RoboLab: A High-Fidelity Simulation Benchmark for Analysis of Task Generalist Policies

Xuning Yang[1], Rishit Dagli[2,4], Alex Zook[1], Hugo Hadfield[1],
Ankit Goyal[1], Stan Birchfield[1], Fabio Ramos[1,3], and Jonathan Tremblay[1]
[1]NVIDIA, [2]University of Toronto, [3]The University of Sydney, [4]Work done during internship at NVIDIA

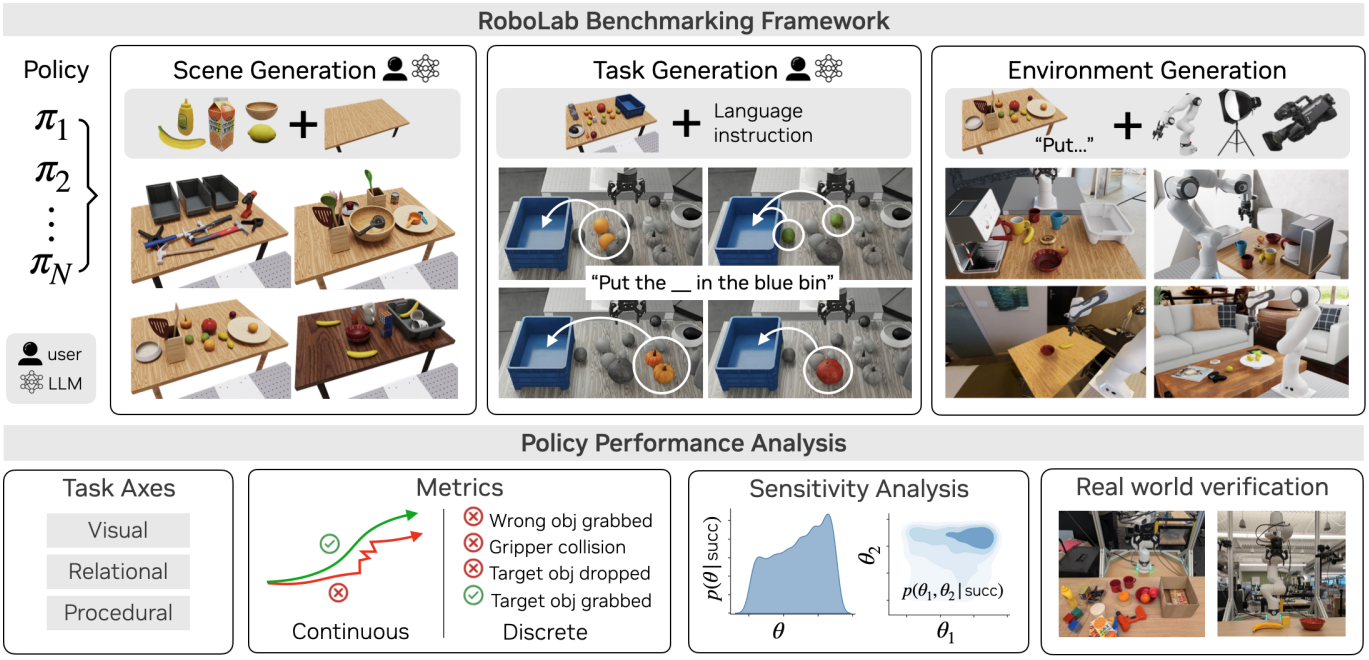

Fig. 1: **Overview of RoboLab.** RoboLab addresses the simulation-to-real gap by evaluating robotics policies on entirely held-out domains. By featuring a streamlined generation pipeline for new scenes and tasks (top row), RoboLab enables rapid extensibility for testing generalization capabilities. Our accompanying benchmark introduces visual, relational, and procedural testing axes, paired with robust metrics designed to reveal how modern models perform when faced with novel, out-of-distribution challenges (bottom row).

***Abstract*—The pursuit of general-purpose robotics has yielded impressive foundation models, yet simulation-based benchmarking remains a bottleneck due to rapid performance saturation and a lack of true generalization testing. Existing benchmarks often exhibit significant domain overlap between training and evaluation, trivializing success rates and obscuring insights into robustness. We introduce RoboLab, a simulation benchmarking framework designed to address these challenges. Concretely, our framework is designed to answer two questions: (1) to what extent can we understand the performance of a real-world policy by analyzing its behavior in simulation, and (2) which external factors most strongly affect that behavior under controlled perturbations. First, RoboLab enables human-authored and LLM-enabled generation of scenes and tasks in a robot- and policy-agnostic manner within a physically realistic and photorealistic simulation. With this, we propose the RoboLab-120 benchmark, consisting of 120 tasks categorized into three competency axes: visual, procedural, relational competency, across three difficulty levels. Second, we introduce a systematic analysis of real-world policies that quantify both their performance and the sensitivity of their behavior to controlled perturbations, indicating that high-fidelity simulation can serve as a proxy for analyzing performance and its dependence on external factors. Evaluation with RoboLab exposes significant performance gap in current state-of-the-art models. By providing granular metrics and a scalable toolset, RoboLab offers a scalable framework for evaluating the true generalization capabilities of task-generalist robotic policies.**

I. Introduction

The pursuit of generality has been a longstanding challenge in modern robotics. Recent advances have produced impressive generalist robot policies that demonstrate success in challenging and novel tasks in the real-world. Despite this progress, benchmarks for evaluating whether these policies are truly task-general has been slow. Evaluating models in the real world remains prohibitively expensive and logistically intractable, motivating the rise of simulation-based benchmarks as an appealing alternative.

Current robotics benchmarks [19, 35, 11, 16] face several critical limitations: (1) a lack of high-fidelity simulation capable of supporting real-world policies; (2) rapid performance saturation on static task sets; and (3) a lack of granular analysis regarding policy failure modes. For instance, popular benchmarks like

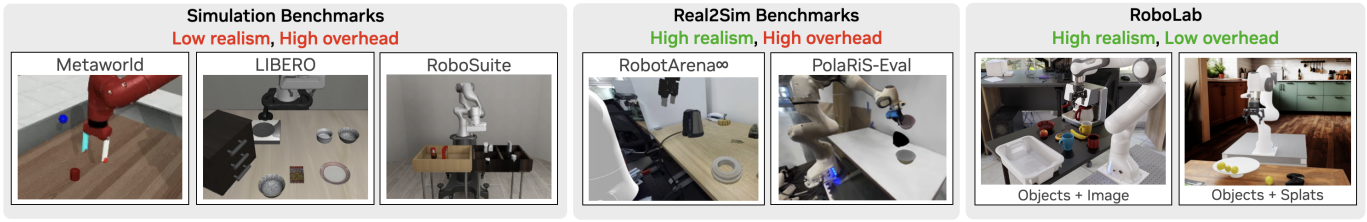

Fig. 2: Three approaches for robotic benchmarks. LEFT: To date, pure simulation based benchmarks have exhibited low visual quality, creating a large sim2real transfer gap. MIDDLE: Real2sim benchmarks address this issue by using techniques to bring real-world visual texture into simulation. However, these environments are extremely costly with reported per-scene generation time of ∼1hr [10]. RIGHT: Our approach achieves a high degree of realism with low overhead.

LIBERO [19] often utilize nearly identical environments for both training and evaluation. When policies are fine-tuned on these simulation-specific demonstrations, the lack of a meaningful domain gap trivializes the evaluation process and obscures the model's true generalization capabilities. Many existing platforms have limited realism or are difficult to extend due to using rigid architectures that make it cumbersome to introduce new objects, tasks, or robots (Fig. 2).

To address these limitations, we present RoboLab (Fig. 1), a simulation platform and benchmarking suite designed for rigorous robotics evaluation. Unlike prior benchmarks that rely on PDDL or rigid scene-graph definitions [19], RoboLab introduces an easy-to-use interface that enables human-authored and LLM-scaled scene and task generation. RoboLab enables generation and validation of new scenes and tasks from natural language prompts. This system enables the creation of over 800 diverse scenarios (see supplemental), providing a scalable framework that mitigates benchmark saturation and ensures long-term value.

RoboLab introduces novel task axes and robust metrics to provide deeper diagnostic insights, paired with a streamlined toolset for generation of scenes and tasks (see Fig. 1).. To provide a granular assessment of policy behavior in our proposed benchmark, we evaluate three axes: Visual (perceptual attributes like color and size), Procedural (action-oriented logic such as stacking and reorientation), Relational (spatial and linguistic logic like "and/or") spanning across three difficulty levels depending on task length and language nuance. Policy execution on these tasks is then evaluated with metrics on graded task completion, failure and error occurrences, and trajectory quality. Finally, we highlight novel metrics for evaluation; including sensitivity analysis to identify environmental factors that most strongly influence policy performance, *e.g.*, camera placement.

We introduce the RoboLab-120 benchmark, comprising 120 tasks generated via our automated workflow and verified by humans. These tasks span varying difficulties (35 simple, 28 moderate, 16 complex) and multiple competency axes (33 relational, 54 visual, and 9 procedural). To prevent overfitting to the simulation domain, we evaluate policies trained exclusively on the real-world DROID [13] dataset. This creates an environment that reflects "in the wild" conditions; for instance, the state-of-the-art $\pi_{0.5}$ [9] achieves only a ∼30% success rate on RoboLab-120, highlighting the benchmark's difficulty.

In summary, our contributions are:

1) **RoboLab**: A novel simulation platform designed for evaluating modern robotics policies with a scalable, LLM-based workflow capable of procedurally generating over 800 unique scenes and tasks using human-readable USD and Python interfaces in IsaacLab[21].
2) **RoboLab-120 Benchmark**: Comprising 120 tasks evaluated across three distinct competency axes (visual, procedural, relational) and supported by four new robustness metrics. We also present five policies evaluated on RoboLab-120.
3) **Policy Analysis**: We introduce a suite of analysis tools that gives insight into the model performance beyond binary success rates and broader understanding of policy performance.

II. RELATED WORK

Simulation-Based Benchmarks. Simulation provides a scalable and reproducible environment for evaluating robot manipulation policies. Widely used benchmarks such as RLBench [11], MetaWorld [33], and robosuite [36], ManiSkill2 [7], CALVIN [20], LIBERO [19], and BEHAVIOR-1K [15], offer standardized task suites for learning and evaluation in simulation across pre-defined task families and object configurations. However, in these settings, policies are typically trained and evaluated in the same simulated environments, which encourages overfitting to simulator-specific quirks, leads to rapid benchmark saturation, and makes real-world generalization hard to assess. [35]. In our setting, policies are instead trained on large-scale real-world data (e.g., DROID [13]), while high-fidelity simulation is used only as a controlled evaluation environment, so training and evaluation domains are decoupled and measured performance more closely reflects robustness in the real world.

Real-to-sim Evaluation. Recent work have focused on leveraging 3D reconstruction to build photorealistic simulation scenes from real-world videos in order to achieve closer visual alignment between simulation and real world photorealism [16, 12, 10, 37]. These works typically use Gaussian splatting, 3D segmentation, and multi-view inpainting, often operated at a per-scene level, which entails costly optimization and

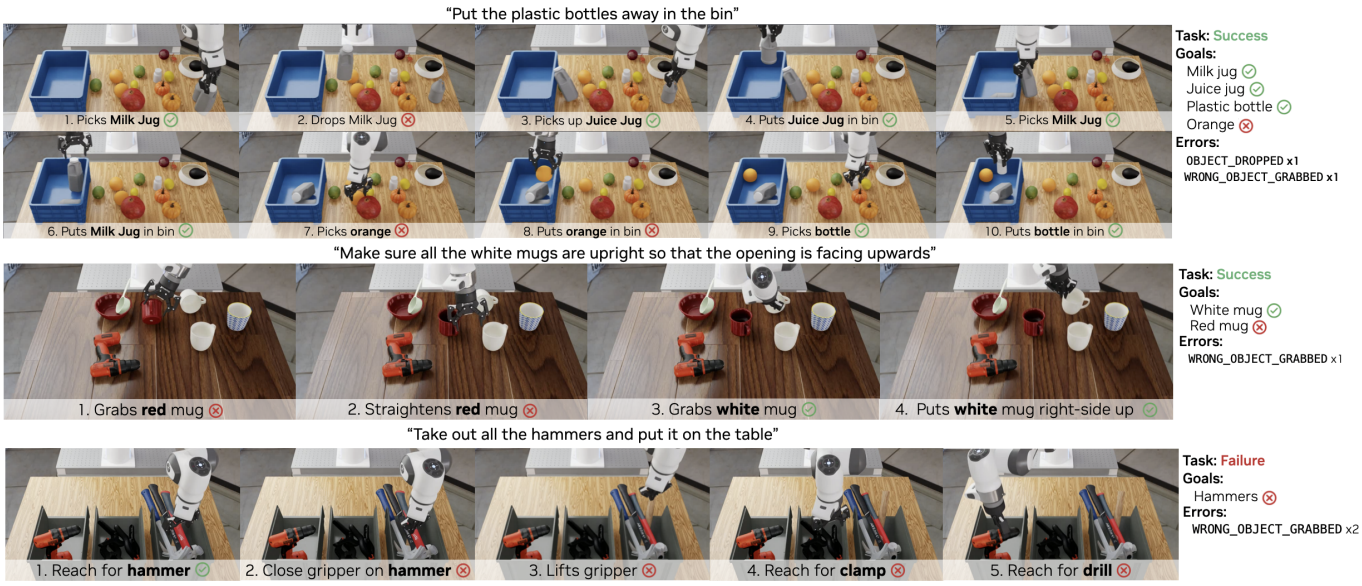

Fig. 3: **Task progression of a few tasks, illustrating errors encountered during policy rollout.** Top row: Although the task is successfully completed, errors were encountered during execution: 1) The robot drops the milk jug too early, missing the bin. 2) the robot grasps an orange (wrong object) and puts it in the bin. Mid row: An extraneous object was reoriented before the actual intended object. Final row: Intended objects were attempted unsuccessfully, and the policy tended to two wrong other objects.

makes it slow to scale beyond a small number of environments [10, 34, 28]. In contrast, our framework produces large-scale, photorealistic scenes and tasks within minutes rather than hours, while preserving sufficient geometric and visual fidelity for policy evaluation, thereby making real-to-sim benchmarking practical at the scale needed for modern generalist robot policies.

III. RoboLab

Evaluating real-world, generalist robotics policies in simulation remains a significant challenge. RoboLab is a benchmarking framework that introduces three novel task axes and three original metrics tailored for modern robotics systems. RoboLab enables a multifaceted analysis of Vision-Language-Action (VLA) models, providing deeper insights into their scalability and task generalization.

A. RoboLab-120

Inspired by the Large Language Model (LLM) community's use of Visual Question and Answering (VQA) benchmarks, we introduce RoboLab-120 Benchmark that focus on evaluating specific competency axes spanning three difficulty levels. This taxonomic decomposition enables fine-grained analysis of policy capabilities by systematically assessing performance. Figure 4 shows examples of these questions accompanied by scene examples.

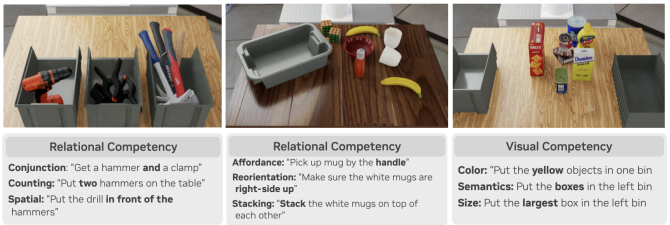

Fig. 4: Example of language instructions in RoboLab-120.

Visual Competency: Assesses recognition of *color*, *semantics*, and *size*, capturing the policy's capability to link perceptual attributes with higher-level reasoning.

Procedural Competency: Evaluates the ability to perform tasks that involve action-oriented reasoning, including *affordances*, *reorientation*, or *stacking*.

Relational Competency: Tests understanding of language *conjunctions* (e.g., 'and', 'or'), *counting*, and *spatial* relationships, measuring how effectively the policy interprets multi-object instructions and scene structure.

Tasks from these competencies can span one of the following difficulty levels: simple, medium, complex. These are determined as a function of two aspects: whether if the language was straightforward in describing the task, as well as the number of required reasoning steps for the task.

B. Metrics for Evaluation

We establish a comprehensive suite of evaluation metrics that captures the full spectrum of policy performance characteristics. While task success rate remains a fundamental metric, prior work [14] has demonstrated that they fail to reveal nuanced aspects of policy behavior and failure modes. Unlike approaches relying on human judgment [12], we define a set of discrete and continuous metrics to characterize policy performance. We regroup these novel metrics as follow, failure cases scoring, trajectory metrics, and sensitivity analysis.

Failure cases. In addition to *success rate*, we compute

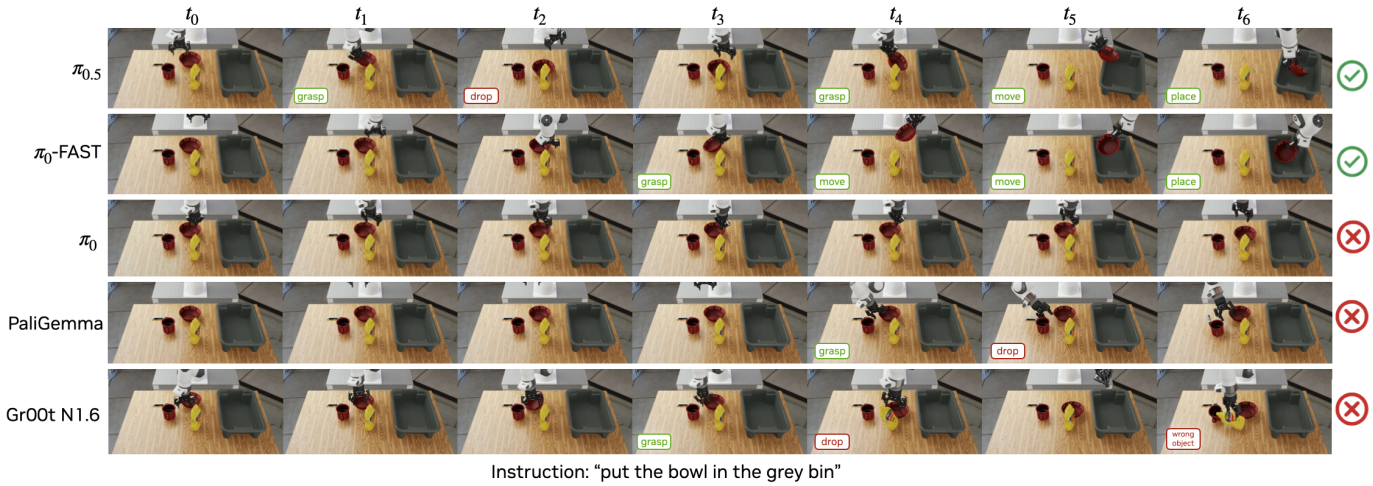

Fig. 5: Comparison of policy performance for bowl-in-bin manipulation. Rows represent distinct policies shown in chronological order (left to right). Successful execution involves grasping the central red bowl and depositing it into the gray bin on the right. Unsuccessful attempts are characterized by aimless arm trajectories and a lack of object interaction.

a *normalized graded score* $Sc(\mathcal{T}) = \frac{1}{|\mathcal{T}|}\sum_{\tau\in\mathcal{T}} Sc(\tau)$. For example, for the instruction "pick the lemon and the lime," the subtasks τ "pick lemon" and "pick lime" includes steps such as "grasp" and "drop". The final task score is the normalized subtask scores. Our benchmark automatically records instances of events; including wrong object grasped, object dropped, and gripper collisions. Fig. 3 demonstrates a successful episode; however, the policy incorrectly grasped an extraneous object. Such errors highlight potential biases in the policy not captured by other metrics.

Trajectory Metrics. Trajectory quality metrics capture characteristics of motion efficiency and optimality. We compute the following: *Spectral arc-length (SPARC)*, which evaluates motion smoothness [2] via the arc length of the normalized Fourier magnitude spectrum of the velocity profile. Given a speed profile $v(t)$ of the end effector over time interval $[0, T]$

$$\text{SPARC} = -\int_0^{\omega_c} \sqrt{\left(\frac{1}{\omega_c}\right)^2 + \left(\frac{d\hat{V}(\omega)}{d\omega}\right)^2}\, d\omega \qquad (1)$$

where $\hat{V}(\omega) = V(\omega)/V(0)$ represents the normalized Fourier magnitude spectrum. Smoother motions yield values closer to zero, while jerkier trajectories produce more negative values. We employ an adaptive cutoff frequency $\omega_c = \min(10\text{ Hz}, \omega_\alpha)$, where $\omega_\alpha = \max_{k\in\mathcal{K}} \omega_k$ and $\mathcal{K} = \{k \mid \hat{V}(\omega_k) \geq \alpha\}$ denotes the set of frequency bins exceeding threshold $\alpha = 0.05$. This adaptive strategy ensures that the smoothness evaluation focuses on relevant frequency components. Lastly, trajectory optimality is assessed through end effector *speed* $v(t)$, and *path length* $l = \sum_{k=0}^{N-1} \|p_{k+1} - p_k\|$, where p_k denotes the end-effector position at timestep k. Shorter path lengths indicate more direct trajectories and generally reflect superior motion quality.

C. Sensitivity Analysis

We present a Bayesian framework for evaluating policy robustness across diverse environmental conditions using Simulation-Based Inference (SBI). This analysis provides insight into which scene parameters are most strongly linked to success and failure outcomes by learning an approximate posterior distribution over them given evaluation data. Let $\theta = (\theta^{\text{cont}}, \theta^{\text{disc}})$ denote the environment parameters comprising of continuous variables (e.g., object distance, camera displacement) and/or discrete variables. After evaluating policy π under varied conditions, we generate episodes $\mathcal{D} = \{(\theta_i, x_i)\}_{i=1}^N$ with observed outcomes x_i (e.g., task success). The posterior distribution $p(\theta \mid x) \propto p(x \mid \theta)p(\theta)$ is approximated using Mixed Neural Posterior Estimation (MNPE), which trains a neural density estimator $q_\phi(\theta \mid x)$ to directly learn the mapping from observations to parameter distributions. The resulting posterior $q_\phi(\theta \mid x)$ characterizes which scene variables are most associated with a target observation x. Our approach provides systematic assessment of which variables most strongly influence performance outcomes. Further details are in Appendix B.

D. Robolab scene and task generation

RoboLab offers a user-friendly workflow that mirrors the process of preparing a real-world robot evaluation (Fig. 1): 1) create a **scene** by positioning and orienting objects in a workspace; 2) define a **task** as language instructions for a goal state in the scene; 3) instantiate an **environment** by selecting a robot, policy, and variations of scene features including camera, lighting, and backgrounds for a task. We make this process reproducible and scalable by decoupling task definitions from environments, allowing reuse over new embodiments and policies. Our approach automates the process of environment assembly, reducing manual labor when evaluating a new robot or policy. In addition, we developed an automated workflow

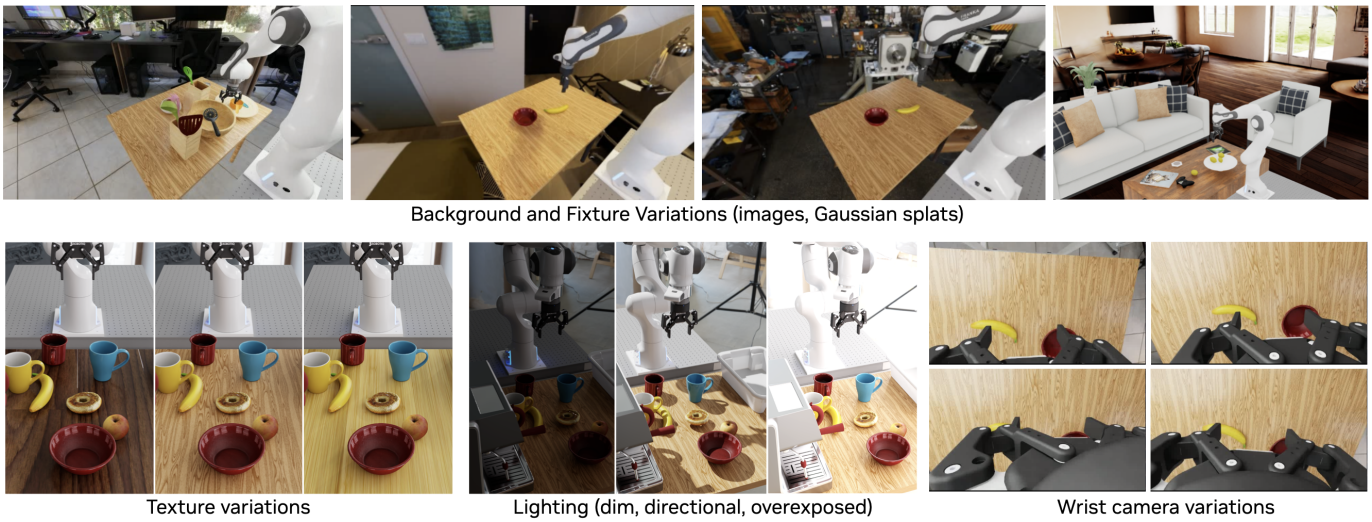

Fig. 6: Example scene variations, lighting variations, and camera pose variations in RoboLab.

to generate new scenes and tasks to facilitate extension of our evaluations and to mitigate benchmark saturation in the future.

Formally, define a *scene* $S = \{(b_i, \mathbf{p}_i, \mathbf{q}_i)\}_{i=1}^{N}$, where b_i represents an object instance selected from the available catalog of objects $\mathcal{B}$ and $\mathbf{p}_i \in \mathbb{R}^3, \mathbf{q}_i \in SO(3)$ denote its position and orientation. Define a *task* $\mathcal{T} = \{S, l\}$, where l is the *language instruction* to complete in the scene. Define a policy $\pi : \mathcal{O} \to \mathcal{A}$ where the action space $\mathcal{A} \in \{\mathcal{A}^{\text{joint}}, \mathcal{A}^{\text{EE}}, \dots\}$ and observation space $\mathcal{O} = (\mathcal{O}^{\text{proprio}}, \mathcal{O}^{\text{rgb}}, \mathcal{O}^{\text{depth}} \dots)$ is policy dependent. An environment $E = (\mathcal{T}, \mathcal{R}, \mathcal{O}, \mathcal{A}, \xi)$ consists of a task, robot embodiment $\mathcal{R}$, policy parameters $(\mathcal{A}, \mathcal{O})$, and scene variations $\xi = (\xi^{\text{camera}}, \xi^{\text{light}}, \xi^{\text{background}}, \xi^{\text{pose}})$. More details on the specific objects, scenes and tasks in RoboLab can be found in Appendix A.

1) Scaling Scene Generation: We enable scaling scene generation through an automated pipeline that: 1) prompts an LLM to generate a structured scene plan for asset placement; 2) uses a geometric solver and physics simulation to check asset placement validity; and 3) refines the scene if it is not valid. First, the LLM is prompted with a theme (e.g., "messy counter") to generate a structured scene plan consisting of a subset of objects $B \subset \mathcal{B}$ and spatial predicates $\mathcal{P}$ governing the layout. The LLM is provided with the full catalog of objects $\mathcal{B}$ containing names and bounding box dimensions $\mathbf{d}_i \in \mathbb{R}^3$.

Second, a spatial solver converts the relational predicates $\mathcal{P}$ into valid pose configurations $(\mathbf{p}, \mathbf{q})$). Objects are processed in dependency order, with support surfaces placed before objects on those surfaces (Algorithm 1). To check physical stability, the scene is then forward simulated in Isaac Sim [23] for 300 steps under gravity. An object b_i is flagged as *unstable* if it's maximum Euclidean displacement is larger than a threshold (typically 0.02m). Third, If any object is unstable, we generate a text error describing the failure (e.g., "Object 'apple' fell off 'plate' with displacement 0.15m"). This feedback is provided to the LLM to refine the scene plan and repeat the process. Further details, including on the spatial and physical solvers, are provided in Appendix C.

2) Scaling Task Generation: We enable scaling task generation through an automated pipeline that: 1) generates task code from information including the scene and competency axes; 2) validates code syntax; 3) validates asset selections in the scene; and 4) refines the task if it is not valid. First, we prompt an LLM with detailed task information: 1) the scene object catalog B_S and metadata (including bounding boxes and semantics) with dimensions; 2) task examples demonstrating the task structure; 3) the complete predicate library defining sub-task success and termination; 4) Competency-axes language templates with placeholders for objects, spatial verbs, and attributes; and 5) constraints including difficulty levels and physical feasibility requirements (e.g., containment size constraints, stacking stability). The prompt forbids referencing objects not present in B_S and includes previously generated tasks to prevent duplicates (see Appendix C for details). Second, tasks

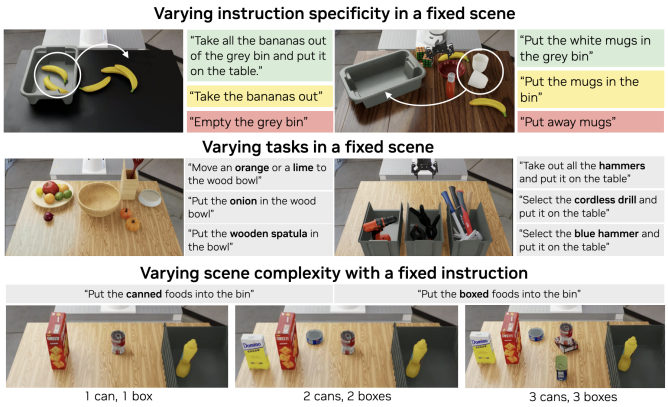

Fig. 7: Examples of language ablation experiments. Top: Same scene and goal, but the instruction wording ranges from precise to increasingly vague. Middle: Same scene, but the instruction specifies different tasks to perform. Bottom: Same instruction, but the scene becomes progressively more complex.

TABLE I: **Overall performance of VLAs on RoboLab.** While recent VLAs exhibit emerging capabilities across diverse task dimensions, overall success rates and consistency remain limited.†

	Overall Metrics				**Difficulty** (succ%)			**Procedural** (succ%)			**Relational** (succ%)			**Visual** (succ%)		
Model	Succ% (↑)	Score (↑)	SPARC (↑)	Speed (↑)	simple	moderate	complex	affordance	reorientation	stacking	conjunction	counting	spatial	color	semantics	size
$\pi_{0.5}$ [9]	**31.9**	**0.11**	−9.10 ±4.5	**5.7** ±1.9	**40.6**	**28.2**	**19.4**	**20.0**	**53.3**	**16.0**	**76.0**	**72.0**	**23.9**	**30.0**	**21.5**	**35.0**
π_0-FAST [27]	18.2	0.05	−9.14 ±4.7	4.7 ±1.9	26.0	17.1	3.1	0.0	20.0	0.0	42.0	56.0	20.0	8.3	12.1	**35.0**
GR00T N1.6 [24]	8.0	0.06	**−6.44** ±2.0	4.2 ±1.3	12.3	7.1	0.0	0.0	0.0	0.0	4.0	38.0	7.0	5.6	5.9	0.0
π_0 [5]	5.9	0.04	−9.45 ±3.4	4.4 ±1.4	11.7	2.1	0.0	0.0	0.0	0.0	26.0	10.0	3.0	1.7	1.8	0.0
PaliGemma [4]	1.9	0.01	−17.6 ±11.8	1.4 ±1.2	1.1	3.9	0.0	0.0	0.0	0.0	0.0	22.0	1.3	0.0	2.1	5.0

†Current paper results were from an older version of the benchmark which contained 80 tasks; updated results on the full benchmark (120 tasks) will be available soon.

are check for syntax validity as code. Third, asset validation checks that all objects are not in the forbidden set and, for containment tasks (e.g., "place b_i inside b_j"), that inner objects fit inside containers with some clearance. Fourth, if validation fails, feedback is gathered into a fix prompt $\mathcal{Q}_{\text{fix}}$ that includes the original prompt $\mathcal{Q}$, the invalid output, and an error message $\mathcal{E}$ describing syntax errors or invalid asset references. The fix prompt is provided to the LLM to refine the task and repeat the process.

We evaluated our task generation approach using an LLM-as-judge framework. We generated 812 tasks across 59 scenes evenly across the competency axes with o1 [26]. We then extracted each natural-language instruction and its programmatic termination conditions from the generated code, and prompted a second o1 judge to score instruction–criterion alignment across relation, target, object, and quantifier match, plus instruction clarity and physical feasibility (each on a 0–1 scale), and to assign an aligned/partial/misaligned verdict. Overall, tasks achieved 0.91 alignment, 0.96 clarity, 0.92 feasibility, and 0.95 semantic match, with 76% judged fully aligned (misaligned $\approx 1\%$) and covered 88% of objects. These results show our approach can scale to generate diverse tasks that are semantically aligned to their language instructions (see Appendix D).

IV. Experiments

We evaluate several off-the-shelf VLA policies on RoboLab-120, controlled ablations, and environmental perturbations to identify which competencies generalize and where failures concentrate. The experiments are designed to address the following questions: **Q1:** How well does a real-world policy perform in our simulated benchmark? **Q2:** How well does a policy generalize with language variations? **Q3:** When and why does a policy fail?

A. Experiment Setup

We evaluate 80 tasks† of varying difficulty levels (35 simple, 28 moderate, 16 complex) and spanning competency axes (33 relational, 54 visual, and 9 procedural). Each task was assigned to one or more competency axes. Our experiments used the DROID robot [13], which is commonly used to benchmark VLAs [10, 1]. DROID has a 7-DOF Franka Panda robot arm with a Robotiq-2F-85 gripper, an externally mounted ZED 2i camera with f=2.1mm, and a ZED mini as the wrist camera. We evaluated VLAs with off-the-shelf checkpoints fine-tuned on the DROID dataset [13]: $\pi_{0.5}$ [9], π_0-FAST [27], π_0 [5], PaliGemma [4], and GR00T N1.6 [24]. The action space is 7-DOF Franka joint positions and a 1-DOF binary gripper command. The environments were composed of a default office-like background and natural lighting to mimic typical setups in the DROID dataset [13], with wrist and external camera poses designed to match the real-world DROID robot. Each task was repeated 10 times with a fixed seed to address uncontrolled stochasticity in the physics simulation and robot policy.

B. Task Results

Table I shows the overall results on our benchmark. Overall success rates were low, with the best policy ($\pi_{0.5}$) reaching 31.9% success. This matches prior observations on out-of-domain generalization for VLAs [35]. Below, we discuss $\pi_{0.5}$ to illustrate how RoboLab supports targeted diagnosis of policy capabilities and suggests concrete directions for improvement.

Competency axes also highlighted asymmetric generalization in **relational reasoning** tasks: $\pi_{0.5}$ handled conjunctions (76.0% success) and counting (60.0%) better than spatial relations (23.9%). In **visual grounding**, performance remained low across attribute types (35.0% for size, 30.0% for color, and 21.5% for semantics), indicating brittle language-to-object binding beyond a narrow set of familiar object descriptions. **Procedural understanding** proved most challenging: $\pi_{0.5}$ achieved modest success on reorientation (53.3%) but struggled with affordances (20.0%) and stacking (16.0%). Together, these results show how RoboLab isolates where generalization fails, supporting diagnosis that can inform data collection and training priorities.

C. Ablation Experiments

To further probe robustness and language grounding, we performed controlled ablations that varied the instruction, scene, or task in isolation (Fig. 7).

Varying instruction specificity in a fixed scene. Table IIa shows how VLAs respond to varying levels of instruction specificity. Results reveal that VLAs lack grounding for abstract or implied goals. Interestingly, we observe that the for the vague command "Empty the grey bin", $\pi_{0.5}$ tries to grab the bin instead of clearing its content. These results indicate that current VLAs may rely on keyword matching within instructions rather than demonstrating the linguistic inference necessary to identify implicit task goals. As shown in Table IIa, $\pi_{0.5}$ exhibits higher performance on specific instructions than on vague ones, indicating a sensitivity to unde-rspecified task goals.

TABLE II: Language understanding ablations. (a) VLA performance degrades with abstract or vague language instructions. (b) Performance drops as scene complexity increases. (c) VLAs show brittle language grounding, with consistent object confusion patterns across different instructions in the same scene.

(a) Effect of language specificity on task performance.

	$\pi_{0.5}$		π_0**-FAST**	
Task	Succ %	Score	Succ %	Score
Bananas Out Of Bin Task				
"Take all the bananas out of the grey bin and put it on the table."	50	0.13	30	0.05
"Take the bananas out"	40	0.22	10	0.15
"Empty the grey bin"	10	0.07	70	0.11
White Mugs In Bin Task				
"Put the white mugs in the grey bin"	80	0.50	20	0.22
"Put the mugs in the bin"	90	0.50	10	0.11
"Put away mugs"	0	0.00	0	0.00
Remove Measuring Spoons from the Plate Task				
"Put the orange measuring cup and the blue measuring cup outside of the plate"	20	0.47	0	0.31
"Clear the plate"	0	0.08	0	0.10

(b) Effect of scene complexity on task performance.

Scene	$\pi_{0.5}$	π_0**-FAST**	π_0	**GR00T N1.6**
Task: *"Pack boxed foods into the bin"*				
1 Box/Can	10	0	0	0
2 Boxes/Cans	0	0	0	0
3 Boxes/Cans	0	0	0	0
Task: "Pack canned foods into the bin"				
1 Box/Can	70	30	0	0
2 Boxes/Cans	30	10	0	0
3 Boxes/Cans	20	0	0	0

(c) Instruction sensitivity within fixed scenes.

Task / Prompt	$\pi_{0.5}$	π_0**-FAST**	π_0	**GR00T N1.6**
Fruit Plate Scene				
"Move an orange or a lime to the wood bowl"	50	0	0	0
"Move an orange to the white bowl"	0	0	0	0
"Put the onion in the wood bowl"	70	10	20	20
"Put the onion on the plate"	0	0	0	0
Tools Cleanup Scene				
"Put hammers in the right bin"	20	0	0	0
"Put hammers in the left bin"	10	0	0	0
Tools Selection Scene				
"Select the cordless drill and put it on the table"	70	50	20	30
"Select the blue hammer and put it on the table"	0	0	10	0

Varying scene complexity with a fixed instruction. Table IIb isolates the effect of scene complexity by increasing the number of objects to manipulate. Success rates drop as object count increases: from 70% with a single target object to 20% with three objects. More revealing is the *type* of failure: VLAs exhibit systematic geometric biases, frequently grasping cylindrical objects (cans) when instructed to manipulate boxes. This suggests that training data distributions create strong shape priors that override language-specified targets, a critical limitation for real-world deployment where objects vary widely in geometry.

TABLE III: Robustness to controlled environmental variations over two simple tasks (BananaInBowl, BananaAndCubeInBowl). PaliGemma is excluded as it fails to achieve meaningful results.

	$\pi_{0.5}$		π_0**-FAST**		π_0	
Variation	Succ.%	Time (s)	Succ.%	Time (s)	Succ.%	Time (s)
Lighting						
Color	96.7	14.5 $\pm$7.9	93.3	17.9 $\pm$10.7	6.7	31.1 $\pm$4.3
Shadows	100.0	16.0 $\pm$6.0	90.0	12.4 $\pm$3.5	0.0	-
Dim	90.0	9.1 $\pm$2.1	70.0	13.1 $\pm$2.8	70.0	35.5 $\pm$9.7
Overexposed	100.0	13.9 $\pm$4.4	100.0	9.6 $\pm$1.7	0.0	-
Visual Variations						
Background	85.0	14.4 $\pm$8.7	70.0	21.3 $\pm$10.7	25.0	31.6 $\pm$11.8
Table texture	87.5	19.0 $\pm$13.8	60.0	19.0 $\pm$12.9	22.5	28.1 $\pm$6.9
Object Pose						
10cm	95.0	16.2 $\pm$9.2	55.0	26.5 $\pm$13.8	22.5	34.7 $\pm$13.3
20cm	95.0	19.7 $\pm$10.2	40.0	21.8 $\pm$9.5	20.0	37.4 $\pm$11.2
30cm	62.5	18.9 $\pm$8.7	35.0	24.3 $\pm$11.9	17.5	24.3 $\pm$11.9
Camera Pose						
external	85.0	17.4 $\pm$11.3	45.0	27.7 $\pm$10.6	50.0	27.4 $\pm$16.0
wrist	60.0	21.9 $\pm$13.4	25.0	20.1 $\pm$9.9	10.0	35.3 $\pm$10.4

Varying tasks in a fixed scene. Table IIc shows whether VLAs can flexibly respond to different instructions while holding the scene fixed. Results expose brittle language grounding: $\pi_{0.5}$ was highly sensitive to object choice; 70% success for "Select the cordless drill and put it on the table" but 0% when replacing "cordless drill" with "blue hammer". Error analysis reveals consistent object confusion patterns: visually similar distractors (pumpkin vs. orange, drill vs. hammer) frequently override language-specified targets (see Table. VI in Appendix for detailed analysis). These findings indicate that VLA language grounding is highly sensitive to the specific object-instruction pairings seen during training, rather than reflecting generalizable language-to-object binding.

D. Sensitivity and robustness

We perform a set of variations given two basic tasks and observe the outcome, for example, we only change the target object to pick or the target for place, this is akin to domain randomization [29] as illustrated in Fig. 6. The following variations are considered, variations 1) in wrist and external camera poses; 2) object poses; 3) in visual features, including background and table textures; and 4) in lighting, including saturation and hue. Table III illustrates the results for all experiments.

Visual and Lighting variations. We vary the lighting conditions via color temperature shifts, lighting exposure and strong directional light that generates shadows as the robot is moving. **Lighting:** VLAs were robust to changes in lighting conditions, with 90–100% success across shadow variations, color temperature shifts, and 500$\times$ intensity changes. **Visual appearance:** Variations over 10 background textures and 4 table textures had minimal impact ($<$5% degradation), suggesting generalization to scene appearance changes.

Camera variation sensitivity analysis. We infer posteriors over camera displacement conditioned on task success (Fig. 8).

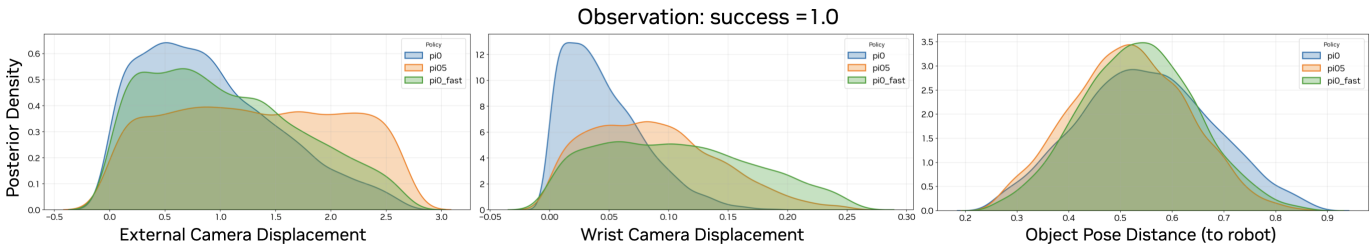

Fig. 8: Results of the sensitivity analysis using MNPE. Policies were highly sensitive to wrist-camera displacement from the nominal pose, indicating strong dependence on wrist-mounted camera calibration. Success also peaked for objects placed at approximately 0.5m from the robot, likely due to robot reachability.

TABLE IV: Overall success rates across real and simulation environments across 6 selected *simple* tasks.

Environment	$\pi_{0.5}$	π_0**-FAST**	π_0	**PaliGemma**
Real	79.5	34.1	63.2	0.0
Sim	74.0	42.0	18.0	4.0

Camera poses were randomized in both orientation and position for 10 episodes each. Displacement is calculated with respect to the nominal position of the cameras. Across all policies, the wrist-camera posterior is sharply concentrated near zero, indicating that successful execution often required the wrist camera to remain close to its nominal pose, while performance is more tolerant to external camera position changes. This indicates performance is critically dependent on wrist camera than external camera.

Object pose variation sensitivity analysis. We randomize initial object poses via a uniform distribution of 10cm, 20cm, and 30cm within its nominal placement (usually in front of the robot) for 10 episodes each. We then infer posteriors over initial object poses conditioned on task success (see Fig. 8), relative to the robot pose. We observe a strong peak over 0.5m from the robot's origin, suggesting that objects placed at this distance has the highest probability of success, likely due to reachability.

E. Real Robot Verification

We evaluated the same policies on a small set of six simple real-robot tasks and compared success rates to matched simulation evaluations (Table IV). $\pi_{0.5}$ achieved 79.5% success in the real world, close to its 74.0% success in simulation, suggesting that RoboLab can provide a reasonable proxy for this policy on these task types. π_0-FAST achieved 34.1% success in the real world and 42.0% in simulation, showing a similar trend. π_0 was a notable outlier, reaching 63.2% success on the real robot but only 18.0% in simulation; qualitatively, this policy appeared tuned to reliably grasp single objects, which matched the selected real-robot tasks. We leave deeper investigation of policy-specific sim-to-real deviations to future work.

V. Limitations

While RoboLab provides a flexible and scalable framework for evaluating language-conditioned manipulation, it currently focuses on rigid-body tabletop scenes and does not fully capture the challenges of deformable object manipulation (e.g., cloth, cables, bags). Moreover, many contact-rich skills that require precise force control, compliant interaction, or complex frictional dynamics are underrepresented and dependent on the physics simulation fidelity, limiting Robolab's coverage of fine-grained, low-level control tasks. Finally, although evaluation in high-fidelity simulation is a strong proxy for real-world performance, a residual visual distribution shift remains. This gap needs to be characterized further both by analyzing the behavior and robustness of the visual perception stack and through extensive validation on real-world deployments.

VI. Conclusion

Recent benchmarking efforts have made significant strides in scalable robot evaluation, but they primarily assess robustness to perturbations of training environments rather than true task generalization to novel scenarios. RoboLab addresses this gap by evaluating real world policies in a high-fidelity simulation, structured evaluation vectors that decompose policy competence into visual, procedural, and relational dimensions, and a set of sensitivity analysis set of novel analysis that provides insight into policy behavior for robotics. Our benchmarking framework enables the community to critically answer the question of *generalization* and *performance*. At the same time, the framework is designed to be pragmatically usable: new tasks can be authored in minutes by arranging objects on a tabletop and attaching language instructions, and a generative scene–task–environment workflow that supports continuous benchmark evolution.

References

[1] Pranav Atreya, Karl Pertsch, Tony Lee, Moo Jin Kim, Arhan Jain, Artur Kuramshin, Clemens Eppner, Cyrus Neary, Edward Hu, Fabio Ramos, et al. Roboarena: Distributed real-world evaluation of generalist robot policies. In *Proceedings of the Conference on Robot Learning (CoRL 2025)*, 2025.

[2] Sivakumar Balasubramanian, Alejandro Melendez-Calderon, and Etienne Burdet. A robust and sensitive metric for quantifying movement smoothness. *IEEE Transactions on Biomedical Engineering*, 59(8): 2126–2136, 2012. doi: 10.1109/TBME.2011.2179545.

[3] Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Shangchen Han, Fan Zhang, Linguang Zhang, Jade Fountain, Edward Miller, Selen Basol, Richard Newcombe, Robert Wang, Jakob Julian Engel, and Tomas Hodan. HOT3D: Hand and object tracking in 3D from egocentric multi-view videos. *CVPR*, 2025.

[4] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, Julian Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bošnjak, Xi Chen, Matthias Minderer, Paul Voigtlaender, Ioana Bica, Ivana Balazevic, Joan Puigcerver, Pinelopi Papalampidi, Olivier Henaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiaohua Zhai. Paligemma: A versatile 3b vlm for transfer, 2024. URL https://arxiv.org/abs/2407.07726.

[5] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2026. URL https://arxiv.org/abs/2410.24164.

[6] Rishit Dagli, Donglai Xiang, Vismay Modi, Charles Loop, Clement Fuji Tsang, Anka He Chen, Anita Hu, Gavriel State, David I.W. Levin, and Maria Shugrina. Vomp: Predicting volumetric mechanical property fields. *arXiv preprint*, 2025.

[7] Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, et al. Maniskill2: A unified benchmark for generalizable manipulation skills. *arXiv preprint arXiv:2302.04659*, 2023.

[8] Andrew Guo, Bowen Wen, Jianhe Yuan, Jonathan Tremblay, Stephen Tyree, Jeffrey Smith, and Stan Birchfield. HANDAL: A dataset of real-world manipulable object categories with pose annotations, affordances, and reconstructions. In *IROS*, 2023.

[9] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. URL https://arxiv.org/abs/2504.16054.

[10] Arhan Jain, Mingtong Zhang, Kanav Arora, William Chen, Marcel Torne, Muhammad Zubair Irshad, Sergey Zakharov, Yue Wang, Sergey Levine, Chelsea Finn, Wei-Chiu Ma, Dhruv Shah, Abhishek Gupta, and Karl Pertsch. Polaris: Scalable real-to-sim evaluations for generalist robot policies, 2025. URL https://arxiv.org/abs/2512.16881.

[11] Stephen James, Zicong Ma, David R. Arrojo, and Andrew J. Davison. RLBench: The Robot Learning Benchmark & Learning Environment. *RAL*, 2020.

[12] Yash Jangir, Yidi Zhang, Kashu Yamazaki, Chenyu Zhang, Kuan-Hsun Tu, Tsung-Wei Ke, Lei Ke, Yonatan Bisk, and Katerina Fragkiadaki. Robotarena ∞: Scalable robot benchmarking via real-to-sim translation, 2025. URL https://arxiv.org/abs/2510.23571.

[13] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, Peter David Fagan, Joey Hejna, Masha Itkina, Marion Lepert, Yecheng Jason Ma, Patrick Tree Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Beltran Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag R Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Rich Walke, Blake Wulfe, Ted Xiao, Jonathan Heewon Yang, Arefeh Yavary, Tony Z. Zhao, Christopher Agia, Rohan Baijal, Mateo Guaman Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Paul Foster, Jensen Gao, Vitor Guizilini, David Antonio Herrera, Minho Heo, Kyle Hsu, Jiaheng Hu, Muhammad Zubair Irshad, Donovon Jackson, Charlotte Le, Yunshuang Li, Kevin Lin, Roy Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew E. Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph J Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael C. Yip, Yuke Zhu, Thomas Kollar, Sergey Levine, and Chelsea Finn. Droid: A large-scale in-the-wild robot manipulation dataset. 2024.

[14] Hadas Kress-Gazit, Kunimatsu Hashimoto, Naveen Kuppuswamy, Paarth Shah, Phoebe Horgan, Gordon Richardson, Siyuan Feng, and Benjamin Burchfiel. Robot learning as an empirical science: Best practices for policy evaluation, 2024. URL https://arxiv.org/abs/2409.09491.

[15] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Wensi Ai, Benjamin Martinez, et al. Behavior-1k: A human-centered, embodied ai benchmark with 1,000 everyday activities and realistic simulation.

arXiv preprint arXiv:2403.09227, 2024.

[16] Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation. *arXiv preprint arXiv:2405.05941*, 2024.

[17] Yunzhi Lin, Jonathan Tremblay, Stephen Tyree, Patricio A. Vela, and Stan Birchfield. Multi-view fusion for multi-level robotic scene understanding. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6817–6824, 2021. doi: 10.1109/IROS51168.2021.9635994.

[18] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation, 2024. URL https://arxiv.org/abs/2404.01291.

[19] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

[20] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. CALVIN: A Benchmark for Language-Conditioned Policy Learning for Long-Horizon Robot Manipulation Tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

[21] Mayank Mittal, Pascal Roth, James Tigue, Antoine Richard, Octi Zhang, Peter Du, Antonio Serrano-Muñoz, Xinjie Yao, René Zurbrügg, Nikita Rudin, Lukasz Wawrzyniak, Milad Rakhsha, Alain Denzler, Eric Heiden, Ales Borovicka, Ossama Ahmed, Iretiayo Akinola, Abrar Anwar, Mark T. Carlson, Ji Yuan Feng, Animesh Garg, Renato Gasoto, Lionel Gulich, Yijie Guo, M. Gussert, Alex Hansen, Mihir Kulkarni, Chenran Li, Wei Liu, Viktor Makoviychuk, Grzegorz Malczyk, Hammad Mazhar, Masoud Moghani, Adithyavairavan Murali, Michael Noseworthy, Alexander Poddubny, Nathan Ratliff, Welf Rehberg, Clemens Schwarke, Ritvik Singh, James Latham Smith, Bingjie Tang, Ruchik Thaker, Matthew Trepte, Karl Van Wyk, Fangzhou Yu, Alex Millane, Vikram Ramasamy, Remo Steiner, Sangeeta Subramanian, Clemens Volk, CY Chen, Neel Jawale, Ashwin Varghese Kuruttukulam, Michael A. Lin, Ajay Mandlekar, Karsten Patzwaldt, John Welsh, Huihua Zhao, Fatima Anes, Jean-Francois Lafleche, Nicolas Moënne-Loccoz, Soowan Park, Rob Stepinski, Dirk Van Gelder, Chris Amevor, Jan Carius, Jumyung Chang, Anka He Chen, Pablo de Heras Ciechomski, Gilles Daviet, Mohammad Mohajerani, Julia von Muralt, Viktor Reutskyy, Michael Sauter, Simon Schirm, Eric L. Shi, Pierre Terdiman, Kenny Vilella, Tobias Widmer, Gordon Yeoman, Tiffany Chen, Sergey Grizan, Cathy Li, Lotus Li, Connor Smith, Rafael Wiltz, Kostas Alexis, Yan Chang, David Chu, Linxi "Jim" Fan, Farbod Farshidian, Ankur Handa, Spencer Huang, Marco Hutter, Yashraj Narang, Soha Pouya, Shiwei Sheng, Yuke Zhu, Miles Macklin, Adam Moravanszky, Philipp Reist, Yunrong Guo, David Hoeller, and Gavriel State. Isaac Lab: A GPU-accelerated simulation framework for multi-modal robot learning. *arXiv preprint arXiv:2511.04831*, 2025. URL https://arxiv.org/abs/2511.04831.

[22] Nicolas Moenne-Loccoz, Ashkan Mirzaei, Or Perel, Riccardo de Lutio, Janick Martinez Esturo, Gavriel State, Sanja Fidler, Nicholas Sharp, and Zan Gojcic. 3d gaussian ray tracing: Fast tracing of particle scenes. *ACM Transactions on Graphics and SIGGRAPH Asia*, 2024.

[23] NVIDIA. Isaac Sim. URL https://github.com/isaac-sim/IsaacSim.

[24] NVIDIA, Johan Bjorck, Nikita Cherniadev Fernando Castañeda, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An open foundation model for generalist humanoid robots. In *ArXiv Preprint*, March 2025.

[25] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish

Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

[26] OpenAI, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Quiñonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singhal, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov, Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon

Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. Openai o1 system card, 2024. URL https://arxiv.org/abs/2412.16720.

[27] Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models, 2025. URL https://arxiv.org/abs/2501.09747.

[28] Mohammad Nomaan Qureshi, Sparsh Garg, Francisco Yandun, David Held, George Kantor, and Abhishesh Silwal. Splatsim: Zero-shot sim2real transfer of rgb manipulation policies using gaussian splatting, 2024. URL https://arxiv.org/abs/2409.10161.

[29] Jonathan Tremblay, Aayush Prakash, David Acuna, Mark Brophy, Varun Jampani, Cem Anil, Thang To, Eric Cameracci, Shaad Boochoon, and Stan Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 969–977, 2018.

[30] Qi Wu, Janick Martinez Esturo, Ashkan Mirzaei, Nicolas Moenne-Loccoz, and Zan Gojcic. 3dgut: Enabling distorted cameras and secondary rays in gaussian splatting. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[31] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. 2018.

[32] Yandan Yang, Baoxiong Jia, Shujie Zhang, and Siyuan Huang. Sceneweaver: All-in-one 3d scene synthesis with an extensible and self-reflective agent, 2025. URL https://arxiv.org/abs/2509.20414.

[33] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Avnish Narayan, Hayden Shively, Adithya Bellathur, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning, 2021. URL https://arxiv.org/abs/1910.10897.

[34] Kaifeng Zhang, Shuo Sha, Hanxiao Jiang, Matthew Loper, Hyunjong Song, Guangyan Cai, Zhuo Xu, Xiaochen Hu, Changxi Zheng, and Yunzhu Li. Real-to-sim robot policy evaluation with gaussian splatting simulation of soft-body interactions. *arXiv preprint arXiv:2511.04665*, 2025.

[35] Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. Libero-pro: Towards robust and fair evaluation of vision-language-action models beyond memorization. *[arXiv preprint arXiv:2510.03827]*, 2025.

[36] Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. In *arXiv preprint arXiv:2009.12293*, 2020.

[37] Alex Zook, Fan-Yun Sun, Josef Spjut, Valts Blukis, Stan Birchfield, and Jonathan Tremblay. Grs: Generating robotic simulation tasks from real-world images. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 594–603, 2025.

APPENDIX A
DETAILS ON THE ROBOLAB BENCHMARK

In this section we provide detail on the benchmark.

RoboLab provides a set of ∼300 object assets from well-known 3D pose estimation benchmarks; including YCB [31], HOT3D [3], HOPE [17], HANDAL [8], and VoMP [6]. Each asset contains a visual and collision mesh, with mass and friction properties added. Each object has a language description and an object label attached to it. This forms the catalog of objects used in the scenes, in either manual or LLM-scaled scene environments.

RoboLab the RoboLab-120 benchmark contains 120 manually generated tasks. These tasks span one or more categories. More details on the benchmark implementation will be provided in the code repository, which will be open sourced.

We provide additional details on the results reported in the paper. Please refer to Table V for expanded results on overall performance on our benchmark; Table VI for more details on the language ablation; Table VII for details on the real-sim verification experiments; and Tables VIII-XII for detailed per-task results for each policy.

APPENDIX B
DETAILS OF MNPE SENSITIVITY ANALYSIS

MNPE allows us to analyze the relationship between scene parameters and policy outcomes in a likelihood-free Bayesian inference setting.

Variable Definitions. Let $\theta \in \Theta$ denote the vector of variation parameters. In our camera pose sensitivity experiments, $\theta = (d_{\text{ext}}, d_{\text{wrist}}) \in \mathbb{R}^2$ where d_{ext} and d_{wrist} represent the displacement of the external and wrist cameras from their reference configurations, respectively, in SE(3).

Let $x \in \{0, 1\}$ denote the binary task success indicator, and let π denote the robot policy being evaluated.

Handling Mixed Parameters. For experiments involving both continuous parameters (e.g., pose distances) and discrete parameters (e.g., lighting levels, table materials), MNPE handles mixed continuous-discrete parameters through factorization: $q_\phi(\theta \mid x) = q_\phi(\theta^{\text{cont}} \mid \theta^{\text{disc}}, x) \cdot q_\phi(\theta^{\text{disc}} \mid x)$, where discrete components use softmax distributions and continuous components use normalizing flows. In our camera pose experiments, all parameters are continuous.

Pose Distance Metric. Poses are represented as 7-DoF transformations $\mathbf{T} = (\mathbf{p}, \mathbf{q})$ comprising position $\mathbf{p} \in \mathbb{R}^3$ and unit quaternion orientation $\mathbf{q} \in \mathbb{H}$. We compute a weighted distance from the reference configuration:

$$d(\mathbf{T}, \mathbf{T}_{\text{ref}}) = \|\mathbf{p} - \mathbf{p}_{\text{ref}}\|_2 + \beta \cdot d_{\text{SO(3)}}(\mathbf{q}, \mathbf{q}_{\text{ref}}), \tag{2}$$

where the geodesic distance on SO(3) is:

$$d_{\text{SO(3)}}(\mathbf{q}_1, \mathbf{q}_2) = 2 \arccos\left(\min(1, |\mathbf{q}_1 \cdot \mathbf{q}_2|)\right). \tag{3}$$

The weighting factor $\beta = 1.0$ balances translational (meters) and rotational (radians) components. For camera displacement, reference poses correspond to nominal camera mounting positions. For object pose, reference pose is the origin of the robot base.

Prior Specification. We adopt non-informative uniform priors to avoid biasing the inference toward any particular parameter region. For continuous parameters normalized to the unit interval:

$$p(\theta) = \prod_{j=1}^{m} \text{Uniform}(0, 1) = 1, \quad \theta_j \in [0, 1]. \tag{4}$$

Training Objective. The neural network parameters ϕ are optimized by minimizing the negative log-likelihood over the training dataset, $\mathcal{D} = \{(\theta_i, x_i)\}_{i=1}^N$:

$$\mathcal{L}(\phi) = -\frac{1}{N} \sum_{i=1}^{N} \log q_\phi(\theta_i \mid x_i). \tag{5}$$

We train for 50 epochs using the Adam optimizer on data collected from the camera pose variation and initial pose variation experiments (see Table III).

Importance Sampling Correction. Since the experimental data may sample parameters non-uniformly, we apply importance sampling to recover the posterior under a uniform prior: $p(\theta \mid x) \approx \frac{p(\theta)}{\tilde{p}(\theta)} q_\phi(\theta \mid x)$, where $\tilde{p}(\theta)$ is the empirical proposal distribution. We correct posterior samples using importance weights:

$$w_i = \frac{p(\theta_i)}{\tilde{p}(\theta_i)}, \tag{6}$$

where $\tilde{p}(\theta)$ is estimated via Gaussian kernel density estimation on the training data. The effective sample size $\text{ESS} = 1/\sum_i \bar{w}_i^2$ quantifies the efficiency of this correction.

Posterior Inference. Given a query observation x_o (e.g., $x_o = 1$ for successful task completion), we draw $N_s = 5000$ samples from the learned posterior:

$$\left\{\theta^{(i)}\right\}_{i=1}^{N_s} \sim q_\phi(\theta \mid x_o). \tag{7}$$

Posterior Statistics. For each continuous parameter, we compute the posterior mean and 95% credible interval:

$$\hat{\mu}_j = \frac{1}{N_s} \sum_{i=1}^{N_s} \theta_j^{(i)}, \tag{8}$$

$$\text{CI}_{95\%}^{(j)} = \left[Q_{0.025}\left(\{\theta_j^{(i)}\}\right),\ Q_{0.975}\left(\{\theta_j^{(i)}\}\right)\right], \tag{9}$$

where $Q_\alpha(\cdot)$ denotes the α-quantile.

This analysis reveals which variation parameters are most strongly associated with successful task outcomes: a posterior distribution tightly concentrated near zero indicates high sensitivity to that parameter (the policy requires it to remain near the reference value), while a broad posterior indicates robustness to variation.

APPENDIX C
DETAILS ON SCALING SCENE GENERATION

We present additional implementation details on scene generation (Section III-D1).

TABLE V: Expanded results of Table I: Overall performance of VLAs on RoboLab (**Best viewed with zoom**).

Task Categories	#	$\pi_{0.5}$				π_0-FAST				GR00T N1.6				π_0			
		Succ %	Score	SPARC	Speed (cm/s)	Succ %	Score	SPARC	Speed (cm/s)	Succ %	Score	SPARC	Speed (cm/s)	Succ %	Score	SPARC	Speed (cm/s)
Total	120	25.0%	0.00	−10.11 (±7.05)	5.7 (±1.8)	15.4%	0.00	−9.51 (±6.88)	4.6 (±1.8)	8.0%	0.09	−6.44 (±2.04)	4.2 (±1.3)	5.3%	0.00	−9.48 (±4.07)	4.4 (±1.4)
simple	64	26.7%	0.00	−8.46 (±5.73)	5.8 (±1.7)	21.1%	0.00	−8.00 (±6.63)	4.6 (±1.7)	12.3%	0.12	−5.79 (±1.96)	4.3 (±1.5)	7.7%	0.00	−8.52 (±4.31)	4.5 (±1.4)
moderate	39	27.7%	0.01	−10.87 (±8.35)	5.9 (±1.9)	11.3%	0.01	−10.17 (±5.89)	4.7 (±2.0)	7.1%	0.08	−6.70 (±1.91)	4.1 (±1.2)	3.7%	0.00	−10.36 (±3.41)	4.3 (±1.4)
complex	17	12.4%	0.01	−14.55 (±6.06)	5.0 (±1.6)	3.5%	0.00	−13.67 (±7.93)	4.1 (±1.3)	0.0%	0.07	−7.36 (±1.99)	4.3 (±1.1)	0.0%	0.00	−11.67 (±3.94)	4.1 (±1.2)
Procedural	34	17.9%	0.01	−12.85 (±8.28)	5.1 (±1.7)	3.8%	0.00	−11.04 (±6.31)	4.5 (±1.5)	0.0%	0.06	−6.35 (±1.47)	3.9 (±1.2)	0.3%	0.00	−10.77 (±3.43)	4.2 (±1.2)
affordance	12	15.0%	0.00	−13.02 (±10.43)	4.8 (±1.7)	1.7%	0.00	−10.73 (±4.68)	4.1 (±1.5)	0.0%	0.00	−5.39 (±1.59)	4.6 (±1.1)	0.0%	0.00	−10.59 (±3.01)	4.5 (±1.5)
reorientation	6	15.0%	0.07	−14.94 (±9.48)	4.7 (±1.5)	8.3%	0.03	−11.35 (±6.43)	4.7 (±1.5)	0.0%	0.03	−6.89 (±1.44)	3.2 (±0.6)	1.7%	0.01	−12.06 (±2.30)	4.0 (±0.7)
stacking	6	26.7%	0.00	−10.04 (±4.68)	6.2 (±1.5)	1.7%	0.00	−8.82 (±2.46)	5.1 (±1.5)	0.0%	0.09	−6.23 (±1.35)	4.2 (±1.3)	0.0%	0.00	−9.52 (±3.35)	3.8 (±0.9)
Relational	42	32.6%	0.00	−9.09 (±5.84)	6.1 (±1.8)	24.8%	0.00	−8.73 (±6.71)	5.1 (±2.0)	11.2%	0.07	−6.30 (±2.22)	4.2 (±1.4)	7.4%	0.00	−9.26 (±3.46)	4.7 (±1.5)
conjunction	8	48.8%	0.00	−8.23 (±7.00)	6.3 (±1.5)	30.0%	0.00	−7.57 (±4.15)	5.4 (±1.7)	4.0%	0.13	−6.54 (±2.13)	3.7 (±1.1)	17.5%	0.00	−7.90 (±2.63)	4.8 (±1.2)
counting	7	57.1%	0.00	−9.54 (±9.70)	7.4 (±2.4)	42.9%	0.00	−10.04 (±6.83)	5.5 (±2.3)	38.0%	0.14	−6.75 (±2.59)	4.3 (±1.5)	10.0%	0.00	−11.60 (±4.90)	4.2 (±1.3)
spatial	29	20.0%	0.00	−9.21 (±3.97)	5.7 (±1.5)	17.2%	0.00	−8.74 (±7.19)	4.9 (±2.0)	7.0%	0.04	−6.15 (±2.15)	4.3 (±1.4)	3.4%	0.00	−9.07 (±2.95)	4.8 (±1.7)
Visual	84	18.2%	0.00	−10.63 (±6.35)	5.5 (±1.8)	8.9%	0.00	−10.43 (±7.81)	4.2 (±1.5)	5.8%	0.10	−6.73 (±2.05)	4.2 (±1.2)	2.3%	0.00	−9.63 (±3.73)	4.2 (±1.4)
color	26	14.6%	0.01	−10.34 (±6.02)	5.5 (±1.6)	7.3%	0.00	−9.15 (±6.72)	4.6 (±1.5)	5.6%	0.13	−6.05 (±2.00)	4.2 (±1.3)	2.7%	0.00	−9.28 (±3.05)	4.2 (±1.4)
semantics	60	19.3%	0.00	−10.87 (±6.51)	5.6 (±1.8)	11.3%	0.00	−11.16 (±7.72)	4.1 (±1.6)	5.9%	0.09	−7.10 (±2.02)	4.3 (±1.2)	3.1%	0.00	−9.95 (±4.03)	4.3 (±1.4)
size	6	13.3%	0.00	−9.43 (±6.08)	4.7 (±2.1)	8.3%	0.00	−8.66 (±11.42)	4.2 (±1.3)	0.0%	0.00	−6.51 (±1.34)	2.5 (±0.3)	0.0%	0.00	−8.01 (±2.59)	3.7 (±0.9)

Fig. 9: (*Left*) We show one of our Gaussian Splat + Mesh scenes in RoboLab. This scene has a Gaussian splat background with a collision mesh for the splat estimated with 3DGRUT [22, 30], and a mesh foreground. All objects in the scene have spatially varying density, and thus mass is estimated with VoMP [6]. (*Right*) We show a VLA running a task in this scene.

A. Stage I: Predicates for Semantic Planning

The following predicates are used:

- `PlaceIn`(x, y): Object x must be contained within y (e.g., fruit in a bowl).
- `PlaceOn`(x, y): Object x is supported by y (e.g., mug on a coaster).
- `ClusterAround`$(x, \{y_i\})$: Object x acts as an anchor for a group $\{B_i\}$.
- `PlaceAnywhere`(x): Object x is placed freely on the global support surface (table).

If a predicate refers to a non-existent anchor, it is downgraded to a `PlaceAnywhere` constraint to preserve the object in the scene.

B. Stage II: Geometric Constraint Solving

For **Global Placement** (`PlaceAnywhere`), we utilize rejection sampling on the global table surface bounds, checking collision against all currently placed objects using SAT on OBBs. To handle high-density scenes, we employ two strategies: (1) an adaptive relaxation loop that progressively increases collision margins if a valid layout is not found, and (2) a stochastic perturbation step that randomly jitters all object positions when the solver converges to a local minimum (Algorithm 1).

For **Stacking** (`PlaceOn`$(b_i, b_{support})$), we sample positions on the top surface of $b_{support}$ using rejection sampling (up to $K = 20$ attempts) to find a position $\mathbf{p}_{xy}$ such that $\text{OBB}(b_i) \cap \text{OBB}(b_{existing}) = \emptyset$ for all previously placed objects on the same support (Algorithm 2).

For **Containment** (`PlaceIn`$(b_i, b_{container})$), we compute the available interior volume of $b_{container}$ using its bounding box dimensions $\mathbf{d}_b$. We employ a packing heuristic that discretizes the container's floor into a grid with resolution $s = \max(\mathbf{d}_i^x, \mathbf{d}_i^y) + \epsilon_{margin}$. A cell (u, v) is considered valid if it is unoccupied and within the container's bounds scaled by factor $\gamma = 0.7$ to avoid edge collisions. We assign o_i to the first valid cell, setting its height $z_i = z_{container} + h_{container}/2$.

C. Baseline Method

To validate the efficacy of our hierarchical approach, we implement a robust baseline inspired by standard domain randomization techniques. The baseline operates in a single pass without iterative feedback. The process begins with the LLM selecting a list of objects O and suggesting a grid layout (rows $R\times$ columns C) for the table surface. The table surface is then divided into $R \times C$ rectangular cells, and objects are assigned to cells sequentially. Within each cell k, the object's position is jittered uniformly: $\mathbf{p}_{xy} \sim \mathcal{U}(\text{center}_k - w/4, \text{center}_k + w/4)$. This ensures basic separation but precludes complex stacking or containment, as objects are simply placed at a safe height $z = z_{table} + h_{obj}/2$. Finally, we run the same physics simulation pass as in our method to allow objects to settle under gravity, resolving minor inter-penetrations but without the capability to correct semantic or structural failures.

D. Experiments

We compare our scene generation with the baseline method using popular scene generation metrics, VQA score [18], GPT preference where it is shown two images each from the baseline or our method and is asked to pick one, and following Yang et al. [32], we report the visual realism (Real.), functionality (Func.), layout correctness (Lay.), Quality (Qual.) and scene completeness (Comp.) scores. We use GPT-4o [25] to generate scenes from our method and baseline; and use GPT-4.1 [25] for the evaluations. These metrics are computed on rendered RGB images from two viewpoints: a frontal view aligned with the table axis (camera at $(1.0, 0.0, 0.7)$ looking at table center) and an angled perspective view (camera at $(-0.3, 0.3, 0.7)$).

TABLE VI: Expanded results of Table II: Language understanding ablations.(a) VLA performance degrades with abstract or vague language instructions. π_0, PaliGemma and GR00T N1.6 excluded as no tasks were able to be completed. (b) Performance drops as scene complexity increases. (c) VLAs show brittle language grounding, with consistent object confusion patterns across different instructions in the same scene.

(a) Effect of language specificity on task performance.

	$\pi_{0.5}$							π_0-FAST						
Task	Succ %	Score	Time (s)	Wrong Obj	SPARC	l (m)	v (cm/s)	Succ %	Score	Time (s)	Wrong Obj	SPARC	l (m)	v (cm/s)
Bananas Out Of Bin Task														
"Take all the bananas out of the grey bin and put it on the table."	50	0.13	26.96	18	-10.54	2.26	5.7	30	0.05	34.29	4	-6.46	2.71	5.3
"Take the bananas out"	40	0.22	38.77	14	-9.90	2.63	5.2	10	0.15	24.53	3	-9.46	2.89	5.0
"Empty the grey bin"	10	0.07	79.47	120	-13.02	4.06	4.4	70	0.11	51.25	14	-7.46	3.79	6.2
White Mugs In Bin Task														
"Put the white mugs in the grey bin"	80	0.50	44.97	0	-6.50	3.39	7.1	20	0.22	54.00	7	-7.39	3.92	6.3
"Put the mugs in the bin"	90	0.50	35.13	1	-5.79	3.22	8.4	10	0.11	51.80	22	-6.74	4.28	7.0
"Put away mugs"	0	0.00	-	0	-7.51	4.64	7.3	0	0.00	-	53	-7.20	4.05	6.3
Remove Measuring Spoons from the Plate Task														
"Put the orange measuring cup and the blue measuring cup outside of the plate"	20	0.47	95.17	25	-17.91	4.97	3.0	0	0.31	-	5	-12.75	10.20	5.4
"Clear the plate"	0	0.08	-	15	-13.43	9.76	5.1	0	0.10	-	0	-23.86	4.93	2.5

(b) Effect of scene complexity on task performance.

	$\pi_{0.5}$		π_0-FAST		π_0		**GR00T N1.6**	
Scene	Succ %	Wrong object grabbed	Succ %	Wrong object grabbed	Succ %	Wrong object grabbed	Succ %	Wrong object grabbed
Task: *"Pack boxed foods into the bin"*								
1 Box/Can	10	-	0	soup can	0	soup can	0	soup can, mustard
2 Boxes/Cans	0	-	0	soup can	0	soup can	0	soup can, bin
3 Boxes/Cans	0	spam can, soup can	0	soup can	0	soup can, spam can	0	spam can, soup can
Task: *"Pack canned foods into the bin"*								
1 Box/Can	70	bin	30	-	0	-	0	-
2 Boxes/Cans	30	bin	10	-	0	-	0	-
3 Boxes/Cans	20	bin, pudding box	0	-	0	pudding box	0	-

(c) Instruction sensitivity within fixed scenes.

	$\pi_{0.5}$		π_0-FAST		π_0		**GR00T N1.6**	
Task	Succ %	Wrong object grabbed	Succ %	Wrong object grabbed	Succ %	Wrong object grabbed	Succ %	Wrong object grabbed
Fruit Plate Scene								
"Move an orange or a lime to the wood bowl"	50	pumpkin, redonion	0	-	0	-	0	redonion, wooden_bowl
"Move an orange to the white bowl"	0	pumpkin	0	-	0	pumpkin, wooden_bowl	0	redonion, wooden_bowl, pomegranate
"Put the onion in the wood bowl"	70	pumpkin, wooden_bowl	10	-	20	-	20	wooden_bowl, pumpkin, storage_box
"Put the onion on the plate"	0	pumpkin	0	-	0	wooden_bowl, lime	0	wooden_bowl, pumpkin
Tools Cleanup Scene								
"Put hammers in the right bin and ignore everything else"	20	right_bin, drill	0	drill	0	drill	0	-
"Put hammers in the left bin"	10	left_bin	0	drill	0	drill	0	spring_clamp
Tools Selection Scene								
"Take out all the hammers and put it on the table"	0	clamp, left_bin	0	-	0	drill, center_bin, right_bin	0	left_bin, center_bin
"Select the cordless drill and put it on the table"	70	red_hammer, wood_hammer, blue_hammer	50	clamp	20	center_bin, right_bin	30	blue_hammer, red_hammer, clamp
"Select the blue hammer and put it on the table"	0	left_bin	0	red_hammer	10	center_bin, right_bin, left_bin	0	left_bin, center_bin

We show quantitative comparisons across 100 generated scenes for our method compared to the baselines in Table XIII. We show the quantitative comparisons split by the number of objects in the scene ($[1, 5]$ objects, $[6, 15]$ objects, and $[16, 20]$ objects) in Table XIV. We show the quantitative comparisons split across the 10 scene themes we use in Table XV. We find our method consistently outperforms the baseline across all metrics, with particularly large gains in visual realism and semantic functionality.

APPENDIX D
DETAILS ON TASK GENERATION EVALUATION

We evaluated the quality of our task generation method using an LLM-as-judge framework. Tasks were generated by prompting an LLM (o1 [26]) with scene descriptions and our Category templates[1]. For each generated task, we extracted the natural language instruction and the corresponding termination conditions (success criteria implemented as predicate functions) through static analysis of the generated Python code. We then prompted an LLM (also o1 [26]) to assess alignment between the instruction and the programmatic success conditions across six dimensions: relation match (whether the spatial/logical relationship is preserved), target match (correctness of goal state), object match (whether referenced objects are correct), quantifier match (handling of "all," "any," or specific counts), instruction clarity (unambiguous and well-formed language), and physical feasibility (whether the task is achievable given typical robot capabilities). Each dimension was scored on a 0–1 scale, and we computed an aggregate alignment score as the

[1]For 2 simple scenes, we generated 1 task for each of the 7 categories and for the remaining 57 scenes, we generated 2 tasks. This produced $57 * 7 * 2 + 2 * 7 * 1 = 812$ tasks

TABLE VII: Expanded results of Table IV: Comparison of success rates (%) between real and simulation environments per task.

		Success %			
Task	**Env**	$\pi_{0.5}$	π_0**-FAST**	π_0	**PaliGemma**
BananaInBowl	Real	90.9	80.0	80.0	-
	Sim	100.0	70.0	20.0	0.0
BananaAndCubeInBowl	Real	100.0	30.0	90.0	0.0
	Sim	90.0	60.0	50.0	0.0
BananasOutOfBin	Real	83.3	60.0	100.0	-
	Sim	50.0	30.0	0.0	0.0
FoodPacking2Cans	Real	40.0	0.0	40.0	-
	Sim	-	-	-	-
PickOranges	Real	33.3	0.0	40.0	-
	Sim	60.0	0.0	0.0	0.0
ToolsPickingDrill	Real	100.0	0.0	0.0	0.0
	Sim	70.0	50.0	20.0	20.0
Total	Real	79.5	34.1	63.2	0.0
	Sim	74.0	42.0	18.0	4.0

weighted mean. The model additionally provided a categorical verdict—aligned, partially aligned, or misaligned—based on whether the termination conditions would correctly evaluate task success as described in the instruction.

Table XVI shows that our method can successfully generate a variety of types of tasks appropriate to the assets in the scene. The **Alignment** score represents the overall instruction-code alignment, aggregating the six sub-dimensions. **Clarity** measures whether instructions are unambiguous and grammatically well-formed. **Feasibility** assesses physical realizability of the task. **Match** combines the four semantic dimensions (relation, target, object, quantifier) into a single score reflecting how accurately the code captures the instruction's intent. **Verdict** reports the percentage of tasks judged as fully aligned versus partially aligned (misaligned tasks, comprising approximately 1% of the dataset, are omitted for brevity). We additionally compute scene coverage metrics: object coverage measures the fraction of manipulable objects in each scene that appear in at least one generated task, while predicate coverage measures the fraction of available termination predicates used across tasks for that scene. All evaluations use temperature 0 for reproducibility, with automatic retry logic to handle rate limits.

The evaluation reveals strong overall task generation quality, with 0.91 mean alignment and 76% of tasks receiving full alignment verdicts. Performance varies by category: conjunction and recognition tasks achieve the highest alignment (0.97 and 0.96), likely because their success conditions map straightforwardly to compositional predicates, while color-based tasks show lower alignment (0.81), reflecting the challenge of grounding color references to specific object instances. High clarity (0.96) and semantic match (0.95) scores indicate that the generated instructions are well-formed and the termination conditions capture the intended semantics, though feasibility scores are slightly lower for spatial tasks (0.89) where precise placement requirements may exceed typical manipulation tolerances. The 88% object coverage demonstrates good utilization of scene assets, while the lower predicate coverage (29%) suggests the generator favors a subset of reliable predicates rather than exploring the full space of available success conditions—a conservative strategy that likely contributes to the high alignment scores. Overall, these results demonstrate our method can successfully generate a variety of types of tasks appropriate to the assets in the scene.

Algorithm 1 Spatial Constraint Solver

Input: Objects B, Predicates P, Table Bounds L_{max}
Output: 2D coordinates (x, y, θ) for all base objects

Margins $M \leftarrow [\mu, 1.25\mu, 1.5\mu, 2.0\mu]$
for all margin $\in M$ **do**
  {Phase 1: Initialization}
  Randomize (x, y) for all loose objects inside L_{max}
  for all $p \in P$ **do**
    if p.type == place-on-base **then**
      p.object.$(x, y, \theta) \leftarrow (p.x, p.y, p.\text{yaw})$
    else if p.type == cluster-around **then**
      PolarPlace(p.targets, p.anchor, p.radius)
    end if
  end for
  {Phase 2: Relative Constraints}
  while constraints not satisfied **do**
    ApplyRelativeConstraints(P)
  end while
  ApplyOrientations(P)
  {Phase 3: Collision Resolution}
  for $k = 1$ **to** K_{max} **do**
    $C \leftarrow$ FindCollisions(B, margin)
    if $C = \emptyset$ **then**
      return Success
    end if
    if $|C|$ not decreasing for 10 steps **then**
      PerturbPositions(B)
    end if
    for all $(o_i, o_j) \in C$ **do**
      ResolveOverlap(b_i, b_j, margin)
      ClampToBounds(b_i, b_j, L_{max})
    end for
  end for
end for
return Failure

Algorithm 2 Physical Placement Solver

Input: Objects B, Predicates P, Solved Base Poses
Output: 3D coordinates (x, y, z) for all objects

{Solve Stacking:}
for all $p \in P$ where p.type $==$ place-on **do**
  $s \leftarrow p$.support
  $B_{\text{peers}} \leftarrow \{b' \mid b' \text{ is already on } s\}$
  $(x, y) \leftarrow \text{FindSpot}(s, p.\text{object}, B_{\text{peers}})$
  $p.\text{object}.z \leftarrow s.z + s.\text{height} + p.\text{object.height}/2$
  $p.\text{object}.(x, y) \leftarrow (x, y)$
end for
{Solve Containment:}
for all $p \in P$ where p.type $==$ place-in **do**
  $c \leftarrow p$.container
  if $\text{TotalArea}(p.\text{objects}) > 0.8 \times \text{Area}(c)$ **then**
    $p.\text{objects} \leftarrow \text{SortAndFilter}(p.\text{objects}, c.\text{capacity})$
  end if
  $(R, C) \leftarrow \text{ComputeGridDimensions}(c.\text{dims}, |p.\text{objects}|)$
  for $i = 0$ **to** $|p.\text{objects}| - 1$ **do**
    $(r, c) \leftarrow (i//C, i\%C)$
    $(x_{\text{loc}}, y_{\text{loc}}) \leftarrow \text{GridCellCenter}(r, c, c.\text{dims})$
    $\text{Jitter}(x_{\text{loc}}, y_{\text{loc}})$
    $p.\text{objects}[i].(x, y) \leftarrow c.(x, y) + (x_{\text{loc}}, y_{\text{loc}})$
    $p.\text{objects}[i].z \leftarrow c.z + c.\text{height}/2 + \text{buffer}$
  end for
end for
return Success

TABLE VIII: **Detailed results for $\pi_{0.5}$.**

Task Name	Succ%	Score	Time(s)	SPARC	PathLen(m)	Speed(cm/s)	WrongObjNames
TOTAL (120 tasks)	25.0	0.003	27.55 ± 28.49	−10.11 ± 7.05	5.58 ± 10.13	5.7 ± 1.8	
AnimalsInBinTask	0.0	0.000	-	−9.62 ± 2.85	4.58 ± 1.25	5.1 ± 1.4	-
AppleAndYogurtInBowlTask	80.0	0.000	48.43 ± 23.22	−20.74 ± 33.62	21.15 ± 49.40	6.2 ± 1.3	-
BBQSauceInBinTask	0.0	0.000	-	−8.98 ± 1.76	4.08 ± 1.19	4.4 ± 1.2	-
BagelsOnPlateTask	0.0	0.000	-	−8.30 ± 2.31	2.07 ± 0.43	4.2 ± 0.5	-
BananaInBowlTask	100.0	-	14.95 ± 8.96	−4.50 ± 0.84	0.97 ± 0.37	6.6 ± 1.0	-
BananaOnPlateTask	100.0	-	10.43 ± 1.99	−4.10 ± 1.01	0.74 ± 0.06	6.9 ± 1.2	-
BananaThenRubiksCubeTask	70.0	0.000	27.54 ± 8.97	−9.44 ± 10.73	7.43 ± 16.68	6.2 ± 0.5	-
BananasInBinOneMoreTask	100.0	-	10.45 ± 3.24	−4.26 ± 1.19	0.98 ± 0.22	9.3 ± 1.4	-
BananasInBinThreeTotalTask	100.0	-	12.35 ± 5.17	−4.14 ± 0.85	1.11 ± 0.40	8.9 ± 1.1	-
BananasInCrateTask	90.0	0.000	11.03 ± 4.75	−9.87 ± 18.52	9.59 ± 26.92	9.4 ± 2.1	-
BananasOutOfBinTask	90.0	0.000	41.70 ± 9.24	−9.07 ± 2.61	2.67 ± 0.84	5.6 ± 0.8	-
BigPumpkinInBinTask	0.0	0.000	-	−10.15 ± 4.22	2.21 ± 0.97	3.5 ± 1.5	-
BlackItemsInBinTask	0.0	0.000	-	−12.21 ± 2.52	4.55 ± 0.72	3.8 ± 0.5	-
BlockStackingOrderAgnosticTask	30.0	0.000	62.84 ± 15.40	−11.33 ± 4.82	9.40 ± 14.17	6.0 ± 1.5	-
BlockStackingSpecifiedOrderTask	0.0	0.000	-	−12.01 ± 2.77	4.80 ± 0.99	5.0 ± 1.1	-
BlocksInBinTask	0.0	0.000	-	−10.66 ± 3.20	9.55 ± 1.24	6.1 ± 0.8	-
BowlInBinTask	60.0	0.000	28.43 ± 13.32	−12.21 ± 11.78	4.06 ± 7.57	4.6 ± 1.4	-
BowlStackingLeftOnRightTask	10.0	0.000	15.33	−6.34 ± 1.31	1.25 ± 0.31	6.0 ± 1.3	-
BowlStackingRightOnLeftTask	30.0	0.000	14.00 ± 2.43	−6.73 ± 1.96	1.06 ± 0.21	5.6 ± 1.4	-
ButterAboveRaisinTask	0.0	0.000	-	−9.53 ± 2.02	1.92 ± 0.34	4.7 ± 0.8	-
CannedFoodInBinTask	40.0	0.000	30.80 ± 13.63	−12.25 ± 12.39	4.65 ± 6.91	5.0 ± 0.9	-
ClampInRightBinTask	0.0	0.000	-	−7.70 ± 1.48	3.28 ± 0.75	5.2 ± 1.2	-
CleanUpToysTask	0.0	0.000	-	−14.94 ± 1.46	19.68 ± 1.31	6.3 ± 0.4	-
ClearOrganicObjectsTask	0.0	0.000	-	−13.93 ± 1.79	18.60 ± 2.27	7.4 ± 0.8	-
ClutterPlasticTask	10.0	0.000	178.73	−13.05 ± 3.23	11.30 ± 2.27	6.1 ± 1.2	-
ClutterPumpkinTask	0.0	0.000	-	−10.78 ± 2.54	3.73 ± 1.47	4.3 ± 1.3	-
CoffeePotInBinTask	0.0	0.000	-	−9.58 ± 2.06	1.79 ± 0.54	3.0 ± 0.8	-
CondimentsInBinTask	0.0	0.000	-	−17.09 ± 6.06	5.38 ± 2.10	2.9 ± 1.1	-
CookingClearPlateTask	0.0	0.000	-	−18.86 ± 6.10	5.20 ± 2.17	2.8 ± 1.2	-
CookingPickPastaToolTask	0.0	0.000	-	−8.90 ± 1.20	4.18 ± 0.71	6.5 ± 1.1	-
CubesAndBlocksInBinTask	10.0	0.000	127.27	−12.86 ± 2.85	18.51 ± 13.27	5.9 ± 1.0	-
DishesInBinTask	20.0	0.000	132.00 ± 9.99	−13.96 ± 1.91	13.36 ± 4.96	6.7 ± 0.6	-
ElectronicsInBinTask	10.0	0.000	64.60	−14.90 ± 5.29	11.49 ± 14.91	4.3 ± 1.2	-
FoodPacking1BoxesTask	10.0	0.000	27.47	−7.40 ± 1.17	5.18 ± 4.65	6.5 ± 1.2	-
FoodPacking1CansTask	60.0	0.000	28.13 ± 14.07	−6.49 ± 1.50	2.89 ± 1.04	7.3 ± 1.4	-
FoodPacking2BoxesTask	0.0	0.000	-	−13.39 ± 2.31	9.95 ± 1.95	5.3 ± 1.0	-
FoodPacking2CansTask	30.0	0.000	139.09 ± 38.78	−14.15 ± 4.11	12.40 ± 8.32	4.9 ± 0.7	-
FoodPacking3BoxesTask	0.0	0.000	-	−14.86 ± 2.98	10.43 ± 3.22	4.4 ± 1.1	-
FoodPacking3CansTask	10.0	0.000	117.60	−14.06 ± 3.25	22.17 ± 27.30	5.6 ± 1.6	-
FoodPackingByColorTask	0.0	0.000	-	−10.39 ± 2.38	7.24 ± 1.76	5.8 ± 1.3	-
FruitsGreenLimesOnPlateTask	30.0	0.000	73.69 ± 13.51	−10.90 ± 2.36	5.02 ± 3.92	4.4 ± 0.7	-
FruitsMovingOrangeOrLimeTask	70.0	0.000	18.92 ± 10.44	−5.40 ± 2.39	1.55 ± 0.67	6.3 ± 1.8	-
FruitsMovingTask	10.0	0.000	49.67	−8.89 ± 3.05	3.57 ± 1.23	5.8 ± 1.9	-
FruitsOnPlate3Task	30.0	0.000	81.71 ± 88.70	−18.49 ± 11.74	18.86 ± 24.73	4.3 ± 1.6	-
FruitsOnPlateTask	0.0	0.000	-	−18.01 ± 4.10	12.82 ± 2.90	4.2 ± 1.0	-
FruitsOnionTask	60.0	0.000	16.87 ± 6.94	−11.00 ± 17.16	7.03 ± 16.65	6.4 ± 1.6	-
FruitsOnionToPlateTask	60.0	0.000	29.06 ± 12.55	−11.54 ± 13.22	5.88 ± 11.34	5.6 ± 1.3	-
FruitsOrangesOnPlateTask	80.0	0.000	10.58 ± 8.37	−4.36 ± 2.13	1.69 ± 1.80	7.6 ± 1.5	-
GrabABagelTask	0.0	0.000	-	−6.11 ± 0.41	2.07 ± 0.32	6.5 ± 1.0	-
GrabAFruitTask	0.0	0.000	-	−7.02 ± 0.91	1.72 ± 0.11	5.4 ± 0.3	-
GreenSpoonsInPotTask	0.0	0.000	-	−22.14 ± 4.24	4.79 ± 1.97	2.8 ± 1.0	-
HammersInLeftBinTask	0.0	0.000	-	−11.68 ± 2.34	8.06 ± 2.03	4.7 ± 1.0	-
JugsOnShelfTask	0.0	0.000	-	−10.23 ± 2.42	9.49 ± 2.32	7.5 ± 1.8	-
KeyboardOutOfBinTask	0.0	0.000	-	−8.38 ± 1.45	2.40 ± 0.83	3.8 ± 1.3	-
LargerObjectRaisinBoxInBinTask	0.0	0.000	-	−8.26 ± 3.20	1.15 ± 0.49	3.9 ± 1.4	-
MarkerInMugTask	0.0	0.000	-	−10.64 ± 1.99	1.13 ± 0.27	2.7 ± 0.7	-
MouseOnKeyboardTask	50.0	0.000	32.03 ± 19.35	−14.21 ± 13.27	5.22 ± 11.17	4.4 ± 1.1	-
MoveBananaToBagelPlateTask	0.0	0.000	-	−10.75 ± 1.44	5.83 ± 0.66	6.0 ± 0.7	-
MustardAboveRaisinTask	100.0	-	9.42 ± 1.24	−3.38 ± 0.48	0.57 ± 0.04	5.7 ± 0.4	-
MustardInLeftBinTask	50.0	0.000	15.65 ± 8.47	−4.58 ± 0.81	1.46 ± 0.55	6.6 ± 1.6	-
MustardInRightBinTask	100.0	-	7.80 ± 1.58	−4.09 ± 1.07	0.70 ± 0.08	8.6 ± 1.3	-
NonHammerToolsInRightBinTask	0.0	0.000	-	−16.02 ± 4.18	7.27 ± 1.31	4.1 ± 0.6	-
OneBottleInSquarePailTask	100.0	-	11.49 ± 3.62	−4.43 ± 1.17	0.91 ± 0.20	7.6 ± 0.8	-
OneBottleOnShelfTask	0.0	0.000	-	−6.79 ± 1.22	5.63 ± 0.69	8.9 ± 1.1	-
PhoneOrRemoteInBinTask	20.0	0.000	4.10 ± 0.24	−13.12 ± 14.33	3.09 ± 4.88	4.8 ± 2.1	-
PickDrillTask	0.0	0.000	-	−8.68 ± 1.24	2.45 ± 0.21	5.8 ± 0.5	-
PickGlassesTask	0.0	0.000	-	−5.74 ± 0.81	2.70 ± 0.35	8.5 ± 1.2	-
PickOrangeObjectTask	10.0	0.000	14.40	−9.28 ± 2.61	5.13 ± 4.47	6.4 ± 0.6	-
PickUpBluePitcherTask	0.0	0.000	-	−8.39 ± 1.53	1.54 ± 0.17	4.9 ± 0.5	-
PickUpGreenObjectTask	0.0	0.000	-	−8.54 ± 1.48	1.58 ± 0.23	4.9 ± 0.7	-
PinkSpoonInPotTask	50.0	0.000	30.36 ± 11.56	−9.15 ± 5.00	5.12 ± 9.53	5.4 ± 1.6	-
PlasticBottlesInSquarePailTask	30.0	0.000	52.44 ± 13.22	−14.20 ± 12.52	26.68 ± 56.46	6.2 ± 2.0	-
PutBowlOnShelfTopTask	0.0	0.000	-	−9.14 ± 1.32	3.64 ± 0.76	5.7 ± 1.1	-
PutMugsOnShelfTask	0.0	0.000	-	−11.73 ± 2.71	11.27 ± 2.66	6.1 ± 1.4	-
PutTwoMugsOnShelfTask	0.0	0.000	-	−13.98 ± 4.14	10.44 ± 3.30	5.8 ± 1.7	-
RecycleCartonTask	30.0	0.000	46.44 ± 15.70	−10.82 ± 8.60	7.58 ± 9.87	6.0 ± 1.2	-
RecycleCartonsOnBoxTask	0.0	0.000	-	−8.75 ± 0.70	6.45 ± 0.86	6.8 ± 0.9	-

TABLE IX: **Detailed results for π_0-FAST.**

Task Name	Succ%	Score	Time(s)	SPARC	PathLen(m)	Speed(cm/s)	WrongObjNames
TOTAL (120 tasks)	15.4	0.002	21.05 ± 15.90	−9.51 ± 6.88	4.18 ± 7.16	4.6 ± 1.8	
AnimalsInBinTask	0.0	0.000	-	−12.28 ± 3.07	3.68 ± 0.77	3.9 ± 0.8	-
AppleAndYogurtInBowlTask	10.0	0.000	26.20	−12.43 ± 5.45	7.29 ± 8.80	4.0 ± 1.3	-
BBQSauceInBinTask	0.0	0.000	-	−8.34 ± 2.45	3.37 ± 0.72	3.6 ± 0.8	-
BagelsOnPlateTask	0.0	0.000	-	−9.85 ± 2.88	1.46 ± 0.44	2.9 ± 0.6	-
BananaInBowlTask	60.0	0.000	39.19 ± 8.84	−8.13 ± 1.61	2.77 ± 4.28	3.4 ± 0.9	-
BananaOnPlateTask	80.0	0.000	16.14 ± 8.03	−5.27 ± 1.97	1.00 ± 0.32	5.2 ± 1.5	-
BananaThenRubiksCubeTask	50.0	0.000	30.84 ± 12.06	−8.05 ± 8.36	6.76 ± 12.77	6.0 ± 1.0	-
BananasInBinOneMoreTask	80.0	0.000	28.66 ± 13.89	−9.78 ± 12.17	8.27 ± 19.67	6.6 ± 1.6	-
BananasInBinThreeTotalTask	100.0	-	23.27 ± 17.44	−5.04 ± 1.57	1.60 ± 1.00	7.2 ± 1.4	-
BananasInCrateTask	100.0	-	12.50 ± 3.73	−3.64 ± 0.69	1.13 ± 0.24	8.7 ± 1.0	-
BananasOutOfBinTask	10.0	0.111	37.93	−11.91 ± 3.32	4.42 ± 3.63	3.8 ± 1.0	-
BigPumpkinInBinTask	0.0	0.000	-	−7.84 ± 1.40	2.75 ± 0.78	4.4 ± 1.3	-
BlackItemsInBinTask	0.0	0.000	-	−11.04 ± 2.65	3.76 ± 0.82	3.3 ± 0.5	-
BlockStackingOrderAgnosticTask	10.0	0.000	79.67	−8.64 ± 2.32	3.77 ± 1.02	4.0 ± 1.0	-
BlockStackingSpecifiedOrderTask	0.0	0.000	-	−8.54 ± 1.46	4.73 ± 0.91	5.0 ± 1.0	-
BlocksInBinTask	0.0	0.000	-	−11.30 ± 1.57	8.17 ± 1.34	5.2 ± 0.8	-
BowlInBinTask	90.0	0.000	26.01 ± 12.94	−5.11 ± 1.03	1.42 ± 0.70	4.8 ± 0.8	-
BowlStackingLeftOnRightTask	80.0	0.000	8.51 ± 4.80	−3.20 ± 1.14	0.78 ± 0.24	7.9 ± 2.2	-
BowlStackingRightOnLeftTask	100.0	-	9.49 ± 0.85	−2.49 ± 0.16	0.71 ± 0.04	7.1 ± 0.4	-
ButterAboveRaisinTask	0.0	0.000	-	−7.05 ± 1.36	1.57 ± 0.33	4.0 ± 0.6	-
CannedFoodInBinTask	20.0	0.000	40.83 ± 10.23	−6.26 ± 1.93	2.73 ± 0.73	4.8 ± 1.4	-
ClampInRightBinTask	0.0	0.000	-	−7.35 ± 2.06	2.12 ± 0.53	3.5 ± 0.8	-
CleanUpToysTask	0.0	0.000	-	−14.09 ± 1.18	17.95 ± 2.27	5.7 ± 0.7	-
ClearOrganicObjectsTask	0.0	0.000	-	−17.33 ± 4.91	8.19 ± 1.94	3.3 ± 0.8	-
ClutterPlasticTask	0.0	0.000	-	−21.50 ± 15.59	5.05 ± 1.51	2.7 ± 0.8	-
ClutterPumpkinTask	0.0	0.000	-	−10.11 ± 2.48	3.22 ± 0.69	3.4 ± 0.7	-
CoffeePotInBinTask	0.0	0.000	-	−8.23 ± 1.22	2.41 ± 0.40	3.8 ± 0.7	-
CondimentsInBinTask	0.0	0.000	-	−12.66 ± 1.07	6.45 ± 1.07	3.5 ± 0.6	-
CookingClearPlateTask	0.0	0.000	-	−12.47 ± 4.09	8.78 ± 1.91	4.6 ± 1.0	-
CookingPickPastaToolTask	0.0	0.000	-	−8.87 ± 1.80	3.47 ± 0.62	5.3 ± 0.9	-
CubesAndBlocksInBinTask	0.0	0.000	-	−12.50 ± 1.85	12.15 ± 3.40	4.9 ± 1.4	-
DishesInBinTask	0.0	0.000	-	−12.58 ± 2.54	10.25 ± 1.70	5.4 ± 0.9	-
ElectronicsInBinTask	0.0	0.000	-	−13.79 ± 1.87	6.35 ± 1.50	3.4 ± 0.8	-
FoodPacking1BoxesTask	0.0	0.000	-	−8.44 ± 1.82	2.34 ± 0.42	3.7 ± 0.6	-
FoodPacking1CansTask	10.0	0.000	12.40	−8.72 ± 3.39	2.78 ± 2.19	3.9 ± 1.2	-
FoodPacking2BoxesTask	0.0	0.000	-	−13.34 ± 2.21	4.96 ± 1.32	2.7 ± 0.7	-
FoodPacking2CansTask	0.0	0.000	-	−14.42 ± 3.39	4.93 ± 1.28	2.6 ± 0.7	-
FoodPacking3BoxesTask	0.0	0.000	-	−14.20 ± 7.48	7.72 ± 4.19	3.2 ± 1.7	-
FoodPacking3CansTask	0.0	0.000	-	−17.09 ± 2.65	7.40 ± 2.03	3.0 ± 0.8	-
FoodPackingByColorTask	0.0	0.000	-	−10.25 ± 2.42	5.13 ± 1.25	4.1 ± 1.0	-
FruitsGreenLimesOnPlateTask	0.0	0.000	-	−9.44 ± 1.91	3.14 ± 0.39	3.4 ± 0.4	-
FruitsMovingOrangeOrLimeTask	0.0	0.000	-	−10.06 ± 1.99	2.06 ± 0.15	3.3 ± 0.2	-
FruitsMovingTask	0.0	0.000	-	−7.87 ± 1.25	2.27 ± 0.52	3.6 ± 0.8	-
FruitsOnPlate3Task	0.0	0.000	-	−15.06 ± 1.87	6.34 ± 0.92	3.0 ± 0.5	-
FruitsOnPlateTask	0.0	0.000	-	−18.83 ± 4.88	8.34 ± 1.76	2.6 ± 0.6	-
FruitsOnionTask	40.0	0.000	23.93 ± 6.30	−5.18 ± 0.84	1.72 ± 0.47	4.1 ± 1.5	-
FruitsOnionToPlateTask	20.0	0.000	13.23 ± 0.42	−6.81 ± 2.34	1.72 ± 0.53	3.8 ± 1.6	-
FruitsOrangesOnPlateTask	20.0	0.000	24.13 ± 3.77	−13.53 ± 6.62	7.66 ± 15.22	3.7 ± 1.1	-
GrabABagelTask	0.0	0.000	-	−6.38 ± 1.41	1.06 ± 0.30	3.4 ± 0.9	-
GrabAFruitTask	0.0	0.000	-	−6.29 ± 1.40	1.42 ± 0.15	4.4 ± 0.5	-
GreenSpoonsInPotTask	0.0	0.000	-	−21.52 ± 7.96	5.11 ± 2.52	2.9 ± 1.4	-
HammersInLeftBinTask	0.0	0.000	-	−12.74 ± 1.82	6.26 ± 0.74	3.4 ± 0.4	-
JugsOnShelfTask	0.0	0.000	-	−11.30 ± 3.21	6.60 ± 1.82	5.3 ± 1.5	-
KeyboardOutOfBinTask	0.0	0.000	-	−8.43 ± 1.72	3.15 ± 0.32	4.9 ± 0.5	-
LargerObjectRaisinBoxInBinTask	0.0	0.000	-	−6.09 ± 0.78	0.89 ± 0.13	3.6 ± 0.4	-
MarkerInMugTask	0.0	0.000	-	−8.48 ± 2.42	1.38 ± 0.45	3.4 ± 1.1	-
MouseOnKeyboardTask	0.0	0.000	-	−8.61 ± 2.13	3.55 ± 0.61	5.5 ± 0.9	-
MoveBananaToBagelPlateTask	0.0	0.000	-	−14.34 ± 3.44	3.04 ± 0.61	3.1 ± 0.6	-
MustardAboveRaisinTask	90.0	0.000	11.07 ± 4.30	−13.02 ± 30.55	8.49 ± 24.66	5.8 ± 0.8	-
MustardInLeftBinTask	70.0	0.000	10.21 ± 3.33	−4.13 ± 1.38	0.97 ± 0.39	6.3 ± 1.6	-
MustardInRightBinTask	90.0	0.000	12.87 ± 5.87	−3.44 ± 0.48	0.95 ± 0.36	6.5 ± 1.1	-
NonHammerToolsInRightBinTask	0.0	0.000	-	−14.97 ± 6.92	5.34 ± 1.44	3.0 ± 0.8	-
OneBottleInSquarePailTask	90.0	0.000	13.84 ± 10.54	−10.72 ± 22.94	9.02 ± 25.31	7.5 ± 2.2	-
OneBottleOnShelfTask	0.0	0.000	-	−7.24 ± 1.54	2.95 ± 0.87	4.7 ± 1.4	-
PhoneOrRemoteInBinTask	30.0	0.000	11.24 ± 7.98	−7.18 ± 3.68	1.50 ± 0.80	4.2 ± 2.1	-
PickDrillTask	0.0	0.000	-	−7.01 ± 1.43	1.98 ± 0.37	4.7 ± 0.9	-
PickGlassesTask	0.0	0.000	-	−6.17 ± 1.33	1.81 ± 0.32	5.8 ± 0.9	-
PickOrangeObjectTask	0.0	0.000	-	−9.44 ± 1.79	2.82 ± 0.70	4.4 ± 1.0	-
PickUpBluePitcherTask	0.0	0.000	-	−6.17 ± 1.22	1.75 ± 0.40	5.3 ± 1.2	-
PickUpGreenObjectTask	0.0	0.000	-	−5.76 ± 1.18	1.44 ± 0.27	4.5 ± 0.7	-
PinkSpoonInPotTask	0.0	0.000	-	−7.09 ± 1.14	2.89 ± 0.55	4.6 ± 0.8	-
PlasticBottlesInSquarePailTask	50.0	0.000	71.17 ± 22.29	−18.54 ± 26.76	22.00 ± 51.28	5.0 ± 1.9	-
PutBowlOnShelfTopTask	0.0	0.000	-	−7.38 ± 2.36	1.93 ± 0.77	3.5 ± 1.1	-
PutMugsOnShelfTask	0.0	0.000	-	−12.80 ± 3.38	7.91 ± 2.25	4.3 ± 1.2	-
PutTwoMugsOnShelfTask	0.0	0.000	-	−10.75 ± 4.96	8.68 ± 4.03	4.8 ± 2.1	-
RecycleCartonTask	0.0	0.000	-	−11.09 ± 2.68	2.30 ± 0.46	2.4 ± 0.5	-

TABLE X: **Detailed results for π_0.**

Task Name	Succ%	Score	Time(s)	SPARC	PathLen(m)	Speed(cm/s)	WrongObjNames
TOTAL (120 tasks)	5.3	0.001	23.09 ± 16.87	−9.48 ± 4.07	4.24 ± 3.92	4.4 ± 1.4	
AnimalsInBinTask	0.0	0.000	-	−7.29 ± 1.11	5.20 ± 0.76	5.5 ± 0.8	-
AppleAndYogurtInBowlTask	0.0	0.000	-	−8.17 ± 2.47	6.74 ± 1.71	5.3 ± 1.4	-
BBQSauceInBinTask	0.0	0.000	-	−7.25 ± 1.05	4.18 ± 0.73	4.5 ± 0.8	-
BagelsOnPlateTask	0.0	0.000	-	−6.30 ± 1.00	2.96 ± 0.65	4.5 ± 1.0	-
BananaInBowlTask	0.0	0.000	-	−6.81 ± 1.00	2.30 ± 0.26	4.5 ± 0.5	-
BananaOnPlateTask	70.0	0.000	25.59 ± 9.52	−12.71 ± 17.94	6.12 ± 15.21	4.7 ± 1.4	-
BananaThenRubiksCubeTask	10.0	0.000	47.47	−9.06 ± 2.73	3.10 ± 0.63	5.1 ± 1.1	-
BananasInBinOneMoreTask	0.0	0.000	-	−9.46 ± 1.41	2.58 ± 0.42	4.1 ± 0.7	-
BananasInBinThreeTotalTask	20.0	0.000	47.73 ± 4.15	−9.94 ± 1.89	2.36 ± 0.18	4.0 ± 0.7	-
BananasInCrateTask	50.0	0.000	33.97 ± 21.15	−9.00 ± 8.63	3.78 ± 4.62	5.5 ± 1.7	-
BananasOutOfBinTask	0.0	0.083	-	−11.48 ± 1.99	4.95 ± 1.39	5.3 ± 1.5	-
BigPumpkinInBinTask	0.0	0.000	-	−7.56 ± 0.98	2.27 ± 0.40	3.6 ± 0.6	-
BlackItemsInBinTask	0.0	0.000	-	−7.96 ± 1.94	5.96 ± 1.73	4.7 ± 1.3	-
BlockStackingOrderAgnosticTask	0.0	0.000	-	−7.63 ± 0.96	3.01 ± 0.65	3.2 ± 0.7	-
BlockStackingSpecifiedOrderTask	0.0	0.000	-	−7.90 ± 0.84	2.74 ± 0.27	2.9 ± 0.3	-
BlocksInBinTask	0.0	0.000	-	−8.98 ± 1.56	7.51 ± 0.83	4.8 ± 0.5	-
BowlInBinTask	0.0	0.000	-	−5.68 ± 0.60	3.96 ± 0.14	6.2 ± 0.2	-
BowlStackingLeftOnRightTask	0.0	0.000	-	−7.13 ± 1.17	1.11 ± 0.27	5.2 ± 1.2	-
BowlStackingRightOnLeftTask	0.0	0.000	-	−5.43 ± 1.22	1.39 ± 0.17	6.6 ± 0.8	-
ButterAboveRaisinTask	0.0	0.000	-	−6.36 ± 1.06	2.58 ± 0.31	6.1 ± 0.7	-
CannedFoodInBinTask	0.0	0.000	-	−5.94 ± 0.76	2.64 ± 0.40	4.3 ± 0.6	-
ClampInRightBinTask	0.0	0.000	-	−6.98 ± 1.71	2.15 ± 0.71	3.5 ± 1.1	-
CleanUpToysTask	0.0	0.000	-	−18.35 ± 1.86	15.55 ± 2.01	4.8 ± 0.6	-
ClearOrganicObjectsTask	0.0	0.000	-	−14.38 ± 2.24	8.88 ± 0.86	3.5 ± 0.3	-
ClutterPlasticTask	0.0	0.000	-	−11.23 ± 2.08	6.76 ± 0.60	3.7 ± 0.3	-
ClutterPumpkinTask	0.0	0.000	-	−8.04 ± 0.99	3.69 ± 0.66	4.0 ± 0.7	-
CoffeePotInBinTask	0.0	0.000	-	−5.61 ± 0.95	2.84 ± 0.71	4.6 ± 1.2	-
CondimentsInBinTask	0.0	0.000	-	−9.71 ± 2.21	9.36 ± 1.73	5.0 ± 0.9	-
CookingClearPlateTask	0.0	0.000	-	−15.81 ± 3.49	7.91 ± 0.74	4.2 ± 0.3	-
CookingPickPastaToolTask	0.0	0.000	-	−10.52 ± 1.01	3.07 ± 0.52	4.7 ± 0.8	-
CubesAndBlocksInBinTask	0.0	0.000	-	−10.66 ± 3.87	11.11 ± 1.72	4.5 ± 0.7	-
DishesInBinTask	0.0	0.000	-	−12.25 ± 1.91	8.24 ± 1.78	4.4 ± 1.0	-
ElectronicsInBinTask	0.0	0.000	-	−10.63 ± 2.83	7.26 ± 1.88	3.8 ± 1.0	-
FoodPacking1BoxesTask	0.0	0.000	-	−7.07 ± 1.15	2.96 ± 0.43	4.6 ± 0.7	-
FoodPacking1CansTask	0.0	0.000	-	−7.83 ± 0.89	1.89 ± 0.25	3.0 ± 0.4	-
FoodPacking2BoxesTask	0.0	0.000	-	−11.44 ± 1.50	7.17 ± 1.00	3.8 ± 0.5	-
FoodPacking2CansTask	0.0	0.000	-	−12.78 ± 2.42	5.59 ± 0.70	3.0 ± 0.4	-
FoodPacking3BoxesTask	0.0	0.000	-	−10.85 ± 2.56	10.02 ± 1.51	4.0 ± 0.6	-
FoodPacking3CansTask	0.0	0.000	-	−11.95 ± 2.60	8.75 ± 1.08	3.5 ± 0.4	-
FoodPackingByColorTask	0.0	0.000	-	−9.99 ± 1.81	4.31 ± 0.49	3.4 ± 0.4	-
FruitsGreenLimesOnPlateTask	0.0	0.000	-	−8.75 ± 1.15	2.99 ± 0.23	3.1 ± 0.2	-
FruitsMovingOrangeOrLimeTask	0.0	0.000	-	−7.08 ± 0.71	2.85 ± 0.24	4.6 ± 0.4	-
FruitsMovingTask	0.0	0.000	-	−7.07 ± 0.46	3.39 ± 0.30	5.5 ± 0.5	-
FruitsOnPlate3Task	0.0	0.000	-	−11.04 ± 1.74	7.30 ± 0.59	3.5 ± 0.3	-
FruitsOnPlateTask	0.0	0.000	-	−13.38 ± 3.04	10.44 ± 0.60	3.3 ± 0.2	-
FruitsOnionTask	20.0	0.000	42.47 ± 11.22	−7.79 ± 1.59	2.97 ± 0.36	5.1 ± 0.6	-
FruitsOnionToPlateTask	0.0	0.000	-	−7.75 ± 1.19	2.65 ± 0.22	4.2 ± 0.3	-
FruitsOrangesOnPlateTask	0.0	-	-	−19.87 ± 0.79	25.16 ± 3.69	3.1 ± 0.5	-
GrabABagelTask	0.0	0.000	-	−8.29 ± 1.01	1.37 ± 0.12	4.2 ± 0.4	-
GrabAFruitTask	0.0	0.000	-	−7.79 ± 0.82	1.42 ± 0.18	4.4 ± 0.5	-
GreenSpoonsInPotTask	0.0	0.000	-	−14.29 ± 3.25	5.73 ± 1.10	3.0 ± 0.6	-
HammersInLeftBinTask	0.0	0.000	-	−10.88 ± 2.48	5.71 ± 1.03	3.2 ± 0.5	-
JugsOnShelfTask	0.0	0.000	-	−8.75 ± 1.13	7.21 ± 1.27	5.8 ± 1.0	-
KeyboardOutOfBinTask	0.0	0.000	-	−12.61 ± 1.47	1.91 ± 0.37	2.9 ± 0.6	-
LargerObjectRaisinBoxInBinTask	0.0	0.000	-	−5.97 ± 1.01	1.11 ± 0.26	3.5 ± 0.8	-
MarkerInMugTask	0.0	0.000	-	−9.16 ± 2.36	1.30 ± 0.29	3.1 ± 0.7	-
MouseOnKeyboardTask	0.0	0.000	-	−9.17 ± 3.48	2.42 ± 0.91	3.8 ± 1.4	-
MoveBananaToBagelPlateTask	0.0	0.000	-	−15.15 ± 3.53	3.72 ± 1.06	3.8 ± 1.1	-
MustardAboveRaisinTask	0.0	0.000	-	−9.34 ± 0.85	2.13 ± 0.15	4.9 ± 0.3	-
MustardInLeftBinTask	40.0	0.000	11.53 ± 3.82	−4.98 ± 0.87	1.48 ± 0.59	6.4 ± 0.9	-
MustardInRightBinTask	0.0	0.000	-	−5.23 ± 0.78	2.04 ± 0.38	6.5 ± 1.2	-
NonHammerToolsInRightBinTask	0.0	0.000	-	−12.45 ± 4.25	5.01 ± 2.00	2.8 ± 1.0	-
OneBottleInSquarePailTask	40.0	0.000	12.22 ± 6.19	−5.89 ± 2.90	2.16 ± 1.54	5.5 ± 2.3	-
OneBottleOnShelfTask	0.0	0.000	-	−7.60 ± 1.57	3.30 ± 0.94	5.2 ± 1.5	-
PhoneOrRemoteInBinTask	20.0	0.000	42.17 ± 3.72	−6.79 ± 1.14	1.65 ± 0.37	2.7 ± 0.5	-
PickDrillTask	0.0	0.000	-	−8.79 ± 0.89	1.84 ± 0.16	4.3 ± 0.4	-
PickGlassesTask	0.0	0.000	-	−10.01 ± 1.60	1.16 ± 0.18	3.6 ± 0.6	-
PickOrangeObjectTask	0.0	0.000	-	−12.00 ± 1.85	2.15 ± 0.37	3.3 ± 0.6	-
PickUpBluePitcherTask	0.0	0.000	-	−8.66 ± 1.36	0.80 ± 0.10	2.5 ± 0.3	-
PickUpGreenObjectTask	0.0	0.000	-	−8.03 ± 2.34	0.67 ± 0.13	2.1 ± 0.4	-
PinkSpoonInPotTask	0.0	0.000	-	−9.33 ± 1.98	3.00 ± 0.54	4.6 ± 0.8	-
PlasticBottlesInSquarePailTask	0.0	0.000	-	−12.01 ± 2.36	7.95 ± 2.64	4.2 ± 1.4	-
PutBowlOnShelfTopTask	0.0	0.000	-	−10.34 ± 2.44	3.32 ± 0.84	5.3 ± 1.3	-
PutMugsOnShelfTask	0.0	0.000	-	−10.53 ± 2.02	10.45 ± 2.60	5.5 ± 1.4	-
PutTwoMugsOnShelfTask	0.0	0.000	-	−11.34 ± 1.63	6.99 ± 1.38	4.0 ± 0.6	-
RecycleCartonTask	0.0	0.000	-	−8.11 ± 1.89	3.37 ± 0.53	3.6 ± 0.6	-

TABLE XI: **Detailed results for PaliGemma.**

Task Name	Succ%	Score	Time(s)	SPARC	PathLen(m)	Speed(cm/s)	WrongObjNames
TOTAL (79 tasks)	1.9	0.012	33.44 ± 17.42	−17.60 ± 11.83	1.18 ± 1.37	1.4 ± 1.2	
BagelsOnPlateTask	0.0	0.000	-	−25.17 ± 8.08	0.21 ± 0.07	0.3 ± 0.1	-
BananaInBowlTableTask	0.0	0.000	-	−6.30 ± 1.69	0.42 ± 0.27	0.8 ± 0.5	-
BananaOnPlateTableTask	0.0	0.050	-	−9.23 ± 2.46	0.67 ± 0.25	1.6 ± 0.6	-
BananasInBinOneMoreTask	70.0	0.000	41.35 ± 14.72	−10.10 ± 3.43	1.29 ± 0.27	3.2 ± 2.2	grey_bin(1)
BananasInBinThreeTotalTask	0.0	0.000	-	−13.10 ± 3.82	1.15 ± 0.23	1.9 ± 0.3	-
BananasInCrateTask	40.0	0.083	36.43 ± 7.62	−8.38 ± 4.74	2.12 ± 0.93	4.2 ± 1.8	-
BananasOutOfBinSlightlyVagueTask	0.0	0.033	-	−14.15 ± 4.80	1.45 ± 0.68	2.3 ± 1.1	grey_bin(3)
BananasOutOfBinTask	0.0	0.033	-	−21.73 ± 7.62	0.89 ± 0.58	1.4 ± 0.9	grey_bin(1)
BananasOutOfBinVagueTask	0.0	0.017	-	−18.31 ± 5.10	1.87 ± 1.04	2.0 ± 1.1	banana(1), grey_bin(1)
BlockStackingOrderAgnosticTask	0.0	0.000	-	−12.69 ± 3.86	1.65 ± 0.26	1.9 ± 0.3	-
BlockStackingSpecifiedOrderTask	0.0	0.000	-	−19.45 ± 12.84	1.40 ± 0.85	1.5 ± 0.9	-
BowlInBinTask	0.0	0.000	-	−8.67 ± 2.88	1.35 ± 0.72	2.1 ± 1.1	-
BowlStackingLeftOnRightTask	0.0	0.000	-	−6.76 ± 0.22	0.04 ± 0.00	0.2 ± 0.0	-
BowlStackingRightOnLeftTask	0.0	0.000	-	−6.68 ± 1.06	0.05 ± 0.01	0.2 ± 0.0	-
ButterAboveRaisinTask	0.0	0.000	-	−11.91 ± 3.46	0.96 ± 0.33	2.4 ± 0.8	-
ClearOrganicObjectsTask	0.0	0.000	-	−35.14 ± 10.24	1.98 ± 0.80	0.8 ± 0.3	-
ClutterPlasticTask	0.0	0.000	-	−19.77 ± 2.38	1.04 ± 0.13	0.6 ± 0.1	-
ClutterPumpkinTask	0.0	0.000	-	−12.49 ± 2.24	0.45 ± 0.17	0.5 ± 0.2	-
CookingClearPlateSpecificTask	0.0	0.375	-	−14.17 ± 4.75	8.05 ± 3.09	4.2 ± 1.5	-
CookingClearPlateVagueTask	0.0	0.125	-	−17.41 ± 7.22	2.79 ± 1.82	1.6 ± 1.1	-
CookingPickPastaToolTask	0.0	0.000	-	−27.11 ± 6.22	0.38 ± 0.10	0.6 ± 0.2	-
DishesInBinTask	0.0	0.000	-	−14.83 ± 4.73	2.63 ± 1.09	1.5 ± 0.5	-
FoodPacking1BoxesTask	0.0	0.000	-	−12.81 ± 5.17	0.58 ± 0.52	0.7 ± 0.7	-
FoodPacking1CansTask	0.0	0.000	-	−14.32 ± 4.88	0.88 ± 0.17	0.9 ± 0.2	-
FoodPacking2BoxesTask	0.0	0.000	-	−38.48 ± 15.02	1.17 ± 0.24	0.6 ± 0.1	-
FoodPacking2CansTask	0.0	0.000	-	−18.34 ± 8.70	2.39 ± 0.65	1.3 ± 0.3	-
FoodPacking3BoxesTask	0.0	0.000	-	−26.20 ± 13.96	2.45 ± 0.63	1.0 ± 0.3	-
FoodPacking3CansTask	0.0	0.000	-	−23.29 ± 8.81	2.96 ± 0.92	1.2 ± 0.3	-
FoodPackingByColorTask	0.0	0.000	-	−34.80 ± 15.02	0.84 ± 0.45	0.7 ± 0.4	-
FruitsGreenLimesOnPlateTask	0.0	0.000	-	−16.59 ± 3.00	1.17 ± 0.17	1.2 ± 0.2	-
FruitsLimesOnPlateTask	0.0	0.000	-	−12.82 ± 6.33	1.39 ± 0.46	1.6 ± 0.5	-
FruitsMovingOrangeOrLimeTask	0.0	0.000	-	−26.58 ± 10.56	0.93 ± 0.60	1.0 ± 0.6	-
FruitsMovingTask	0.0	0.000	-	−24.39 ± 6.82	0.47 ± 0.22	0.7 ± 0.3	-
FruitsOnPlate3Task	0.0	0.000	-	−16.62 ± 2.25	3.82 ± 1.46	1.9 ± 0.7	-
FruitsOnPlateTask	0.0	0.000	-	−23.04 ± 5.17	4.83 ± 1.83	1.5 ± 0.6	-
FruitsOnionTask	0.0	0.000	-	−30.73 ± 4.41	0.47 ± 0.13	0.7 ± 0.2	-
FruitsOnionToPlateTask	0.0	0.000	-	−42.76 ± 9.36	0.67 ± 0.12	0.7 ± 0.1	-
FruitsOrangesOnPlateTask	0.0	0.000	-	−10.46 ± 3.00	1.35 ± 0.30	1.5 ± 0.3	-
LargerObjectRaisinBoxInBinTask	0.0	0.000	-	−9.01 ± 3.11	0.80 ± 0.22	2.9 ± 0.8	-
MustardAboveRaisinTask	0.0	0.000	-	−22.18 ± 6.58	0.35 ± 0.14	0.8 ± 0.3	-
PickDrillTask	0.0	0.000	-	−9.25 ± 0.85	0.10 ± 0.01	0.2 ± 0.0	-
PickOrangeObjectTask	0.0	0.000	-	−10.79 ± 3.35	3.15 ± 0.51	4.9 ± 0.8	-
PickUpBluePitcherTask	0.0	0.000	-	−27.40 ± 11.71	0.23 ± 0.04	0.7 ± 0.1	-
PickUpGreenObjectTask	0.0	0.000	-	−6.05 ± 3.22	0.68 ± 0.15	2.1 ± 0.5	-
PutBowlOnShelfTopTask	0.0	0.000	-	−11.55 ± 4.80	2.12 ± 0.81	2.2 ± 0.9	-
RecycleCartonTask	0.0	0.000	-	−25.68 ± 15.83	0.34 ± 0.13	0.4 ± 0.2	-
RecycleCartonsOnBoxTask	0.0	0.000	-	−22.53 ± 18.84	1.66 ± 1.31	1.7 ± 1.4	-
RecycleCartonsVerticalCrateTask	0.0	0.000	-	−13.83 ± 6.24	0.56 ± 0.35	0.6 ± 0.4	-
RedDishesInBinTask	0.0	0.050	-	−6.10 ± 0.96	1.47 ± 0.48	2.5 ± 0.8	-
RedItemsInBinTask	0.0	0.000	-	−24.04 ± 6.85	0.17 ± 0.02	0.3 ± 0.0	-
ReorientAllMugsTask	0.0	0.000	-	−12.20 ± 1.61	1.04 ± 0.19	1.1 ± 0.2	-
ReorientRedMugTask	0.0	0.000	-	−19.78 ± 3.30	0.21 ± 0.01	0.3 ± 0.0	-
ReorientWhiteMugsTask	0.0	0.000	-	−17.99 ± 8.43	0.40 ± 0.15	0.6 ± 0.2	-
RubiksCubeAndBananaTask	0.0	0.100	-	−12.45 ± 3.61	0.78 ± 0.67	1.2 ± 1.0	-
RubiksCubeBehindBowlTask	0.0	0.000	-	−12.54 ± 7.83	0.23 ± 0.31	0.7 ± 0.9	rubiks_cube(1)
RubiksCubeInFrontOfBowlTask	0.0	0.000	-	−8.72 ± 2.54	0.09 ± 0.03	0.3 ± 0.1	-
RubiksCubeLeftOfBowlTask	0.0	0.000	-	−9.96 ± 3.20	0.58 ± 0.76	1.8 ± 2.4	rubiks_cube(4), banana(3)
RubiksCubeOrBananaTask	0.0	0.000	-	−7.55 ± 3.14	0.88 ± 0.67	2.7 ± 2.1	-
RubiksCubeRightOfBowlTask	0.0	0.000	-	−10.72 ± 2.31	0.84 ± 0.16	2.5 ± 0.5	-
RubiksCubeTask	10.0	0.000	39.67	−7.49 ± 3.28	0.76 ± 0.46	1.8 ± 1.1	-
RubiksCubeThenBananaTask	0.0	0.100	-	−14.03 ± 8.57	1.33 ± 0.86	2.1 ± 1.3	-
SauceBottlesCrateTask	0.0	0.000	-	−8.03 ± 3.30	1.06 ± 0.39	2.5 ± 0.9	-
SmallerObjectButterInBinTask	10.0	0.000	26.47	−7.79 ± 1.49	0.70 ± 0.08	2.6 ± 0.5	-
Stack3RubiksCubeTask	0.0	0.000	-	−22.98 ± 6.24	0.27 ± 0.02	0.4 ± 0.0	-
StackWhiteMugsTask	0.0	0.000	-	−17.33 ± 7.93	0.94 ± 0.40	1.6 ± 0.7	-
TakeMeasuringSpoonOutTask	0.0	0.000	-	−8.89 ± 1.24	0.14 ± 0.02	0.3 ± 0.0	-
ToolOrganizationBothTask	0.0	0.000	-	−22.23 ± 13.22	2.60 ± 1.11	1.4 ± 0.5	-
ToolOrganizationLeftTask	0.0	0.000	-	−15.34 ± 5.37	1.82 ± 0.94	1.1 ± 0.5	-
ToolOrganizationSpecificTask	0.0	0.000	-	−25.98 ± 9.19	1.91 ± 0.53	1.2 ± 0.4	-
ToolsPickingAllHammersTask	0.0	0.000	-	−57.19 ± 12.12	1.49 ± 0.48	0.6 ± 0.2	-
ToolsPickingDrillTask	20.0	0.000	0.17 ± 0.05	−24.12 ± 12.76	0.30 ± 0.17	1.0 ± 0.8	-
ToolsPickingHammerTask	0.0	0.000	-	−24.27 ± 10.95	0.44 ± 0.09	0.7 ± 0.1	-
UnstackRubiksCubeTask	0.0	0.000	-	−13.78 ± 0.69	0.22 ± 0.01	0.2 ± 0.0	-
WhiteMugInCenterOfTableTask	0.0	0.000	-	−16.69 ± 2.21	0.26 ± 0.01	0.8 ± 0.0	-
WhiteMugsInBinMoreVagueTask	0.0	0.000	-	−12.05 ± 3.45	1.37 ± 0.87	2.1 ± 1.3	banana_far(1)
WhiteMugsInBinTask	0.0	0.000	-	−38.60 ± 16.95	0.40 ± 0.15	0.6 ± 0.2	-
WhiteMugsInBinVagueTask	0.0	0.000	-	−7.83 ± 1.07	0.43 ± 0.54	0.7 ± 0.9	-
WoodSpatulaToBowlTask	0.0	0.000	-	−27.02 ± 5.46	0.51 ± 0.15	0.8 ± 0.2	-
YellowAndWhiteObjectsInBinTask	0.0	0.000	-	−12.78 ± 7.68	0.75 ± 0.52	1.2 ± 0.9	-

TABLE XII: **Detailed results for GR00T N1.6.**

Task Name	Succ%	Score	Time(s)	SPARC	PathLen(m)	Speed(cm/s)	WrongObjNames
TOTAL (79 tasks)	8.0	0.056	19.29 ± 14.27	−6.44 ± 2.04	3.77 ± 3.19	4.2 ± 1.3	
BagelsOnPlateTask	0.0	0.000	-	−5.98 ± 0.68	2.46 ± 0.23	3.9 ± 0.4	banana(2)
BananaInBowlTableTask	10.0	0.000	14.13	−6.25 ± 1.19	1.87 ± 0.46	3.9 ± 0.7	-
BananaOnPlateTableTask	0.0	0.050	-	−6.85 ± 1.31	1.15 ± 0.29	2.8 ± 0.7	-
BananasInBinOneMoreTask	50.0	0.000	26.21 ± 15.72	−8.99 ± 3.22	1.03 ± 0.23	3.2 ± 1.3	-
BananasInBinThreeTotalTask	100.0	-	19.84 ± 13.56	−3.72 ± 0.54	0.99 ± 0.31	5.6 ± 1.4	-
BananasInCrateTask	40.0	0.000	40.43 ± 17.45	−5.65 ± 1.67	2.35 ± 0.60	4.6 ± 1.4	purple_crate(2)
BananasOutOfBinSlightlyVagueTask	0.0	0.000	-	−6.97 ± 2.11	2.21 ± 0.53	3.6 ± 0.7	grey_bin(24), banana_03(16)
BananasOutOfBinTask	0.0	0.033	-	−6.34 ± 1.31	2.55 ± 1.35	4.1 ± 2.1	grey_bin(4)
BananasOutOfBinVagueTask	0.0	0.000	-	−6.29 ± 1.18	4.35 ± 0.75	4.6 ± 0.8	grey_bin(21), banana_03(2)
BlockStackingOrderAgnosticTask	0.0	0.000	-	−6.31 ± 0.97	4.32 ± 1.12	4.5 ± 1.1	yellow_block(28), blue_block(3)
BlockStackingSpecifiedOrderTask	0.0	0.000	-	−6.11 ± 1.03	5.64 ± 0.92	6.0 ± 0.9	blue_block(13)
BowlInBinTask	10.0	0.000	20.00	−4.35 ± 0.85	2.63 ± 0.91	4.7 ± 1.4	mustard(6), mug(2)
BowlStackingLeftOnRightTask	0.0	0.000	-	−3.87 ± 0.43	0.94 ± 0.09	4.5 ± 0.4	bowl_1(30)
BowlStackingRightOnLeftTask	0.0	0.000	-	−3.77 ± 0.48	0.96 ± 0.04	4.6 ± 0.3	bowl_1(36)
ButterAboveRaisinTask	0.0	0.000	-	−7.49 ± 1.12	1.39 ± 0.26	3.4 ± 0.7	raisin_box(8)
ClearOrganicObjectsTask	0.0	0.009	-	−7.41 ± 1.54	12.86 ± 1.60	5.2 ± 0.6	right_bin(1)
ClutterPlasticTask	0.0	0.000	-	−8.75 ± 0.90	11.75 ± 1.53	6.3 ± 0.8	pomegranate01(27), orange_01(1)
ClutterPumpkinTask	0.0	0.000	-	−8.21 ± 1.61	3.55 ± 0.69	3.9 ± 0.7	pomegranate01(17), orange_01(12), lime01(5), right_bin(1)
CookingClearPlateSpecificTask	0.0	0.125	-	−8.16 ± 1.32	6.09 ± 1.01	3.3 ± 0.5	storage_box_01(9), redonion(8), wooden_bowl(6), serving_spoon(6)
CookingClearPlateVagueTask	0.0	0.300	-	−6.50 ± 1.18	7.91 ± 0.69	4.2 ± 0.3	redonion(8), storage_box_01(4), wooden_bowl(3), potato_masher(1)
CookingPickPastaToolTask	0.0	0.000	-	−5.99 ± 1.77	2.15 ± 0.42	3.6 ± 0.7	storage_box_01(17), serving_spoon(15), redonion(2), serving_bowl(1)
DishesInBinTask	0.0	0.100	-	−9.59 ± 4.04	7.50 ± 2.83	4.2 ± 1.3	grey_bin(1)
FoodPacking1BoxesTask	0.0	0.000	-	−6.82 ± 1.36	4.26 ± 0.68	4.5 ± 0.7	tomato_soup_can(13), mustard(1)
FoodPacking1CansTask	0.0	0.000	-	−6.51 ± 0.52	3.73 ± 0.83	4.0 ± 0.8	-
FoodPacking2BoxesTask	0.0	0.000	-	−6.60 ± 1.48	9.15 ± 0.69	4.9 ± 0.3	tomato_soup_can(19), bin_a06(1)
FoodPacking2CansTask	0.0	0.000	-	−7.24 ± 1.45	8.17 ± 1.82	4.3 ± 1.0	-
FoodPacking3BoxesTask	0.0	0.000	-	−8.03 ± 1.09	11.95 ± 1.87	4.8 ± 0.6	spam_can(8), tomato_soup_can(3)
FoodPacking3CansTask	0.0	0.167	-	−7.29 ± 0.76	10.41 ± 1.08	4.2 ± 0.4	-
FoodPackingByColorTask	0.0	0.000	-	−5.48 ± 0.91	6.89 ± 0.73	5.5 ± 0.6	mustard(7), tomato_soup_can(3), sugar_box(1)
FruitsGreenLimesOnPlateTask	0.0	0.000	-	−9.89 ± 1.66	2.44 ± 0.72	2.6 ± 0.7	pomegranate01(2)
FruitsLimesOnPlateTask	0.0	0.000	-	−7.65 ± 2.15	3.73 ± 1.03	4.0 ± 1.1	pomegranate01(10), lemon_01(2), redonion(1)
FruitsMovingOrangeOrLimeTask	0.0	0.015	-	−5.46 ± 2.19	2.99 ± 1.53	4.3 ± 1.1	redonion(12), wooden_bowl(3)
FruitsMovingTask	0.0	0.000	-	−5.85 ± 0.85	3.16 ± 0.44	5.0 ± 0.7	redonion(20), wooden_bowl(2), pomegranate01(1)
FruitsOnPlate3Task	0.0	0.000	-	−8.37 ± 1.32	7.99 ± 1.29	3.9 ± 0.6	pumpkinlarge(7), redonion(2)
FruitsOnPlateTask	0.0	0.000	-	−9.25 ± 2.04	11.96 ± 3.33	3.9 ± 1.1	redonion(4), pumpkinlarge(2)
FruitsOnionTask	20.0	0.000	35.77 ± 22.96	−5.16 ± 1.23	2.90 ± 0.79	5.1 ± 1.2	wooden_bowl(21), pumpkinlarge(7), pumpkinsmall(6), storage_box(4), wooden_spoons(3)
FruitsOnionToPlateTask	0.0	0.050	-	−6.27 ± 1.70	4.55 ± 0.88	4.8 ± 0.9	wooden_bowl(7), pumpkinsmall(2), pumpkinlarge(1)
FruitsOrangesOnPlateTask	0.0	0.000	-	−7.04 ± 1.15	3.88 ± 1.60	4.2 ± 1.7	pomegranate01(13), redonion(2), lemon_01(1), pumpkinlarge(1)
LargerObjectRaisinBoxInBinTask	0.0	0.000	-	−6.57 ± 1.73	0.70 ± 0.12	2.4 ± 0.4	butter(1)
MustardAboveRaisinTask	0.0	0.100	-	−5.45 ± 1.28	1.92 ± 0.47	4.6 ± 1.1	-
PickDrillTask	0.0	0.000	-	−6.54 ± 1.45	1.43 ± 0.20	3.4 ± 0.5	cordless_drill(54)
PickOrangeObjectTask	0.0	0.000	-	−4.98 ± 0.80	2.94 ± 0.59	4.6 ± 0.9	redonion(8), serving_spoon(8), wooden_bowl(5), storage_box_01(4), serving_bowl(1)
PickUpBluePitcherTask	0.0	0.000	-	−5.39 ± 1.59	1.46 ± 0.36	4.6 ± 1.1	pitcher(9)
PickUpGreenObjectTask	0.0	0.000	-	−4.00 ± 1.10	1.05 ± 0.29	3.4 ± 0.9	utilityjug_a02(6)
PutBowlOnShelfTopTask	0.0	0.000	-	−8.97 ± 2.32	2.46 ± 0.39	2.7 ± 0.4	rack_l04(2)
RecycleCartonTask	0.0	0.000	-	−5.65 ± 0.87	4.09 ± 0.91	4.4 ± 1.0	ketchup_bottle(24), container_a01(24)
RecycleCartonsOnBoxTask	0.0	0.000	-	−6.30 ± 1.35	4.13 ± 0.61	4.5 ± 0.7	cubebox_a02(7), ketchup_bottle(3)
RecycleCartonsVerticalCrateTask	0.0	0.000	-	−5.72 ± 1.58	3.68 ± 0.70	4.0 ± 0.7	ketchup_bottle(37), container_a01(21)
RedDishesInBinTask	0.0	0.450	-	−5.43 ± 0.99	3.11 ± 0.46	4.9 ± 0.7	-
RedItemsInBinTask	10.0	0.250	56.53	−5.88 ± 1.35	2.92 ± 0.44	4.7 ± 0.7	grey_bin(4)
ReorientAllMugsTask	0.0	0.100	-	−7.80 ± 1.07	3.01 ± 0.38	3.3 ± 0.4	ceramic_mug(4), red_mug(2), cordless_drill(2)
ReorientRedMugTask	0.0	0.000	-	−7.06 ± 1.32	1.63 ± 0.15	2.6 ± 0.2	sideways_white_mug(3)
ReorientWhiteMugsTask	0.0	0.000	-	−5.81 ± 1.26	2.35 ± 0.32	3.7 ± 0.5	upright_white_mug(8), cordless_drill(4), red_mug(2)
RubiksCubeAndBananaTask	0.0	0.125	-	−6.85 ± 0.81	1.98 ± 0.25	3.2 ± 0.4	-
RubiksCubeBehindBowlTask	20.0	0.000	12.00 ± 3.58	−3.87 ± 0.67	1.27 ± 0.28	4.9 ± 1.5	rubiks_cube(19)
RubiksCubeInFrontOfBowlTask	10.0	0.000	22.33	−5.19 ± 0.99	1.36 ± 0.21	4.5 ± 0.9	rubiks_cube(72), banana(14), bowl(10)
RubiksCubeLeftOfBowlTask	50.0	0.000	13.61 ± 5.11	−4.67 ± 0.55	1.21 ± 0.51	5.3 ± 0.8	rubiks_cube(16), banana(3)
RubiksCubeOrBananaTask	10.0	0.028	12.07	−5.52 ± 1.26	0.97 ± 0.17	3.4 ± 0.8	-
RubiksCubeRightOfBowlTask	40.0	0.000	13.92 ± 2.89	−4.96 ± 1.39	1.54 ± 0.59	6.2 ± 1.1	rubiks_cube(22), bowl(7), banana(1)
RubiksCubeTask	100.0	-	18.23 ± 4.91	−4.69 ± 0.84	0.95 ± 0.24	5.0 ± 0.6	-
RubiksCubeThenBananaTask	10.0	0.139	57.13	−9.12 ± 1.97	1.60 ± 0.40	2.6 ± 0.7	-
SauceBottlesCrateTask	80.0	0.500	8.12 ± 1.11	−3.72 ± 1.13	0.81 ± 0.43	6.5 ± 1.5	-
SmallerObjectButterInBinTask	0.0	0.000	-	−6.45 ± 0.90	0.73 ± 0.04	2.5 ± 0.2	-
Stack3RubiksCubeTask	0.0	0.000	-	−6.03 ± 0.87	2.24 ± 0.32	3.5 ± 0.5	rubiks_cube(10), rubiks_cube_2(7), rubiks_cube_1(3)
StackWhiteMugsTask	0.0	0.000	-	−6.51 ± 2.57	1.76 ± 0.33	3.0 ± 0.5	red_mug(9)
TakeMeasuringSpoonOutTask	30.0	0.143	15.24 ± 5.36	−6.17 ± 1.96	1.36 ± 0.71	4.2 ± 1.6	bowl(9), measuring_cup(8), cordless_drill(7)
ToolOrganizationBothTask	0.0	0.000	-	−8.45 ± 1.79	10.53 ± 1.08	5.0 ± 0.5	-
ToolOrganizationLeftTask	0.0	0.000	-	−7.43 ± 1.42	7.04 ± 1.38	3.8 ± 0.7	spring_clamp(2)
ToolOrganizationSpecificTask	0.0	0.000	-	−8.38 ± 1.45	7.26 ± 0.96	3.8 ± 0.5	-
ToolsPickingAllHammersTask	0.0	0.438	-	−7.06 ± 1.03	9.43 ± 1.26	3.9 ± 0.5	left_bin(6), center_bin(1)
ToolsPickingDrillTask	30.0	0.000	0.13 ± 0.12	−7.04 ± 5.29	2.00 ± 1.41	5.4 ± 4.2	blue_hammer(11), red_hammer(3), clamp(1), husky_hammer(1)
ToolsPickingHammerTask	0.0	0.450	-	−7.09 ± 0.79	2.35 ± 0.31	3.9 ± 0.5	left_bin(8), center_bin(1)
UnstackRubiksCubeTask	0.0	0.425	-	−6.18 ± 0.68	3.57 ± 0.39	3.8 ± 0.4	rubiks_cube_bottom(5)
WhiteMugInCenterOfTableTask	10.0	0.000	29.40	−4.52 ± 0.89	1.34 ± 0.19	4.3 ± 0.6	mug(13)
WhiteMugsInBinMoreVagueTask	0.0	0.000	-	−6.98 ± 1.05	2.08 ± 0.62	3.3 ± 1.0	ketchup_bottle(4), rubiks_cube_bottom(2)
WhiteMugsInBinTask	0.0	0.212	-	−5.50 ± 0.94	2.93 ± 0.64	4.8 ± 1.1	rubiks_cube_middle(3), ketchup_bottle(2), grey_bin(1)
WhiteMugsInBinVagueTask	0.0	0.050	-	−6.92 ± 1.26	2.46 ± 0.46	4.0 ± 0.7	ketchup_bottle(2), grey_bin(2), rubiks_cube_bottom(1), rubiks_cube_middle(1)
WoodSpatulaToBowlTask	0.0	0.025	-	−5.45 ± 0.69	3.85 ± 0.53	6.2 ± 0.8	pumpkinlarge(8)
YellowAndWhiteObjectsInBinTask	0.0	0.250	-	−5.75 ± 1.73	3.05 ± 0.39	4.9 ± 0.6	grey_bin(6)

TABLE XIII: **Quantitative comparison for Scene Generation.** We evaluate our against the baseline across diverse metrics measuring the visual realism (Real.), functionality (Func.), layout correctness (Lay.), Quality (Qual.), VQA score [18], and GPT Preference.

Method	VQA (↑)	Real. (↑)	Func. (↑)	Lay. (↑)	Compl. (↑)	Qual. (↑)	# Obj (↑)	GPT Pref. (↑)
Baseline	0.398 (±0.04)	6.889 (±0.36)	6.221 (±0.58)	6.166 (±0.42)	4.687 (±0.57)	5.991 (±0.24)	13.750 (±10.52)	18.000
Ours	**0.554** (±0.03)	**8.755** (±0.27)	**8.951** (±0.33)	**7.919** (±0.28)	**8.207** (±0.40)	**8.458** (±0.25)	**26.870** (±24.90)	**82.000**

You are a scene generation expert creating REALISTIC robot manipulation scenarios.

REAL-WORLD SCENE PRINCIPLES:
1. Objects form CLUSTERS - not evenly spaced grids
2. Containers (bowls, bins) have objects INSIDE them
3. Supports (plates, trays) have objects ON TOP
4. Objects scatter naturally AROUND containers
5. Orientations VARY - not all aligned to 0°/90°

COORDINATE SYSTEM:
- Table bounds: X=[0.25 to 0.85], Y=[-0.40 to 0.40] (meters)
- Table center: (0.55, 0.0)
- Front=+X, Back=-X, Left=+Y, Right=-Y

PLACEMENT TYPES:
1. **place-on-base**: Object directly on table
{"type": "place-on-base", "object": "bowl_0", "x": 0.4, "y": 0.1, "yaw": 23}
VARY yaw angles (15, 47, 123, not just 0/90/180).
Position matters for anchors, less for loose objects.
2. **place-in**: Objects INSIDE a container
{"type": "place-in", "objects": ["apple_01", "orange_01"], "container": "bowl_0"}
Container MUST be placed first with place-on-base.
Great for fruits in bowls, items in bins.
3. **place-on**: Object ON TOP of support
{"type": "place-on", "object": "banana", "support": "plate_large", "position": "center"}
Support MUST be placed first.
position: "center", "edge", or "random"
Great for food on plates, items on trays.
4. **cluster-around**: Objects scattered NEAR an anchor
{"type": "cluster-around", "objects": ["mug", "spoon"], "anchor": "bowl_0", "radius": 0.15}
Creates natural groupings.

radius: how far from anchor (0.10–0.20m typical)

SCENE STRUCTURE (follow this pattern):
1. Place 1-2 ANCHOR objects (containers/supports) on table
2. Put objects INSIDE containers (place-in)
3. Put objects ON supports (place-on)
4. Cluster objects AROUND anchors (cluster-around)
5. Add a few LOOSE objects to fill space

REALISTIC SPACING:
- Anchors: 0.25-0.35m apart
- Clustered objects: 0.08-0.15m from anchor
- Loose objects: fill remaining space naturally

Fig. 10: System prompt for Stage I (Semantic Planning). This prompt instructs the LLM to generate physically plausible scene layouts using structured predicates rather than raw coordinates.

OUTPUT FORMAT (JSON only, no markdown):

```
{
"objects": [
{"name": "bowl_0"},
{"name": "plate_large"},
{"name": "apple_01"},
{"name": "orange_01"},
{"name": "banana"},
{"name": "mug"},
{"name": "spoon"}
],
"predicates": [
{"type": "place-on-base", "object": "bowl_0", "x": 0.40, "y": 0.15, "yaw": 23},
{"type": "place-on-base", "object": "plate_large", "x": 0.65, "y": -0.10, "yaw": 156},
{"type": "place-in", "objects": ["apple_01", "orange_01"], "container": "bowl_0"},
{"type": "place-on", "object": "banana", "support": "plate_large", "position": "center"},
{"type": "cluster-around", "objects": ["mug", "spoon"], "anchor": "bowl_0", "radius": 0.12}
]
}
```

CRITICAL RULES:

1. Object names MUST match EXACTLY from catalog
2. Containers/supports MUST be placed before objects go in/on them
3. Create INTERESTING scenes with containment, stacking, AND clustering
4. VARY yaw angles - real scenes aren't grid-aligned
5. Return ONLY valid JSON, no markdown

Fig. 11: Continued System prompt for Stage I (Semantic Planning). This prompt instructs the LLM to generate physically plausible scene layouts using structured predicates rather than raw coordinates.

TABLE XIV: **Quantitative comparison across Difficulty Splits.** We evaluate our method against the baseline on Easy, Medium, and Hard splits. Our method consistently outperforms the baseline across all difficulty levels.

Method	VQA (↑)	Real. (↑)	Func. (↑)	Lay. (↑)	Compl. (↑)	Qual. (↑)	# Obj (↑)	GPT Pref. (↑)
Easy ($[0, 5]$ objects)								
Baseline	0.458 (±0.02)	6.767 (±0.36)	6.269 (±0.57)	6.079 (±0.48)	4.737 (±0.61)	5.963 (±0.26)	6.467 (±0.52)	33.333
Ours	**0.525** (±0.05)	**8.331** (±0.24)	**8.423** (±0.27)	**7.636** (±0.21)	**7.626** (±0.25)	**8.004** (±0.09)	**12.800** (±11.54)	**66.667**
Medium ($[6, 15]$ objects)								
Baseline	0.401 (±0.02)	6.933 (±0.37)	6.223 (±0.59)	6.199 (±0.41)	4.634 (±0.56)	5.997 (±0.22)	10.957 (±2.11)	17.143
Ours	**0.561** (±0.02)	**8.779** (±0.18)	**8.996** (±0.23)	**7.932** (±0.24)	**8.235** (±0.29)	**8.485** (±0.13)	**22.414** (±19.15)	**82.857**
Hard ($[16, 20]$ objects)								
Baseline	0.326 (±0.02)	6.808 (±0.30)	6.162 (±0.59)	6.099 (±0.42)	4.883 (±0.58)	5.988 (±0.30)	34.067 (±14.89)	6.667
Ours	**0.553** (±0.03)	**9.067** (±0.13)	**9.271** (±0.17)	**8.142** (±0.27)	**8.659** (±0.25)	**8.785** (±0.07)	**61.733** (±28.80)	**93.333**

SCENE REQUEST: theme from dataset
TARGET: target object count objects

TABLE SIZE: 0.7m $\times$ 1.0m = 0.70m^2 (objects must fit with spacing!)

SIZE LIMITS (max 1-2 large objects, prefer smaller for 8+ items):
Large ($> 0.08\text{m}^2$): computed from catalog footprint
Avoid picking multiple large objects - they won't all fit!

AVAILABLE OBJECTS:
CONTAINERS (can hold objects inside): filled from catalog

SUPPORTS (can stack objects on): filled from catalog

FOOD: filled from catalog

TOOLS: filled from catalog

OTHER: filled from catalog

MEDIUM SCENE STRATEGY (10-14 objects):
- Use 1-2 containers/supports as ANCHORS
- Put 2-4 objects IN containers (place-in)
- Stack 1-2 items ON supports (place-on)
- Cluster 2-3 objects near anchors (cluster-around)
- VARY yaw angles - no grid alignment!

SUGGESTED for diversity (use only if they fit the theme): preselected objects

Fig. 12: User prompt template for Stage I (medium target count). The highlighted fields are populated at runtime (theme, target count, size warnings, catalog subsets, and diversity suggestions). Analogous strategy blocks are used for sparse (fewer than 10) and dense (15+) targets.

PREVIOUS ATTEMPT FAILED:
feedback string produced by spatial/physical solver or grammar checks

Please fix the issues. Common fixes:
- Use MORE containment (place-in) to reduce table crowding
- Use MORE stacking (place-on) to utilize vertical space
- Use clustering (cluster-around) instead of individual placements
- Select SMALLER objects if collisions persist

Fig. 13: Feedback block appended to the user prompt when spatial solving, physical placement, grammar checks, or intersection validation fails. The highlighted region is the dynamic diagnostic message.

TABLE XV: **Per-Scene Quantitative Analysis.** We report the performance breakdown across 10 distinct scene themes. Our method demonstrates robust generalization, outperforming the baseline in nearly all metrics across diverse environments.

Theme	Method	VQA (↑)	Real. (↑)	Func. (↑)	Lay. (↑)	Compl. (↑)	Qual. (↑)	# Obj (↑)	GPT Pref. (↑)
Bathroom Counter	Baseline	0.405 (±0.02)	6.901 (±0.37)	6.196 (±0.73)	6.260 (±0.26)	4.805 (±0.56)	6.041 (±0.19)	15.00 (±0.00)	10.00
	Ours	**0.564** (±0.02)	**8.913** (±0.16)	**9.159** (±0.22)	**7.967** (±0.28)	**8.276** (±0.33)	**8.579** (±0.15)	**28.50** (±18.47)	**90.00**
Classroom Supplies	Baseline	0.401 (±0.02)	6.881 (±0.31)	6.458 (±0.50)	5.931 (±0.41)	5.018 (±0.55)	6.072 (±0.17)	10.40 (±1.26)	**50.00**
	Ours	**0.562** (±0.02)	**8.790** (±0.18)	**8.990** (±0.18)	**7.842** (±0.28)	**8.406** (±0.26)	**8.507** (±0.12)	**32.90** (±23.70)	**50.00**
Craft Station	Baseline	0.399 (±0.02)	7.062 (±0.49)	6.385 (±0.56)	6.171 (±0.39)	4.590 (±0.60)	6.052 (±0.24)	9.70 (±0.48)	10.00
	Ours	**0.560** (±0.03)	**8.761** (±0.19)	**9.045** (±0.28)	**7.938** (±0.22)	**8.179** (±0.33)	**8.481** (±0.06)	**17.70** (±13.61)	**90.00**
Garage Workstation	Baseline	0.410 (±0.01)	7.032 (±0.34)	6.308 (±0.63)	6.246 (±0.47)	4.432 (±0.62)	6.005 (±0.26)	9.00 (±0.47)	0.00
	Ours	**0.566** (±0.02)	**8.796** (±0.21)	**9.004** (±0.20)	**7.881** (±0.28)	**8.207** (±0.29)	**8.472** (±0.13)	**13.00** (±11.68)	**100.00**
Garden Tools	Baseline	0.400 (±0.02)	7.018 (±0.45)	6.155 (±0.53)	6.315 (±0.47)	4.489 (±0.53)	5.994 (±0.21)	10.30 (±0.95)	30.00
	Ours	**0.561** (±0.02)	**8.778** (±0.16)	**8.949** (±0.15)	**7.950** (±0.20)	**8.175** (±0.27)	**8.463** (±0.12)	**13.80** (±11.36)	**70.00**
Kitchen Cabinet	Baseline	0.327 (±0.02)	6.862 (±0.31)	6.281 (±0.61)	6.008 (±0.39)	5.069 (±0.48)	6.055 (±0.26)	37.38 (±19.66)	12.50
	Ours	**0.554** (±0.03)	**9.052** (±0.13)	**9.313** (±0.17)	**8.070** (±0.25)	**8.710** (±0.25)	**8.786** (±0.06)	**48.88** (±25.63)	**87.50**
Laundry Sorting	Baseline	0.396 (±0.01)	6.795 (±0.24)	6.327 (±0.63)	6.269 (±0.34)	4.536 (±0.47)	5.982 (±0.21)	12.30 (±1.70)	0.00
	Ours	**0.558** (±0.03)	**8.742** (±0.17)	**8.975** (±0.26)	**8.021** (±0.16)	**8.202** (±0.28)	**8.485** (±0.12)	**35.30** (±27.34)	**100.00**
Office Desk	Baseline	0.457 (±0.02)	6.747 (±0.36)	6.183 (±0.25)	6.101 (±0.52)	4.697 (±0.59)	5.932 (±0.25)	7.00 (±0.00)	**57.14**
	Ours	**0.499** (±0.05)	**8.296** (±0.22)	**8.535** (±0.16)	**7.524** (±0.16)	**7.609** (±0.32)	**7.991** (±0.08)	**17.57** (±16.03)	42.86
Storage Room	Baseline	0.324 (±0.02)	6.747 (±0.29)	6.027 (±0.58)	6.203 (±0.45)	4.670 (±0.64)	5.912 (±0.35)	30.29 (±5.91)	0.00
	Ours	**0.552** (±0.02)	**9.085** (±0.14)	**9.224** (±0.17)	**8.224** (±0.28)	**8.601** (±0.24)	**8.783** (±0.08)	**76.43** (±26.40)	**100.00**
Tea Time	Baseline	0.458 (±0.02)	6.784 (±0.39)	6.345 (±0.77)	6.061 (±0.48)	4.773 (±0.67)	5.991 (±0.29)	6.00 (±0.00)	12.50
	Ours	**0.547** (±0.03)	**8.361** (±0.26)	**8.325** (±0.31)	**7.735** (±0.21)	**7.642** (±0.18)	**8.016** (±0.10)	**8.63** (±1.85)	**87.50**
Workshop Bench	Baseline	0.394 (±0.02)	6.838 (±0.35)	5.733 (±0.38)	6.199 (±0.47)	4.564 (±0.52)	5.833 (±0.22)	10.00 (±0.47)	20.00
	Ours	**0.556** (±0.03)	**8.675** (±0.10)	**8.846** (±0.22)	**7.928** (±0.26)	**8.200** (±0.30)	**8.412** (±0.16)	**15.70** (±10.36)	**80.00**

TABLE XVI: LLM-judged quality metrics for 812 automatically generated manipulation tasks across 59 scenes and 7 competency axes.

		LLM Judge						**Coverage**	
Category	N	Alignment	Clarity	Feasibility	Match	Aligned%	Partial%	Object	Predicate
color	116	0.81	0.94	0.80	0.90	57	40	0.88	0.29
conjunction	116	0.97	0.98	1.00	0.98	91	9	0.88	0.29
counting	116	0.87	0.97	0.90	0.92	60	38	0.88	0.29
recognition	116	0.96	0.97	0.96	0.97	85	15	0.88	0.29
semantics	116	0.89	0.95	0.94	0.94	72	27	0.88	0.29
sorting	116	0.94	0.95	0.97	0.96	86	14	0.88	0.29
spatial	116	0.92	0.98	0.89	0.95	80	17	0.88	0.29
Overall	**812**	**0.91**	**0.96**	**0.92**	**0.95**	**76**	**23**	**0.88**	**0.29**