Benchmarking Vision, Language, & Action Models on Robotic Learning Tasks

Pranav Guruprasad *, Harshvardhan Sikka *, Jaewoo Song, Yangyue Wang, Paul Liang

Posted Date: 7 November 2024

doi: 10.20944/preprints202411.0494.v1

Keywords: benchmark; machine learning; vision language model; large language model; vision language action; vla; robotic learning; offline RL; robotics; control

 doi:10.20944/preprints202411.0494.v1

Article

Benchmarking Vision, Language, & Action Models on Robotic Learning Tasks

Pranav Guruprasad [1,2,*], Harshvardhan Sikka [1,2,3,*], Jaewoo Song [1], Yangyue Wang [1] and Paul Pu Liang [4]

[1] Manifold Research
[2] Metarch.ai
[3] Georgia Tech
[4] MIT
* Correspondence: pranav@metarch.ai (P.G.); harshsikka@gatech.edu (H.S.)

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.

Keywords: benchmark; machine learning; vision language model; large language model; vision language action; vla; robotic learning; offline RL; robotics; control

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 [1,2], struggle with spatial variations in object configurations [3], perform poorly under variable lighting or occlusion [4], and show degraded performance when interacting with novel objects or in cluttered environments [5,6]. 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 [7,8], sophisticated reasoning about interactions between objects and agents [9–11], software development [12], 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

 doi:10.20944/preprints202411.0494.v1

of data volume, computational resources, and model complexity [13,14]. This pattern aligns with historical observations suggesting that general-purpose models efficiently utilizing computational resources tend to outperform specialized solutions [15]. The advantages of unified sequence 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 [16,17]. 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 (Ahn et al., 2022; Driess et al., 2023; Vemprala et al., 2023). 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 [18–20]. These components have also been integrated into planning systems [10,21]. 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 [3], cotraining [6], or fine-tuning [1,22,23]. 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 generalization
- 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, 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 [24] represents one of the first systematic attempts to evaluate multimodal learning across diverse domains, spanning healthcare, robotics, affective computing, and finance. Similar to our

 doi:10.20944/preprints202411.0494.v1

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 [25]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 [26], OK-VQA [27], MSCOCO [28], and GQA [29] to more comprehensive evaluation frameworks. Recent benchmarks span various capabilities, from basic OCR to adversarial robustness and hallucination detection (e.g., POPE [30] and HaELM [31]). More holistic evaluations have emerged through benchmarks like LAMM [32], LVLM-eHub [33], SEED [34], MMBench [35], and MM-Vet [36]. Specialized benchmarks such as MathVista [37] focus on specific domains like mathematical reasoning, while GAIA [38] 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 [39] addresses this gap by providing a systematic evaluation framework focused on robotic manipulation, evaluating generalization across 14 different environmental perturbations. Similar efforts include FactorWorld [5], which examines 11 variation factors across 19 tasks, and KitchenShift [40], which evaluates zero-shot generalization across 7 variation factors in kitchen environments.Several other specialized robotics benchmarks have emerged: RLBench [41] offers a suite of 100 manipulation tasks in simulation; RAVENS [42] focuses on vision-based manipulation; and FurnitureBench [43] provides reproducible real-world benchmarks for long-horizon complex manipulation. LIBERO [44] offers benchmarks for knowledge transfer in lifelong robot learning, while FMB [45] emphasizes generalizable robotic learning across complex tasks. Recent work has also introduced DUDE [46] for robotic document manipulation and ProcTHOR[47] 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.

 doi:10.20944/preprints202411.0494.v1

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 Figure [X]. 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, along with details about our evaluation split creation methodology.

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, 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. The complete list of included and excluded datasets, along with the specific rationale for each curation decision, is provided in Appendix [X].

 doi:10.20944/preprints202411.0494.v1

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 in robotic control tasks.

JAT [48] 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 [49] 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:

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

 doi:10.20944/preprints202411.0494.v1

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 comprehensive performance analysis:

1. **Average MSE (AMSE):** Computed as the mean MSE across all trajectories in a dataset, AMSE enables direct comparison of model performance across different datasets and architectures.
2. **Normalized AMSE (NAMSE):** Calculated as $(timestep\_MSE - min\_MSE)/(max\_MSE - min\_MSE)$, this metric normalizes predictions to each model's prediction range, facilitating more equitable cross-dataset for a single model comparisons by accounting for different scales in model outputs.
3. **Completion Rate:** We assess successful completion by comparing final predicted actions with ground truth final actions. 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:

 doi:10.20944/preprints202411.0494.v1

  - 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 occured with 2 datasets in particular, conq_hose, 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.

 doi:10.20944/preprints202411.0494.v1

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

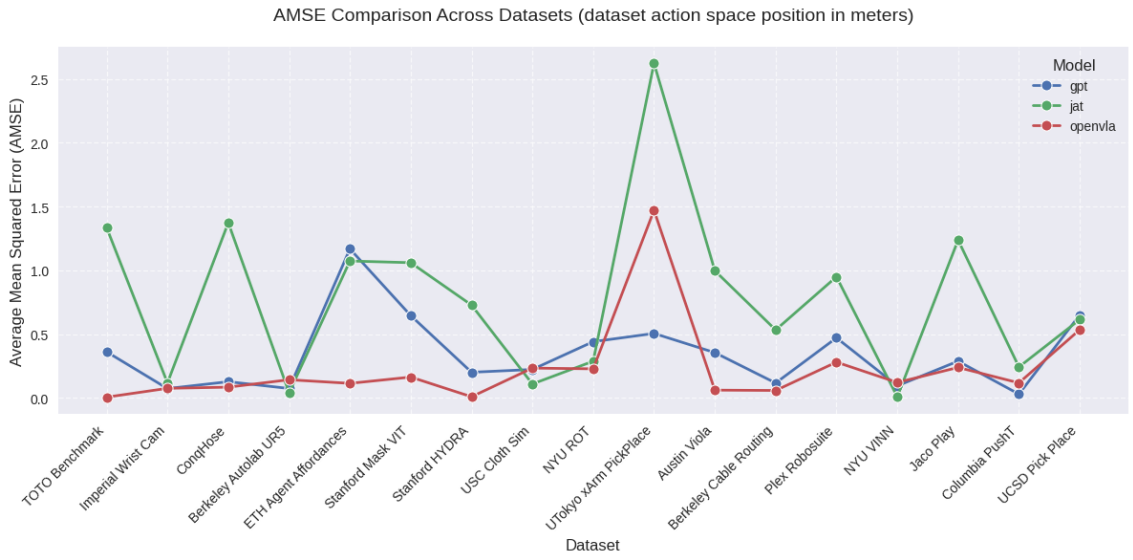

Figure 1. AMSE Across All Datasets.

 doi:10.20944/preprints202411.0494.v1

Table 1. Dataset Coverage and Action Space Characteristics

Dataset Name	Registered Dataset Name	In Pretraining	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

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

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 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)

 doi:10.20944/preprints202411.0494.v1

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.
- 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

 doi:10.20944/preprints202411.0494.v1

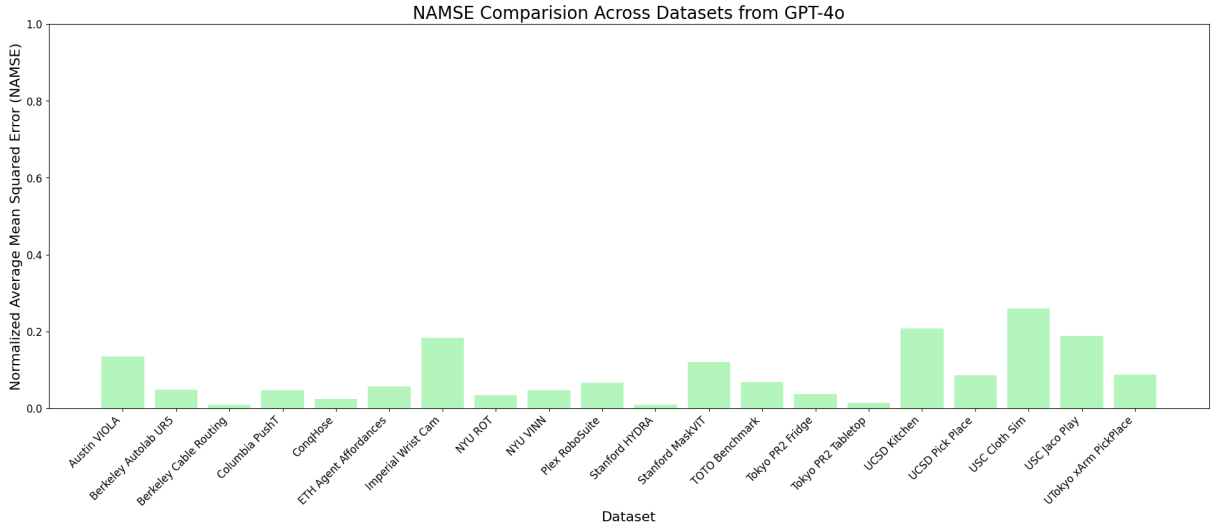

Figure 2. Normalized AMSE For GPT4o

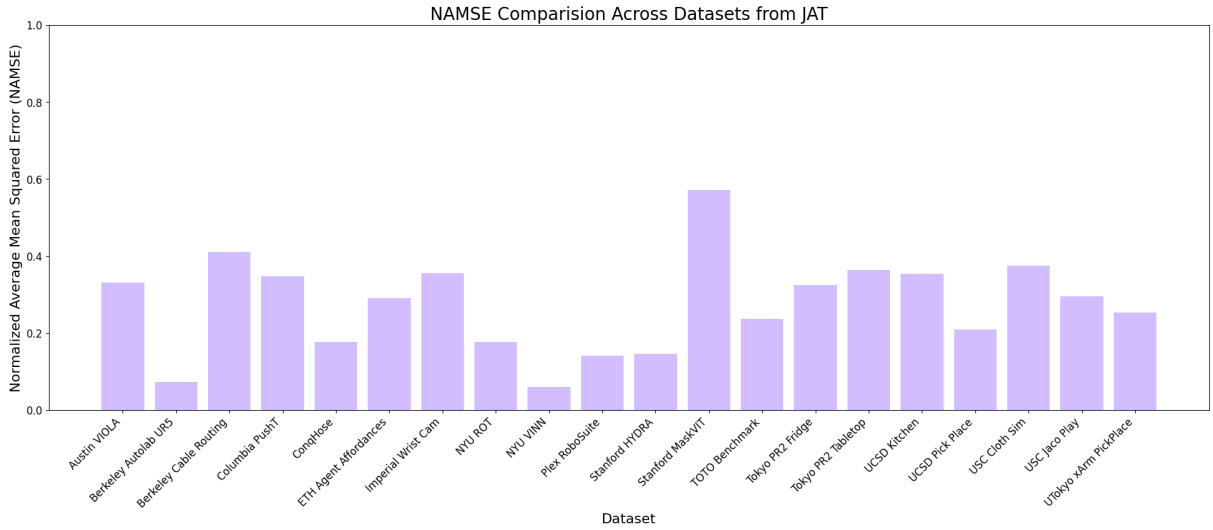

Figure 3. Normalized AMSE For JAT

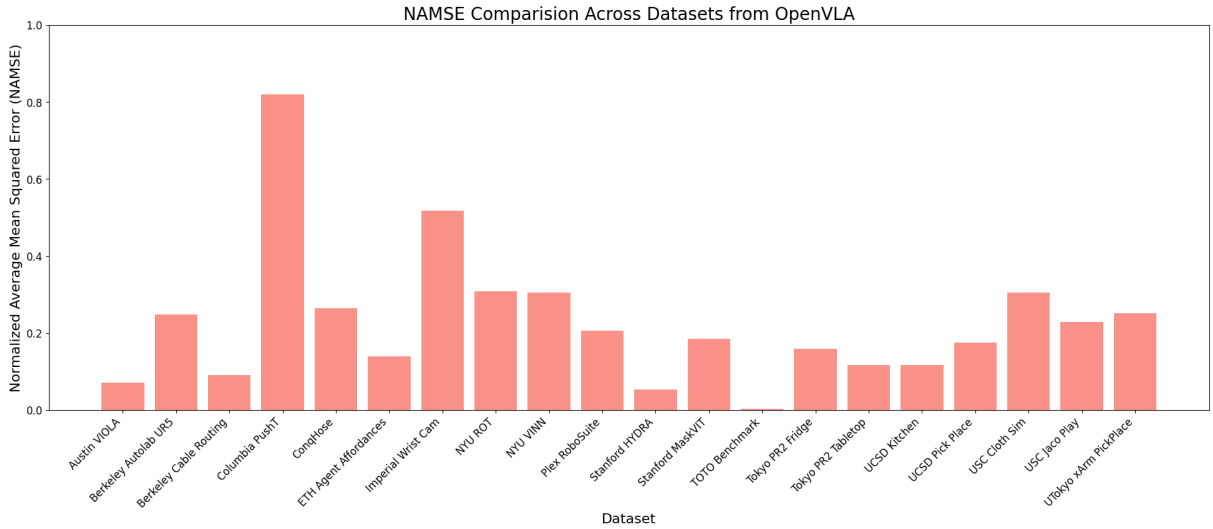

Figure 4. Normalized AMSE For OpenVLA

 doi:10.20944/preprints202411.0494.v1

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
- 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 certain 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 ucsd_kitchen_dataset (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:

- Kitchen manipulation tasks and complex 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

 doi:10.20944/preprints202411.0494.v1

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.

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.

 doi:10.20944/preprints202411.0494.v1

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. Conclusions

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.

 doi:10.20944/preprints202411.0494.v1

Appendix A.

Appendix A.1. Dataset Coverage, Completion Rate, and Additional AMSE Recordings

Table A1. 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 A2. 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.

 doi:10.20944/preprints202411.0494.v1

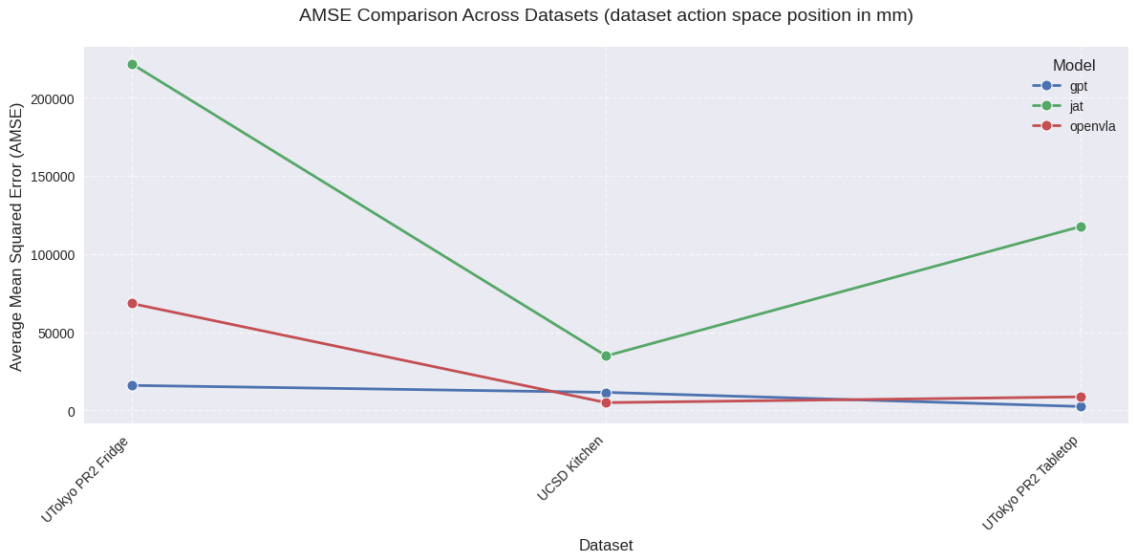

Figure A1. AMSE Across Datasets with Action Space Unit in Millimeter

References

1. Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Chen, X.; Choromanski, K.; Ding, T.; Driess, D.; Dubey, A.; Finn, C.; others. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818* **2023**.
2. Walke, H.R.; Black, K.; Zhao, T.Z.; Vuong, Q.; Zheng, C.; Hansen-Estruch, P.; He, A.W.; Myers, V.; Kim, M.J.; Du, M.; others. Bridgedata v2: A dataset for robot learning at scale. Conference on Robot Learning. PMLR, 2023, pp. 1723–1736.
3. Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Dabis, J.; Finn, C.; Gopalakrishnan, K.; Hausman, K.; Herzog, A.; Hsu, J.; others. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817* **2022**.
4. Chi, C.; Xu, Z.; Feng, S.; Cousineau, E.; Du, Y.; Burchfiel, B.; Tedrake, R.; Song, S. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research* **2023**, p. 02783649241273668.
5. Xie, A.; Lee, L.; Xiao, T.; Finn, C. Decomposing the generalization gap in imitation learning for visual robotic manipulation. 2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2024, pp. 3153–3160.
6. Team, O.M.; Ghosh, D.; Walke, H.; Pertsch, K.; Black, K.; Mees, O.; Dasari, S.; Hejna, J.; Kreiman, T.; Xu, C.; others. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213* **2024**.
7. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.Y.; others. Segment anything. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4015–4026.
8. Minderer, M.; Gritsenko, A.; Stone, A.; Neumann, M.; Weissenborn, D.; Dosovitskiy, A.; Mahendran, A.; Arnab, A.; Dehghani, M.; Shen, Z.; Wang, X.; Zhai, X.; Kipf, T.; Houlsby, N. Simple Open-Vocabulary Object Detection with Vision Transformers, 2022, [arXiv:cs.CV/2205.06230].
9. Alayrac, J.B.; Miech, A.; Laptev, I.; Sivic, J.; others. Multi-Task Learning of Object States and State-Modifying Actions from Web Videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2024**.
10. Driess, D.; Xia, F.; Sajjadi, M.S.; Lynch, C.; Chowdhery, A.; Ichter, B.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; others. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378* **2023**.
11. Wang, J.; Yang, Z.; Hu, X.; Li, L.; Lin, K.; Gan, Z.; Liu, Z.; Liu, C.; Wang, L. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100* **2022**.
12. Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; Pinto, H.P.D.O.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; others. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374* **2021**.
13. Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T.B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361* **2020**.

 doi:10.20944/preprints202411.0494.v1

14. Hoffmann, J.; Borgeaud, S.; Mensch, A.; Buchatskaya, E.; Cai, T.; Rutherford, E.; Casas, D.d.L.; Hendricks, L.A.; Welbl, J.; Clark, A.; others. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556* **2022**.
15. Sutton, R. The bitter lesson. *Incomplete Ideas (blog)* **2019**, *13*, 38.
16. Collaboration, O.X.E.; O'Neill, A.; Rehman, A.; Gupta, A.; Maddukuri, A.; Gupta, A.; Padalkar, A.; Lee, A.; Pooley, A.; Gupta, A.; Mandlekar, A.; Jain, A.; Tung, A.; Bewley, A.; Herzog, A.; Irpan, A.; Khazatsky, A.; Rai, A.; Gupta, A.; Wang, A.; Kolobov, A.; Singh, A.; Garg, A.; Kembhavi, A.; Xie, A.; Brohan, A.; Raffin, A.; Sharma, A.; Yavary, A.; Jain, A.; Balakrishna, A.; Wahid, A.; Burgess-Limerick, B.; Kim, B.; Schölkopf, B.; Wulfe, B.; Ichter, B.; Lu, C.; Xu, C.; Le, C.; Finn, C.; Wang, C.; Xu, C.; Chi, C.; Huang, C.; Chan, C.; Agia, C.; Pan, C.; Fu, C.; Devin, C.; Xu, D.; Morton, D.; Driess, D.; Chen, D.; Pathak, D.; Shah, D.; Büchler, D.; Jayaraman, D.; Kalashnikov, D.; Sadigh, D.; Johns, E.; Foster, E.; Liu, F.; Ceola, F.; Xia, F.; Zhao, F.; Frujeri, F.V.; Stulp, F.; Zhou, G.; Sukhatme, G.S.; Salhotra, G.; Yan, G.; Feng, G.; Schiavi, G.; Berseth, G.; Kahn, G.; Yang, G.; Wang, G.; Su, H.; Fang, H.S.; Shi, H.; Bao, H.; Amor, H.B.; Christensen, H.I.; Furuta, H.; Bharadhwaj, H.; Walke, H.; Fang, H.; Ha, H.; Mordatch, I.; Radosavovic, I.; Leal, I.; Liang, J.; Abou-Chakra, J.; Kim, J.; Drake, J.; Peters, J.; Schneider, J.; Hsu, J.; Vakil, J.; Bohg, J.; Bingham, J.; Wu, J.; Gao, J.; Hu, J.; Wu, J.; Wu, J.; Sun, J.; Luo, J.; Gu, J.; Tan, J.; Oh, J.; Wu, J.; Lu, J.; Yang, J.; Malik, J.; Silvério, J.; Hejna, J.; Booher, J.; Tompson, J.; Yang, J.; Salvador, J.; Lim, J.J.; Han, J.; Wang, K.; Rao, K.; Pertsch, K.; Hausman, K.; Go, K.; Gopalakrishnan, K.; Goldberg, K.; Byrne, K.; Oslund, K.; Kawaharazuka, K.; Black, K.; Lin, K.; Zhang, K.; Ehsani, K.; Lekkala, K.; Ellis, K.; Rana, K.; Srinivasan, K.; Fang, K.; Singh, K.P.; Zeng, K.H.; Hatch, K.; Hsu, K.; Itti, L.; Chen, L.Y.; Pinto, L.; Fei-Fei, L.; Tan, L.; Fan, L.J.; Ott, L.; Lee, L.; Weihs, L.; Chen, M.; Lepert, M.; Memmel, M.; Tomizuka, M.; Itkina, M.; Castro, M.G.; Spero, M.; Du, M.; Ahn, M.; Yip, M.C.; Zhang, M.; Ding, M.; Heo, M.; Srirama, M.K.; Sharma, M.; Kim, M.J.; Kanazawa, N.; Hansen, N.; Heess, N.; Joshi, N.J.; Suenderhauf, N.; Liu, N.; Palo, N.D.; Shafiullah, N.M.M.; Mees, O.; Kroemer, O.; Bastani, O.; Sanketi, P.R.; Miller, P.T.; Yin, P.; Wohlhart, P.; Xu, P.; Fagan, P.D.; Mitrano, P.; Sermanet, P.; Abbeel, P.; Sundaresan, P.; Chen, Q.; Vuong, Q.; Rafailov, R.; Tian, R.; Doshi, R.; Mart'in-Mart'in, R.; Baijal, R.; Scalise, R.; Hendrix, R.; Lin, R.; Qian, R.; Zhang, R.; Mendonca, R.; Shah, R.; Hoque, R.; Julian, R.; Bustamante, S.; Kirmani, S.; Levine, S.; Lin, S.; Moore, S.; Bahl, S.; Dass, S.; Sonawani, S.; Tulsiani, S.; Song, S.; Xu, S.; Haldar, S.; Karamcheti, S.; Adebola, S.; Guist, S.; Nasiriany, S.; Schaal, S.; Welker, S.; Tian, S.; Ramamoorthy, S.; Dasari, S.; Belkhale, S.; Park, S.; Nair, S.; Mirchandani, S.; Osa, T.; Gupta, T.; Harada, T.; Matsushima, T.; Xiao, T.; Kollar, T.; Yu, T.; Ding, T.; Davchev, T.; Zhao, T.Z.; Armstrong, T.; Darrell, T.; Chung, T.; Jain, V.; Kumar, V.; Vanhoucke, V.; Zhan, W.; Zhou, W.; Burgard, W.; Chen, X.; Chen, X.; Wang, X.; Zhu, X.; Geng, X.; Liu, X.; Liangwei, X.; Li, X.; Pang, Y.; Lu, Y.; Ma, Y.J.; Kim, Y.; Chebotar, Y.; Zhou, Y.; Zhu, Y.; Wu, Y.; Xu, Y.; Wang, Y.; Bisk, Y.; Dou, Y.; Cho, Y.; Lee, Y.; Cui, Y.; Cao, Y.; Wu, Y.H.; Tang, Y.; Zhu, Y.; Zhang, Y.; Jiang, Y.; Li, Y.; Li, Y.; Iwasawa, Y.; Matsuo, Y.; Ma, Z.; Xu, Z.; Cui, Z.J.; Zhang, Z.; Fu, Z.; Lin, Z. Open X-Embodiment: Robotic Learning Datasets and RT-X Models. https://arxiv.org/abs/2310.08864, 2023.
17. Khazatsky, A.; Pertsch, K.; Nair, S.; Balakrishna, A.; Dasari, S.; Karamcheti, S.; Nasiriany, S.; Srirama, M.K.; Chen, L.Y.; Ellis, K.; others. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945* **2024**.
18. Shridhar, M.; Manuelli, L.; Fox, D. Cliport: What and where pathways for robotic manipulation. Conference on robot learning. PMLR, 2022, pp. 894–906.
19. Nair, S.; Rajeswaran, A.; Kumar, V.; Finn, C.; Gupta, A. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601* **2022**.
20. Karamcheti, S.; Nair, S.; Chen, A.S.; Kollar, T.; Finn, C.; Sadigh, D.; Liang, P. Language-driven representation learning for robotics. *arXiv preprint arXiv:2302.12766* **2023**.
21. Stone, A.; Xiao, T.; Lu, Y.; Gopalakrishnan, K.; Lee, K.H.; Vuong, Q.; Wohlhart, P.; Kirmani, S.; Zitkovich, B.; Xia, F.; others. Open-world object manipulation using pre-trained vision-language models. *arXiv preprint arXiv:2303.00905* **2023**.
22. Kim, M.J.; Pertsch, K.; Karamcheti, S.; Xiao, T.; Balakrishna, A.; Nair, S.; Rafailov, R.; Foster, E.; Lam, G.; Sanketi, P.; others. OpenVLA: An Open-Source Vision-Language-Action Model. *arXiv preprint arXiv:2406.09246* **2024**.
23. Marcu, A.M.; Chen, L.; Hünermann, J.; Karnsund, A.; Hanotte, B.; Chidananda, P.; Nair, S.; Badrinarayanan, V.; Kendall, A.; Shotton, J.; Arani, E.; Sinavski, O. LingoQA: Visual Question Answering for Autonomous Driving, 2024, [arXiv:cs.RO/2312.14115].

 doi:10.20944/preprints202411.0494.v1

24. Liang, P.P.; Lyu, Y.; Fan, X.; Wu, Z.; Cheng, Y.; Wu, J.; Chen, L.; Wu, P.; Lee, M.A.; Zhu, Y.; others. Multibench: Multiscale benchmarks for multimodal representation learning. *Advances in neural information processing systems* **2021**, *2021*, 1.
25. Yue, X.; Ni, Y.; Zhang, K.; Zheng, T.; Liu, R.; Zhang, G.; Stevens, S.; Jiang, D.; Ren, W.; Sun, Y.; others. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 9556–9567.
26. Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Zitnick, C.L.; Parikh, D. Vqa: Visual question answering. Proceedings of the IEEE international conference on computer vision, 2015, pp. 2425–2433.
27. Marino, K.; Rastegari, M.; Farhadi, A.; Mottaghi, R. Ok-vqa: A visual question answering benchmark requiring external knowledge. Proceedings of the IEEE/cvf conference on computer vision and pattern recognition, 2019, pp. 3195–3204.
28. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. Springer, 2014, pp. 740–755.
29. Hudson, D.A.; Manning, C.D. Gqa: A new dataset for real-world visual reasoning and compositional question answering. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 6700–6709.
30. Li, Y.; Du, Y.; Zhou, K.; Wang, J.; Zhao, W.X.; Wen, J.R. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355* **2023**.
31. Wang, J.; Zhou, Y.; Xu, G.; Shi, P.; Zhao, C.; Xu, H.; Ye, Q.; Yan, M.; Zhang, J.; Zhu, J.; others. Evaluation and analysis of hallucination in large vision-language models. *arXiv preprint arXiv:2308.15126* **2023**.
32. Yin, Z.; Wang, J.; Cao, J.; Shi, Z.; Liu, D.; Li, M.; Sheng, L.; Bai, L.; Huang, X.; Wang, Z.; others. LAMM: Language-Assisted Multi-Modal Instruction-Tuning Dataset. *Framework, and Benchmark* **2023**, pp. 1–37.
33. Xu, P.; Shao, W.; Zhang, K.; Gao, P.; Liu, S.; Lei, M.; Meng, F.; Huang, S.; Qiao, Y.; Luo, P. Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models. *arXiv preprint arXiv:2306.09265* **2023**.
34. Ge, Y.; Ge, Y.; Zeng, Z.; Wang, X.; Shan, Y. Planting a seed of vision in large language model. *arXiv preprint arXiv:2307.08041* **2023**.
35. Zhang, Y.Z.B.L.S.; Wang, W.Z.Y.Y.J.; Chen, C.H.Z.L.K.; Liu, D.L.Y.; Duan, H. Mmbench: Is your multi-modal model an all-around player. *arXiv preprint arXiv:2307.06281* **2023**, *2*.
36. Yu, W.; Yang, Z.; Li, L.; Wang, J.; Lin, K.; Liu, Z.; Wang, X.; Wang, L. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490* **2023**.
37. Lu, P.; Bansal, H.; Xia, T.; Liu, J.; Li, C.; Hajishirzi, H.; Cheng, H.; Chang, K.W.; Galley, M.; Gao, J. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255* **2023**.
38. Mialon, G.; Fourrier, C.; Swift, C.; Wolf, T.; LeCun, Y.; Scialom, T. Gaia: a benchmark for general ai assistants. *arXiv preprint arXiv:2311.12983* **2023**.
39. Pumacay, W.; Singh, I.; Duan, J.; Krishna, R.; Thomason, J.; Fox, D. The colosseum: A benchmark for evaluating generalization for robotic manipulation. *arXiv preprint arXiv:2402.08191* **2024**.
40. Xing, E.; Gupta, A.; Powers, S.; Dean, V. Kitchenshift: Evaluating zero-shot generalization of imitation-based policy learning under domain shifts. NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications, 2021.
41. James, S.; Ma, Z.; Arrojo, D.R.; Davison, A.J. RLBench: The Robot Learning Benchmark & Learning Environment. arXiv e-prints, art. *arXiv preprint arXiv:1909.12271* **2019**.
42. Huang, J.; Ping, W.; Xu, P.; Shoeybi, M.; Chang, K.C.C.; Catanzaro, B. Raven: In-context learning with retrieval augmented encoder-decoder language models. *arXiv preprint arXiv:2308.07922* **2023**.
43. Heo, M.; Lee, Y.; Lee, D.; Lim, J.J. Furniturebench: Reproducible real-world benchmark for long-horizon complex manipulation. *arXiv preprint arXiv:2305.12821* **2023**.
44. Liu, B.; Zhu, Y.; Gao, C.; Feng, Y.; Liu, Q.; Zhu, Y.; Stone, P. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems* **2024**, *36*.
45. Luo, J.; Xu, C.; Liu, F.; Tan, L.; Lin, Z.; Wu, J.; Abbeel, P.; Levine, S. FMB: A functional manipulation benchmark for generalizable robotic learning. *The International Journal of Robotics Research* **2023**, p. 02783649241276017.

 doi:10.20944/preprints202411.0494.v1

46. Van Landeghem, J.; Tito, R.; Borchmann, Ł.; Pietruszka, M.; Joziak, P.; Powalski, R.; Jurkiewicz, D.; Coustaty, M.; Anckaert, B.; Valveny, E.; others. Document understanding dataset and evaluation (dude). Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 19528–19540.
47. Deitke, M.; VanderBilt, E.; Herrasti, A.; Weihs, L.; Ehsani, K.; Salvador, J.; Han, W.; Kolve, E.; Kembhavi, A.; Mottaghi, R. ProcTHOR: Large-Scale Embodied AI Using Procedural Generation. *Advances in Neural Information Processing Systems* **2022**, *35*, 5982–5994.
48. Gallouédec, Q.; Beeching, E.; Romac, C.; Dellandréa, E. Jack of All Trades, Master of Some, a Multi-Purpose Transformer Agent, 2024, [arXiv:cs.AI/2402.09844].
49. Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; others. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* **2023**.