BENCHMARKING VISION, LANGUAGE, & ACTION MODELS ON ROBOTIC LEARNING TASKS

Pranav Guruprasad[*12], **Harshvardhan Sikka**[*123], **Jaewoo Song**[*1], **Yangyue Wang**[1], **Paul Pu Liang**[4]

[1]Manifold Research
[2]Metarch.ai
[3]Georgia Tech
[4]MIT

ABSTRACT

Vision-language-action (VLA) models represent a promising direction for developing general-purpose robotic systems, demonstrating the ability to combine visual understanding, language comprehension, and action generation. However, systematic evaluation of these models across diverse robotic tasks remains limited. In this work, we present a comprehensive evaluation framework and benchmark suite for assessing VLA models. We profile three state-of-the-art VLM and VLAs —GPT-4o, OpenVLA, and JAT—across 20 diverse datasets from the Open-X-Embodiment collection, evaluating their performance on various manipulation tasks. Our analysis reveals several key insights: (1) current VLA models show significant variation in performance across different tasks and robot platforms, with GPT-4o demonstrating the most consistent performance through sophisticated prompt engineering, (2) all models struggle with complex manipulation tasks requiring multi-step planning, and (3) model performance is notably sensitive to action space characteristics and environmental factors. We release our evaluation framework and findings to facilitate systematic assessment of future VLA models and identify critical areas for improvement in the development of general-purpose robotic systems.

1 Introduction

The quest for robust, generalizable robotic systems continues to pose a fundamental challenge in machine learning and robotics research. Despite significant progress in controlled environments, current systems exhibit limited generalization beyond their training conditions. These limitations span numerous dimensions: systems fail when encountering unfamiliar task descriptions [6, 42], struggle with spatial variations in object configurations [5], perform poorly under variable lighting or occlusion [8], and show degraded performance when interacting with novel objects or in cluttered environments [45, 39]. These generalization challenges significantly hinder the deployment of learned robotic systems in unconstrained environments.

Recent breakthroughs in foundation models, especially in vision and language processing, suggest a promising path forward. These models, trained on web-scale datasets, have achieved remarkable capabilities in visual understanding [23, 33], sophisticated reasoning about interactions between objects and agents [3, 11, 43], software development [7], and cross-modal comprehension. The robust generalization exhibited by these models addresses precisely the challenges that have historically limited robotics systems. Their advanced capabilities in semantic understanding, problem-solving, and visual processing could revolutionize the development of versatile robots capable of handling diverse tasks in dynamic environments.

This approach corresponds with a broader trend in machine learning toward unified neural sequence architectures. These models demonstrate continued performance gains at the boundaries of data volume, computational resources, and model complexity [19, 15]. This pattern aligns with historical observations suggesting that general-purpose models efficiently utilizing computational resources tend to outperform specialized solutions [38]. The advantages of unified sequence

[*]equal contribution, alphabetical order. Corresponding Author: pranav@metarch.ai **Sponsored by Metarch.ai**

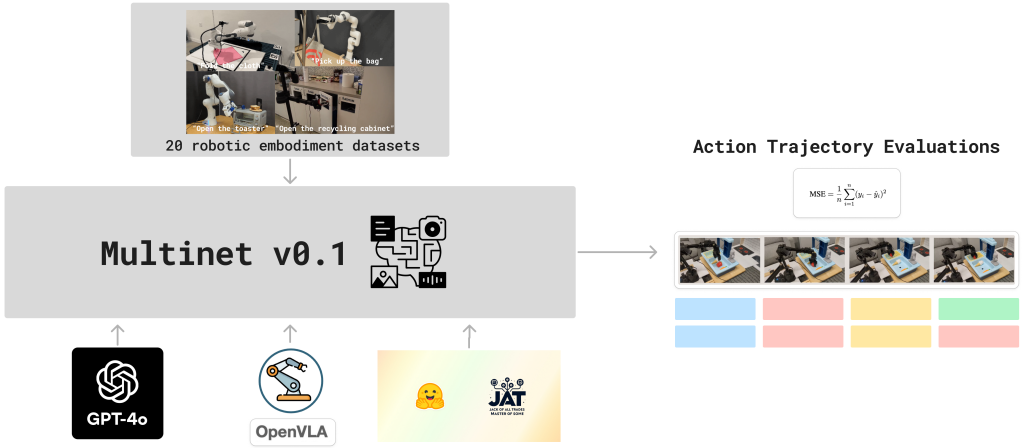

Figure 1: Multinet v0.1 overview. Benchmarking a SoTA VLM, SoTA VLA, and novel generalist model on 20 real-world robotics datasets by comparing the action predicted by the models, with the ground truth action from the dataset, at every timestep.

models are multifaceted: they remove the requirement for custom policy architectures with domain-specific assumptions, enable the use of diverse training data through sequence-based representation, and show reliable improvements with increasing scale.

Nevertheless, adapting these models for robotics applications presents substantial challenges. The vast scale of training data available for foundation models - billions of tokens and images from the internet - far exceeds what is currently feasible to collect for robot interactions [9, 21]. Moreover, while foundation models excel at abstract reasoning and high-level comprehension, robotic control requires precise, physically grounded actions, such as specific end-effector movements. Recent research has explored integrating language models (LLMs) and vision-language models (VLMs) into robotics frameworks [2][11][41]. However, many current approaches limit foundation models to high-level planning roles, using them essentially as advanced state machines that convert commands into basic actions, executed by separate low-level controllers unable to access the models' rich semantic understanding.

Current research initiatives have investigated leveraging pretrained language and vision-language models to enhance robotic representations [36, 34, 20]. These components have also been integrated into planning systems [11, 37]. A particularly promising development has been the emergence of vision-language-action models (VLAs), which extend foundation models for robotics through various approaches including pretraining [5][39] or fine-tuning [22, 6, 30]. These models have shown encouraging results in transferring to novel tasks, marking an important advancement toward developing generally capable robotic systems.

As these models continue to evolve, there is a critical need for systematic evaluation of their capabilities across both their intended multimodal training domains and out-of-distribution scenarios.

Our primary contributions in this paper are:

- Detailed profiling results for an initial set of VLM, VLA, and emerging "generalist" models, providing insights into their capabilities and limitations.
- Analysis of model generalization to a diverse set of real-world robotics datasets comprising a wide variety of tasks and environments.
- A systematic set of evaluation splits and metrics specifically designed for robotics learning tasks in the widely-used OpenX Dataset.
- A general framework for mapping VLMs to other modality classes, with particular emphasis on action spaces.
- Open-source software infrastructure for downloading, managing, and utilizing the benchmark data.

Through this work, we aim to provide the robotics learning community with robust tools and methodologies for assessing and comparing these emerging approaches, facilitating progress in this rapidly evolving field and helping to

bridge the gap between foundation models and practical robotics applications. Importantly, this is the first foray into a new large-scale generalist action model benchmark, called MultiNet v0.1, which we discuss in the context of Future Work.

2 Related Work

Recent years have seen a proliferation of benchmarks aimed at evaluating multimodal models across different domains and capabilities. We organize our discussion of related work into three categories: general multimodal benchmarks, robotics-specific benchmarks, and multimodal language model evaluations.

General Multimodal Benchmarks MultiBench [25] represents one of the first systematic attempts to evaluate multimodal learning across diverse domains, spanning healthcare, robotics, affective computing, and finance. Similar to our work, MultiBench emphasizes the importance of evaluating multiple aspects of model performance, including generalization, complexity, and robustness. However, while MultiBench covers a broad range of domains, its robotics evaluation is limited in scope. MMMU [50]provides another comprehensive benchmark focused on college-level multimodal understanding. The authors evaluate models across technical disciplines like engineering and science through expert-level problems requiring nuanced perception and domain-specific knowledge, but do not specifically address robotics control tasks.

Multimodal Language Model Evaluations The evolution of multimodal evaluation has progressed from single-task benchmarks like VQA [4], OK-VQA [31], MSCOCO [26], and GQA [17] to more comprehensive evaluation frameworks. Recent benchmarks span various capabilities, from basic OCR to adversarial robustness and hallucination detection (e.g., POPE [24] and HaELM [44]). More holistic evaluations have emerged through benchmarks like LAMM [48], LVLM-eHub [47], SEED [13], MMBench [51], and MM-Vet [49]. Specialized benchmarks such as MathVista [28] focus on specific domains like mathematical reasoning, while GAIA [32] tests fundamental abilities in reasoning and multimodality handling.

Robotics-Specific Benchmarks The evolution of robotics datasets has demonstrated considerable diversity across various dimensions, particularly with the advancement of imitation learning and behavior cloning (BC). While many robotics benchmarks focus on evaluating model adaptability to new tasks, functionalities, or environments, there remains a gap in systematically evaluating different BC models at scale in both simulated and real-world settings. THE COLOSSEUM [35] addresses this gap by providing a systematic evaluation framework focused on robotic manipulation, evaluating generalization across 14 different environmental perturbations. Similar efforts include FactorWorld [45], which examines 11 variation factors across 19 tasks, and KitchenShift [46], which evaluates zero-shot generalization across 7 variation factors in kitchen environments.Several other specialized robotics benchmarks have emerged: RLBench [18] offers a suite of 100 manipulation tasks in simulation; RAVENS [16] focuses on vision-based manipulation; and FurnitureBench [14] provides reproducible real-world benchmarks for long-horizon complex manipulation. LIBERO [27] offers benchmarks for knowledge transfer in lifelong robot learning, while FMB [29] emphasizes generalizable robotic learning across complex tasks. Recent work has also introduced DUDE [40] for robotic document manipulation and ProcTHOR[10] for large-scale embodied AI using procedural generation.

Our work differs from these previous benchmarks in several key aspects. First, we focus specifically on evaluating models' ability to process and generate actions from real-world robotic trajectories, rather than simulated environments or static vision-language tasks. Second, by leveraging the OpenX dataset, we evaluate across a diverse range of robot platforms and tasks, providing a more comprehensive view of model capabilities. Third, our evaluation framework specifically measures models' ability to perform zero-shot generalization across different action spaces and robot morphologies, a crucial capability for general-purpose robotic systems.

3 Evaluating VLMs and VLAs

3.1 Data

Our evaluation framework leverages the Open X-Embodiment Dataset (OpenX), currently the largest open-source repository of real robot trajectories. OpenX represents a significant collaborative effort across 21 institutions, aggregating over 1 million real robot trajectories from 22 distinct robot embodiments, ranging from single-arm manipulators to bi-manual systems and quadrupedal robots. The dataset's comprehensive nature makes it particularly suitable for evaluating generalist models, as it spans a diverse range of manipulation and locomotion tasks, environmental conditions, and robot configurations.

The dataset utilizes the Reinforcement Learning Datasets (RLDS) format, storing data in serialized tfrecord files. This standardized format efficiently accommodates the heterogeneous nature of robotics data, handling varied action spaces and input modalities across different robot setups. For instance, the format seamlessly integrates data from systems with different sensor configurations, including varying numbers of RGB cameras, depth sensors, and point cloud generators.

For version 0.1 of our benchmark, we utilize 53 of the 72 available OpenX datasets, as detailed in Table 3. We present results for 20 of these datasets for all 3 models, and have the full 53 for JAT.This subset was selected to ensure comprehensive coverage across different task types, embodiments, and environmental conditions while maintaining data quality and consistency. For datasets that did not include pre-defined evaluation sets, we have created and provided new evaluation splits to ensure robust assessment of model performance. The training splits of these 53 datasets comprise approximately 32 terabytes of data.

This careful curation of the OpenX dataset provides several advantages for our evaluation framework:

1. Scale and Diversity: The large number of trajectories and varied robot embodiments allows for comprehensive assessment of model generalization capabilities.
2. Real-World Relevance: Being composed entirely of real robot data rather than simulated interactions, the dataset better reflects the challenges of physical robot deployment.
3. Standardization: The consistent RLDS format facilitates systematic evaluation across different robot platforms and task types.
4. Cross-Domain Assessment: The inclusion of both manipulation and locomotion tasks enables evaluation of model performance across fundamentally different types of robot control.

The complete list of included datasets and their characteristics is provided in the appendix.

3.1.1 Dataset Curation

To ensure the quality and utility of our benchmark, we implemented a systematic curation process for the OpenX datasets. This process was designed to maximize the diversity and relevance of the included data while maintaining practical considerations for large-scale evaluation.

Our curation methodology consisted of several steps. First, we conducted a high-level review of dataset quality and accessibility, which resulted in the exclusion of three datasets: Austin BUDS, Austin Sailor, and Stanford Kuka Multimodal. For datasets that contained only training splits, we performed a detailed comparative analysis based on the robot platform used for data collection. This analysis considered multiple features: Robot model and morphology, Gripper specifications, Action space characteristics, Sensor configuration (number and type of RGB cameras, depth cameras, and wrist-mounted cameras), Presence of language annotations, Availability of camera calibration data, and Inclusion of proprioceptive information.

When multiple datasets shared identical values across all these features for the same robot platform, we retained only the dataset with the larger number of episodes. This decision was made to minimize redundancy while maximizing the diversity of our evaluation set. This approach ensures that each included dataset contributes unique information to the benchmark, either through different robot configurations, sensor setups, or task specifications.

Several additional datasets were excluded from version 0.1 of our benchmark due to technical limitations in their accessibility through the TensorFlow Datasets (TFDS) builder, which is the recommended data loading mechanism for OpenX. These compatibility issues will be addressed in future versions of the benchmark as the underlying infrastructure evolves. This careful curation process results in a benchmark that balances comprehensive coverage with practical considerations, ensuring that the included datasets provide meaningful evaluation scenarios while maintaining manageable computational requirements.

3.2 Models

In our evaluation, we focus on three recent vision-language-action (VLA) models that represent the current state-of-the-art in generalist robot learning: JAT (Jack of All Trades), GPT-4o, and OpenVLA. These models are particularly noteworthy for their ability to handle multiple modalities and their demonstrated capabilities across a wide variety of tasks.

JAT [12] is a transformer-based model optimized for handling sequential decision-making tasks and multi-modal data types. With 768-dimensional hidden states and 12 layers, JAT employs a dual attention mechanism inspired by the Longformer architecture, combining global attention with a 512-token window and local attention with a 256-token

window. The model was trained for 250,000 steps on a diverse dataset spanning robotics control, computer vision, and natural language processing tasks. JAT's architecture is specifically designed to provide wider attention windows for timesteps compared to previous approaches, making it particularly suitable for long-horizon robotics tasks.

GPT-4o [1] represents a significant advancement in omni-modal modeling, accepting combinations of text, audio, image, and video inputs while generating multi-modal outputs. The model demonstrates strong performance in robotic manipulation tasks, particularly in scenarios requiring generalization to novel objects and environments. GPT-4o incorporates advanced safety measures and has been extensively evaluated across multiple risk categories, including cybersecurity, persuasion, and model autonomy.

OpenVLA, a 7B-parameter open-source vision-language-action model, was trained on 970,000 robot episodes from the Open X-Embodiment dataset. Its architecture combines a 600M-parameter visual encoder (utilizing both SigLIP and DinoV2 models) with a 7B-parameter Llama 2 language model backbone. OpenVLA is notable for its strong performance in generalist robot manipulation tasks, outperforming larger models while using significantly fewer parameters. The model particularly excels in multi-task environments involving multiple objects and demonstrates strong language grounding abilities.

Each of these models represents different approaches to the challenge of generalist robot learning:

JAT emphasizes broad "generalist" multi-modal capabilities. GPT-4o is a powerful VLM, and allows for various approaches to map language output to action & control tasks. OpenVLA prioritizes open-source accessibility while maintaining competitive performance with larger closed-source models

This diversity in approaches provides valuable insights into different architectural and training strategies for generalist robot learning. The models also represent different points on the spectrum of model size and computational requirements, allowing us to evaluate the relationship between model scale and performance across various robotics tasks.

3.3 Evaluation Metrics

Mean Squared Error (MSE) serves as our primary metric for evaluating model performance on offline robotics trajectories. In the context of offline reinforcement learning, MSE has proven to be a reliable metric for estimating optimal value functions and has demonstrated strong empirical performance. For our benchmark, MSE is particularly appropriate due to several key properties:

1. Non-Negativity: The metric remains non-negative, ensuring that errors are consistently accounted for without potential cancellation effects from opposing signs.
2. Sensitivity to Large Errors: The squared term in MSE emphasizes larger deviations, providing clear indication of significant prediction errors.
3. Bias-Variance Trade-off: MSE inherently captures both bias and variance components, offering a comprehensive measure of prediction accuracy.

For a given prediction, MSE is calculated as:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{1}$$

where y_i represents the ground truth action, $\hat{y}_i$ is the predicted action, and n is the number of observations.

For our benchmark, we employ MSE to evaluate how accurately models predict actions given the observation states, image observation, and language instruction at each timestep. Given the offline nature of the OpenX dataset and the inability to evaluate models on physical robots, comparing predicted and ground truth action tensors provides the most direct assessment of model performance.

We report several variations of MSE to provide a comprehensive performance analysis:

1. **Average MSE (AMSE):** Computed as the mean MSE across all timesteps in a dataset, AMSE enables direct comparison of model performance across different datasets and architectures.
2. **Normalized AMSE (NAMSE):** Calculated as an average of the min-max normalized MSEs over all the timesteps in the dataset, this metric normalizes prediction errors to each model's error range, facilitating more equitable cross-dataset for a single model comparison by accounting for different scales in model outputs and dataset action spaces.

3. **Completion Rate:** We assess successful completion by comparing final predicted actions with ground truth final actions for all episodes in the dataset. While this serves as an approximate measure of task completion, it provides valuable insights into models' ability to reach target states across trajectories.

The combination of these metrics allows us to evaluate both the fine-grained accuracy of action predictions and the overall task-completion capabilities of different models. This is particularly important in offline robotics, where environments and rewards are not available for policy evaluation.

4 Experimental Setup

4.1 Profiling Configuration

We established specific configurations for each model to ensure consistent and fair evaluation across the diverse OpenX datasets. Below, we detail the precise setup for each model, including handling of inputs, processing decisions, and any necessary adaptations.

JAT Configuration The JAT model was evaluated in a zero-shot setting, where predictions are made using only the current timestep information without access to previous states. For each prediction, the model receives the observation state, observation image, and language instruction. Several key preprocessing steps were implemented:

- **Image Processing:** JAT requires 4-channel images. For 3-channel RGB inputs, we create an RGBA image by duplicating the red channel as the alpha channel. For 2-channel inputs, we duplicate both channels to create a 4-channel representation.
- **Observation Processing:** For dictionary-type observations, we concatenate all floating-point observations (excluding image and language instruction embeddings) into a single tensor. In cases where no floating-point observations exist, we pass a zero-filled dummy tensor.
- **Action Processing:** Ground truth actions are processed by concatenating all floating-point actions into a single tensor when the action space is represented as a dictionary.
- **Multi-Image Handling:** For timesteps with multiple available images, we select the primary image (typically designated with the keyword 'image').

GPT Configuration GPT was also evaluated in a zero-shot configuration, with several specific processing requirements:

- **Prompt Construction:** Each prediction is based on a comprehensive prompt including:
  - Floating-point observation states with their corresponding keys as descriptors for specific datasets like Berkeley Autolab where there are such observation states available.
  - Primary image observation
  - Natural language instruction
  - Verbal descriptions for each action space dimension
  - The official action space statistics if available or statistical information (min, max, mean) for each action dimension.
  - Environmental and task descriptions when available
- **Output Processing:** To handle GPT's VLM-native outputs, which may be incompatible with the required floating-point action tensor format, we implemented error handling:
  - For incompatible outputs (incorrect tensor sizes, string elements, mixed text-tensor outputs, or non-scalar elements), we generate a random action tensor with values in $[0.0, 1.0)$ as a fallback.
- **Multi-Image Processing:** For timesteps with multiple available images, we select the primary image (typically designated with the keyword 'image').

OpenVLA Configuration OpenVLA's configuration focused primarily on action space handling and gripper command conversions:

- **Gripper Command Standardization:** We implemented several conversion protocols:
  - Binary discretization: For $[0, 1]$ to $\{0, 1\}$ conversion, we threshold at 0.5

  - Ternary discretization: For $[0, 1]$ to $\{-1, 0, 1\}$ conversion, values < 0.05 map to -1 (closed), > 0.95 to 1 (open), and $[0.05, 0.95]$ to 0 (no change)
  - Continuous normalization: For $[0, 1]$ to $[-1, 1]$ conversion, we apply the formula: $y = 2 \cdot (x - orig_{low})/(orig_{high} - orig_{low}) - 1$. This was used by the authors in [22].
- **Special Cases:**
  - For the UCSD pick-and-place dataset, we used dataset statistics to scale gripper commands to the appropriate torque space
  - For ETH agent affordances, we applied the transformation: $unnormalized = 0.5 \cdot (normalized + 1) \cdot (high - low) + low$, where high and low are the 99th and 1st percentiles respectively
- **Action Space Handling:**
  - For datasets using velocity, angular velocity, or torque-based action spaces (e.g., ETH agent affordances and UCSD datasets), we note potential compatibility issues with OpenVLA's position-based predictions
  - We exclude 'Terminal' tensors from action spaces, as OpenVLA predicts only XYZ, RPY, and gripper commands

Additional Considerations We encountered cases where image observations were unavailable due to non-standard image key naming (e.g., 'agentview_rgb', 'frontright_fisheye_image') in some datasets. These were utilized for OpenVLA, but not the other models, as OpenVLA requires an image as part of its input. This specific case occurred with 2 datasets in particular, conq_hose_manipulation, and viola.

4.2 Inference Infrastructure

To facilitate reproducible evaluation of these models, we detail the infrastructure requirements and setup for each model's inference pipeline.

JAT and GPT Infrastructure For JAT evaluation and GPT API interfacing, we utilized a Google Cloud Platform (GCP) e2-standard-8 instance with 8 vCPU (4 physical cores), 32 GB memory, and x86/64 architecture. While this configuration exceeds the minimum requirements, the additional computational resources enabled efficient parallelization of evaluation runs. For GPT specifically, as inference occurs through API endpoints, the local infrastructure requirements are minimal. Storage was provided through GCP's standard persistent disk service.

OpenVLA Infrastructure OpenVLA inference was conducted on a GCP g2-standard-8 instance equipped with a single NVIDIA L4 GPU, 8 vCPU (4 physical cores), 32 GB system memory, and x86/64 architecture. The NVIDIA L4 GPU, featuring the Ada Lovelace architecture, was specifically chosen for two key advantages: compatibility with Flash Attention 2.x for efficient attention computation, and 24 GB of GDDR6 memory, sufficient for full-model inference of OpenVLA without optimization. Storage was similarly provided through GCP's standard persistent disk service.

5 Results & Discussion

In our evaluation of vision-language-action models, we seek to answer several fundamental questions about their capabilities and limitations: (1) How do current VLM & VLA models perform across diverse robotics tasks and platforms, particularly in zero-shot settings? (2) What impact do different model architectures and training approaches (e.g., prompt engineering, robotics-specific training) have on performance? (3) How well do these models handle different action spaces and robot morphologies? (4) What are the current limitations and failure modes of these models in real-world robotics tasks? Through systematic analysis of three state-of-the-art models across 20 diverse datasets, we provide insights into these questions below.

5.1 Average Model Performance Analysis

Our evaluation reveals significant variations in performance across models and datasets. We observe that while JAT consistently shows higher AMSE (indicating worse performance) across most datasets, OpenVLA and GPT demonstrate more comparable performance levels, with AMSE typically below 0.5 for most datasets.

Overall Performance Patterns For OpenVLA, we observe generally consistent performance across most datasets with AMSE in the 0.1-0.5 range, with best performance of all 3 models for tasks that fall within its training distribution, with notable exceptions in complex manipulation tasks. GPT shows comparable or slightly better performance on many

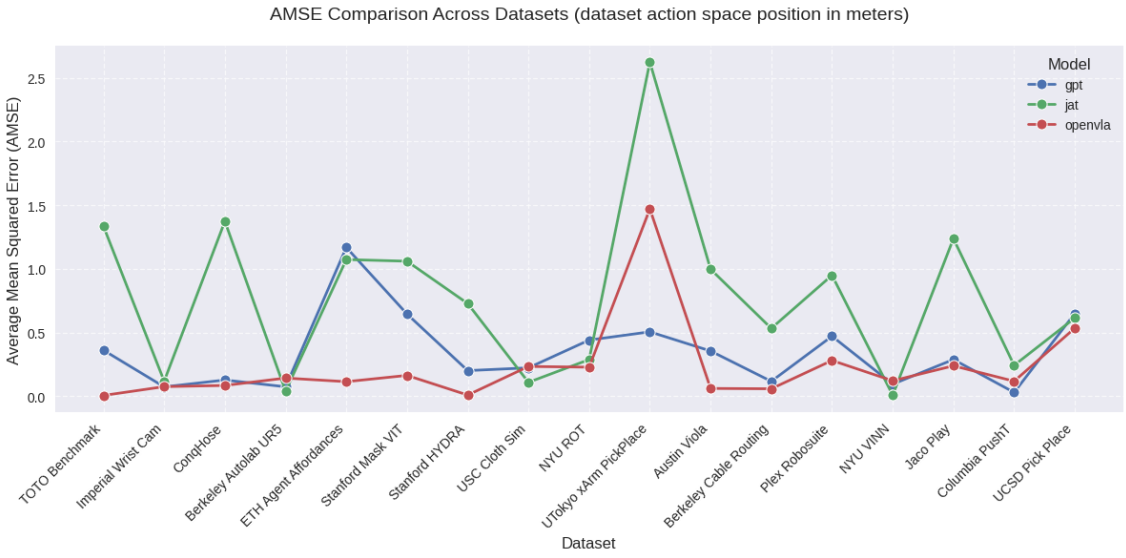

Figure 2: AMSE values of GPT-4o, JAT, and OpenVLA Across 20 OpenX Datasets. JAT displays the poorest performance out of the 3 models with higher AMSE scores, while OpenVLA and GPT-4o demonstrate similar performance. OpenVLA displays consistent performance across most datasets

Table 1: Dataset Coverage and Action Space Characteristics

Dataset Name	Registered Dataset Name	In Pretraining OpenVLA	Action Space Type
Jaco Play	jaco_play	✓	4D (1 grip, 3 pos)
Berkeley Cable Routing	berkeley_cable_routing	✓	7D (3 ang, 3 pos, 1 term)
NYU Door Opening	nyu_door_opening_surprising_effectiveness		8D (1 grip, 3 ang, 3 pos, 1 term)
VIOLA	viola	✓	8D (1 grip, 3 ang, 3 pos, 1 term)
Berkeley Autolab UR5	berkeley_autolab_ur5	✓	8D (1 grip, 3 ang, 3 pos, 1 term)
TOTO	toto	✓	7D (3 ang, 3 pos, 1 term)
Columbia PushT	columbia_cairlab_pusht_real		8D (1 grip, 3 ang, 3 pos, 1 term)
NYU ROT	nyu_rot_dataset_converted_externally_to_rlds		7D (3 pos, 3 ang, 1 grip)
Stanford HYDRA	stanford_hydra_dataset_converted_externally_to_rlds	✓	7D (3 pos, 3 ang, 1 grip)
UCSD Kitchen	ucsd_kitchen_dataset_converted_externally_to_rlds	✓	8D (3 pos, 3 ang, 1 grip, 1 term)
UCSD Pick Place	ucsd_pick_and_place_dataset_converted_externally_to_rlds		4D (3 vel, 1 grip torque)
USC Cloth Sim	usc_cloth_sim_converted_externally_to_rlds		4D (3 pos, 1 grip)
Tokyo PR2 Fridge	utokyo_pr2_opening_fridge_converted_externally_to_rlds		8D (3 pos, 3 ang, 1 grip, 1 term)
Tokyo PR2 Tabletop	utokyo_pr2_tabletop_manipulation_converted_externally_to_rlds		8D (3 pos, 3 ang, 1 grip, 1 term)
UTokyo xArm Pick-Place	utokyo_xarm_pick_and_place_converted_externally_to_rlds		7D (3 pos, 3 ang, 1 grip)
Stanford MaskVIT	stanford_mask_vit_converted_externally_to_rlds		5D (3 pos, 1 ang, 1 grip)
ETH Agent Affordances	eth_agent_affordances		6D (3 vel, 3 ang vel)
Imperial Sawyer	imperialcollege_sawyer_wrist_cam		8D (3 pos, 3 ang, 1 grip, 1 term)
ConqHose	conq_hose_manipulation		7D (3 pos, 3 ang, 1 grip)
Plex RoboSuite	plex_robosuite		7D (3 pos, 3 ang, 1 grip)

pos: position, orient: orientation, grip: gripper, term: terminate, vel: velocity, ang: angular

datasets, particularly excelling in precise manipulation tasks. Both models maintain relatively stable performance across similar task types, though with different error profiles.

GPT demonstrates strongest performance on:

- berkeley_autolab_ur5 (AMSE: 0.074)
- columbia_cairlab_pusht_real (AMSE: 0.030)
- imperialcollege_sawyer_wrist_cam (AMSE: 0.073)

Common Challenges Both models exhibit significant challenges with certain task types:

- Complex manipulation tasks, particularly those involving large movements or multi-step sequences like Kitchen manipulation tasks.

Table 2: Performance Metrics Comparison across Models

	GPT		OpenVLA		JAT	
Dataset Name	AMSE	NAMSE	AMSE	NAMSE	AMSE	NAMSE
Jaco Play	0.288	0.188	0.239	0.228	1.237	0.295
Berkeley Cable Routing	0.117	0.010	0.058	0.091	0.533	0.411
NYU Door Opening	0.094	0.046	0.121	0.304	0.008	0.061
VIOLA	0.355	0.134	0.061	0.072	0.997	0.331
Berkeley Autolab UR5	0.074	0.049	0.142	0.249	0.040	0.073
TOTO	0.361	0.069	0.006	0.004	1.335	0.238
Columbia PushT	0.030	0.046	0.118	0.820	0.242	0.347
NYU ROT	0.441	0.034	0.228	0.308	0.288	0.177
Stanford HYDRA	0.201	0.009	0.009	0.054	0.728	0.147
UCSD Kitchen	11580.963	0.207	5018.936	0.116	34890.635	0.353
UCSD Pick Place	0.650	0.086	0.535	0.175	0.614	0.210
USC Cloth Sim	0.223	0.260	0.234	0.305	0.109	0.375
Tokyo PR2 Fridge	16035.136	0.037	68433.175	0.159	221666.531	0.324
Tokyo PR2 Tabletop	2550.878	0.014	8728.959	0.116	117663.493	0.364
UTokyo xArm Pick-Place	0.505	0.088	1.471	0.252	2.623	0.254
Stanford MaskVIT	0.645	0.120	0.163	0.184	1.060	0.571
ETH Agent Affordances	1.168	0.057	0.114	0.139	1.073	0.290
Imperial Sawyer	0.073	0.183	0.075	0.517	0.118	0.356
ConqHose	0.127	0.024	0.084	0.264	1.373	0.178
Plex RoboSuite	0.471	0.067	0.280	0.206	0.950	0.142

AMSE: Average Mean Squared Error, NAMSE: Normalized Average Mean Squared Error
Large AMSE values (e.g., for Kitchen and PR2 tasks) reflect different action space scales

- Tasks requiring significant temporal reasoning or complex action sequences. This follows naturally as the models were assessed in a zero shot fashion.

5.1.1 Model-Specific Analysis

The performance patterns we observe can may be attributable to several architectural and training differences between the models:

OpenVLA The combination of SigLIP and DinoV2 visual encoders appears to provide robust visual features, contributing to consistent performance across tasks. However, this comes at the cost of absolute precision in some cases. The model's specific training on robotics data from OpenX likely contributes to its stability across different task types, though it may not always achieve optimal performance on any single task type.

GPT GPT's sophisticated prompt construction and ability to handle detailed statistical information about action spaces appears to help in making more precise predictions for well-defined tasks. Its strong performance on precise manipulation tasks suggests that its general-purpose capabilities transfer well to robotics control in structured scenarios. However, it shows similar limitations to OpenVLA in complex, multi-step tasks.

JAT JAT's significantly higher AMSE across datasets suggests that its architecture, while suitable for general-purpose tasks, may not be optimized for precise robotics control.

5.1.2 Implications for Future Development

These results suggest several directions for improvement in VLA model development:

- The variation in performance across robot platforms suggests that more work is needed in developing platform-agnostic control capabilities
- The superior performance of GPT and OpenVLA in their respective strengths suggests that combining their approaches - sophisticated prompt engineering with robotics-specific training - might yield better overall performance

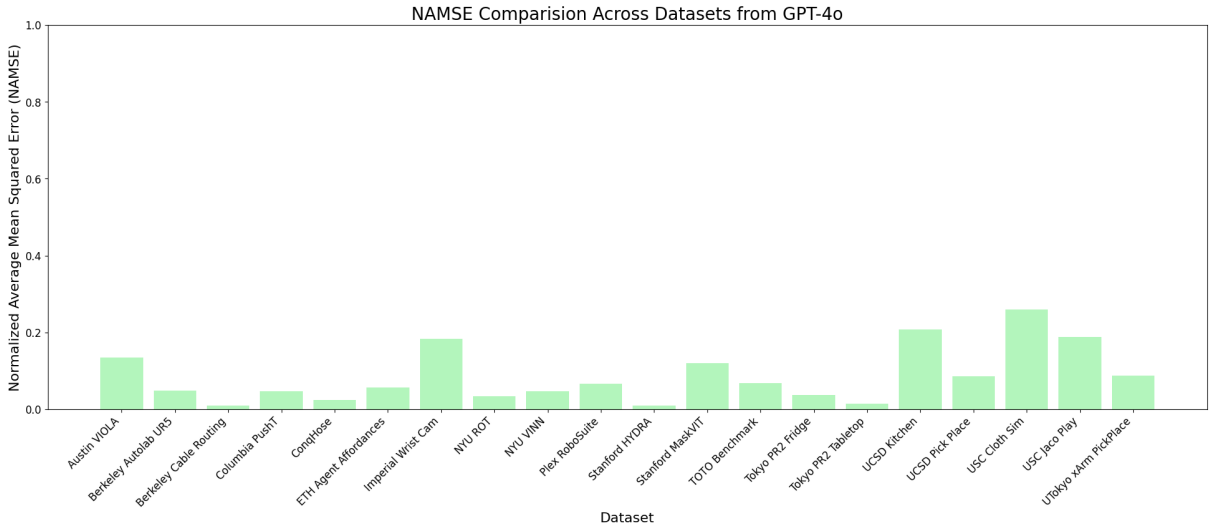

Figure 3: Normalized AMSE For GPT4o. GPT-4o demonstrates consistent NAMSE across all datasets, suggesting that the prompt engineering framework which provides detailed information about the action space, task, and environment, may be a key factor.

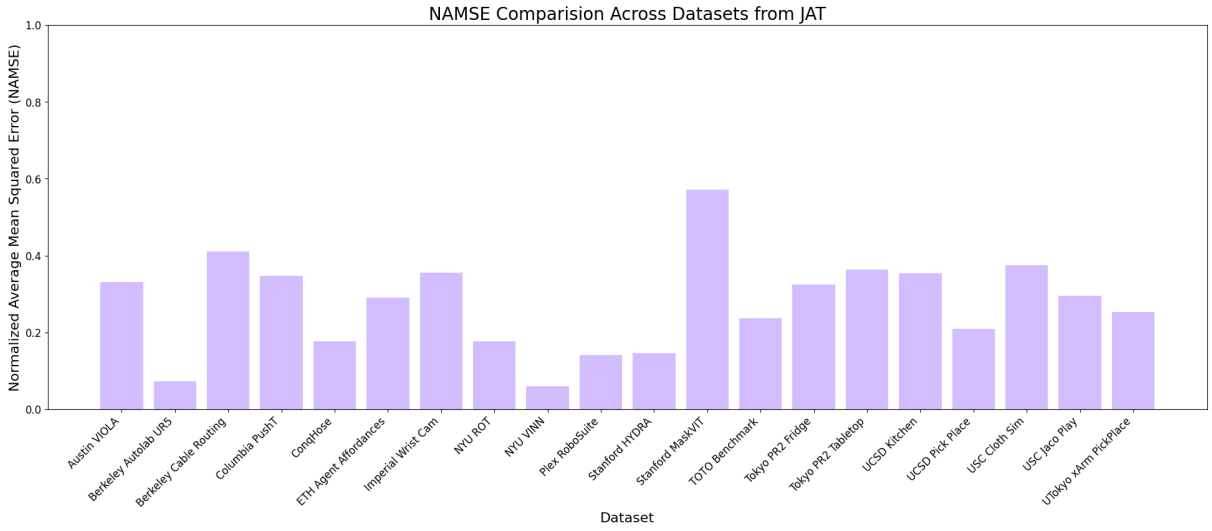

Figure 4: Normalized AMSE For JAT. JAT exhibits moderate variation in NAMSE across tasks, with a spike in the Stanford MaskVIT Dataset, while maintaining realtively consistent preformance for similar task types.

5.2 Normalized Performance Analysis

While absolute performance metrics like AMSE provide insight into task-specific capabilities, normalized average mean squared error (NAMSE) allows us to understand how each model performs across different tasks relative to its own capabilities. NAMSE is particularly valuable for understanding inherent task difficulty and model behavior patterns independent of action space scale.

5.2.1 Model-Specific Performance Patterns

GPT-4o GPT-4o demonstrates remarkably consistent normalized performance across datasets, with NAMSE generally remaining below 0.2. This stability is particularly noteworthy given the diversity of tasks in the benchmark. The model's sophisticated prompt engineering approach appears to be a key factor in this consistency, as it includes:

- Explicit action space statistics (min, max, mean) for each dimension

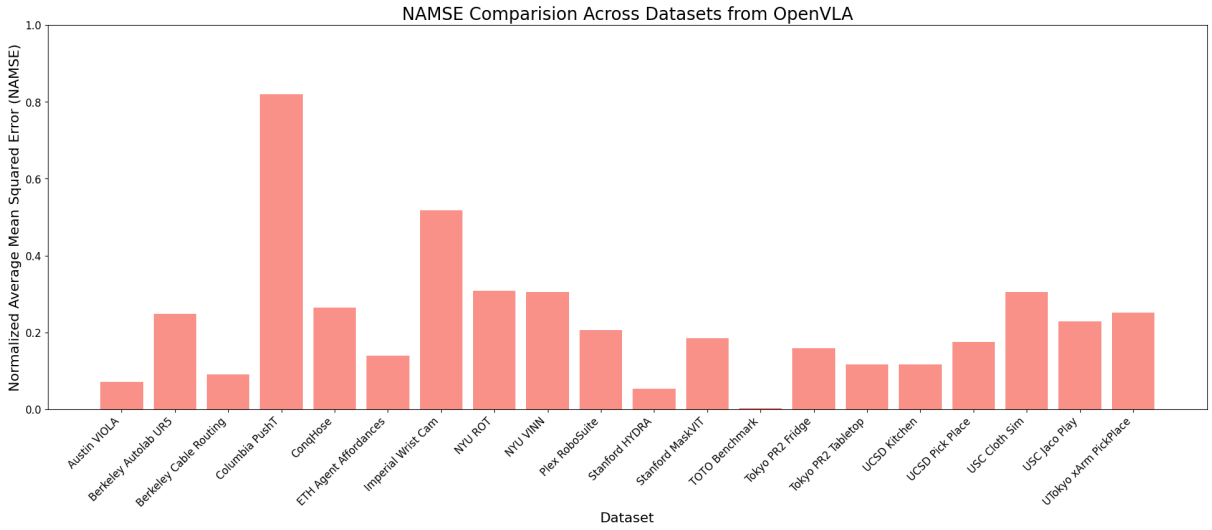

Figure 5: Normalized AMSE For OpenVLA. OpenVLA shows high variation in NAMSE values. As expected it displays very strong performance on tasks such as toto which is within its training distribution. OpenVLA shows a clear pattern of task-specific performance differences.

- Verbal descriptions for each action dimension
- Detailed environment and task descriptions when available

This comprehensive prompting strategy provides clear constraints and context for each prediction, likely contributing to the model's ability to maintain consistent relative performance across diverse tasks.

OpenVLA OpenVLA shows the most dramatic variation in normalized performance:

- Highest normalized error on columbia_cairlab_pusht_real (NAMSE: 0.82)
- Exceptionally strong performance on pretrained tasks (e.g., toto with NAMSE: 0.003)
- Clear pattern of task-specific performance variations

This variation suggests that OpenVLA's architecture and training approach may lead to stronger task specialization compared to other models.

JAT JAT exhibits moderate variation across tasks, with NAMSE typically ranging from 0.2 to 0.4:

- Notable performance spike on stanford_mask_vit (NAMSE $\sim$0.57)
- Relatively consistent performance band for similar task types
- Higher baseline NAMSE compared to GPT-4o but more stable than OpenVLA

5.2.2 Cross-Model Insights

The normalized analysis reveals several key patterns about task difficulty and model architecture:

Task Difficulty Patterns Certain tasks consistently show higher normalized error across all models, independent of architecture:

- Complex manipulation tasks and multi-step operations consistently show higher NAMSE
- Simple pick-and-place operations tend to show lower normalized error
- Tasks requiring precise control generally result in higher normalized error

Architectural Implications The variation in normalized performance across models provides insights into their architectural strengths:

- GPT-4o's consistent normalized performance suggests its architecture and prompting strategy create a more generally robust system
- OpenVLA's high variation indicates stronger task specialization, possibly due to its training approach and dual visual encoder
- JAT's moderate but consistent variation suggests a middle ground between specialization and generalization

This normalized analysis reveals that while absolute performance varies significantly, there are consistent patterns in what tasks are relatively more challenging for each model architecture. The success of GPT-4o's prompt engineering approach, in particular, suggests that providing structured context about action spaces and environmental constraints may be a key factor in achieving consistent performance across diverse tasks. This observation could inform future development of VLA models, suggesting that incorporating more explicit task and action space information could improve robustness and generalization capabilities.

5.3 Key Takeaways

Our evaluation of VLM & VLA models reveals several fundamental insights about the current state of VLM & VLA models transferring to robotics tasks:

1. Prompt Engineering Impact: GPT-4o's consistent performance across diverse tasks demonstrates that structured prompting with explicit action space information and environmental context may be as important as architectural choices. This suggests that future VLA development should consider incorporating structured task representations as a core design principle.

2. Specialization vs. Generalization: We observe a clear trade-off between specialized and general performance. OpenVLA shows superior performance on tasks within its training distribution but higher variation across tasks, while GPT-4o maintains more consistent but sometimes suboptimal performance. This highlights the ongoing challenge of developing models that can both specialize and generalize effectively.

3. Task Complexity Barriers: All models, regardless of architecture, struggle with complex manipulation tasks requiring multi-step planning or precise control. This consistent limitation suggests that current approaches may be missing key capabilities needed for complex robotics tasks.

4. Action Space Sensitivity: Performance varies significantly with different action space characteristics, particularly in tasks requiring precise control or complex action sequences. This suggests the need for more robust methods of handling diverse action spaces and robot morphologies.

6 Future Work

While our current results provide valuable insights into the capabilities and limitations of these models, we envision several important directions for expanding and enhancing this benchmark. We present these as a subset of a larger benchmark we are developing, dubbed MultiNet. We contextualize the opportunities ahead in the context of this benchmark below.

A critical question in the development of generalist models is whether the integration of control capabilities comes at the cost of performance in other domains. To address this, future versions of MultiNet will evaluate SOTA VLAs on pure vision-language and language tasks, allowing us to assess whether fine-tuning or co-training on control tasks impacts their performance in these foundational modalities. This analysis will help inform architectural and training strategies that maintain strong performance across all modalities.

We also plan to expand beyond the OpenX dataset to evaluate these models on a broader range of control tasks. This expansion will allow us to better understand how VLAs and generalist models perform on completely out-of-distribution data, providing insights into their true generalization capabilities. While our current evaluations focus on zero-shot performance, future work will investigate few-shot learning and fine-tuning scenarios, offering a more complete picture of these models' adaptability.

A particularly promising direction is the exploration of VLA transfer to non-robotic domains. We are especially interested in investigating how these models can be fine-tuned for software environments, potentially enabling the development of more capable digital agents. This research could reveal insights about the generalization of embodied learning principles to virtual environments.

Additionally, we identify several novel directions for future investigation:

- **Compositional Generalization:** Evaluating how well VLAs can combine learned primitives to solve novel tasks, particularly in scenarios requiring multi-step reasoning or tool use.
- **Long Sequence Reliability:** Developing metrics to assess the consistency of model behavior over extended sequences, including the ability to maintain goals and adapt to changing conditions.
- **Cross-Embodiment Transfer:** Further investigating how knowledge transfers between different robot morphologies, potentially leading to more efficient training strategies for new platforms.
- **Memory and Long-Term Planning:** Assessing models' capabilities in tasks requiring long-term memory and strategic planning, particularly in multi-phase manipulation tasks.
- **Multi-Agent Interaction:** Extending the benchmark to scenarios involving multiple agents, evaluating coordination and collaborative manipulation capabilities.

Finally, while MultiNet currently operates as an offline benchmark, we plan to develop online evaluation capabilities. This expansion will include the integration of simulation environments for both 2D and 3D control tasks, enabling more dynamic and interactive assessment of model performance. Such environments will allow for more comprehensive evaluation of model capabilities in real-time decision-making scenarios.

Through these future developments, we aim to establish MultiNet as a comprehensive and rigorous benchmark for assessing and advancing the field of vision-language-action models. This expanded scope will provide researchers and practitioners with valuable tools for understanding and improving these increasingly important models.

7 Conclusion

In this work, we presented a comprehensive evaluation framework for vision-language-action models and conducted a systematic analysis of their performance across a diverse range of robotics tasks. Our study reveals several important insights about the current state of VLA models and highlights critical areas for future development.

We find that current VLA models demonstrate varying levels of capability across different tasks, with notable strengths and limitations. GPT-4o shows remarkable consistency in normalized performance across datasets, likely due to its sophisticated prompt engineering approach that provides structured context about action spaces and environmental constraints. OpenVLA demonstrates strong performance on certain tasks but shows higher variation across different scenarios, suggesting task-specific specialization. JAT, while showing moderate consistency, generally achieves higher error rates, indicating potential limitations in its architecture for precise control tasks.

Our analysis reveals several critical challenges that need to be addressed in future work. First, all models struggle significantly with complex manipulation tasks. Second, the performance of these models varies substantially across different robot platforms and action spaces, suggesting a need for more robust architectures that can better handle diverse control scenarios. Third, the notable impact of prompt engineering on performance, as demonstrated by GPT-4o, suggests that developing more sophisticated ways to provide context and constraints to these models could be a promising direction for improvement.

Looking forward, our results suggest several promising directions for future research. The development of more robust architectures that can maintain consistent performance across diverse tasks while handling complex, multi-step manipulations remains a key challenge. Additionally, the integration of structured task representations and better handling of temporal dependencies could help address the current limitations in complex manipulation tasks. Finally, our open-source evaluation framework provides a foundation for systematic assessment of future VLA models, enabling more rigorous comparison and benchmarking of new approaches. We are excited to engage with the broader research community to extend these results and advance the emerging class of Multimodal VLA models.

References

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

[2] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Daniel Ho, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Eric Jang, Rosario Jauregui Ruano, Kyle Jeffrey, Sally Jesmonth, Nikhil J Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Kuang-Huei Lee, Sergey Levine, Yao Lu, Linda Luu, Carolina

Parada, Peter Pastor, Jornell Quiambao, Kanishka Rao, Jarek Rettinghouse, Diego Reyes, Pierre Sermanet, Nicolas Sievers, Clayton Tan, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Mengyuan Yan, and Andy Zeng. Do as i can, not as i say: Grounding language in robotic affordances, 2022. URL `https://arxiv.org/abs/2204.01691`.

[3] Jean-Baptiste Alayrac, Antoine Miech, Ivan Laptev, Josef Sivic, et al. Multi-task learning of object states and state-modifying actions from web videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[4] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015.

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

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

[7] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

[8] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

[9] Open X-Embodiment Collaboration, Abby O'Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, Albert Tung, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anchit Gupta, Andrew Wang, Andrey Kolobov, Anikait Singh, Animesh Garg, Aniruddha Kembhavi, Annie Xie, Anthony Brohan, Antonin Raffin, Archit Sharma, Arefeh Yavary, Arhan Jain, Ashwin Balakrishna, Ayzaan Wahid, Ben Burgess-Limerick, Beomjoon Kim, Bernhard Schölkopf, Blake Wulfe, Brian Ichter, Cewu Lu, Charles Xu, Charlotte Le, Chelsea Finn, Chen Wang, Chenfeng Xu, Cheng Chi, Chenguang Huang, Christine Chan, Christopher Agia, Chuer Pan, Chuyuan Fu, Coline Devin, Danfei Xu, Daniel Morton, Danny Driess, Daphne Chen, Deepak Pathak, Dhruv Shah, Dieter Büchler, Dinesh Jayaraman, Dmitry Kalashnikov, Dorsa Sadigh, Edward Johns, Ethan Foster, Fangchen Liu, Federico Ceola, Fei Xia, Feiyu Zhao, Felipe Vieira Frujeri, Freek Stulp, Gaoyue Zhou, Gaurav S. Sukhatme, Gautam Salhotra, Ge Yan, Gilbert Feng, Giulio Schiavi, Glen Berseth, Gregory Kahn, Guangwen Yang, Guanzhi Wang, Hao Su, Hao-Shu Fang, Haochen Shi, Henghui Bao, Heni Ben Amor, Henrik I Christensen, Hiroki Furuta, Homanga Bharadhwaj, Homer Walke, Hongjie Fang, Huy Ha, Igor Mordatch, Ilija Radosavovic, Isabel Leal, Jacky Liang, Jad Abou-Chakra, Jaehyung Kim, Jaimyn Drake, Jan Peters, Jan Schneider, Jasmine Hsu, Jay Vakil, Jeannette Bohg, Jeffrey Bingham, Jeffrey Wu, Jensen Gao, Jiaheng Hu, Jiajun Wu, Jialin Wu, Jiankai Sun, Jianlan Luo, Jiayuan Gu, Jie Tan, Jihoon Oh, Jimmy Wu, Jingpei Lu, Jingyun Yang, Jitendra Malik, João Silvério, Joey Hejna, Jonathan Booher, Jonathan Tompson, Jonathan Yang, Jordi Salvador, Joseph J. Lim, Junhyek Han, Kaiyuan Wang, Kanishka Rao, Karl Pertsch, Karol Hausman, Keegan Go, Keerthana Gopalakrishnan, Ken Goldberg, Kendra Byrne, Kenneth Oslund, Kento Kawaharazuka, Kevin Black, Kevin Lin, Kevin Zhang, Kiana Ehsani, Kiran Lekkala, Kirsty Ellis, Krishan Rana, Krishnan Srinivasan, Kuan Fang, Kunal Pratap Singh, Kuo-Hao Zeng, Kyle Hatch, Kyle Hsu, Laurent Itti, Lawrence Yunliang Chen, Lerrel Pinto, Li Fei-Fei, Liam Tan, Linxi "Jim" Fan, Lionel Ott, Lisa Lee, Luca Weihs, Magnum Chen, Marion Lepert, Marius Memmel, Masayoshi Tomizuka, Masha Itkina, Mateo Guaman Castro, Max Spero, Maximilian Du, Michael Ahn, Michael C. Yip, Mingtong Zhang, Mingyu Ding, Minho Heo, Mohan Kumar Srirama, Mohit Sharma, Moo Jin Kim, Naoaki Kanazawa, Nicklas Hansen, Nicolas Heess, Nikhil J Joshi, Niko Suenderhauf, Ning Liu, Norman Di Palo, Nur Muhammad Mahi Shafiullah, Oier Mees, Oliver Kroemer, Osbert Bastani, Pannag R Sanketi, Patrick "Tree" Miller, Patrick Yin, Paul Wohlhart, Peng Xu, Peter David Fagan, Peter Mitrano, Pierre Sermanet, Pieter Abbeel, Priya Sundaresan, Qiuyu Chen, Quan Vuong, Rafael Rafailov, Ran Tian, Ria Doshi, Roberto Mart'in-Mart'in, Rohan Baijal, Rosario Scalise, Rose Hendrix, Roy Lin, Runjia Qian, Ruohan Zhang, Russell Mendonca, Rutav Shah, Ryan Hoque, Ryan Julian, Samuel Bustamante, Sean Kirmani, Sergey Levine, Shan Lin, Sherry Moore, Shikhar Bahl, Shivin Dass, Shubham Sonawani, Shubham Tulsiani, Shuran Song, Sichun Xu, Siddhant Haldar, Siddharth Karamcheti, Simeon Adebola, Simon Guist, Soroush Nasiriany, Stefan Schaal, Stefan Welker, Stephen Tian, Subramanian Ramamoorthy, Sudeep Dasari, Suneel Belkhale, Sungjae Park, Suraj Nair, Suvir Mirchandani, Takayuki Osa, Tanmay Gupta, Tatsuya Harada, Tatsuya Matsushima, Ted Xiao, Thomas Kollar, Tianhe Yu, Tianli Ding, Todor Davchev, Tony Z. Zhao, Travis Armstrong, Trevor Darrell, Trinity Chung, Vidhi Jain, Vikash Kumar, Vincent Vanhoucke, Wei Zhan, Wenxuan Zhou, Wolfram Burgard, Xi Chen, Xiangyu Chen, Xiaolong Wang,

Xinghao Zhu, Xinyang Geng, Xiyuan Liu, Xu Liangwei, Xuanlin Li, Yansong Pang, Yao Lu, Yecheng Jason Ma, Yejin Kim, Yevgen Chebotar, Yifan Zhou, Yifeng Zhu, Yilin Wu, Ying Xu, Yixuan Wang, Yonatan Bisk, Yongqiang Dou, Yoonyoung Cho, Youngwoon Lee, Yuchen Cui, Yue Cao, Yueh-Hua Wu, Yujin Tang, Yuke Zhu, Yunchu Zhang, Yunfan Jiang, Yunshuang Li, Yunzhu Li, Yusuke Iwasawa, Yutaka Matsuo, Zehan Ma, Zhuo Xu, Zichen Jeff Cui, Zichen Zhang, Zipeng Fu, and Zipeng Lin. Open X-Embodiment: Robotic learning datasets and RT-X models. `https://arxiv.org/abs/2310.08864`, 2023.

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

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

[12] Quentin Gallouédec, Edward Beeching, Clément Romac, and Emmanuel Dellandréa. Jack of all trades, master of some, a multi-purpose transformer agent, 2024. URL `https://arxiv.org/abs/2402.09844`.

[13] Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. Planting a seed of vision in large language model. *arXiv preprint arXiv:2307.08041*, 2023.

[14] Minho Heo, Youngwoon Lee, Doohyun Lee, and Joseph J Lim. Furniturebench: Reproducible real-world benchmark for long-horizon complex manipulation. *arXiv preprint arXiv:2305.12821*, 2023.

[15] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[16] Jie Huang, Wei Ping, Peng Xu, Mohammad Shoeybi, Kevin Chen-Chuan Chang, and Bryan Catanzaro. Raven: In-context learning with retrieval augmented encoder-decoder language models. *arXiv preprint arXiv:2308.07922*, 2023.

[17] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.

[18] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J Davison. Rlbench: The robot learning benchmark & learning environment. arxiv e-prints, art. *arXiv preprint arXiv:1909.12271*, 2019.

[19] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[20] Siddharth Karamcheti, Suraj Nair, Annie S Chen, Thomas Kollar, Chelsea Finn, Dorsa Sadigh, and Percy Liang. Language-driven representation learning for robotics. *arXiv preprint arXiv:2302.12766*, 2023.

[21] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[22] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[23] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.

[24] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.

[25] Paul Pu Liang, Yiwei Lyu, Xiang Fan, Zetian Wu, Yun Cheng, Jason Wu, Leslie Chen, Peter Wu, Michelle A Lee, Yuke Zhu, et al. Multibench: Multiscale benchmarks for multimodal representation learning. *Advances in neural information processing systems*, 2021(DB1):1, 2021.

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[27] 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, 2024.

[28] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.

[29] Jianlan Luo, Charles Xu, Fangchen Liu, Liam Tan, Zipeng Lin, Jeffrey Wu, Pieter Abbeel, and Sergey Levine. Fmb: A functional manipulation benchmark for generalizable robotic learning. *The International Journal of Robotics Research*, page 02783649241276017, 2023.

[30] Ana-Maria Marcu, Long Chen, Jan Hünermann, Alice Karnsund, Benoit Hanotte, Prajwal Chidananda, Saurabh Nair, Vijay Badrinarayanan, Alex Kendall, Jamie Shotton, Elahe Arani, and Oleg Sinavski. Lingoqa: Visual question answering for autonomous driving, 2024. URL `https://arxiv.org/abs/2312.14115`.

[31] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204, 2019.

[32] Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. Gaia: a benchmark for general ai assistants. *arXiv preprint arXiv:2311.12983*, 2023.

[33] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, Xiao Wang, Xiaohua Zhai, Thomas Kipf, and Neil Houlsby. Simple open-vocabulary object detection with vision transformers, 2022. URL `https://arxiv.org/abs/2205.06230`.

[34] Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

[35] Wilbert Pumacay, Ishika Singh, Jiafei Duan, Ranjay Krishna, Jesse Thomason, and Dieter Fox. The colosseum: A benchmark for evaluating generalization for robotic manipulation. *arXiv preprint arXiv:2402.08191*, 2024.

[36] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on robot learning*, pages 894–906. PMLR, 2022.

[37] Austin Stone, Ted Xiao, Yao Lu, Keerthana Gopalakrishnan, Kuang-Huei Lee, Quan Vuong, Paul Wohlhart, Sean Kirmani, Brianna Zitkovich, Fei Xia, et al. Open-world object manipulation using pre-trained vision-language models. *arXiv preprint arXiv:2303.00905*, 2023.

[38] Richard Sutton. The bitter lesson. *Incomplete Ideas (blog)*, 13(1):38, 2019.

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

[40] Jordy Van Landeghem, Rubèn Tito, Łukasz Borchmann, Michał Pietruszka, Pawel Joziak, Rafal Powalski, Dawid Jurkiewicz, Mickaël Coustaty, Bertrand Anckaert, Ernest Valveny, et al. Document understanding dataset and evaluation (dude). In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19528–19540, 2023.

[41] Sai Vemprala, Rogerio Bonatti, Arthur Bucker, and Ashish Kapoor. Chatgpt for robotics: Design principles and model abilities, 2023. URL `https://arxiv.org/abs/2306.17582`.

[42] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023.

[43] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022.

[44] Junyang Wang, Yiyang Zhou, Guohai Xu, Pengcheng Shi, Chenlin Zhao, Haiyang Xu, Qinghao Ye, Ming Yan, Ji Zhang, Jihua Zhu, et al. Evaluation and analysis of hallucination in large vision-language models. *arXiv preprint arXiv:2308.15126*, 2023.

[45] Annie Xie, Lisa Lee, Ted Xiao, and Chelsea Finn. Decomposing the generalization gap in imitation learning for visual robotic manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3153–3160. IEEE, 2024.

[46] Eliot Xing, Abhinav Gupta, Sam Powers, and Victoria Dean. Kitchenshift: Evaluating zero-shot generalization of imitation-based policy learning under domain shifts. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.

[47] Peng Xu, Wenqi Shao, Kaipeng Zhang, Peng Gao, Shuo Liu, Meng Lei, Fanqing Meng, Siyuan Huang, Yu Qiao, and Ping Luo. Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models. *arXiv preprint arXiv:2306.09265*, 2023.

[48] Zhenfei Yin, Jiong Wang, Jianjian Cao, Zhelun Shi, Dingning Liu, Mukai Li, Lu Sheng, Lei Bai, Xiaoshui Huang, Zhiyong Wang, et al. Lamm: Language-assisted multi-modal instruction-tuning dataset. *Framework, and Benchmark*, pages 1–37, 2023.

[49] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023.

[50] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567, 2024.

[51] Yuanhan Zhang Bo Li-Songyang Zhang, Wangbo Zhao Yike Yuan Jiaqi Wang, Conghui He Ziwei Liu Kai Chen, Dahua Lin Yuan Liu, and Haodong Duan. Mmbench: Is your multi-modal model an all-around player. *arXiv preprint arXiv:2307.06281*, 2, 2023.

8 Appendix

8.1 Dataset Coverage, Completion Rate, and Additional AMSE Recordings

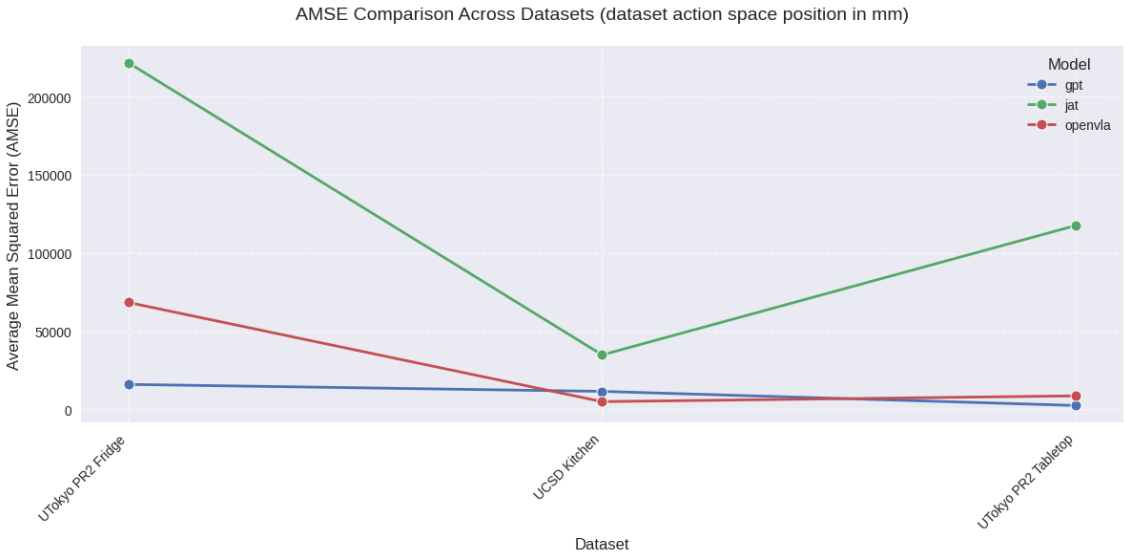

Figure 6: AMSE Across Datasets with Action Space Unit in Millimeter

Table 3: Dataset Coverage and Action Space Types

Dataset Name	Registered Dataset Name	JAT	GPT	OpenVLA	Action Space Type
RT-1 Robot Action	fractal20220817_data	✓			10D (2 pos for base, 1 ang for base, 1 grip, 3 ang for arm, 3 pos for arm)
QT-Opt	kuka	✓			10D (2 pos for base, 1 ang for base, 1 grip, 3 ang for arm, 3 pos for arm)
Berkeley Bridge	bridge	✓			7D (3 pos, 3 ang,1 term)
Freiburg Franka Play	taco_play	✓			–
USC Jaco Play	jaco_play	✓	✓	✓	4D (1 grip, 3 pos)
Berkeley Cable Routing	berkeley_cable_routing	✓	✓	✓	7D (3 ang, 3 pos, 1 term)
Roboturk	roboturk	✓			–
NYU VINN	nyu_door_opening_surprising_effectiveness	✓	✓	✓	8D (1 grip, 3 ang, 3 pos, 1 term)
Austin VIOLA	viola	✓	✓	✓	8D (1 grip, 3 ang, 3 pos, 1 term)
Berkeley Autolab UR5	berkeley_autolab_ur5	✓	✓	✓	8D (1 grip, 3 ang, 3 pos, 1 term)
TOTO Benchmark	toto	✓	✓	✓	7D (3 ang, 3 pos, 1 term)
Language Table	language_table	✓			2D
Columbia PushT	columbia_cairlab_pusht_real	✓	✓	✓	8D (1 grip, 3 ang, 3 pos, 1 term)
NYU ROT	nyu_rot_dataset_converted_externally_to_rlds	✓	✓	✓	7D (3 pos, 3 ang, 1 grip)
Stanford HYDRA	stanford_hydra_dataset_converted_externally_to_rlds	✓	✓	✓	7D (3 pos, 3 ang, 1 grip)
NYU Franka Play	nyu_franka_play_dataset_converted_externally_to_rlds	✓			–
Maniskill	maniskill_dataset_converted_externally_to_rlds	✓			–
Furniture Bench	furniture_bench_dataset_converted_externally_to_rlds	✓			8D (3 pos, 4 quat, 1 grip)
CMU Franka Exploration	cmu_franka_exploration_dataset_converted_externally_to_rlds	✓			–
UCSD Kitchen	ucsd_kitchen_dataset_converted_externally_to_rlds	✓	✓	✓	8D (3 pos, 3 ang, 1 grip, 1 term)
UCSD Pick Place	ucsd_pick_and_place_dataset_converted_externally_to_rlds	✓	✓	✓	4D (3 vel, 1 grip torque)
Austin Sirius	austin_sirius_dataset_converted_externally_to_rlds	✓			–
BC-Z	bc_z	✓			61D (30 pos, 30 ang, 1 grip)
USC Cloth Sim	usc_cloth_sim_converted_externally_to_rlds	✓	✓	✓	4D (3 pos, 1 grip)
Tokyo PR2 Fridge	utokyo_pr2_opening_fridge_converted_externally_to_rlds	✓	✓	✓	8D (3 pos, 3 ang, 1 grip, 1 term)
Tokyo PR2 Tabletop	utokyo_pr2_tabletop_manipulation_converted_externally_to_rlds	✓	✓	✓	8D (3 pos, 3 ang, 1 grip, 1 term)
Saytap	utokyo_saytap_converted_externally_to_rlds	✓			–
UTokyo xArm PickPlace	utokyo_xarm_pick_and_place_converted_externally_to_rlds	✓	✓	✓	7D (3 pos, 3 ang, 1 grip)
UTokyo xArm Bimanual	utokyo_xarm_bimanual_converted_externally_to_rlds	✓	✓		14D (dual arm 7D control)
Berkeley MVP Data	berkeley_mvp_converted_externally_to_rlds	✓			–
Berkeley RPT Data	berkeley_rpt_converted_externally_to_rlds	✓			–
KAIST Nonprehensile	kaist_nonprehensile_converted_externally_to_rlds	✓	✓		20D (3 pos, 3 ang, 7 gain coeff, 7 damping ratio coeff)
Stanford MaskVIT	stanford_mask_vit_converted_externally_to_rlds	✓	✓	✓	5D (3 pos, 1 ang, 1 grip)
LSMO Dataset	tokyo_u_lsmo_converted_externally_to_rlds	✓			–
ConqHose	conq_hose_manipulation	✓	✓	✓	7D (3 pos, 3 ang, 1 grip)
ETH Agent Affordances	eth_agent_affordances	✓	✓	✓	6D (3 vel, 3 ang vel)
Imperial Wrist Cam	imperialcollege_sawyer_wrist_cam	✓	✓	✓	8D (3 pos, 3 ang, 1 grip, 1 term)
Plex RoboSuite	plex_robosuite	✓	✓	✓	7D (6 pose, 1 grip)
DLR Sara Grid Clamp Dataset	dlr_sara_grid_clamp_converted_externally_to_rlds	✓			–
DLR Sara Pour Dataset	dlr_sara_pour_converted_externally_to_rlds	✓			–
DLR Wheelchair Shared Control	dlr_edan_shared_control_converted_externally_to_rlds	✓			–
ASU TableTop Manipulationl	asu_table_top_converted_externally_to_rlds	✓			–
CMU Franka Pick-Insert Data	iamlab_cmu_pickup_insert_converted_externally_to_rlds	✓			–
Austin Mutex	utaustin_mutex	✓			–
Stanford Robocook	stanford_robocook_converted_externally_to_rlds	✓			–
CMU Play Fusion	cmu_play_fusion	✓			–
CMU Stretch	cmu_stretch	✓			–
RECON	berkeley_gnm_recon	✓			–
CoryHall	berkeley_gnm_cory_hall	✓			–
SACSoN	berkeley_gnm_sac_son	✓			–
DobbE	dobbe	✓			–
IO-AI Office PicknPlace	io_ai_tech	✓			–
RoboSet	robo_set	✓			–

pos: position, orient: orientation, grip: gripper, term: terminate, vel: velocity, ang: angular, quat: quaternion
Some datasets have been excluded due to space constraints or incomplete information

Table 4: Task Completion Rates Across Models and Datasets

Dataset Name	GPT	OpenVLA	JAT
Jaco Play	0.917%	29.358%	0.000%
Berkeley Cable Routing	0.000%	0.000%	0.000%
NYU Door Opening	0.000%	0.000%	0.000%
VIOLA	0.000%	0.000%	0.000%
Berkeley Autolab UR5	1.923%	0.000%	0.000%
TOTO	0.000%	0.000%	0.000%
Columbia PushT	0.000%	0.000%	0.000%
NYU ROT	7.143%	0.000%	0.000%
Stanford HYDRA	0.833%	0.000%	0.000%
UCSD Kitchen	0.000%	0.000%	0.000%
UCSD Pick Place	0.000%	0.000%	0.000%
USC Cloth Sim	0.000%	0.000%	0.000%
Tokyo PR2 Fridge	0.000%	0.000%	0.000%
Tokyo PR2 Tabletop	0.000%	0.000%	0.000%
UTokyo xArm Pick-Place	0.000%	0.000%	0.000%
Stanford MaskVIT	0.000%	0.000%	0.000%
ETH Agent Affordances	0.000%	0.000%	0.000%
Imperial Sawyer	0.000%	0.000%	0.000%
ConqHose	0.000%	0.000%	0.000%
Plex RoboSuite	0.000%	0.000%	0.000%

Success rates reported as percentage of episodes where final action matched ground truth