VLA-OS: Structuring and Dissecting Planning Representations and Paradigms in Vision-Language-Action Models

Chongkai Gao[1], Zixuan Liu[1], Zhenghao Chi[1], Junshan Huang[2], Xin Fei[3],
Yiwen Hou[1], Yuxuan Zhang[1], Yudi Lin[1], Zhirui Fang[1], Zeyu Jiang[4], Lin Shao[1†]

***Abstract*— Recent studies on Vision-Language-Action (VLA) models have shifted from the end-to-end action-generation paradigm toward a pipeline involving task planning followed by action generation, demonstrating improved performance on various complex, long-horizon manipulation tasks. However, existing approaches vary significantly in terms of network architectures, planning paradigms, representations, and training data sources, making it challenging for researchers to identify the precise sources of performance gains and components to be further improved. To systematically investigate the impacts of different planning paradigms and representations isolating from network architectures and training data, in this paper, we introduce VLA-OS, a unified VLA architecture series capable of various task planning paradigms, and design a comprehensive suite of controlled experiments across diverse object categories (rigid and deformable), visual modalities (2D and 3D), environments (simulation and real-world), and end-effectors (grippers and dexterous hands). Our results demonstrate that: 1) visually grounded planning representations are generally better than language planning representations; 2) the Hierarchical-VLA paradigm generally achieves superior or comparable performance than other paradigms on task performance, pretraining, generalization ability, scalability, and continual learning ability, albeit at the cost of slower training and inference speeds. Experiment results are in https://nus-lins-lab.github.io/vlaos/.**

I. Introduction

Building intelligent and generalizable robots capable of perceiving, reasoning about, and interacting with physical environments remains a persistent challenge in the robotics community [1], [2]. Recent studies have increasingly emphasized the development of foundational models for robot manipulation tasks by training large Vision-Language-Action models (VLAs) on extensive datasets [3], [4], [5], [6], [7], [8], [9], [10]. Different from end-to-end foundation models in computer vision [11], [12], [13] and natural language processing tasks [14], [15], [16], recent studies of VLAs have shifted toward a new paradigm capable of performing task planning and policy learning either simultaneously or sequentially [17], [18], [19], [20], [21], [22], [23], [4]. This shift arises from the inherent complexity of robotic manipulation tasks, which naturally exhibit hierarchical structures involving both high-level task planning and low-level physical interactions [24]. Compared to end-to-end VLAs that only generate actions, these methods demonstrate stronger capabilities in task reasoning and comprehension for long-horizon tasks [25], [4], better success rates [18], [20], and higher sample efficiency [26], [19], [27].

However, current task-planning approaches in VLA are mainly based on intuitive designs and lack fair and systematic comparisons, as these methods vary along multiple dimensions, including network architectures, planning paradigms, data representations, and training data sources. This makes it difficult for researchers to clearly identify which specific component contributes to performance gains or requires further improvement, hindering progress in the field.

Among these challenges, five core questions stand out: 1) **Representation**: What representation should we adopt for task planning and policy learning? Does using multiple representations yield better results, or could they conflict with one another? 2) **Paradigm**: Should we employ a monolithic model that jointly performs task planning and policy learning, or should we opt for a hierarchical paradigm where two separate models handle these tasks independently? 3) **Bottleneck**: Between task planning and policy learning, which presents a greater challenge for current manipulation tasks? 4) **Scalability and Pretraining**: Do VLAs that incorporate task planning preserve the advantageous properties of end-to-end foundation models, such as model and data scalability, as well as benefits derived from pretraining? and 5) **Performance**: Do VLAs employing task planning have better generalization and continual learning ability than end-to-end VLAs? Addressing these questions will provide the community with a clearer understanding of how task planning works in VLA models, and offer empirical evidence and guidance for future developments.

In this work, we aim to answer these questions with systematic and controllable experiments. First, to avoid biases introduced by specific neural network choices, we develop VLA-OS[1] model series: a unified and composable family of VLA models for general-purpose manipulation tasks capable of different task planning paradigms. Concretely, we designed VLA-OS-A, VLA-OS-I, and VLA-OS-H that correspond to three mainstream VLA paradigms (ActionOnly-VLA, Integrated-VLA, Hierarchical-VLA), respectively, as illustrated in Figure 1. VLA-OS series features a unified, interchangeable VLM backbone that can be directly downloaded from HuggingFace, various plug-and-play planning heads for different representations, and two different action heads both supporting 2D/3D tasks, as shown in Figure 2.

[1]National University of Singapore, [2]University of Science and Technology of China, [3]Tsinghua University, [4]Nanyang Technological University.

† Corresponding to `linshao@nus.edu.sg`

[1] "OS" stands for "Operating System" and designates that our model family provides unified and organized interfaces of advanced VLA architectures with various planning heads and different paradigms for users.

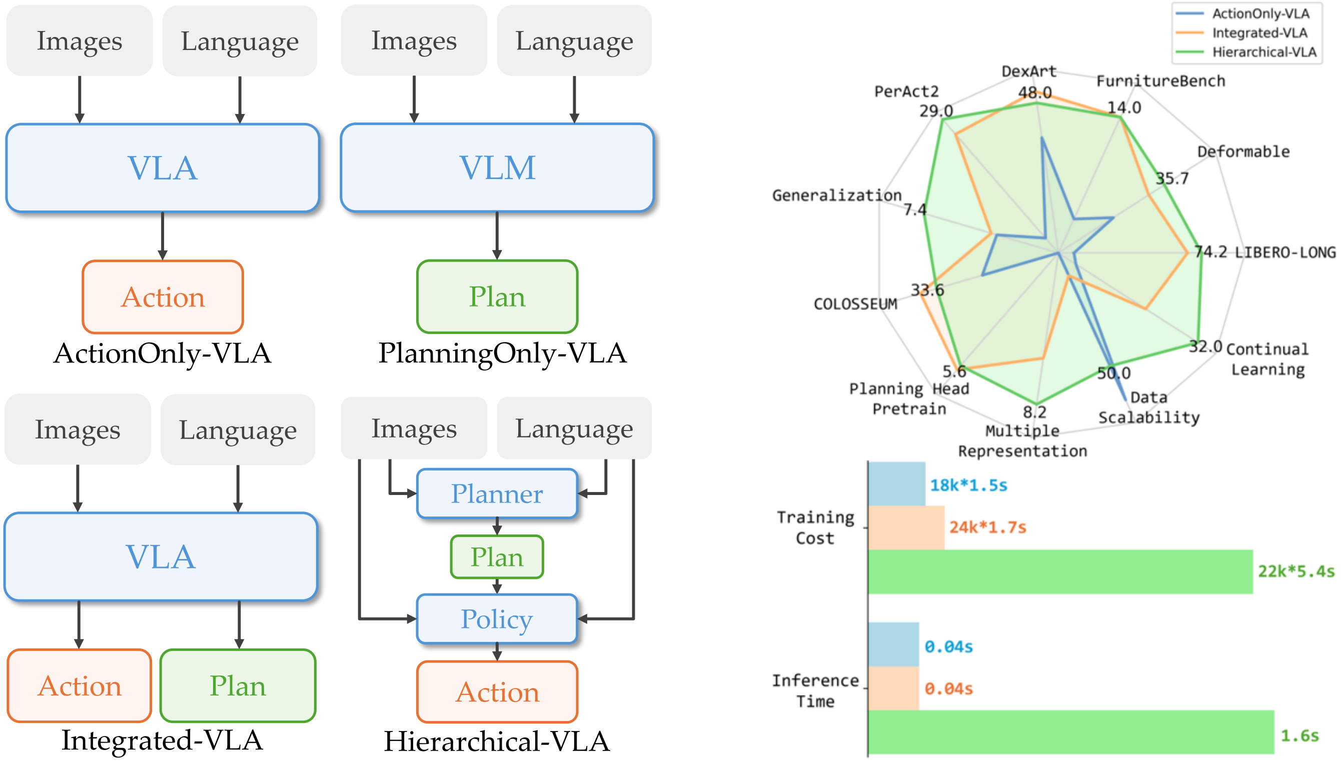


Fig. 1: Left: four different VLA paradigms. Note in this paper, we didn't explore PlanningOnly-VLA since they usually cannot be trained with the provided datasets and perform worse than others. Right: VLA paradigm comparison results. Hierarchical-VLA exhibits a generally better performance than ActionOnly-VLA and Integrated-VLA, while it incurs larger training and inference costs. This motivates future work on improving training and inference speeds for them.

We show in our experiments that VLA-OS exhibits superior performance compared to most existing VLA methods with fewer parameters and without pretraining.

Next, to answer the **representation** question, we annotate three kinds of task reasoning representations, including language reasoning, visual reasoning, and goal images, and conducted exhaustive combinatorial experiments with Integrated-VLA and Hierarchical-VLA models on LIBERO [28] benchmark to identify representations that yield optimal performance. Subsequently, employing the optimal representations identified, we conducted performance comparisons among three VLA paradigms on six benchmarks to answer the **paradigm** question, including rigid body manipulation tasks [28], visual generalization tasks [29], complex long-horizon tasks [30], real-world deformable manipulation tasks, dexterous manipulation tasks [31], and dual-arm manipulation tasks [32]. Furthermore, to answer the **bottleneck** question, we designed a novel set of evaluation metrics tailored to separately assess the performance of task planning and policy learning parts. To answer the **scalability** question, we use LIBERO [28] to test the model and data scalability as well as the effects of pretraining among different paradigms. And lastly, we test the generalization capabilities and continual learning ability of different VLA paradigms to answer the **performance** question.

Our experiments yield three primary findings: 1) Visually grounded planning representations (visual reasoning and image foresight planning) outperform language-based planning representations across multiple dimensions including task performance, generalization, training and speed, and low-level policy execution; 2) Hierarchical-VLA matches or exceeds the performance of Integrated-VLA and ActionOnly-VLA in terms of task performance, generalization, scalability, planning scores, continual learning, and gains from task-planning pretraining, albeit at the expense of increased training cost and slower inference; 3) On LIBERO [28] benchmark tasks, policy learning is consistently more challenging than task planning, regardless of which planning representation is used. We believe that our findings (as well as source codes, annotated datasets, and checkpoints) will provide significant help and guidance for future research within the VLA community and the broader robotics community.

II. Related Works

A. VLA Paradigms for Robot Manipulation

a) PlanningOnly-VLA: These works leverage pretrained LLMs or VLMs to reason and perform task planning without generating the low-level action. They break up the given task into simpler sub-tasks that can be performed by either using a set of pre-trained sub-skills [33], [7], [34], [35], [36], or outputting the parameters of pre-defined motions or cost functions for optimization [37], [38], [39], [40], [41], [42], [43], [44]. The problem is that their VLMs and low-level skills usually cannot be trained with further datasets, which frequently places them at a disadvantage compared to other VLA paradigms capable of training on given datasets [45], [46]. So we do not include PlanningOnly-VLA in this study.

b) ActionOnly-VLA: These works employ an end-to-end fashion to directly map visual and language inputs to robot actions with a multi-modal network. Pioneering works mainly focus on verifying the effectiveness of large-scale robot learning [47], [48], [49], [50], while later works start to explore different model architectures, training objectives, and extra multi-modal representations and information fusion designs to make this paradigm more effective and efficient [3], [6], [5], [51], [52], [53], [54], [55], [56], [57], [58], [59]. In this work, we design VLA-OS-A for this paradigm by synthesizing several advanced model designs that have been verified to be superior in recent works [60], [3], [57].

c) Integrated-VLA: These works use a single model to perform task planning and policy learning simultaneously. According to whether the action generation process is conditioned on the planning embeddings or results, they can be further divided into explicit planning and implicit planning. For explicit planning, EmbodiedCoT [18] and CotVLA [17] generate either language-based or goal-image-based embodied chain-of-thought [61] reasoning before generating actions, and the action generation process is conditioned on the embeddings of CoT. For implicit planning, MDT [62] and PIDM [23] use goal image foresight generation loss as an auxiliary objective for planning, while RoboBrain [26] and ChatVLA [25] train VLA with auxiliary task reasoning loss in language representations. Some recent works also seek to use latent action tokens [63], [64], [65], [66], [67] that serve as forward dynamics representations to generate future images as image foresight planning, and decode these latent actions to real actions with another action head. The inputs to the action head are from the VLM encoder, and they do not need the planning heads (decoder) during inference [63], [64], [65], [66] or they only need one planning forward pass [67], so we also see these methods as implicit planning. In this work, we design VLA-OS-I for this paradigm with various plug-and-play planning heads upon VLA-OS-A for different planning representations.

d) Hierarchical-VLA: These works use two separate models for task planning and policy learning, with no connection or gradient between them. The idea of hierarchical models has always existed in robotics research [43], [44], [27], [68], [69]. RT-H [22] is the first work of this paradigm, where they use two identical VLMs to generate languages and actions respectively. Later works [20], [70], [4] also follow this idea but use different model architectures for task planning and action generation. Other works seek to generate multi-modal planning results for policy learning, such as image flows or trajectories [71], [19], [21], future videos [72], [73], affordance [74], [75], keypose [76], and keypoints [77]. In this work, we design VLA-OS-H for them.

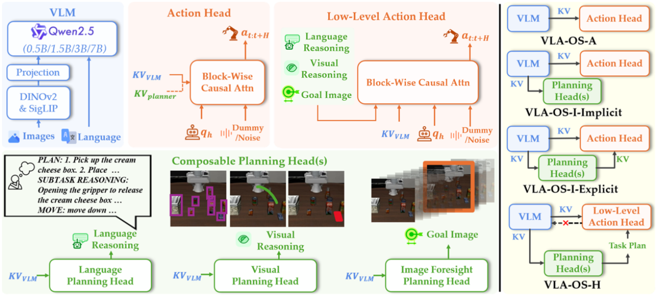


Fig. 2: The VLA-OS model family. Left: the VLM and the composable heads. Our VLM has the same architecture with different numbers of parameters. Although we only draw Qwen2.5 here, our code supports any kind of LLM backbone from HuggingFace. Right: four VLA-OS architectures used in our experiments. To minimize the effects of different numbers of parameters in different models, we restrict the number of parameters of all heads to about 5% of the VLM.

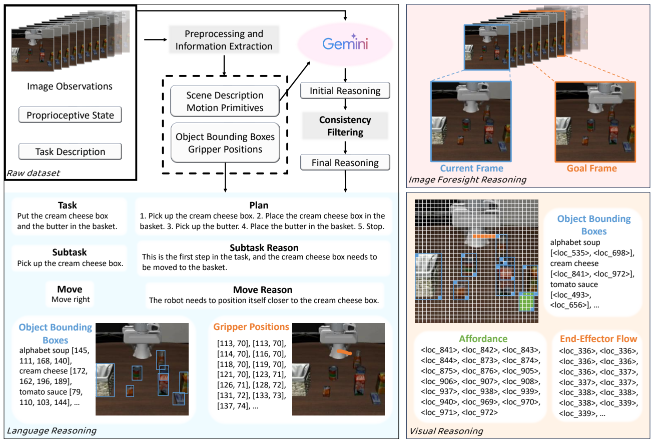
III. VLA-OS Model Family Design

A. Preliminaries

We study imitation learning for robot manipulation tasks. Specifically, for each task $\mathscr{T}$, we assume a set of demonstrations $\mathscr{D}_{\mathscr{T}} = \{(o_i^1, a_i^1), (o_i^2, a_i^2), \cdots, (o_i^{T_i}, a_i^{T_i})\}_{i=1}^N$ and a language goal are given, where T_i is the episode length, o is the observation, a is the robot action, and N is the number of demonstrations. We use a history of multi-view images and proprioception information as observations. In this work, we set the image resolution as 224×224. For actions, we use a normalized continuous delta end-effector pose δ_p action space and gripper open/close action σ for training. We also let the policy generate action chunks, i.e., $a_t = ([\delta_p, \sigma]^t, \cdots, [\delta_p, \sigma]^{t+L-1})$. For dexterous hands, we use the delta joint values as the action space. We train the policy with either flow matching [78], [79] loss (for multi-modal demonstration datasets) or L1 loss (for simple and uni-modal demonstration datasets) under the suggestion of previous works [80], [3], [57], [60].

B. VLA-OS-A for ActionOnly-VLA Paradigm

VLA-OS-A model series directly generates actions without task planning stages. It is also used as the base model for other paradigms. We design a block-wise causal attention VLA drawing inspiration from [3], as shown in Figure 2. First, a VLM encodes the visual and language inputs, where the vision encoder will encode input image patches and project them into language embedding space with an MLP. Then, we use a separate set of weights as an action head for the robotics-specific tokens (action and proprioception states). The action head is a transformer decoder that has the same number of layers as the LLM, and for each layer, the queries of the proprioception tokens can attend to both the keys and values from the LLM and the proprioception keys and values, and the queries of the action tokens can attend to the keys and values from the LLM, proprioception tokens, and themselves.



Put the cream cheese box and the butter in the basket.

1. Pick up the cream cheese box. 2. Place the cream cheese box in the basket. 3. Pick up the butter. 4. Place the butter in the basket. 5. Stop.

Pick up the cream cheese box.

This is the first step in the task, and the cream cheese box needs to be moved to the basket.

Move right

The robot needs to position itself closer to the cream cheese box.

alphabet soup [145, 111, 168, 140], cream cheese [172, 162, 196, 189], tomato sauce [79, 110, 103, 144], ...

[113, 70], [113, 70], [114, 70], [116, 70], [118, 70], [119, 70], [121, 70], [123, 71], [126, 71], [128, 72], [131, 72], [133, 73], [137, 74], ...

alphabet soup [<loc_535>, <loc_698>], cream cheese [<loc_841>, <loc_972>], tomato sauce [<loc_493>, <loc_656>], ...

<loc_841>, <loc_842>, <loc_843>, <loc_844>, <loc_873>, <loc_874>, <loc_875>, <loc_876>, <loc_905>, <loc_906>, <loc_907>, <loc_908>, <loc_937>, <loc_938>, <loc_939>, <loc_940>, <loc_969>, <loc_970>, <loc_971>, <loc_972>

<loc_336>, <loc_336>, <loc_336>, <loc_336>, <loc_336>, <loc_337>, <loc_337>, <loc_337>, <loc_338>, <loc_338>, <loc_338>, <loc_339>, <loc_339>, ...

Fig. 3: The formats and contents of the language reasoning dataset, the visual reasoning dataset, and the image foresight reasoning dataset in this work. We use various vision-language models for data annotation. We illustrate the language reasoning data annotation process on the top left part.

C. VLA-OS-I for Integrated-VLA Paradigm

To perform task planning with different kinds of representations, we design three kinds of task planning heads for VLA-OS. We first annotate three kinds of task reasoning datasets corresponding to each planning representation, as shown in Figure 3. Details of the dataset can be found on our website.

We then design language planning head, visual planning head, and image foresight planning head for each kind of representation, as shown in Figure 2. All of them are transformers that have the same number of layers with the LLM backbone, and use the block-wise causal attention mechanism to acquire the keys and values from each layer of the LLM backbone as conditions. The language planning head uses the LLM's tokenizer for decoding, whereas the visual planning head uses an extended tokenizer vocabulary to predict location tokens. The image foresight planning head is an autoregressive image generation model similar to the recent SOTA image generator [81]. It auto-regressively generates the image in a coarse-to-fine paradigm proposed by VAR [82]. The language and visual planning heads are trained with cross-entropy loss, while the image foresight planning head is trained with the special loss in [81].

D. VLA-OS-H for Hierarchical-VLA Paradigm

This model uses two networks for task planning and policy learning respectively. As shown in Figure 2, we use the VLM together with planning heads for task planning, and modify the action head to an encoder-decoder transformer for policy learning. This action head can take as input the images, proprioception observations, and the planning representations to generate actions. To keep the comparison fair, we make the layer of the encoder and decoder of the action head half of the other two VLA-OS paradigms. We also give frozen image features from AM-Radio [83] and language features from Qwen2.5 [16] for the inputs of the action head to compensate for deficiencies in visual and linguistic features not captured by the VLM. Training details are on our website.

IV. Experiments and Findings

In this section, we perform systematic and controllable experiments with the VLA-OS model series on various manipulation tasks to answer the research questions in Section I. Detailed experimental settings are on our website. All models are trained on 8×NVIDIA A100 80G GPUs.

V. Conclusion and Limitation

We provide a systematic investigation across different VLA paradigms and task planning representations through various kinds of manipulation tasks. Experiments show the superiority of visually grounded planning representations and the Hierarchical-VLA paradigm. Specifically, our findings can be summarized as follows:

1. The time has not yet come to scale up VLA model sizes.
2. Visually grounded representations (visual and image foresight) are better than language planning representations in terms of success rates, low-level following, and continual learning.
3. Integrated-VLA and Hierarchical-VLA outperform ActionOnly-VLA on task performance and generalization ability, but incur faster forgetting.
4. Integrated-VLA and Hierarchical-VLA perform comparably on task performance and Planning Head Pretraining, but Hierarchical-VLA generalizes better and has better task-planning performance.
5. All VLA paradigms have the data scalability. For tasks trained from scratch with roughly 5,000 demonstrations, the LLM backbone should be limited to 0.5B parameters, or keeping the total model size under 1B parameters.

We believe our findings offer meaningful insights that can inform future research in VLA and the broader robotics community. We recommend the following research directions for the community based on our findings:

1. Why are visually grounded representations better than language?
2. How to avoid gradient conflict between planning head losses and action head losses on the VLM backbone? This is because that in both explicit v.s. implicit and Hierarchical v.s. Integrated comparisons, reducing the influence of action head training on VLM improves the performance.
3. How to design network architectures to effectively extract information from VLM? There could be better mechanism than the current KV extraction method.
4. How to design faster planning heads for autoregressive planning heads?
5. How to design better low-level action heads with better planning-following ability?
6. How to construct large-scale task planning datasets for VLA? How to transfer current datasets to task planning datasets? This is because that our finding 6 shows that task planning pretraining is useful.

References

[1] Y. Hu, Q. Xie, V. Jain, J. Francis, J. Patrikar, N. Keetha, S. Kim, Y. Xie, T. Zhang, H.-S. Fang *et al.*, “Toward general-purpose robots via foundation models: A survey and meta-analysis,” *arXiv preprint arXiv:2312.08782*, 2023.

[2] R. Firoozi, J. Tucker, S. Tian, A. Majumdar, J. Sun, W. Liu, Y. Zhu, S. Song, A. Kapoor, K. Hausman *et al.*, “Foundation models in robotics: Applications, challenges, and the future,” *The International Journal of Robotics Research*, p. 02783649241281508, 2023.

[3] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter *et al.*, “pi0: A vision-language-action flow model for general robot control,” *arXiv preprint arXiv:2410.24164*, 2024.

[4] J. Wen, M. Zhu, Y. Zhu, Z. Tang, J. Li, Z. Zhou, C. Li, X. Liu, Y. Peng, C. Shen *et al.*, “Diffusion-vla: Scaling robot foundation models via unified diffusion and autoregression,” *arXiv preprint arXiv:2412.03293*, 2024.

[5] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, “Openvla: An open-source vision-language-action model,” *arXiv preprint arXiv:2406.09246*, 2024.

[6] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu, “Rdt-1b: a diffusion foundation model for bimanual manipulation,” *arXiv preprint arXiv:2410.07864*, 2024.

[7] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman *et al.*, “Do as i can, not as i say: Grounding language in robotic affordances,” *arXiv preprint arXiv:2204.01691*, 2022.

[8] Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, S. Gao, X. He, X. Huang, S. Jiang *et al.*, “Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems,” *arXiv preprint arXiv:2503.06669*, 2025.

[9] J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, L. Fan, Y. Fang, D. Fox, F. Hu, S. Huang *et al.*, “Gr00t n1: An open foundation model for generalist humanoid robots,” *arXiv preprint arXiv:2503.14734*, 2025.

[10] FigureAI, “Helix: A vision-language-action model for generalist humanoid control,” https://www.figure.ai/news/helix, 2025, accessed: 2025-02-20.

[11] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, “Dinov2: Learning robust visual features without supervision,” *arXiv preprint arXiv:2304.07193*, 2023.

[12] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, “Segment anything,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[13] N. Karaev, I. Makarov, J. Wang, N. Neverova, A. Vedaldi, and C. Rupprecht, “Cotracker3: Simpler and better point tracking by pseudo-labelling real videos,” *arXiv preprint arXiv:2410.11831*, 2024.

[14] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, “Gpt-4 technical report,” *arXiv preprint arXiv:2303.08774*, 2023.

[15] D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi *et al.*, “Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning,” *arXiv preprint arXiv:2501.12948*, 2025.

[16] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei *et al.*, “Qwen2. 5 technical report,” *arXiv preprint arXiv:2412.15115*, 2024.

[17] Q. Zhao, Y. Lu, M. J. Kim, Z. Fu, Z. Zhang, Y. Wu, Z. Li, Q. Ma, S. Han, C. Finn, A. Handa, M.-Y. Liu, D. Xiang, G. Wetzstein, and T.-Y. Lin, “Cot-vla: Visual chain-of-thought reasoning for vision-language-action models,” *arXiv preprint arXiv:2503.22020*, 2025.

[18] M. Zawalski, W. Chen, K. Pertsch, O. Mees, C. Finn, and S. Levine, “Robotic control via embodied chain-of-thought reasoning,” *arXiv preprint arXiv:2407.08693*, 2024.

[19] C. Gao, H. Zhang, Z. Xu, Z. Cai, and L. Shao, “Flip: Flow-centric generative planning for general-purpose manipulation tasks,” *arXiv preprint arXiv:2412.08261*, 2024.

[20] L. X. Shi, B. Ichter, M. Equi, L. Ke, K. Pertsch, Q. Vuong, J. Tanner, A. Walling, H. Wang, N. Fusai *et al.*, “Hi robot: Open-ended instruction following with hierarchical vision-language-action models,” *arXiv preprint arXiv:2502.19417*, 2025.

[21] Y. Li, Y. Deng, J. Zhang, J. Jang, M. Memmel, R. Yu, C. R. Garrett, F. Ramos, D. Fox, A. Li *et al.*, “Hamster: Hierarchical action models for open-world robot manipulation,” *arXiv preprint arXiv:2502.05485*, 2025.

[22] S. Belkhale, T. Ding, T. Xiao, P. Sermanet, Q. Vuong, J. Tompson, Y. Chebotar, D. Dwibedi, and D. Sadigh, “Rt-h: Action hierarchies using language,” *arXiv preprint arXiv:2403.01823*, 2024.

[23] Y. Tian, S. Yang, J. Zeng, P. Wang, D. Lin, H. Dong, and J. Pang, “Predictive inverse dynamics models are scalable learners for robotic manipulation,” *arXiv preprint arXiv:2412.15109*, 2024.

[24] M. Brady, *Robot motion: Planning and control.* MIT press, 1982.

[25] Z. Zhou, Y. Zhu, M. Zhu, J. Wen, N. Liu, Z. Xu, W. Meng, R. Cheng, Y. Peng, C. Shen *et al.*, “Chatvla: Unified multimodal understanding and robot control with vision-language-action model,” *arXiv preprint arXiv:2502.14420*, 2025.

[26] Y. Ji, H. Tan, J. Shi, X. Hao, Y. Zhang, H. Zhang, P. Wang, M. Zhao, Y. Mu, P. An *et al.*, “Robobrain: A unified brain model for robotic manipulation from abstract to concrete,” *arXiv preprint arXiv:2502.21257*, 2025.

[27] Z. Xu, C. Gao, Z. Liu, G. Yang, C. Tie, H. Zheng, H. Zhou, W. Peng, D. Wang, T. Hu *et al.*, “Manifoundation model for general-purpose robotic manipulation of contact synthesis with arbitrary objects and robots,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 10 905–10 912.

[28] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone, “Libero: Benchmarking knowledge transfer for lifelong robot learning,” *Advances in Neural Information Processing Systems*, vol. 36, pp. 44 776–44 791, 2023.

[29] W. Pumacay, I. Singh, J. Duan, R. Krishna, J. Thomason, and D. Fox, “The colosseum: A benchmark for evaluating generalization for robotic manipulation,” *arXiv preprint arXiv:2402.08191*, 2024.

[30] M. Heo, Y. Lee, D. Lee, and J. J. Lim, “Furniturebench: Reproducible real-world benchmark for long-horizon complex manipulation,” *The International Journal of Robotics Research*, p. 02783649241304789, 2023.

[31] C. Bao, H. Xu, Y. Qin, and X. Wang, “Dexart: Benchmarking generalizable dexterous manipulation with articulated objects,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 21 190–21 200.

[32] M. Grotz, M. Shridhar, Y.-W. Chao, T. Asfour, and D. Fox, “Peract2: Benchmarking and learning for robotic bimanual manipulation tasks,” in *CoRL 2024 Workshop on Whole-body Control and Bimanual Manipulation: Applications in Humanoids and Beyond*, 2024.

[33] W. Huang, P. Abbeel, D. Pathak, and I. Mordatch, “Language models as zero-shot planners: Extracting actionable knowledge for embodied agents,” in *International conference on machine learning*. PMLR, 2022, pp. 9118–9147.

[34] D. Qiu, W. Ma, Z. Pan, H. Xiong, and J. Liang, “Open-vocabulary mobile manipulation in unseen dynamic environments with 3d semantic maps,” *arXiv preprint arXiv:2406.18115*, 2024.

[35] R. Shah, A. Yu, Y. Zhu, Y. Zhu, and R. Martín-Martín, “Bumble: Unifying reasoning and acting with vision-language models for building-wide mobile manipulation,” *arXiv preprint arXiv:2410.06237*, 2024.

[36] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang *et al.*, “Palm-e: An embodied multimodal language model,” 2023.

[37] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, “Code as policies: Language model programs for embodied control,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9493–9500.

[38] I. Singh, V. Blukis, A. Mousavian, A. Goyal, D. Xu, J. Tremblay, D. Fox, J. Thomason, and A. Garg, “Progprompt: Generating situated robot task plans using large language models,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11 523–11 530.

[39] W. Huang, C. Wang, Y. Li, R. Zhang, and L. Fei-Fei, “Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation,” *arXiv preprint arXiv:2409.01652*, 2024.

[40] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei, “Voxposer: Composable 3d value maps for robotic manipulation with language models,” *arXiv preprint arXiv:2307.05973*, 2023.

[41] S. Wang, M. Han, Z. Jiao, Z. Zhang, Y. N. Wu, S.-C. Zhu, and H. Liu, “Llm^ 3: Large language model-based task and motion planning with motion failure reasoning,” in *2024 IEEE/RSJ International Conference*

on Intelligent Robots and Systems (IROS). IEEE, 2024, pp. 12 086–12 092.

[42] S. Nasiriany, F. Xia, W. Yu, T. Xiao, J. Liang, I. Dasgupta, A. Xie, D. Driess, A. Wahid, Z. Xu *et al.*, "Pivot: Iterative visual prompting elicits actionable knowledge for vlms," *arXiv preprint arXiv:2402.07872*, 2024.

[43] C. Gao, Y. Jiang, and F. Chen, "Transferring hierarchical structures with dual meta imitation learning," in *Conference on Robot Learning*. PMLR, 2023, pp. 762–773.

[44] C. Gao, Z. Li, H. Gao, and F. Chen, "Iterative interactive modeling for knotting plastic bags," in *Conference on Robot Learning*. PMLR, 2023, pp. 571–582.

[45] S. Zhang, Z. Xu, P. Liu, X. Yu, Y. Li, Q. Gao, Z. Fei, Z. Yin, Z. Wu, Y.-G. Jiang *et al.*, "Vlabench: A large-scale benchmark for language-conditioned robotics manipulation with long-horizon reasoning tasks," *arXiv preprint arXiv:2412.18194*, 2024.

[46] R. Yang, H. Chen, J. Zhang, M. Zhao, C. Qian, K. Wang, Q. Wang, T. V. Koripella, M. Movahedi, M. Li *et al.*, "Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents," *arXiv preprint arXiv:2502.09560*, 2025.

[47] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu *et al.*, "Rt-1: Robotics transformer for real-world control at scale," *arXiv preprint arXiv:2212.06817*, 2022.

[48] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," *arXiv preprint arXiv:2307.15818*, 2023.

[49] A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain *et al.*, "Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 6892–6903.

[50] O. M. Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, T. Kreiman, C. Xu *et al.*, "Octo: An open-source generalist robot policy," *arXiv preprint arXiv:2405.12213*, 2024.

[51] J. Wen, Y. Zhu, J. Li, M. Zhu, Z. Tang, K. Wu, Z. Xu, N. Liu, R. Cheng, C. Shen *et al.*, "Tinyvla: Towards fast, data-efficient vision-language-action models for robotic manipulation," *IEEE Robotics and Automation Letters*, 2025.

[52] Q. Li, Y. Liang, Z. Wang, L. Luo, X. Chen, M. Liao, F. Wei, Y. Deng, S. Xu, Y. Zhang *et al.*, "Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation," *arXiv preprint arXiv:2411.19650*, 2024.

[53] J. Zheng, J. Li, D. Liu, Y. Zheng, Z. Wang, Z. Ou, Y. Liu, J. Liu, Y.-Q. Zhang, and X. Zhan, "Universal actions for enhanced embodied foundation models," *arXiv preprint arXiv:2501.10105*, 2025.

[54] K. Pertsch, K. Stachowicz, B. Ichter, D. Driess, S. Nair, Q. Vuong, O. Mees, C. Finn, and S. Levine, "Fast: Efficient action tokenization for vision-language-action models," *arXiv preprint arXiv:2501.09747*, 2025.

[55] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," *arXiv preprint arXiv:2304.13705*, 2023.

[56] T. Z. Zhao, J. Tompson, D. Driess, P. Florence, K. Ghasemipour, C. Finn, and A. Wahid, "Aloha unleashed: A simple recipe for robot dexterity," *arXiv preprint arXiv:2410.13126*, 2024.

[57] S. Belkhale and D. Sadigh, "Minivla: A better vla with a smaller footprint," https://ai.stanford.edu/blog/minivla/, 2024.

[58] R. Zheng, Y. Liang, S. Huang, J. Gao, H. Daumé III, A. Kolobov, F. Huang, and J. Yang, "Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies," *arXiv preprint arXiv:2412.10345*, 2024.

[59] D. Qu, H. Song, Q. Chen, Y. Yao, X. Ye, Y. Ding, Z. Wang, J. Gu, B. Zhao, D. Wang *et al.*, "Spatialvla: Exploring spatial representations for visual-language-action model," *arXiv preprint arXiv:2501.15830*, 2025.

[60] X. Li, P. Li, M. Liu, D. Wang, J. Liu, B. Kang, X. Ma, T. Kong, H. Zhang, and H. Liu, "Towards generalist robot policies: What matters in building vision-language-action models," *arXiv preprint arXiv:2412.14058*, 2024.

[61] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in neural information processing systems*, vol. 35, pp. 24 824–24 837, 2022.

[62] M. Reuss, Ö. E. Yağmurlu, F. Wenzel, and R. Lioutikov, "Multimodal diffusion transformer: Learning versatile behavior from multimodal goals," *arXiv preprint arXiv:2407.05996*, 2024.

[63] S. Ye, J. Jang, B. Jeon, S. Joo, J. Yang, B. Peng, A. Mandlekar, R. Tan, Y.-W. Chao, B. Y. Lin *et al.*, "Latent action pretraining from videos," *arXiv preprint arXiv:2410.11758*, 2024.

[64] D. Schmidt and M. Jiang, "Learning to act without actions," *arXiv preprint arXiv:2312.10812*, 2023.

[65] Q. Bu, Y. Yang, J. Cai, S. Gao, G. Ren, M. Yao, P. Luo, and H. Li, "Univla: Learning to act anywhere with task-centric latent actions," *arXiv preprint arXiv:2505.xxxxx*, 2025.

[66] X. Chen, J. Guo, T. He, C. Zhang, P. Zhang, D. C. Yang, L. Zhao, and J. Bian, "Igor: Image-goal representations are the atomic control units for foundation models in embodied ai," *arXiv preprint arXiv:2411.00785*, 2024.

[67] Y. Hu, Y. Guo, P. Wang, X. Chen, Y.-J. Wang, J. Zhang, K. Sreenath, C. Lu, and J. Chen, "Video prediction policy: A generalist robot policy with predictive visual representations," *arXiv preprint arXiv:2412.14803*, 2024.

[68] H. Chen, J. Li, R. Wu, Y. Liu, Y. Hou, Z. Xu, J. Guo, C. Gao, Z. Wei, S. Xu *et al.*, "Metafold: Language-guided multi-category garment folding framework via trajectory generation and foundation model," *arXiv preprint arXiv:2503.08372*, 2025.

[69] Z. Wei, Z. Xu, J. Guo, Y. Hou, C. Gao, Z. Cai, J. Luo, and L. Shao, "D (r, o) grasp: A unified representation of robot and object interaction for cross-embodiment dexterous grasping," *arXiv preprint arXiv:2410.01702*, 2024.

[70] J. Wen, Y. Zhu, J. Li, Z. Tang, C. Shen, and F. Feng, "Dexvla: Vision-language model with plug-in diffusion expert for general robot control," *arXiv preprint arXiv:2502.05855*, 2025.

[71] J. Gu, S. Kirmani, P. Wohlhart, Y. Lu, M. G. Arenas, K. Rao, W. Yu, C. Fu, K. Gopalakrishnan, Z. Xu *et al.*, "Rt-trajectory: Robotic task generalization via hindsight trajectory sketches," *arXiv preprint arXiv:2311.01977*, 2023.

[72] Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. Tenenbaum, D. Schuurmans, and P. Abbeel, "Learning universal policies via text-guided video generation," *Advances in neural information processing systems*, vol. 36, pp. 9156–9172, 2023.

[73] M. Yang, Y. Du, K. Ghasemipour, J. Tompson, D. Schuurmans, and P. Abbeel, "Learning interactive real-world simulators," *arXiv preprint arXiv:2310.06114*, vol. 1, no. 2, p. 6, 2023.

[74] R. Mendonca, S. Bahl, and D. Pathak, "Structured world models from human videos," *arXiv preprint arXiv:2308.10901*, 2023.

[75] S. Nasiriany, S. Kirmani, T. Ding, L. Smith, Y. Zhu, D. Driess, D. Sadigh, and T. Xiao, "Rt-affordance: Affordances are versatile intermediate representations for robot manipulation," *arXiv preprint arXiv:2411.02704*, 2024.

[76] Y. Chen, Z. Chen, J. Yin, J. Huo, P. Tian, J. Shi, and Y. Gao, "Gravmad: Grounded spatial value maps guided action diffusion for generalized 3d manipulation," *arXiv preprint arXiv:2409.20154*, 2024.

[77] W. Yuan, J. Duan, V. Blukis, W. Pumacay, R. Krishna, A. Murali, A. Mousavian, and D. Fox, "Robopoint: A vision-language model for spatial affordance prediction for robotics," *arXiv preprint arXiv:2406.10721*, 2024.

[78] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," *arXiv preprint arXiv:2210.02747*, 2022.

[79] Q. Liu, "Rectified flow: A marginal preserving approach to optimal transport," *arXiv preprint arXiv:2209.14577*, 2022.

[80] M. J. Kim, C. Finn, and P. Liang, "Fine-tuning vision-language-action models: Optimizing speed and success," *arXiv preprint arXiv:2502.19645*, 2025.

[81] J. Han, J. Liu, Y. Jiang, B. Yan, Y. Zhang, Z. Yuan, B. Peng, and X. Liu, "Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis," *arXiv preprint arXiv:2412.04431*, 2024.

[82] K. Tian, Y. Jiang, Z. Yuan, B. Peng, and L. Wang, "Visual autoregressive modeling: Scalable image generation via next-scale prediction," *Advances in neural information processing systems*, vol. 37, pp. 84 839–84 865, 2024.

[83] M. Ranzinger, G. Heinrich, J. Kautz, and P. Molchanov, "Am-radio: Agglomerative vision foundation model reduce all domains into one," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 12 490–12 500.