [45], which showcase how reflective feedback loops facilitate progressive refinement and correction of initial predictions. Extending these reflective approaches into robotics, several recent frameworks [46, 47] integrate self-reflection into robotic task planning and action execution, enabling robotic agents to dynamically identify and correct errors, thereby progressively enhancing their performance during tasks. Moreover, additional studies have advanced this concept by incorporating multimodal reflection mechanisms, effectively bridging high-level cognitive reasoning with low-level motor control adjustments. This multimodal integration significantly improves robot robustness and adaptability, enabling robots to better manage uncertainties and effectively generalize across diverse manipulation scenarios and real-world conditions [48–50].

2.3 Foundation Models Paradigms for Robotic Manipulation

Recent advances in foundation models have significantly influenced robotics, especially in robotic manipulation tasks, by leveraging LLMs and VLMs for high-level planning and decision-making [51–55]. Current research can be broadly categorized into three main paradigms. The first paradigm employs foundation models to guide robotic execution by linking high-level instructions to predefined low-level skills. Approaches such as SayCan [22], PALM-E [24], and Code as Policies [23] utilize LLM outputs combined with skill libraries or executable code generation to bridge high-level planning and robotic actions. However, these methods often struggle to generalize predefined skill sets. The second paradigm introduces intermediate visual representations or subgoals to enhance generalization and task execution. Methods such as ReKep [21], SuSIE [18], 3D-VLA [19], and Gen2Act [20] use foundation models to generate intermediate goals, like keypoints, subgoal images, or demonstration videos, that guide robotic policies, thus improving adaptability to novel scenarios. The third paradigm directly integrates foundation models into end-to-end frameworks that map visual and linguistic inputs directly to continuous low-level robot actions. Notable methods include RT-1 [12], RT-2 [13], OpenVLA [14], and RDT-1B [15], which bypass hierarchical structuring to provide more flexible and generalized manipulation capabilities through joint training on large-scale demonstration data and multimodal tasks.

3 Method

Our framework is illustrated in Figure 2. In this section, we provide a detailed explanation of problem formulation (Sec. 3.1), agent collaboration (Sec. 3.2), reflective world model (Sec. 3.3), and low-level policy (Sec. 3.4).

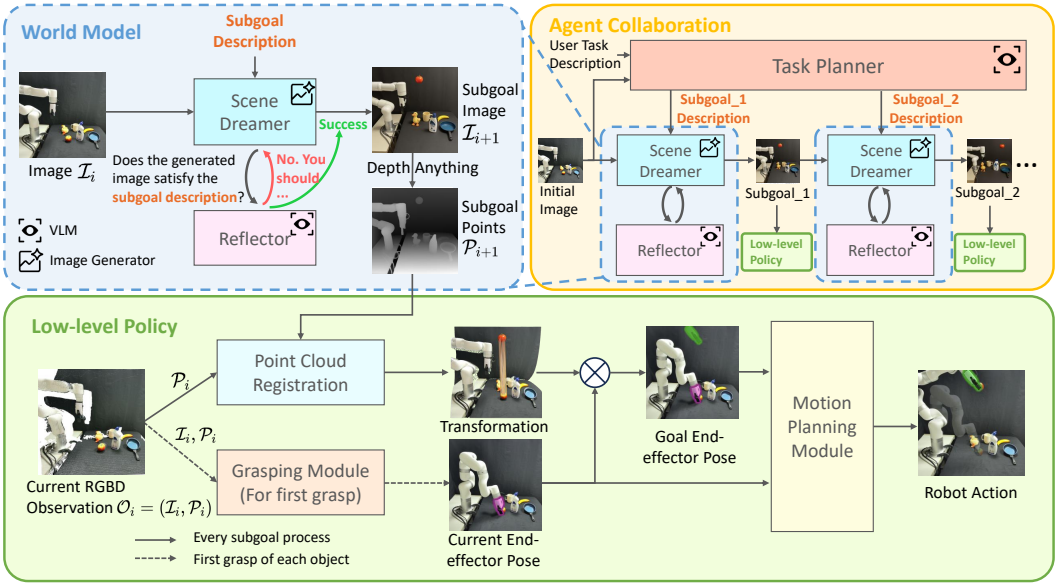


Figure 2: **An instantiation of our framework.** Agent Collaboration involves the Task Planner, Scene Dreamer, and Reflector agents working together **(yellow)**. The Scene Dreamer and Reflector together form the Reflective World Model, which produces subgoal images and corresponding point clouds **(blue)**. The Low-level Policy consumes the current and goal observations and outputs the robot actions **(green)**.

3.1 Problem Formulation

Given a single-view RGB image $\mathcal{I} \in \mathbb{R}^{H\times W\times 3}$, a point cloud $\mathcal{P} \in \mathbb{R}^{N\times 3}$, and a natural language task description $\mathcal{L}$, the objective is to generate an action sequence $\{\boldsymbol{a}\}_i$ that completes the manipulation task described in $\mathcal{L}$.

The **Task Planner Agent** takes an RGB image $\mathcal{I}$ and a task description $\mathcal{L}$ as input and outputs a sequence of subtask descriptions $\{\mathcal{L}\}_i$.

The **World Model** receives an RGB image $\mathcal{I}$ and a subtask description $\mathcal{L}_i$ to produce a subgoal image $\mathcal{I}' \in \mathbb{R}^{H\times W\times 3}$ and a subgoal depth map $\mathcal{D}' \in \mathbb{R}^{H\times W\times 1}$. The resulting future point cloud $\mathcal{P}'$ is obtained by back-projecting $\mathcal{D}'$.

The **Low-level Policy** is provided with the current observation $\mathcal{O} = (\mathcal{I}, \mathcal{P})$ and the future observation $\mathcal{O}' = (\mathcal{I}', \mathcal{P}')$, and outputs a sequence of actions $\{\boldsymbol{a}\}_i$ to drive the system from $\mathcal{O}$ to $\mathcal{O}'$.

Notably, the low-level policy may use any non-empty subset of the modalities from the current observation $\mathcal{O} = (\mathcal{I}, \mathcal{P})$ (i.e., $\mathcal{I}$, $\mathcal{P}$, or both) and any non-empty subset from the target observation $\mathcal{O}' = (\mathcal{I}', \mathcal{P}')$ (i.e., $\mathcal{I}'$, $\mathcal{P}'$, or both) to produce the action sequence $\{\boldsymbol{a}\}_i$. This shows our framework is compatible with a variety of different low-level policy settings.

3.2 Agent Collaboration

Given the user's task description $\mathcal{L}$ and a scene image $\mathcal{I}$, we employ a VLM (GPT-o4-mini-high) as the Task Planner Agent to decompose the task into a sequence of subtask descriptions $\{\mathcal{L}_i\}$ in text form. For example, given the initial scene image and the instruction "Put the tomato in the pan", the Task Planner Agent might decompose the task into the following subtasks: (1) $\mathcal{L}_0$: "Move the tomato vertically upward"; (2) $\mathcal{L}_1$: "Move the tomato horizontally to the right, positioning it above the pan"; (3) $\mathcal{L}_2$: "Move the tomato downward into the pan".

At the start, we input the initial image $\mathcal{I}_0$ (specifically, $\mathcal{I}_0 = \mathcal{I}$) along with its corresponding subtask description $\mathcal{L}_0$ into the Reflective World Model. The Reflective World Model consists of a Scene Dreamer Agent and a Reflector Agent (detailed in Sec. 3.3), and produces a predicted future scene

image $\mathcal{I}_1$. We refer to each such predicted future scene image as a subgoal image. By the same process, for each input image $\mathcal{I}_i$ together with its subtask description $\mathcal{L}_i$, the Reflective World Model outputs the subgoal image $\mathcal{I}_{i+1}$. As illustrated in Figure 2, the initial desktop image and the first subtask instruction $\mathcal{L}_0$ are input to the World Model, which outputs a subgoal image showing the tomato moving upward. This subgoal image, together with the next subtask instruction $\mathcal{L}_1$, is then fed into the World Model to produce the next subgoal image of the tomato moving to the right, and this process continues until the task is complete.

Subgoal images serve two roles: (1) they can be used directly by a low-level policy or converted into another modality for a low-level policy (see Sec. 3.3); and (2) they provide the input for the next Scene Dreamer Agent to generate the subsequent subgoal image. By using subgoal images rather than ground-truth scene images, we avoid issues where objects of interest are occluded by the robot arm or gripper, allowing the Reflective World Model to output more consistent images.

3.3 Reflective World Model

The Reflective World Model consists of a Scene Dreamer Agent, which leverages a large-scale pre-trained image-generation model, and a Reflector Agent, which is built on a VLM. The Scene Dreamer Agent receives the scene image $\mathcal{I}_i$ and the subtask description $\mathcal{L}_i$ from the Task Planner, then employs GPT-4o to generate an image of the future scene $\mathcal{I}_{i+1}$. The Reflector Agent employs GPT-o4-mini-high to understand both the generated image and the subtask semantics, evaluating whether the output of the Scene Dreamer Agent is consistent. If the image passes this check, the Reflector emits a success signal; if not, it produces a revised prompt to steer the Scene Dreamer toward a more accurate generation.

Taking the future scene generation in Fig. 2 as an example, if the output of Scene Dreamer Agent shows incorrect movement of the target object or disregards the surrounding context, the Reflector Agent issues a revised prompt $\mathcal{L}'_i$ and submits it along with the current scene image $\mathcal{I}_i$ to the Scene Dreamer Agent. Then the Scene Dreamer Agent generates a new image $\mathcal{I}_{i+1}$. This reflective loop mitigates hallucinations and goal inconsistencies in image outputs of the Scene Dreamer Agent. Finally, the resulting scene images $\mathcal{I}_{i+1}$ can be converted into depth maps $\mathcal{P}_{i+1}$ using Depth-Anything [36], allowing flexible support for different inputs from low-level models.

In robotic manipulation tasks, we concentrate on the object of interest; accordingly, our generated images are object-centric. Although image-generation models may introduce inconsistencies in background elements, we disregard these artifacts and concentrate solely on the target object. To ensure a clean, focused representation for the low-level policy, we use Grounded SAM [56] to segment out the object of interest.

3.4 Low-level Policy

In our framework, the high-level model provides the low-level policy with both the current observation $\mathcal{O}_i = (\mathcal{I}_i, \mathcal{P}_i)$ and the target observation $\mathcal{O}_{i+1} = (\mathcal{I}_{i+1}, \mathcal{P}_{i+1})$, forming the complete set of available inputs. The low-level policy may then select any non-empty subset of these observations as its input. Our framework is adaptive to different low-level policies—any policy that can operate on these inputs can be seamlessly integrated. Specifically, as an instantiation of the low-level policy of our framework, we employ the Grasping+Planning approach, which comprises three core components: Point Cloud Registration, a Grasping Module, and a Motion Planning Module.

Point Cloud Registration. We adopt GeoAware-SC [57] as our principal framework for point-cloud registration, exploiting its geometry-aware semantic correspondence module to establish dense, per-pixel matches between the initial and subgoal images. Concurrently, we apply Depth-Anything [36] to both views to infer high-resolution depth maps. To delineate object extents within both scenes, we integrate Grounded SAM [56], generating robust segmentation masks for all salient entities. Finally, given the fused semantic correspondences and their associated metric depths, we first lift the depth into 3d point clouds and employ the Umeyama algorithm [58] to estimate the optimal rigid transformation that aligns the initial and subgoal configurations.

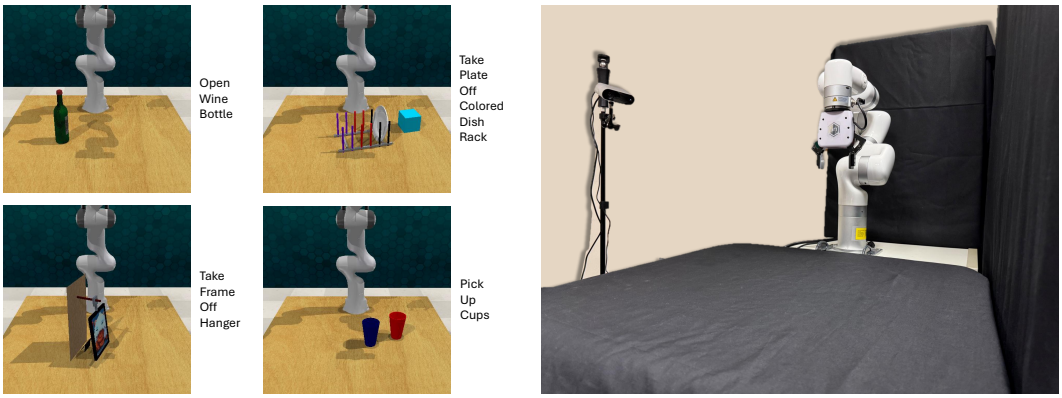
Grasping Module. Our low-level policy framework employs pre-trained GraspNet [59] as its grasping module, which takes a point cloud input and generates top-K grasp poses in the camera frame in an end-to-end manner. For implementation, we first use Grounded SAM [56] to generate

masks and segment target object points from the original point cloud. We then filter the GraspNet-generated poses, keeping only those within a distance threshold of these target points, and select the one with the highest score.

Motion Planning Module. Our motion planning strategies differ between simulation and real-world execution. In the simulated environment, we leverage a sample-based motion planning module built in the simulator, which can interpolate trajectories from the initial to the target pose. **[TODO]**

4 Experiment

4.1 Experimental Setup



(a) Simulation Setup for Four Tasks (b) Real world experiments setup

Figure 3: **Experiment setup in simulation and real world.**

Simulation. Simulation experiments were carried out in RLBench [60] using a Franka Emika Panda 7-DoF arm and several RGB-D cameras. The robot arm is fixed to the tabletop, and the objects are randomly placed on the table. At the beginning of every trial, no objects are held.

Some typical robot manipulation tasks in RLBench are selected. In Table 1 and Table 2, Tasks 1–4 correspond to OpenWineBottle, TakePlateOffColoredDishRack, TakeFrameOffHanger, and PickUpCups, respectively(Figure 3a). To ensure consistency with the baselines, we assign each RLBench task a random seed from 1 to 5, corresponding to five different arrangements for the task. For each seed, we conduct 10 trials, resulting in a total of 50 runs for each task.

All simulations ran on a single NVIDIA A100 GPU with 40 GB of memory. Executing our low-level policy for one task requires approximately 3 minutes, and generating subgoal images and point clouds with the Reflective World Model takes about ten minutes.

Real World. Our real-world experimental setup, as depicted in Figure 3b, comprises a 7-DoF UFACTORY X-ARM 7 and an Orbbec Femto Bolt RGB-D camera. At the start of each trial, the robot does not grasp any object. We evaluate our pipeline on two real-world robotic manipulation tasks. 1)Move the tomato into the pan. 2)Take the plate off the rack.

Our real-world experiments were conducted on a single NVIDIA RTX 4090 with 24 GB of memory.

4.2 Baselines

Representative methods from different paradigms are selected as our baselines.

- For cross-paradigm comparison, we adopt the end-to-end OpenVLA [14] approach as a baseline.
- For intra-paradigm comparisons, our baselines are SuSIE [18], which leverages a pre-trained image-editing model.

We also compare various high-level world models and low-level policies within our framework.

- For world models, we compare GPT-4o [61], DALL·E 3 [62], and Gemini 2.5 Pro [63] for their image-generation performance as well as Sora [64, 65] for video generation.
- For zero-shot low-level policies, we compare our Grasping+Planning method with the Octo [66] foundation model that conditions on both initial and goal images.

Neither the baselines nor our method undergoes any additional training in RLBench or on real-world setups; all comparisons are conducted under zero-shot settings.

Table 1: Cross-paradigm and intra-paradigm experiment results in simulation

Method	Task1	Task2	Task3	Task4	Average Success Rate
OpenVLA	0 / 50	0 / 50	0 / 50	0 / 50	0%
SUSIE	0 / 50	0 / 50	0 / 50	0 / 50	0%
Ours	10 / 50	20 / 50	10 / 50	30 / 50	35%

Table 2: Zero-shot low-level policy evaluation results in simulation

Method	Task1	Task2	Task3	Task4	Average Success Rate
Octo	0 / 50	0 / 50	0 / 50	0 / 50	0%
Grasping+Planning	10 / 50	20 / 50	10 / 50	30 / 50	35%

4.3 Cross-Paradigm Comparison

Table 1 presents the results of our cross-paradigm and intra-paradigm evaluations. As described in 4.1, each method was tested on 50 trials for the RLBench tasks OpenWineBottle, TakePlateOffColoredDishRack, and PickUpCups. In the table, results are reported as "number of successes/number of trials", and the rightmost column shows the average success rate across tasks.

For the end-to-end OpenVLA approach, the average success rate was 0%, indicating its inability to execute RLBench tasks in a zero-shot setting. This underscores their limited ability to generalize, making it challenging to transfer to unseen scenes and tasks. Moreover, their fully integrated, closed-box design offers no observable internal states, preventing us from pinpointing the exact causes of failure.

4.4 Intra-Paradigm Comparison

For the hierarchical models SuSIE, the average success rates on RLBench were 0%, demonstrating that they struggle to complete these tasks in a zero-shot setting. For SuSIE, we observed severe hallucinations in its predicted images (see Fig. 4), making it impossible to generate correct goal images. For example, in the TakePlateOffColoredDishRack task, the generated image is completely different from the original image. Consequently, its low-level policy lacked reliable targets and failed to complete the task.

By contrast, the image generation world model in our framework produces more accurate future images that can effectively guide the zero-shot low-level policy (see Fig. 4). The results in Table 1 demonstrate that our framework achieves strong generalization, completing manipulation tasks without any additional training.

4.5 Image-Generation World Model Comparison

We compared several widely used large multimodal image-generation models. As illustrated in Figure 5, the leftmost panel shows the initial input image. We provided each model with the same subgoal description and conducted a qualitative comparison of their generated images. The top row illustrates the generated simulation images for the subgoal "Take Plate Off Colored Dish Rack." The bottom row illustrates the generated real-world images for the subgoal "Move the tomato upwards."

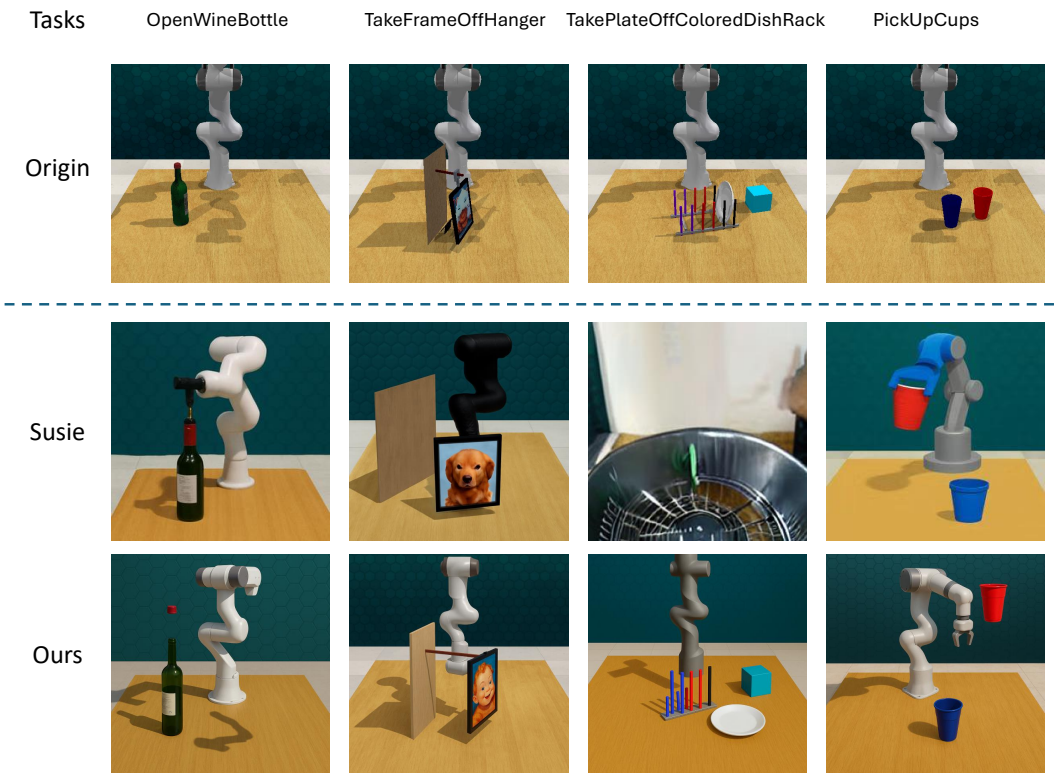


Figure 4: Intra-Paradigm Comparison of generated future images.

We focused exclusively on the correctness of the target object's placement, without considering the consistency of other scene elements.

We observed that GPT-4o's outputs often exhibit stylistic variation while still positioning the object at the intended location. Gemini preserves the original style more faithfully, yet tends to render multiple copies of the target object. The video-generation model Sora suffers from pronounced hallucinations, resulting in drastic scene alterations and poor temporal coherence. Lastly, DALL·E 3 demonstrates limited understanding of scene structure and spatial relationships, resulting in incorrect object placements relative to the environment.

4.6 Zero-shot Low-level Policy Comparison

To compare low-level policies, we evaluated Octo, which takes both the current and predicted future images as input and outputs the actions needed to move the robot toward the goal view. Our experiments show that Octo is unable to zero-shot generalize to the Franka manipulator in RLBench, resulting in task failures.

In contrast, our Grasping+Planning approach achieved a success rate of 35%, surpassing Octo. This demonstrates that the Grasping+Planning module is capable of zero-shot generalization. Most failures in this approach result from the GraspNet module's inability to generate a successful grasp, leading to task failure.

5 Limitation and Future Work

Although the World4Omni framework can execute a variety of robotic manipulation tasks in a zero-shot, cross-embodiment fashion, it makes several trade-offs to achieve this level of generalization. Using the large pre-trained image generation model improves generalization, but it sometimes fails to maintain spatial accuracy, making precise operations such as insertion challenging. Inconsistencies

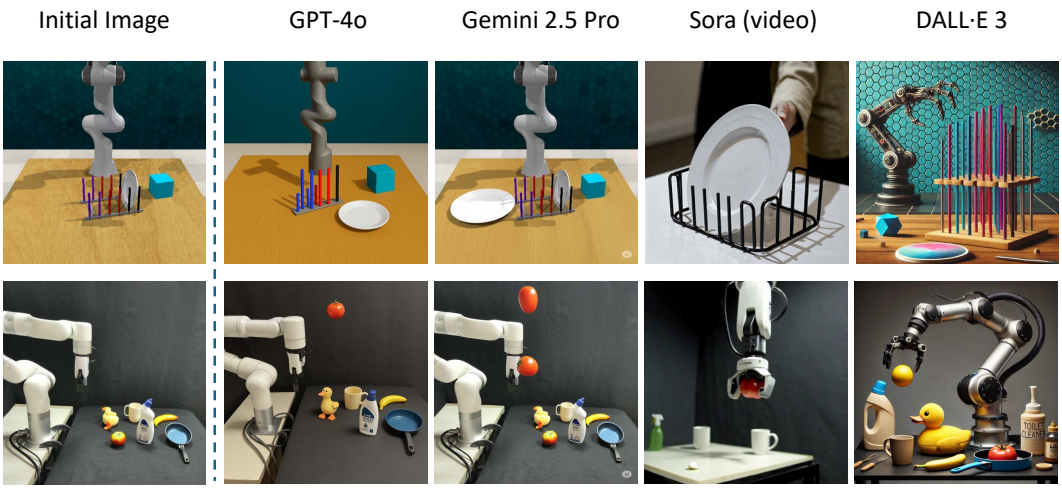


Figure 5: Image generation world model comparison.

in the generated images further complicate execution, and occlusions from the gripper or robot arm impede point-cloud matching, making closed-loop control difficult. Moreover, the foundation grasping module struggles to perform functional grasps on articulated and deformable objects. Although high-level models can generate reasonable future images, the limitations of the low-level policy prevent these tasks from being completed successfully. These findings highlight the need for future research to develop more powerful and general-purpose low-level foundation models.

6 Conclusion

In this work, we introduce **World4Omni**, a hierarchical robot manipulation framework that uses images as intermediate representations. The framework uses a Reflective World Model to generate future scene images and point clouds, where a VLM provides reflective feedback to refine the image quality produced by the pre-trained image-generation model. A zero-shot low-level policy then consumes current and predicted future images (or their corresponding point clouds) to produce robot actions without any additional training. Our cross-paradigm and intra-paradigm evaluations show that World4Omni surpasses representative methods in zero-shot generalization. Moreover, we achieve success in both simulation and real-world settings without any additional training, demonstrating strong generalization and cross-embodiment capabilities. We demonstrate that, by using images generated by foundation models as intermediate representations and executing low-level policies with no additional training, robots can achieve both strong generalization and cross-embodiment across diverse manipulation tasks. This result points the field toward a promising path for realizing general-purpose embodied intelligence.

References

[1] Zhiyuan Xu, Kun Wu, Junjie Wen, Jinming Li, Ning Liu, Zhengping Che, and Jian Tang. A survey on robotics with foundation models: toward embodied ai. *arXiv preprint arXiv:2402.02385*, 2024.

[2] Yang Liu, Weixing Chen, Yongjie Bai, Xiaodan Liang, Guanbin Li, Wen Gao, and Liang Lin. Aligning cyber space with physical world: A comprehensive survey on embodied ai. *arXiv preprint arXiv:2407.06886*, 2024.

[3] Zhou Xian, Theophile Gervet, Zhenjia Xu, Yi-Ling Qiao, Tsun-Hsuan Wang, and Yian Wang. Towards generalist robots: A promising paradigm via generative simulation. *arXiv preprint arXiv:2305.10455*, 2023.

[4] Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3803–3810. IEEE, 2018.

[5] Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE transactions on knowledge and data engineering*, 34(12):5586–5609, 2021.

[6] Tianyu Wang, Dwait Bhatt, Xiaolong Wang, and Nikolay Atanasov. Cross-embodiment robot manipulation skill transfer using latent space alignment. *arXiv preprint arXiv:2406.01968*, 2024.

[7] Nicholas Roy, Ingmar Posner, Tim Barfoot, Philippe Beaudoin, Yoshua Bengio, Jeannette Bohg, Oliver Brock, Isabelle Depatie, Dieter Fox, Dan Koditschek, et al. From machine learning to robotics: Challenges and opportunities for embodied intelligence. *arXiv preprint arXiv:2110.15245*, 2021.

[8] Oliver Kroemer, Scott Niekum, and George Konidaris. A review of robot learning for manipulation: Challenges. *Representations, and Algorithms*, page 82, 2019.

[9] 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. *arXiv preprint arXiv:2307.05973*, 2023.

[10] Jiange Yang, Wenhui Tan, Chuhao Jin, Keling Yao, Bei Liu, Jianlong Fu, Ruihua Song, Gangshan Wu, and Limin Wang. Transferring foundation models for generalizable robotic manipulation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1999–2010. IEEE, 2025.

[11] Dingzhe Li, Yixiang Jin, Yuhao Sun, Hongze Yu, Jun Shi, Xiaoshuai Hao, Peng Hao, Huaping Liu, Fuchun Sun, Jianwei Zhang, et al. What foundation models can bring for robot learning in manipulation: A survey. *arXiv preprint arXiv:2404.18201*, 2024.

[12] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[13] 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 preprint arXiv:2307.15818*, 2023.

[14] 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 preprint arXiv:2406.09246*, 2024.

[15] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.

[16] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[17] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

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

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

[20] Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation. *arXiv preprint arXiv:2409.16283*, 2024.

[21] Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. *arXiv preprint arXiv:2409.01652*, 2024.

[22] 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 preprint arXiv:2204.01691*, 2022.

[23] 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 *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9493–9500. IEEE, 2023.

[24] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 8469–8488, Honolulu, Hawaii, USA, 2023. PMLR. URL `https://proceedings.mlr.press/v202/driess23a.html`.

[25] Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. ivideogpt: Interactive videogpts are scalable world models. *Advances in Neural Information Processing Systems*, 37:68082–68119, 2024.

[26] Zhiyuan Yan, Junyan Ye, Weijia Li, Zilong Huang, Shenghai Yuan, Xiangyang He, Kaiqing Lin, Jun He, Conghui He, and Li Yuan. Gpt-imgeval: A comprehensive benchmark for diagnosing gpt4o in image generation. *arXiv preprint arXiv:2504.02782*, 2025.

[27] Jingxiang Guo and Haonan Chen. Can gpt-4o image generation unlock new potential in robotic manipulation? *TechRxiv*, April 2025. doi: 10.36227/techrxiv.174535631.14854732/v1. URL `http://dx.doi.org/10.36227/techrxiv.174535631.14854732/v1`. Preprint.

[28] Sixiang Chen, Jinbin Bai, Zhuoran Zhao, Tian Ye, Qingyu Shi, Donghao Zhou, Wenhao Chai, Xin Lin, Jianzong Wu, Chao Tang, et al. An empirical study of gpt-4o image generation capabilities. *arXiv preprint arXiv:2504.05979*, 2025.

[29] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

[30] Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

[31] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. Daydreamer: World models for physical robot learning. In *Conference on robot learning*, pages 2226–2240. PMLR, 2023.

[32] Russell Mendonca, Shikhar Bahl, and Deepak Pathak. Structured world models from human videos. *arXiv preprint arXiv:2308.10901*, 2023.

[33] Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. *arXiv preprint arXiv:2504.02792*, 2025.

[34] Chongkai Gao, Haozhuo Zhang, Zhixuan Xu, Zhehao Cai, and Lin Shao. Flip: Flow-centric generative planning for general-purpose manipulation tasks. *arXiv preprint arXiv:2412.08261*, 2024.

[35] Achint Soni, Sreyas Venkataraman, Abhranil Chandra, Sebastian Fischmeister, Percy Liang, Bo Dai, and Sherry Yang. Videoagent: Self-improving video generation. *arXiv preprint arXiv:2410.10076*, 2024.

[36] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10371–10381, 2024.

[37] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 2786–2793. IEEE, 2017.

[38] 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 preprint arXiv:1812.00568*, 2018.

[39] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[40] Marc Rigter, Tarun Gupta, Agrin Hilmkil, and Chao Ma. Avid: Adapting video diffusion models to world models. *arXiv preprint arXiv:2410.12822*, 2024.

[41] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652, 2023.

[42] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594, 2023.

[43] Shreyas Sundara Raman, Vanya Cohen, Ifrah Idrees, Eric Rosen, Raymond Mooney, Stefanie Tellex, and David Paulius. Cape: Corrective actions from precondition errors using large language models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14070–14077. IEEE, 2024.

[44] Pei Ke, Bosi Wen, Zhuoer Feng, Xiao Liu, Xuanyu Lei, Jiale Cheng, Shengyuan Wang, Aohan Zeng, Yuxiao Dong, Hongning Wang, et al. Critiquellm: Towards an informative critique generation model for evaluation of large language model generation. *arXiv preprint arXiv:2311.18702*, 2023.

[45] Zhengyuan Yang, Jianfeng Wang, Linjie Li, Kevin Lin, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. Idea2img: Iterative self-refinement with gpt-4v for automatic image design and generation. In *European Conference on Computer Vision*, pages 167–184. Springer, 2024.

[46] Yunhai Feng, Jiaming Han, Zhuoran Yang, Xiangyu Yue, Sergey Levine, and Jianlan Luo. Reflective planning: Vision-language models for multi-stage long-horizon robotic manipulation. *arXiv preprint arXiv:2502.16707*, 2025.

[47] Zhutian Yang, Caelan Garrett, Dieter Fox, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Guiding long-horizon task and motion planning with vision language models. *arXiv preprint arXiv:2410.02193*, 2024.

[48] Jiaming Liu, Chenxuan Li, Guanqun Wang, Xiaoqi Li, Sixiang Chen, Chuyan Xiong, Jiaxin Ge, Kaichen Zhou, and Shanghang Zhang. Self-corrected multimodal large language model for robot manipulation and reflection. In *International Conference on Learning Representations (ICLR)*, September 2025. URL `https://openreview.net/forum?id=TLWbNfbkxj`. Withdrawn Submission.

[49] Chuyan Xiong, Chengyu Shen, Xiaoqi Li, Kaichen Zhou, Jeremy Liu, Ruiping Wang, and Hao Dong. Aic mllm: Autonomous interactive correction mllm for robust robotic manipulation. *arXiv preprint arXiv:2406.11548*, 2024.

[50] Wenke Xia, Ruoxuan Feng, Dong Wang, and Di Hu. Phoenix: A motion-based self-reflection framework for fine-grained robotic action correction. *arXiv preprint arXiv:2504.14588*, 2025.

[51] Roya Firoozi, Johnathan Tucker, Stephen Tian, Anirudha Majumdar, Jiankai Sun, Weiyu Liu, Yuke Zhu, Shuran Song, Ashish Kapoor, Karol Hausman, et al. Foundation models in robotics: Applications, challenges, and the future. *The International Journal of Robotics Research*, page 02783649241281508, 2023.

[52] Lihe Li, Lei Yuan, Pengsen Liu, Tao Jiang, and Yang Yu. Llm-assisted semantically diverse teammate generation for efficient multi-agent coordination. In *Proceedings of the Forty-second International Conference on Machine Learning*, 2025.

[53] Haonan Chen, Junxiao Li, Ruihai Wu, Yiwei Liu, Yiwen Hou, Zhixuan Xu, Jingxiang Guo, Chongkai Gao, Zhenyu Wei, Shensi Xu, et al. Metafold: Language-guided multi-category garment folding framework via trajectory generation and foundation model. *arXiv preprint arXiv:2503.08372*, 2025.

[54] Chenrui Tie, Shengxiang Sun, Jinxuan Zhu, Yiwei Liu, Jingxiang Guo, Yue Hu, Haonan Chen, Junting Chen, Ruihai Wu, and Lin Shao. Manual2skill: Learning to read manuals and acquire robotic skills for furniture assembly using vision-language models. *arXiv preprint arXiv:2502.10090*, 2025.

[55] 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. *arXiv preprint arXiv:2502.08643*, 2025.

[56] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

[57] Junyi Zhang, Charles Herrmann, Junhwa Hur, Eric Chen, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. Telling left from right: Identifying geometry-aware semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[58] Shinji Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE transactions on pattern analysis and machine intelligence*, 10(5):695–703, 1988.

[59] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 11444–11453, 2020.

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

[61] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[62] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

[63] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[64] Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024.

[65] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. *OpenAI Blog*, 2024. URL `https://openai.com/research/video-generation-models-as-world-simulators`. Online; accessed 16 May 2025.

[66] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: Section 1.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Section 5.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Section 3.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: See supplemental materials.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: Section 4.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [No]

Justification: For robotic manipulation tasks, performance is typically evaluated solely by task success, without reporting error bars or other measures of uncertainty.

Guidelines:

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

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Section 4.1.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [No]

    Justification: This work is not related to any private or personal data, and there are no explicit negative social impacts.

    Guidelines:

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

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

11. **Safeguards**

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

    Answer: [No]

    Justification: OpenAI has published detailed safety documentation for both the language-only and multimodal models used in the research, explicitly outlining the safeguards applied before any public release.

    For GPT-4o: `https://openai.com/index/gpt-4o-system-card/`

    For GPT-o4-mini-high: `https://openai.com/index/o3-o4-mini-system-card/`

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: Yes, we credited them in appropriate ways. And the licenses are below:

    OpenAI: `https://openai.com/policies/row-terms-of-use/`

    GraspNet: [59]

    GeoAware: [57]

    SAM2: `https://github.com/facebookresearch/sam2/blob/main/LICENSE`

    Grounded-SAM-2: `https://github.com/IDEA-Research/Grounded-SAM-2/blob/main/LICENSE`

    cuRoBo: `https://github.com/NVlabs/curobo/blob/main/LICENSE`

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

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

13. **New assets**

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

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

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

14. **Crowdsourcing and research with human subjects**

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

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

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

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

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

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research, the declaration is not required.

    Answer: [Yes]

    Justification: Section 3.2.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.