

3D-CAVLA: Leveraging Depth and 3D Context to Generalize Vision–Language Action Models for Unseen Tasks

Anonymous CVPR submission

Paper ID 12

Abstract

Robotic manipulation in 3D requires learning an N degree-of-freedom joint space trajectory of a robot manipulator. Robots must possess semantic and visual perception abilities to transform real-world mappings of their workspace into the low-level control necessary for object manipulation. Recent work has demonstrated the capabilities of fine-tuning large Vision-Language Models (VLMs) to learn the mapping between RGB images, language instructions, and joint space control. These models typically take as input RGB images of the workspace and language instructions, and are trained on large datasets of teleoperated robot demonstrations. In this work, we explore methods to improve the scene context awareness of a popular recent Vision-Language-Action model by integrating chain-of-thought reasoning, depth perception, and task-oriented region of interest detection. Our experiments in the LIBERO simulation environment show that our proposed model, 3D-CAVLA, improves the success rate across various LIBERO task suites, achieving an average success rate of 98.1%. We also evaluate the zero-shot capabilities of our method, demonstrating that 3D scene awareness leads to robust learning and adaptation for completely unseen tasks. 3D-CAVLA achieves an absolute improvement of 8.8% on unseen tasks. We will open-source our code and the unseen tasks dataset to promote community-driven research.

1. Introduction

The ability to perceive the environment, respond dynamically, and manipulate objects effectively remains a challenging task in robotics. Humans demonstrate this capability effortlessly, emerging from extensive experiential learning during adolescence, where individuals develop visual, reasoning and manipulation skills necessary for interacting with both familiar and novel scenarios. Replicating this robust adaptability in robots is inherently difficult, though recent advancements in artificial intelligence, particularly

in vision and language understanding, have shown promising progress. Vision-Language Models (VLMs), such as ChatGPT, leverage extensive pre-training on internet-scale data, enabling them to interpret real-world images, comprehend conversations, and generate contextually relevant responses. These models have since been used in tasks such as visual question answering [37, 42], visual grounding [51], and task planning [20, 44]—applications that are directly relevant to the field of robotics.

Recent works have explored Vision-Language-Action models (VLAs) which modify VLMs to output robot joint space parameters instead of text tokens [4, 5]. When pre-trained on diverse real-world datasets and fine-tuned on high-quality teleoperated demonstrations, VLAs demonstrate high success rates ($\approx 95\%$) on in-distribution tasks such as “scoop pretzels into bowl.” VLAs typically use RGB images and text instructions as inputs and learn a policy to predict N DOF joint angles at each step required for task execution. A notable recent advancement, OpenVLA-OFT [23], further integrates proprioceptive robot joint-state parameters, concatenating it with visual and textual features from the current timestep. While the performance is impressive for in-distribution tasks, a detailed analysis into the behaviour of these models on unseen tasks has been lacking. Additional sensor modalities have the potential to further improve spatial and logical reasoning of VLAs necessary to generalize to unseen tasks.

In this work, we enhance the architecture introduced by OpenVLA-OFT by exploring effective modifications to existing modalities boosting spatial and contextual understanding. Specifically, we introduce chain-of-thought style narrative prompts to enrich task context, 3D features derived from workspace point clouds to enhance spatial perception, and task-oriented region of interest pooling to effectively focus on visually pertinent patches for each task. Our proposed network, 3D-CAVLA is benchmarked against popular VLAs in the LIBERO benchmark tasks as well as evaluated on unseen tasks to demonstrate improved generalization. Our contributions include:

1. Integrating chain-of-thought prompts and region-of-

interest pooling to learn effective vision-language embeddings for task execution. We introduce 3D point cloud derived depth features into policy training to boost LIBERO in-distribution success rate to 98.1%

2. Benchmarking against existing state-of-the-art approaches for zero-shot language guided object manipulation. Our proposed model, 3D-CAVLA, shows an absolute improvement of 8.8% on 10 novel tasks designed within the LIBERO simulation environment.

2. Related Works

Foundational Models in Robotics. LLMs can generate high-level robotic execution plans based on task inputs and environmental context [19, 45]. However, a recurring challenge with LLMs is their tendency to hallucinate, generating plans that are not physically feasible [40]. To enhance robustness, LLMs require real-world grounding, which can be achieved through feedback from the environment [2, 20, 44], integration with visual perception systems [11, 27, 62], or human-in-the-loop interventions such as question-answering [32, 60]. VLMs, trained on vast image-text datasets, excel at visual reasoning tasks [56] and have been applied to a range of robotics grounding problems such as encoding 3D semantic memory [13, 39], vision-based robot pose estimation [12], guiding object manipulation based on language instructions [43], and enabling robotic navigation [14, 16].

Vision-Language Action Models. VLMs pretrained on internet-scale real-world data possess a vast knowledge base. They can be fine-tuned using robot demonstration datasets, which include images and language instructions, to directly predict robotic joint parameters in the action space [1, 4]. An N degree of freedom robot requires N variables to define its position at any given time step. VLAs are trained on large datasets of robotic demonstration videos and language instructions to predict these N variables at each time step during robot manipulation. Early VLAs demonstrated strong performance in simulation and single robot manipulation [18, 25]. However, many of these models are limited by their closed-source nature or extremely large parameter sizes [5, 10]. Modular systems that integrate planning, grounding, control, and feedback mechanisms are emerging as promising strategies for more robust and adaptable robotic automation [30, 31]. OpenVLA [22], a representative open-source autoregressive VLA, stands out as one of the first approaches to release a compute efficient and scalable VLA with a moderate parameter size ($\approx 7B$), specifically fine-tuned on robot demonstration data from the Open-X Embodiment corpus [34]. Building on top of OpenVLA, OpenVLA-OFT [23] further improves inference efficiency and task performance by incorporating parallel decoding, action chunking, continuous action representations, with an L1 regression objective.

Improving Generalization of VLAs. Recent studies highlight the scaling challenges of directly translating visual frames and language instructions into robot joint states, particularly as the volume of task demonstrations increases [9]. To address these limitations, various approaches have been proposed to enhance the generalization capabilities of Vision-Language Actions (VLAs) for out-of-domain tasks. One prominent direction involves a dual-stage pipeline: an initial pre-training phase where multi-modal encoders are trained with self-supervision using unlabeled human task demonstrations and diverse video planning datasets [26, 29, 55]. This stage aims to learn robust representations without relying on explicit action labels. Complementing this, some works employ teacher-student frameworks to refine action policies. Here, a teacher model leverages reinforcement learning to learn robotic trajectories, which are subsequently distilled into a student model for pose prediction [47]. Other approaches integrate pre-trained models such as CLIP [38]. For instance, [21] uses CLIP’s visual and textual encoders to associate RGB frames with instructions and textual actions during pre-training. Fine-tuning then focuses on selecting actions from a fixed set of classes at each timestep. Similarly, DynaMo [7] adopts a self-supervised strategy, employing both forward and inverse dynamics models to train visual encoders for future observation prediction, enabling robust action forecasting. To improve task execution success rates, researchers have explored integrating proprioception and feedback mechanisms to dynamically correct erroneous actions [28, 49]. Depth information has also proven valuable for robotic manipulation, as it enhances the model’s geometric understanding and spatial reasoning [46]. However, a key limitation in the generalization of VLAs is their reliance on direct input-output mappings without intermediate reasoning. To enhance reasoning capabilities, recent works adopt chain-of-thought prompting, encouraging step-by-step thinking grounded in language, visual observations, and physical actions. Progress has been made by incorporating intermediate reasoning steps such as textual descriptions [58], keypoints [52], or subgoal images [61], which provide structured guidance for planning and action prediction.

3. Methodology

Recent VLAs endow robots with free-form language-following capabilities. We build our model based on the OpenVLA-OFT [23], which reports impressive performance on the LIBERO simulation environment. We first summarize this baseline architecture before detailing our additions, which yields the **3D Context Aware Vision-Language Action model (3D-CAVLA)**.

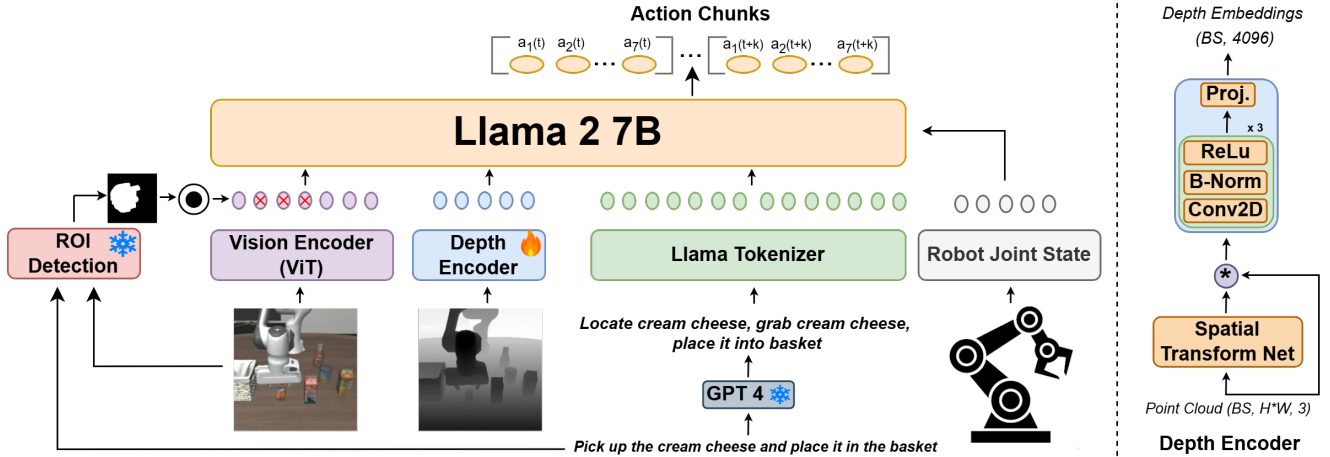


Figure 1. Our proposed model, 3D-CAVLA, integrates chain-of-thought style narrative task descriptions, depth embeddings and Region Of Interest (ROI) pooling to improve the scene awareness of vision-language-action modeling. While GPT4 and ROI Detection are frozen components, our depth encoder is a lightweight PointNet [36] inspired trainable network with spatial invariance transformation, convolution blocks and linear projections to project the embeddings to match the input dimensions of LLaMA 2 7B [50]

3.1. OpenVLA-OFT

OpenVLA-OFT [23] builds on top of OpenVLA [22] and consists of vision, language and robot joint state encoders with optional feature-wise linear modulation (FiLM) layers [35] to improve vision-language feature extraction. The model can be trained on video demonstrations of tele-operated robots completing a task described by a text instruction. Authors use a combination of SigLIP [59] and DinoV2 [33] vision encoders to obtain patch level image embeddings for images captured through the robot’s end effector camera and a stationary 3rd person camera. The task instruction is tokenized and transformed into text embeddings using the LLM’s tokenizer. Robot proprioception, which consists of 8 dimensional joint and gripper states pass through MLP layers. The vision, language and joint embeddings are projected to match the input dimension of the LLM to be fine-tuned. For efficient training, the authors implement LoRA [15] based finetuning which only modifies a small fraction of the trainable parameters by learning trainable projection matrices of larger dimensional inner layers of the LLM. OpenVLA-OFT demonstrates high success rate on seen tasks due to three key features: (i) *parallel decoding* in place of autoregressive prediction for faster inference, (ii) *action chunking* that predicts the next K actions jointly, and (iii) *continuous*, rather than discretized, outputs optimized with an ℓ_1 loss. LLaMA 2 7B [50] serves as the backend LLM for fine-tuning.

3.2. Our Approach: 3D-CAVLA

Motivated to improve generalizability beyond seen tasks, our proposed model adopts the base architecture of OpenVLA-OFT and incorporates modifications to improve

task relevant context capture and spatial information. Our architecture is shown in Figure 1.

Chain-of-Thought Narrative Instructions. Humans learn object manipulation and environment perception through expert guided demonstrations by other humans. However we do not need a separate demonstration to handle each new object. For example, when a child learns to grasp and manipulate a ball, it may not need another lesson on grasping an orange. Similarly, a robot deployed in an unknown environment can benefit with chain-of-thought steps instead of plain task instructions which may not capture the generalization it has learnt to solve the problem. For example, consider a task the policy has trained for such as “Grab the ball and place it in the basket” decomposed into steps - “Locate ball, grab it from the center, move over basket, drop inside basket”. Now when the robot is deployed in an unseen environment to complete the task - “Move the orange into the basket”, the policy may benefit by breaking down the unseen task into steps - “Locate the orange, grab it from the center, move over basket, drop inside basket”. When these tasks are compared, the only difference lies in locating the unseen target object, which can be handled by a powerful object detector or a vision encoder with robust generalization. We test this hypothesis by transforming plain task instructions into task-relevant chain-of-thought steps using GPT 4’s reasoning capabilities. The format of our prompt is shown in Figure 2.

Integrating Depth Features. Majority VLAs learn policies that map language and 2D visual data captured through images into real-world actions. However, depth perception is a critical skill that is needed to robustly manipulate objects of different shapes and sizes. Modern cameras cap-

Chain Of Thought Prompting with GPT4

You are in the command of a robot manipulator to complete a task involving various objects in scene. Your job is to break down the given instruction into smaller steps based on your real-world intuition to ensure precise object grasping and placement

Some examples are given below:

Task Instruction: Put both pots on the stove

Steps: Grasp first pot, place on stove leaving some space, grasp second pot, place on stove next to first pot.

{more examples}

Use the information above to create step-by-step plan for given task instruction. Remember to only use the given objects and standard grasping, moving and placement actions that can be achieved with a parallel gripper.

Task Instruction: {task_input}

Steps:

Figure 2. LLM prompt to decompose task instructions into executable steps that can be generalized across seen and unseen tasks.

ture RGB-D images and thus an effective depth encoder can improve spatial and geometric awareness of VLAs. We introduce a small but efficient trainable depth encoder to transform depth maps into embeddings that are concatenated with vision, language and proprioception information. Given a batch depth map $D \in \mathbb{R}^{B \times H \times W}$, camera intrinsics (f_x, f_y, c_x, c_y) , and integer pixel grids $U \in \mathbb{R}^{H \times W}$ and $V \in \mathbb{R}^{H \times W}$, we recover metric 3-D coordinates for every pixel (h, w) in every image b as

$$\begin{aligned} Z_{b,h,w} &= D_{b,h,w}, \\ X_{b,h,w} &= \frac{U_{h,w} - c_x}{f_x} Z_{b,h,w}, \\ Y_{b,h,w} &= \frac{V_{h,w} - c_y}{f_y} Z_{b,h,w}. \end{aligned}$$

Stacking (X, Y, Z) along the last axis yields a point cloud $P \in \mathbb{R}^{B \times H \times W \times 3}$, which is fed to the subsequent trainable layers. As shown on the right side of Figure 1, the point clouds pass through a spatial transformer network composed of MLP layers, converting the embeddings into a spatially invariant representation. Following a residual batch matrix product, the embeddings pass through 3 blocks of Conv2D, BatchNorm and ReLU and finally a linear layer to project the embeddings to match the dimension of other modalities. Our depth encoder draws inspiration from PointNet [36], which has shown remarkable performance in

depth perception related tasks. Since our depth encoder is lightweight ($\approx 1M$), we use separate encoders for each camera view.

Task Aware Region of Interest Detection. VLAs learn motion trajectories for the end effector during training. The visual embeddings which pass through the LLM contain representations of every patch of the image, however not all patches are relevant for a given task. By choosing the appropriate patches and thus the region of manipulation for the robot, we can constrain the motion to be within that region. This capability can be extremely useful especially in unseen tasks where the robot encounters many out-of-distribution objects and thus can benefit with a region of importance to focus on. During training, we use ground truth demonstrations to approximate such a region for pooling the visual features. Given a task instruction, we apply named entity recognition [57] to identify target objects and locations important for the task. This passes through a powerful object detector, Molmo [8], to generate bounding boxes for the extracted entities. Then we leverage object tracking capabilities of SAMURAI [54] to estimate the image regions in which the entity bounding boxes move. This determines the region of motion for the task, and the resulting binary mask is used to pool visual features. Our overall region of interest detection pipeline is shown in Figure 3. A downfall of such a method may be the removal of background context and distractors that are necessary for the task. To prevent over-dependence on such masks, we randomly perturb this pipeline to only use pooling 25% during training. We empirically observed that ROI detection deteriorates the performance slightly when tested with in-distribution tasks (see Table 2) while it strongly contributes to better results on out-of-distribution tasks.

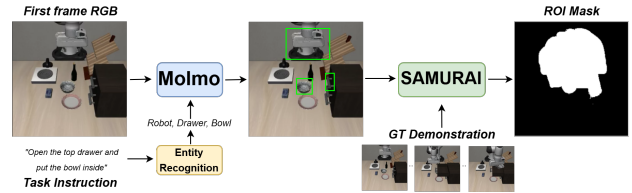


Figure 3. Our framework for task aware region of interest detection using entity recognition, object detection and tracking.

Experimental Setup. We use a single Nvidia A100 GPU with a batch size of 8 for all our experiments. Chain-of-thought steps and binary masks for region pooling are computed offline for efficient fine-tuning. We follow data loading and LoRA pipelines from OpenVLA-OFT [23].

4. Results

Our experiments are divided into two phases. First, we evaluate our method on the LIBERO benchmark, which con-

tains four task suites with ten tasks each, and compare 3D-CAVLA against established baselines under in-distribution settings. Second, we assess generalization by comparing 3D-CAVLA with OpenVLA and OpenVLA-OFT on ten unseen tasks. To enable this zero-shot evaluation, we create the LIBERO-Unseen benchmark by modifying the Behavior Domain Definition Language (BDDL) files of the original LIBERO-90 dataset. We release this benchmark publicly to encourage community-driven zero-shot testing.

4.1. Comparisons on LIBERO Benchmarks

The LIBERO benchmark comprises four task suites, each testing a unique capability of a trained policy -

1. **LIBERO-Spatial:** Tasks that manipulate the *same* object but require placing it in different locations.
Example: Pick up the black bowl between the plate and the ramekin and place it on the plate.
2. **LIBERO-Object:** Tasks with a fixed target location but a *different* object to manipulate each time.
Example: Pick up the milk and place it in the basket.
3. **LIBERO-Goal:** Tasks in which the robot must achieve a higher-level goal beyond simple pick-and-place.
Example: Open the top drawer and put the bowl inside.
4. **LIBERO-Long:** Long-horizon tasks that sequentially manipulate multiple objects, testing extended reasoning.
Example: Turn on the stove and put the moka pot on it.

As in prior works [22, 24], we fine-tune our model independently on each of the task suites and deploy it in simulation. Each task is trained with 50 demonstrations and evaluated with 50 trials per task. The results are shown in Table 1. The first six rows of the table compare our method with policies that only use a third person image and language instruction as input modalities. Under this setting, our model improves success rate on the Spatial and Long task suites, and slightly outperforms Diffusion Transformers policy [6] on average. Qualitative analysis highlights that our policy results in improved precision due to related chain-of-thought instructions across different tasks under the same task suite. This helps the model learn a more robust policy by sharing semantic and logical perception across related tasks. The last four rows show our results when adding an additional camera and robot states. 3D-CAVLA consistently outperforms competitive baselines in the four task suites, highlighting the significance of adding an additional sensor modality to the VLA policy through depth maps. We observed that the resulting policy performs better in precise object manipulation such as cases where target object is situated at a crowded location. Addition of depth maps transforms the input modalities from 2D to 3D, and our results motivate the exploration of more sophisticated depth information extraction pipelines such as 3D meshes [53].

Ablation Studies. 3D-CAVLA builds upon the base architecture of OpenVLA-OFT [24] by adding three key mod-

ules: chain-of-thought style narrative instructions, a depth encoder to learn robust point cloud derived features and task aware ROI detection to constrict the motion to relevant parts of the scene. We ablate over each of these components and provide results across the four LIBERO task suites in Table 2. Removing depth maps causes the highest performance drop, underscoring the importance of 3-D features for policy learning. Eliminating chain-of-thought instructions lowers scores on LIBERO-Long by 1.3%, confirming that this module is beneficial for long-horizon tasks. We also observe a slight decline on seen tasks when using the region-of-interest (TA-ROI) module. TA-ROI can exclude contextual cues, such as nearby obstacles or distractors, essential for effective learning. For instance, in the task “open the drawer and move the bowl inside,” our ROI-pooling module correctly highlights the robot, bowl, and drawer handle, but its binary mask omits the stationary parts of the drawer. As a result, at test time the policy cannot locate the drawer itself because it is absent from the input.

4.2. Zero-Shot Evaluation

While the LIBERO task suites test the spatial, goal-awareness, semantic and long-horizon capabilities of VLAs, we suspect a significant overfitting especially since the task instructions and demonstrations are quite small compared to the number of trainable parameters of some of the larger models we test. Some of these observations were confirmed when we tried evaluating fine-tuned models on LIBERO-Object with LIBERO-Goal, where all baselines fail to successfully complete any task. To evaluate the zero-shot capabilities of finetuned VLAs, we follow the following pipeline: 1) First we finetune the VLAs on Libero-90, which is a larger collection of tasks spanning all four LIBERO suites, 2) We design 10 tasks which the model has not seen during training, and specify the end goals for success using the BDDL format and 3) We evaluate the LIBERO-90 fine-tuned models on these 10 tasks and interpret the performance both qualitatively and quantitatively using task success rates. While [17] also follow a similar framework for zero-shot evaluations, we are unable to use their tasks since they have not been released publicly yet.

We designed ten unseen tasks after analyzing the limitations of fine-tuned VLAs. Preliminary tests showed that when a task introduces entirely new objects, distractors, or motions, both OpenVLA-OFT and our 3D-CAVLA fail to generalize. Consequently, we adopted a milder protocol: task instructions are novel, yet every object still appears in training data and demonstrations cover related skills. For example, in the unseen task “Grab the white bowl and place it on the stove,” the model has learnt to grasp the white bowl and, separately, placing items on the stove, but never both actions together. Even under this relaxed setting, OpenVLA and 3D-CAVLA trained and evaluated with a single cam-

Policy Setup: Single stationary third person camera + Language Instruction					
	Spatial	Object	Goal	Long	Average
Diffusion Policy [6]	78.3	92.5	68.3	50.5	72.4
Octo [48]	78.9	85.7	84.6	51.1	75.1
Diffusion Transformers [41]	84.2	96.3	85.4	63.8	82.4
OpenVLA [22]	84.7	88.4	79.2	53.7	76.5
OTTER [17]	84.0	89.0	82.0	-	-
Ours: 3D-CAVLA (with depth maps)	86.1	94.7	82.9	66.8	82.6

Policy Setup: Third person camera + Wrist camera + Robot states + Language Instruction					
	Spatial	Object	Goal	Long	Average
Multimodal Diffusion Transformer [41]	78.5	87.5	73.5	64.8	76.1
π_0 [3]	96.8	98.8	95.8	85.2	94.2
OpenVLA-OFT [24]	97.6	98.4	97.9	94.5	97.1
Ours: 3D-CAVLA (with depth maps)	98.2	99.8	98.2	96.1	98.1

Table 1. Results on the LIBERO Benchmark. 3D-CAVLA shows consistent improvement across all task suites in the dual camera setup. Most baselines overfit to the tasks and thus the margins are quite narrow. The strongest improvements are shown in long-horizon tasks (column 5) where chain-of-thought instructions helps the policy focus on one sub-task at a time. All scores are reported in success rate (%)

Method	Spatial	Object	Goal	Long
3D-CAVLA	98.2	99.8	98.2	96.1
w/o CoT	97.8	99.4	97.9	94.8
w/o Depth	97.6	99.0	98.0	95.2
w TA-ROI	98.0	99.4	97.4	94.2

Table 2. Ablation Studies. Row 2 shows the results of removing LLM prompted CoT instructions from our method, row 3 shows results of removing the depth projector while row 4 shows the small dip in scores we observe when we add TA-ROI detection when evaluated on seen tasks.

era fail on all tasks. Their greatly reduced training accuracy on the original 90 tasks also indicates poor scalability. We therefore restrict our LIBERO-Unseen comparison to OpenVLA-OFT and 3D-CAVLA using two cameras; results are shown in Table 3.

3D-CAVLA with two cameras outperforms OpenVLA-OFT by 8.8% absolute improvement when considering 50 trials per task. With chain-of-thought reasoning, the model is able to break down the unseen task into sub-steps, some of which may be seen during training. Additionally, the task aware region pooling module provides an approximate binary mask over the input region to constrain the model to generate motion confined to this region. This combined with depth information allows the model to transfer knowledge learnt during training to unseen situations. Our results clearly show an improvement in zero-shot settings as our proposed method 3D-CAVLA (with 2 camera views) is able to generalize better, significantly improving performance.

We showcase some success and failure cases of OpenVLA-OFT and 3D-CAVLA on unseen tasks in Table 4.

5. Conclusion and Future Work

In this paper, we propose a novel method for vision-language action modeling which builds upon a popular open-sourced method OpenVLA-OFT, transforming the problem from 2D to 3D. Our key changes improve the reasoning, geometric and zero-shot capabilities over competitive baselines while maintaining strong performance on in-domain LIBERO simulation software. Our experiments reveal the significant performance gap of VLAs on unseen tasks, motivating further research into efficient input feature extraction, real-time error correction, and the development of generalizable learning strategies that avoid overfitting to training tasks.

Future work will proceed in two directions. First, we will add a VLM-guided, closed-loop feedback module that supplies real-time environment cues to the policy, reducing erroneous motions and boosting performance on unseen tasks; an efficient retrieval mechanism will further exploit prior knowledge acquired during fine-tuning. Second, because LIBERO’s tasks are relatively simple and prone to model saturation, we plan to perform extensive real-world experiments and benchmark the results against other open-source VLAs, aiming for methods that can be deployed zero-shot on truly novel tasks.

Task Instruction	OpenVLA-OFT	3D-CAVLA
Place the white and yellow mug on the plate	32	60
Put the ketchup on top of the cabinet	74	82
Pick up the chocolate pudding at the back and put it in the top drawer of the cabinet	58	52
Stack the right bowl on the left bowl and put the chocolate pudding in the tray	0	0
Put the chocolate pudding on the plate	78	80
Place the cream cheese and soup inside the basket	66	74
Grab the white bowl and keep it on the stove	12	10
Grab the chocolate pudding and place it on the bowl. Then place both items on the tray	6	24
Turn on the stove and put the bowl on it	14	38
Place the mug inside the right compartment of the caddy	24	32
Average	36.4	45.2 (+8.8)

Table 3. Success-rate (in %) of OpenVLA-OFT and 3D-CAVLA on 10 unseen tasks. Both models do not replicate the performance observed on seen tasks. 3D-CAVLA decomposes unseen tasks into seen steps and applies task-aware region-of-interest detection, enabling better generalization.































Task Instruction	OpenVLA-OFT					3D-CAVLA						
Put the chocolate pudding on the plate		...		...		✓		...		...		✓
Place the white and yellow mug on the plate		...		...		✗		...		...		✓
Turn on the stove and put the bowl on it		...		...		✗		...		...		✓
Pick up the chocolate pudding at the back and put it in the top drawer of the cabinet		...		...		✗		...		...		✗
Grab the white bowl and keep it on the stove		...		...		✗		...		...		✗

Table 4. Qualitative comparisons of OpenVLA-OFT and 3D-CAVLA on unseen LIBERO tasks. We show first, middle, and last frames of each inference. The final two rows depict failures where both models misidentify target object or get distracted by previously seen objects.

References

- [1] Michael Ahn, Anthony Brohan, Noah Brown, et al. Do as I can and not as I say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022. 2
- [2] Vineet Bhat, Ali Umut Kaypak, Prashanth Krishnamurthy, et al. Grounding llms for robot task planning using closed-loop state feedback. *ArXiv*, abs/2402.08546, 2024. 2
- [3] Kevin Black, Noah Brown, Danny Driess, et al. π_0 : A vision-language-action flow model for general robot control, 2024. 2

2024. 6
- [4] Anthony Brohan, Noah Brown, Justice Carbajal, et al. Rt-1: Robotics transformer for real-world control at scale. In *arXiv preprint arXiv:2212.06817*, 2022. 1, 2
- [5] Anthony Brohan, Noah Brown, Justice Carbajal, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *arXiv preprint arXiv:2307.15818*, 2023. 1, 2
- [6] Cheng Chi, Zhenjia Xu, Siyuan Feng, et al. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023. 5, 6
- [7] Zichen Jeff Cui, Hengkai Pan, Aadithya Iyer, et al. Dynamo: In-domain dynamics pretraining for visuo-motor control. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2
- [8] Matt Deitke, Christopher Clark, Sangho Lee, et al. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*, 2024. 4
- [9] Sombit Dey, Jan-Nico Zaeche, Nikolay Nikolov, et al. Revla: Reverting visual domain limitation of robotic foundation models. *arXiv preprint arXiv:2409.15250*, 2024. 2
- [10] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, et al. Palm-e: An embodied multimodal language model. In *arXiv preprint arXiv:2303.03378*, 2023. 2
- [11] Jensen Gao, Bidipta Sarkar, Fei Xia, et al. Physically grounded vision-language models for robotic manipulation. *ArXiv*, abs/2309.02561, 2023. 2
- [12] Raktim Gautam Goswami, Prashanth Krishnamurthy, Yann LeCun, and Farshad Khorrani. Robopepp: Vision-based robot pose and joint angle estimation through embedding predictive pre-training. *arXiv preprint arXiv:2411.17662*, 2024. 2
- [13] Qiao Gu, Alihusein Kuwajerwala, Sacha Morin, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. *arXiv*, 2023. 2
- [14] Yicong Hong, Qi Wu, Yuankai Qi, et al. Vln bert: A recurrent vision-and-language bert for navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1643–1653, 2021. 2
- [15] Edward J Hu, Yelong Shen, Phillip Wallis, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. 3
- [16] Chenguang Huang, Oier Mees, Andy Zeng, et al. Visual language maps for robot navigation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10608–10615, 2023. 2
- [17] Huang Huang, Fangchen Liu, Letian Fu, et al. Otter: A vision-language-action model with text-aware visual feature extraction, 2025. 5, 6
- [18] Jiangyong Huang, Silong Yong, Xiaojian Ma, et al. An embodied generalist agent in 3d world. In *ICLR 2024 Workshop: How Far Are We From AGI*, 2024. 2
- [19] Wenlong Huang, Pieter Abbeel, Deepak Pathak, et al. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. *arXiv preprint arXiv:2201.07207*, 2022. 2
- [20] Wenlong Huang et al. Inner monologue: Embodied reasoning through planning with language models. In *arXiv preprint arXiv:2207.05608*, 2022. 1, 2
- [21] Gi-Cheon Kang, Junghyun Kim, Kyuhwan Shim, et al. Clip-rt: Learning language-conditioned robotic policies from natural language supervision. *arXiv preprint arXiv:2411.00508*, 2024. 2
- [22] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 2, 3, 5, 6
- [23] Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025. 1, 2, 3, 4
- [24] Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success, 2025. 5, 6
- [25] Xinghang Li et al. Vision-language foundation models as effective robot imitators. In *The Twelfth International Conference on Learning Representations*, 2024. 2
- [26] Zhuoling Li, Liangliang Ren, Jinrong Yang, et al. Virt: Vision instructed transformer for robotic manipulation. *arXiv preprint arXiv:2410.07169*, 2024. 2
- [27] Fangchen Liu, Kuan Fang, Pieter Abbeel, et al. Moka: Open-vocabulary robotic manipulation through mark-based visual prompting. *arXiv preprint arXiv:2403.03174*, 2024. 2
- [28] Jiaming Liu, Chenxuan Li, Guanqun Wang, et al. Self-corrected multimodal large language model for end-to-end robot manipulation. *arXiv preprint arXiv:2405.17418*, 2024. 2
- [29] Jiaming Liu, Mengzhen Liu, Zhenyu Wang, et al. Robomamba: Multimodal state space model for efficient robot reasoning and manipulation. *arXiv preprint arXiv:2406.04339*, 2024. 2
- [30] Peiqi Liu, Yaswanth Orru, Chris Paxton, et al. OK-Robot: What really matters in integrating open-knowledge models for robotics. *arXiv preprint arXiv:2401.12202*, 2024. 2
- [31] Andrew Melnik et al. Uniteam: Open vocabulary mobile manipulation challenge. *ArXiv*, abs/2312.08611, 2023. 2
- [32] Yuchen Mo et al. Towards open-world interactive disambiguation for robotic grasping. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8061–8067, 2023. 2
- [33] Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3
- [34] Abhishek Padalkar et al. Open x-embodiment: Robotic learning datasets and rt-x models. *ArXiv*, abs/2310.08864, 2023. 2
- [35] Ethan Perez, Florian Strub, Harm de Vries, et al. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018. 3
- [36] Charles R. Qi, Hao Su, Kaichun Mo, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3, 4
- [37] Tianwen Qian, Jingjing Chen, Linhai Zhuo, et al. Nuscenesqa: A multi-modal visual question answering benchmark

- for autonomous driving scenario. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4542–4550, 2024. 1
- [38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2
- [39] Krishan Rana et al. Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning. In *Proceedings of The 7th Conference on Robot Learning*, pages 23–72. PMLR, 2023. 2
- [40] Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, et al. Robots that ask for help: Uncertainty alignment for large language model planners. In *Proceedings of the Conference on Robot Learning*, 2023. 2
- [41] Moritz Reuss, Ömer Erdiñç Yağmurlu, Fabian Wenzel, et al. Multimodal diffusion transformer: Learning versatile behavior from multimodal goals. *arXiv preprint arXiv:2407.05996*, 2024. 6
- [42] Pierre Sermanet et al. Robovqa: Multimodal long-horizon reasoning for robotics. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 645–652, 2024. 1
- [43] Mohit Shridhar et al. Cliport: What and where pathways for robotic manipulation. In *Proceedings of the 5th Conference on Robot Learning*, pages 894–906. PMLR, 2022. 2
- [44] Ishika Singh et al. Progprompt: Generating situated robot task plans using large language models. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11523–11530, 2023. 1, 2
- [45] Chan Hee Song et al. Llm-planner: Few-shot grounded planning for embodied agents with large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2998–3009, 2023. 2
- [46] Priya Sundaresan, Hengyuan Hu, Quan Vuong, et al. What’s the move? hybrid imitation learning via salient points. *arXiv preprint arXiv:2412.05426*, 2024. 2
- [47] Hengkai Tan, Xuezhou Xu, Chengyang Ying, et al. Manibox: Enhancing spatial grasping generalization via scalable simulation data generation. *arXiv preprint arXiv:2411.01850*, 2024. 2
- [48] Octo Model Team, Dibya Ghosh, Homer Walke, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024. 6
- [49] Yang Tian, Sizhe Yang, Jia Zeng, et al. Predictive inverse dynamics models are scalable learners for robotic manipulation, 2024. 2
- [50] Hugo Touvron, Louis Martin, Kevin Stone, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 3
- [51] Georgios Tzifas et al. Language-guided robot grasping: Clip-based referring grasp synthesis in clutter. In *7th Annual Conference on Robot Learning*, 2023. 1
- [52] Chuan Wen, Xingyu Lin, John So, et al. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2023. 2
- [53] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024. 5
- [54] Cheng-Yen Yang, Hsiang-Wei Huang, Wenhao Chai, et al. Samurai: Adapting segment anything model for zero-shot visual tracking with motion-aware memory, 2024. 4
- [55] Jiange Yang, Haoyi Zhu, Yating Wang, et al. Tra-moe: Learning trajectory prediction model from multiple domains for adaptive policy conditioning. *arXiv preprint arXiv:2411.14519*, 2024. 2
- [56] Zhengyuan Yang et al. The dawn of lmms: Preliminary explorations with gpt-4v(ision). *ArXiv*, abs/2309.17421, 2023. 2
- [57] Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. GLiNER: Generalist model for named entity recognition using bidirectional transformer. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5364–5376, Mexico City, Mexico, 2024. Association for Computational Linguistics. 4
- [58] Michał Zawalski et al. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024. 2
- [59] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, et al. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023. 3
- [60] Hanbo Zhang et al. Invigorate: Interactive visual grounding and grasping in clutter. In *Proceedings of Robotics: Science and Systems*, Virtual, 2021. 2
- [61] Qingqing Zhao, Yao Lu, Moo Jin Kim, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. *arXiv preprint arXiv:2503.22020*, 2025. 2
- [62] Peiyuan Zhi et al. Closed-loop open-vocabulary mobile manipulation with gpt-4v, 2024. 2