MetaVLA: Unified Meta Co-Training for Efficient Embodied Adaptation

Chen Li
Carnegie Mellon University
chenli4@andrew.cmu.edu

Han Zhang
Carnegie Mellon University
hanz3@andrew.cmu.edu

Zhantao Yang
Carnegie Mellon University
zhantaoy@andrew.cmu.edu

Fangyi Chen
Carnegie Mellon University
fangyic@andrew.cmu.edu

Anudeepsekhar Bolimera
Carnegie Mellon University
abolimer@andrew.cmu.edu

Marios Saavides
Carnegie Mellon University
marioss@andrew.cmu.edu

Abstract

Vision–Language–Action (VLA) models show promise in embodied reasoning, yet remain far from *true generalists*—they often require task-specific fine-tuning, incur high compute costs, and generalize poorly to unseen tasks. We propose **MetaVLA**, a unified, backbone-agnostic post-training framework for efficient and scalable alignment. MetaVLA introduces *Context-Aware Meta Co-Training*, which consolidates diverse target tasks into a single fine-tuning stage while leveraging structurally diverse auxiliary tasks to improve in-domain generalization. Unlike naive multi-task SFT, MetaVLA integrates a lightweight meta-learning mechanism—derived from Attentive Neural Processes—to enable rapid adaptation from diverse contexts with minimal architectural change or inference overhead. On the LIBERO benchmark, MetaVLA with six auxiliary tasks outperforms OpenVLA by up to 8.0% on long-horizon tasks, reduces training steps from 240K to 75K, and cuts GPU time by $\sim$76%. These results show that scalable, low-resource post-training is achievable, paving the way to general-purpose embodied agents.

1 Introduction

Recent years have seen rapid progress in embodied Vision–Language–Action (VLA) modeling driven by supervised fine-tuning (SFT) or reinforcement learning (RL) of large language models to perform new embodiment task transfer. In these pipelines, a pretrained VLA backbone is adapted to decode action tokens autoregressive by training on annotated video or image demonstrations with instructions on the new embodiment tasks Kim et al. [2024], Brohan et al. [2022, 2023], O'Neill et al. [2024].

Despite these gains, SFT-based VLA methods face practical limits on benchmarks with scarce per-task data. Current practice Kim et al. [2024] fine-tunes each embodiment task independently, increasing total training cost and preventing transfer across related tasks. Such task-specific schedules are brittle: many gradient steps are needed before stable action sequences emerge, raising overfitting risks and slowing adaptation to new variants or limited context. For example, training all four LIBERO suites requires 240K steps, with long-horizon tasks like LIBERO Long further dominating iterations and becoming the bottleneck OpenVLA Team [2024].

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Space in Vision, Language, and Embodied AI.

While recent work Black et al. [2024], Intelligence et al. [2025], Qu et al. [2025] expands datasets and explores backbone or pretraining innovations, we approach the problem from an orthogonal perspective—post-training. Starting with a vanilla multi-task co-training baseline across the LIBERO suites, we observe reduced GPU hours and improved success rates. However, naive inclusion of auxiliary tasks with diverse domains slows convergence and degrades performance. We attribute this to the heterogeneous optimization instability, where misalignments in both the feature space (e.g., camera views) and action space (e.g., degrees of freedom) hinder the benefits of co-training.

Building on these ideas, we propose **MetaVLA**, a unified framework that fills a critical gap in VLA post-training by introducing auxiliary tasks without the inefficiencies of per-task SFT or the performance drop of naive multi-task SFT. MetaVLA trains a single model across target tasks (e.g., LIBERO suites) to harness cross-task gradients, while auxiliary tasks are incorporated through ***Meta-Action-Reasoner (MAR)*** that injects out-of-domain information gain without disrupting target optimization, enabling scalable and robust adaptation. Experiments show that MetaVLA with six auxiliary tasks outperforms the OpenVLA baseline by 4.4% and its joint-task SFT counterpart by 3.7% on average. On LIBERO Long, gains reach 8.0% and 5.1%, respectively. Moreover, MetaVLA reduces model count to one and cuts total training steps from 240K to 75K. In summary, this work explores an underexamined direction: enhancing post-training efficiency and generalization by incorporating diverse auxiliary tasks with minimal optimization overhead. MetaVLA offers a plug-in module and training recipe for scalable, backbone-agnostic adaptation with strong generalization, and extensive experiments demonstrate that it achieves superior performance and efficiency.

2 Related Work

2.1 Embodiment VLA

Recent VLA progress stems from supervised fine-tuning (SFT) of pretrained language and multi-modal backbones to map visual context and instructions to action sequences. Large systems (e.g., OpenVLA Kim et al. [2024]) show strong generalist policies via pretraining and SFT, but highlight key limitations: SFT typically requires per-task tuning and high compute. Benchmarks like LIBERO Liu et al. [2023] emphasize transfer and lifelong learning across related tasks, motivating more compute-efficient, cross-task training approaches.

2.2 Meta-Learning

Meta-learning trains models to rapidly adapt using few-shot context, often via episodic training Finn et al. [2017], Koch [2015], Santoro et al. [2016], Ravi and Larochelle [2016]. In VLA, both gradient-based (fast fine-tuning) and inference-based (context-to-prediction mapping) approaches are used. Attentive Neural Processes (ANP) Kim et al. [2019] are amortized meta-learners that use attention to produce context-conditioned predictions. For VLA, ANP is appealing due to its task-invariance, ability to focus on relevant demonstrations, and avoidance of direct context optimization during adaptation. This simplifies cross-domain training and promotes stability—crucial for achieving strong results with auxiliary data, as shown later.

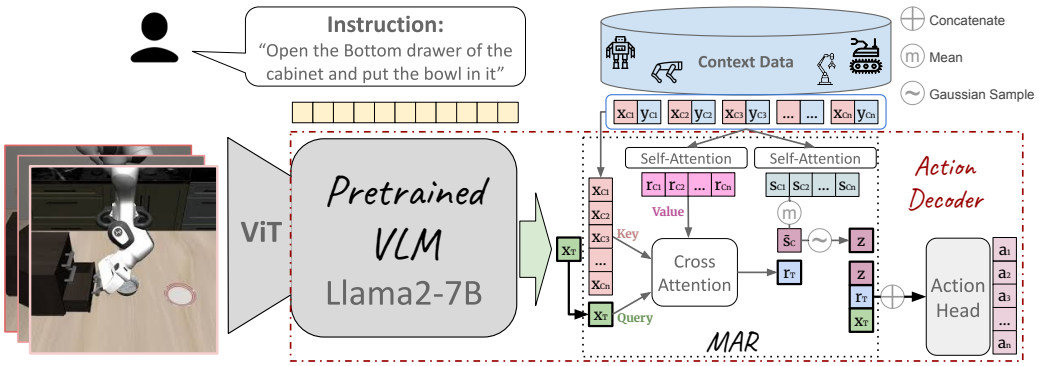

Figure 1: **MetaVLA architecture: VLA backbone married with ANP-based Meta-Learning**

3 Method

3.1 Meta-Learning with VLA

To improve convergence and generalization in low-data task adaptation, we base our architecture on Attentive Neural Processes (ANP) Kim et al. [2019]—a meta-learner inspired by Gaussian Processes that models a distribution over functions conditioned on both context and target representations. These latent codes capture global and task-specific semantics, aggregated via self-attention and cross-attention Vaswani et al. [2023], respectively.

We introduce a compact module, ***Meta-Action-Reasoner (MAR)***, integrated into the Llama2 Touvron et al. [2023] action decoder. Following the original ANP formulation, *MAR* first applies self-attention across context examples to extract a global prior, which is then fused with target queries through cross-attention to form task-aware hybrid representations. Formally, given the target feature x_T, contextual feature-action pairs $(x_{Ci}, y_{Ci}) \in (x_C, y_C)$, *MAR* models the conditional distribution of functions over target action y_T given global and task-specific observations:

$$p(\mathbf{y}_T|\mathbf{x}_T, \mathbf{x}_C, \mathbf{y}_C) := \int p(\mathbf{y}_T|\mathbf{x}_T, \mathbf{r}_T, z)\, q(z|\bar{\mathbf{s}}_C)\, dz \tag{1}$$

Here, $\mathbf{r}_{Ci} \in \mathbf{r}_C$ and $\mathbf{s}_{Ci} \in \mathbf{s}_C$ are per-context representations aggregated from all contexts data pairs (x_C, y_C) through self-attention. r_T is the cross-attention output of query x_T with context keys x_{Ci} and values $\mathbf{r}_{Ci}$. $\bar{\mathbf{s}}_C$ is the mean of all $\mathbf{s}_{Ci}$, while z is a stochastic latent drawn from the approximate posterior $q(z|\bar{\mathbf{s}}_C)$ computed over the context. During training, an additional condensed target representation $\bar{\mathbf{s}}_T$ is produced by the same self-attention and mean process as for $\bar{\mathbf{s}}_C$, with ground truth pair (x_T, y_T). By reparameterizing the Gaussian latent z, the training objective maximizes a variational lower bound:

$$\log p(\mathbf{y}_T|\mathbf{x}_T, \mathbf{x}_C, \mathbf{y}_C) \geq \mathbb{E}_{q(z|\mathbf{s}_T)}[\log p(\mathbf{y}_T|\mathbf{x}_T, \mathbf{r}_T, z)] - D_{\mathrm{KL}}(q(z|\bar{\mathbf{s}}_T) \,\|\, q(z|\bar{\mathbf{s}}_C)) \tag{2}$$

This formulation enables MetaVLA to reconstruct target actions, regularized by a KL divergence that prevents the target distribution from drifting too far from the context distribution.

Unlike standard ANP, which uses smaller-scale neural networks, we integrate a pretrained Llama-2 Touvron et al. [2023] backbone from OpenVLA. *MAR* generates both stochastic and deterministic contextual latent vectors, which are concatenated with the Llama hidden states before the final output layer. The combined representations are then passed through the LM head to produce output logits, enabling end-to-end training via standard Llama decoding. See Figure 1 for a framework overview

3.2 Adding Auxiliary Tasks

To increase context diversity and strengthen meta-learning, we add auxiliary tasks NVIDIA et al. [2025] NVIDIA [2025] into context set. First, GR00T is entirely unseen during OpenVLA pretraining, making it a valuable source of additional information gain. Second, it offers partial domain relevance to LIBERO while differing structurally—striking a balance between familiarity and diversity. Examples of task difference among these three types are showing in Figure 8 and Section A.2.

Unlike Zhao et al. [2025], which carefully select tasks highly similar to LIBERO, our method is less strict in data varieties, leading to a more scalable adaption framework. Ablation study on the effect of auxiliary task selection is presented in Section 4.2.

4 Experiments

4.1 Main Results of MetaVLA

As shown in Table 1, MetaVLA—with or without auxiliary tasks—outperforms all baselines, including OpenVLA and SFT-4LIBERO, across all LIBERO tasks and on average. With six auxiliary tasks, it improves over OpenVLA by 4.4% and SFT-4LIBERO by 3.7%, **notably on LIBERO-Long**, with gains of +8.0% and +5.1%, respectively. Moreover, MetaVLA reduces model count to one and cuts training steps from 240K to 75K, reducing GPU time by 76%, from ~100 hours to ~24 hours.

Model	Training Steps	Goal (%)	Spatial (%)	Object (%)	Long (%)	Average (%)
Diffusion Policy Chi et al. [2023], Kim et al. [2024]	-	68.3	78.3	**92.5**	50.5	72.4
ATM Wen et al. [2023]	-	77.8	68.5	68.0	39.3	63.4
OpenVLA Kim et al. [2024]	240K	76.2	84.7	87.0	51.8	74.9
SFT-4LIBERO	75K	77.8	84.8	87.4	54.7	76.2
SFT-4LIBERO+1single+1bimanual		59.7	68.0	65.2	30.0	55.7
SFT-4LIBERO+3single		24.6	16.8	9.7	1.5	13.2
SFT-4LIBERO+5single+1bimanual		15.2	5.6	12.0	1.6	8.6
MetaVLA-Pretrained-Context-ONLY		74.4	85.4	85.4	52.3	74.4
MetaVLA (ours)		**78.9**	88.5	88.5	55.3	77.8
MetaVLA+Stochastic (ours)		**78.9**	88.9	88.5	53.0	77.3
MetaVLA+1single+1bimanual (ours)		78.5	89.0	87.4	59.0	78.5
MetaVLA+3single (ours)		78.0	88.0	87.2	59.7	78.2
MetaVLA+5single+1bimanual (ours)		78.7	**89.9**	88.9	**59.8**	**79.3**

Table 1: **Comparison with prior methods.** All MetaVLA variants are single models trained for 75k steps. MetaVLA excludes stochastic latent, while MetaVLA+Stochastic includes them. SFT-4LIBERO baselines are single models without meta-learning. OpenVLA baselines are four LIBERO fine-tuned models with over 200k steps. On average, MetaVLA with six auxiliary tasks excels OpenVLA by 4.4%, SFT-4LIBERO by 3.7%, and on LIBERO-Long by 8.0% and 5.1%, respectively.

4.2 Ablation Study

Effect of Auxiliary Task Selection As shown in Table 1, MetaVLA with all three auxiliary settings outperforms its SFT-4-LIBERO counterparts, demonstrating robust generalization to changes in camera views, action spaces, and the number of context tasks.

Effect of Context Batch Size As shown in Figure 2, a context batch size of 32 yields the best performance, with success rates increasing monotonically with batch size under our setting. Larger sizes remain unexplored due to GPU memory limits and are left for future work.

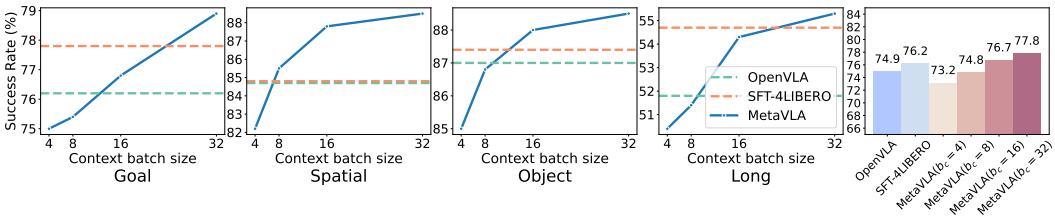

Figure 2: **Comparison of context batch sizes across LIBERO.** Baseline is the OpenVLA released SFT models, and SFT-4LIBERO are baseline single models trained without meta-learning. Within GPU memory limits, larger context batches lead to higher success rates.

Parameter Change and Efficiency Our method adds only 10% more trainable parameters. An ablation with pretrained-only contexts (MetaVLA-Pretrained-Context-ONLY) performs worse (Table 1), confirming the gains come from meta-learning rather than parameter increase. Moreover, MetaVLA matches OpenVLA in LIBERO simulation latency (Figure 7), adding no computational overhead.

4.3 Why Our Method Works?

Joint training promotes knowledge sharing across in-domain tasks, while *MAR* leverages diverse auxiliary data and mitigates gradient conflicts from diverse auxiliary tasks. Section A.1 (Appendix) shows convergence trends for three MetaVLA variants from Table 1, supporting this design. Table 1 further shows that MetaVLA+Stochastic improves over MetaVLA on Spatial, performs similarly on Goal and ObJect, but degrades on Long—likely due to its higher complexity and precision demands. We leave deeper analysis to future work due to computational constraints.

5 Conclusion and Limitation

We introduced MetaVLA, a meta-learning framework that trains a single unified VLA model for efficient cross-task transfer and out-of-the-box generalization. Using Attentive Neural Processes, it adapts to diverse tasks without per-task tuning, boosting sample efficiency, long-horizon performance, and stability. MetaVLA removes the need for multiple models, reducing compute and enabling carbon savings. On LIBERO, it outperforms baselines in both success and convergence. More rigorous equal-step comparisons for auxiliary-task models are left to future work due to compute limits.

References

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2024. URL https://arxiv.org/abs/2410.24164.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil J. Joshi, Ryan C. Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Anand Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Ho Vuong, F. Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-1: Robotics transformer for real-world control at scale. *ArXiv*, abs/2212.06817, 2022. URL https://api.semanticscholar.org/CorpusID:254591260.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Krzysztof Choromanski, Tianli Ding, Danny Driess, Kumar Avinava Dubey, Chelsea Finn, Peter R. Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil J. Joshi, Ryan C. Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Sergey Levine, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Ho Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Ted Xiao, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *ArXiv*, abs/2307.15818, 2023. URL https://api.semanticscholar.org/CorpusID:260293142.

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

Chelsea Finn, P. Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 2017. URL https://api.semanticscholar.org/CorpusID:6719686.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

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

Hyunjik Kim, Andriy Mnih, Jonathan Schwarz, Marta Garnelo, S. M. Ali Eslami, Dan Rosenbaum, Oriol Vinyals, and Yee Whye Teh. Attentive neural processes. *ArXiv*, abs/1901.05761, 2019. URL https://api.semanticscholar.org/CorpusID:58014184.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Gregory R. Koch. Siamese neural networks for one-shot image recognition. 2015. URL https://api.semanticscholar.org/CorpusID:13874643.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qian Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *ArXiv*, abs/2306.03310, 2023. URL https://api.semanticscholar.org/CorpusID:259089508.

NVIDIA. Physicalai-robotics-gr00t-x-embodiment-sim dataset. https://huggingface.co/datasets/nvidia/PhysicalAI-Robotics-GR00T-X-Embodiment-Sim, 2025. Retrieved August 31, 2025.

NVIDIA, :, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. Gr00t n1: An open foundation model for generalist humanoid robots, 2025. URL https://arxiv.org/abs/2503.14734.

OpenVLA Team. Openvla-7b fine-tuned models on libero tasks. https://huggingface.co/openvla, 2024. Checkpoints used:
Goal: https://huggingface.co/openvla/openvla-7b-finetuned-libero-goal,
Spatial: https://huggingface.co/openvla/openvla-7b-finetuned-libero-spatial,
Object: https://huggingface.co/openvla/openvla-7b-finetuned-libero-object,
LIBERO-10: https://huggingface.co/openvla/openvla-7b-finetuned-libero-10.

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

Nair, Suvir Mirchandani, Takayuki Osa, Tanmay Gupta, Tatsuya Harada, Tatsuya Matsushima, Ted Xiao, Thomas Kollar, Tianhe Yu, Tianli Ding, Todor Davchev, Tony Z. Zhao, Travis Armstrong, Trevor Darrell, Trinity Chung, Vidhi Jain, Vincent Vanhoucke, Wei Zhan, Wenxuan Zhou, Wolfram Burgard, Xi Chen, Xiaolong Wang, Xinghao Zhu, Xinyang Geng, Xiyuan Liu, Xu Liangwei, Xuanlin Li, Yao Lu, Yecheng Jason Ma, Yejin Kim, Yevgen Chebotar, Yifan Zhou, Yifeng Zhu, Yilin Wu, Ying Xu, Yixuan Wang, Yonatan Bisk, Yoonyoung Cho, Youngwoon Lee, Yuchen Cui, Yue Cao, Yueh-Hua Wu, Yujin Tang, Yuke Zhu, Yunchu Zhang, Yunfan Jiang, Yunshuang Li, Yunzhu Li, Yusuke Iwasawa, Yutaka Matsuo, Zehan Ma, Zhuo Xu, Zichen Jeff Cui, Zichen Zhang, and Zipeng Lin. Open x-embodiment: Robotic learning datasets and rt-x models : Open x-embodiment collaboration0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903, 2024. doi: 10.1109/ICRA57147.2024.10611477.

Delin Qu, Haoming Song, Qizhi Chen, Zhaoqing Chen, Xianqiang Gao, Xinyi Ye, Qi Lv, Modi Shi, Guanghui Ren, Cheng Ruan, Maoqing Yao, Haoran Yang, Jiacheng Bao, Bin Zhao, and Dong Wang. Embodiedonevision: Interleaved vision-text-action pretraining for general robot control, 2025. URL https://arxiv.org/abs/2508.21112.

Sachin Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *International Conference on Learning Representations*, 2016. URL https://api.semanticscholar.org/CorpusID:67413369.

Adam Santoro, Sergey Bartunov, Matthew M. Botvinick, Daan Wierstra, and Timothy P. Lillicrap. Meta-learning with memory-augmented neural networks. In *International Conference on Machine Learning*, 2016. URL https://api.semanticscholar.org/CorpusID:6466088.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

Chuan Wen, Xingyu Lin, John So, Kai Chen, Qi Dou, Yang Gao, and Pieter Abbeel. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2023.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, Ankur Handa, Ming-Yu Liu, Donglai Xiang, Gordon Wetzstein, and Tsung-Yi Lin. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models, 2025. URL https://arxiv.org/abs/2503.22020.

A Technical Appendices and Supplementary Material

A.1 Training Convergence

Figure 3 presents Training Accuracy, Imitation Loss (cross-entropy over generated discrete action tokens), and L1 Loss (on the transformed continuous actions) for three auxiliary task settings: 1single+1bimanual, 5single+1bimanual, and 3single. In all cases, MetaVLA consistently converges to higher performance across all three metrics.

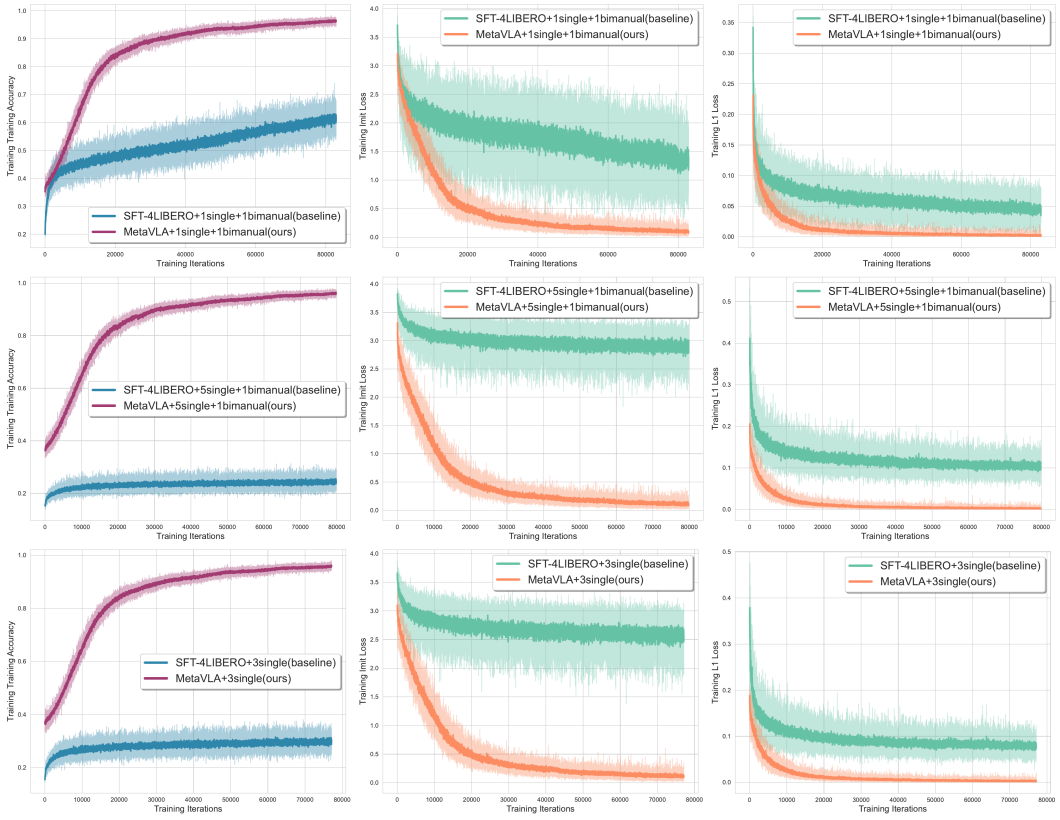

Figure 3: **Training convergence comparison for models trained with 75K steps.** Training Accuracy, Imitation Loss, and L1 Loss are compared between *MetaVLA* variants and *SFT-4LIBERO* under different auxiliary-task settings. All *MetaVLA* variants consistently converges to superior performance across all three metrics, while *SFT-4LIBERO* fails to adapt effectively—highlighting the robustness and scalability of our approach.

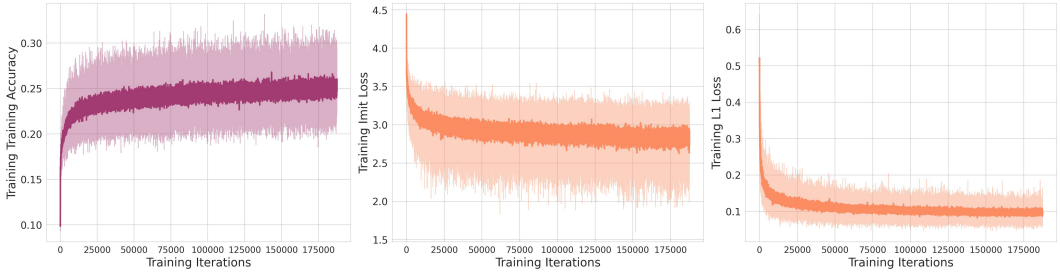

Figure 4: Training convergence of MetaVLA with six auxiliary tasks (one bimanual and five single-arm) trained with 187.5K steps. All three metrics—Accuracy, Imitation Loss, and L1 Loss—converge to suboptimal levels.

A.2 Context Task Details

We use the LIBERO dataset Liu et al. [2023] as both target and context tasks, and GR00T NVIDIA [2025] as auxiliary context tasks only. A detailed breakdown of the datasets is provided in Table 2. Example tasks from LIBERO and GR00T are visualized in Figures 5 and 6, respectively.

Dataset	Tasks
LIBERO Liu et al. [2023]	LIBERO-Goal LIBERO-Spatial LIBERO-Object LIBERO-Long
GR00T NVIDIA [2025]	bimanual_panda_gripper.Threading
	single_panda_gripper.CoffeeServeMug single_panda_gripper.OpenDrawer single_panda_gripper.PnPCabToCounter single_panda_gripper.PnPCounterToMicrowave single_panda_gripper.TurnSinkSpout

Table 2: **Summary of datasets and tasks used in the experiments.**

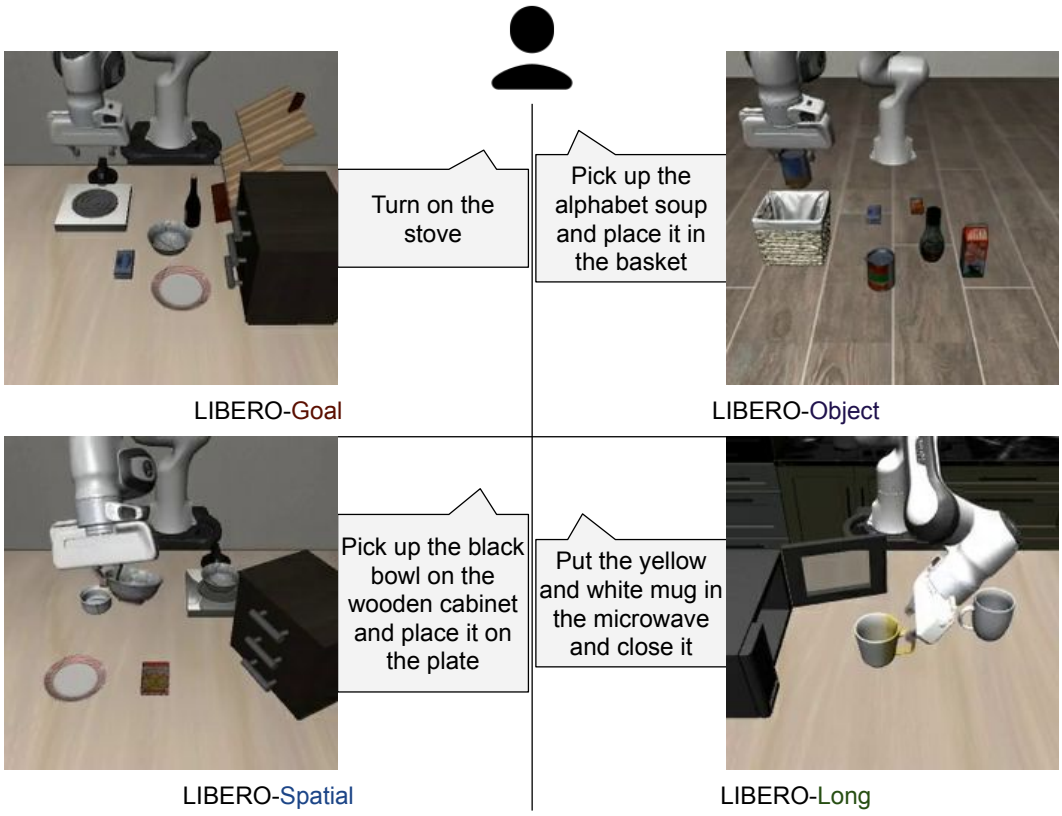

Figure 5: **LIBERO examples.** Each suite example includes a frame from the primary camera view together with its task instruction.

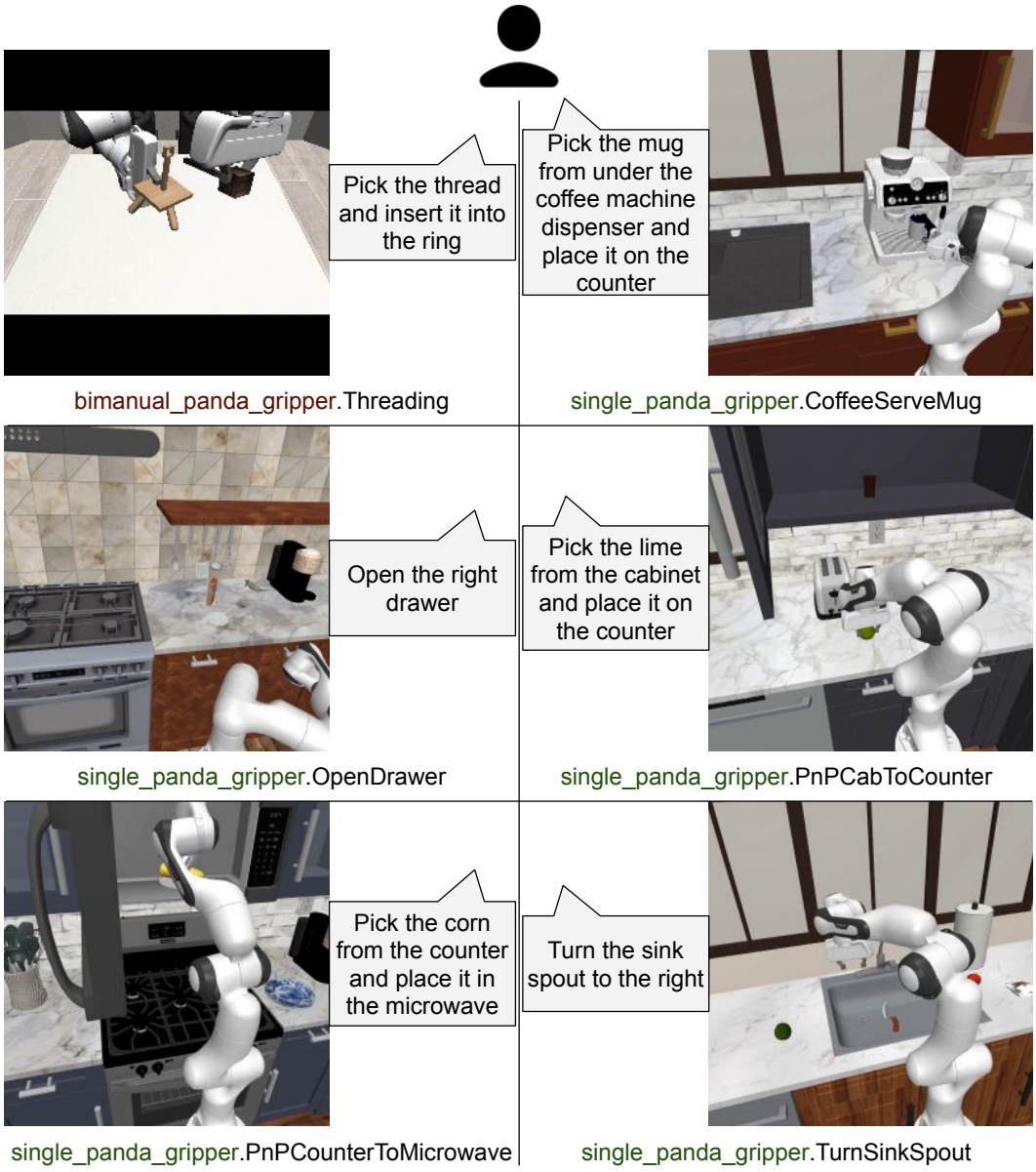

Figure 6: **GR00T examples.** Each task example includes a frame from the primary camera view paired with its task instruction.

A.3 Model Architecture and Training Details

Model Architecture We build on OpenVLA-7B Kim et al. [2024] as the base model, integrating *MAR*, a lightweight, memory-based meta-learning module. In *MAR*, global prior representations are encoded via self-attention, while cross-attention Vaswani et al. [2023] fuses target and context to produce a final hybrid latent representation. Each attention block is followed by Layer Normalization and a final MLP projection.

Training Settings We trained all MetaVLA variants with LoRA Hu et al. [2021] on 8 A100-80 GB GPUs with 75K training steps, taking approximately 24 GPU hours, using 8 x 80GB VRAM. Training hyperparameters are in Table 3.

Hyperparameter	Value
Shuffle Buffer Size	100000
FlashAttention-2	Enabled
LoRA Rank	32
LoRA Dropout	0.0
Total Batch Size	128
Gradient Accumulation Steps	1
Learning Rate	5e-4
Context Batch Size	32
MAR Latent Dimension	2048

Table 3: **Training Hyperparameters.** Total batch size is computed as 16 samples per GPU across 8 GPUs. Context batch size refers to the batch size used for each individual context task.

A.4 Experiment Details

A.4.1 Inference Efficiency

Our method is engineering-friendly and computationally lightweight. We measure both token throughput and latency of the model end-to-end, on one 24GB RTX-4090 GPU against OpenVLA Kim et al. [2024]. All environments and packages are kept the same throughout the experiment to ensure fair comparison. Our efficiency results are shown in Figure 7. Our *MAR* module introduces approximately 5.5% more latency compared to OpenVLA, making MetaVLA an ideal practical choice for achieving a higher success rate.

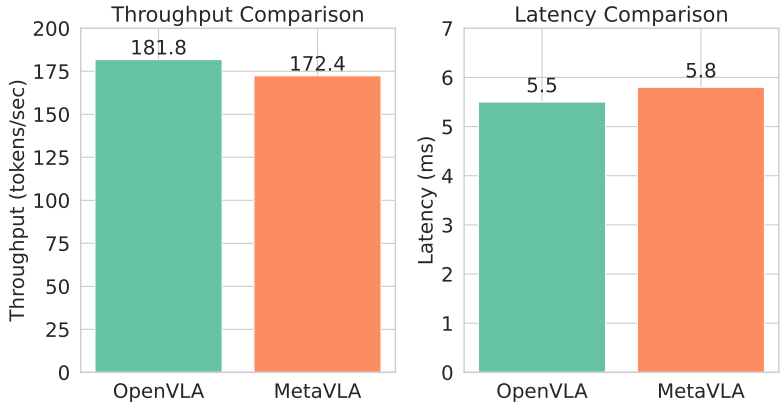

Figure 7: **Efficiency Metrics.** Our lightweight module only adds negligible overhead to inference cost, making MetaVLA practical for deployment and usage.

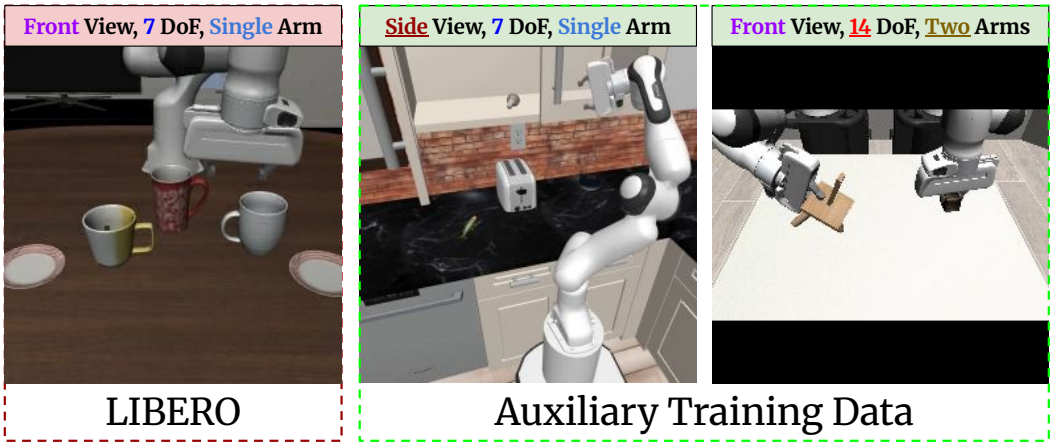

Figure 8: **Comparison between auxiliary tasks and LIBERO evaluation benchmark.** LIBERO tasks use third-person front-view images and 7-DoF actions for a single-arm robot. In contrast, our auxiliary data from GR00T introduces variation through side-view observations and a two-arm robot with 14-DoF actions. MetaVLA benefits from this data diversity, while OpenVLA struggles with the domain mismatch.

A.4.2 Effect of Context Batch Size

Table 4 shows the success rates of MetaVLA across different LIBERO tasks using different context batch sizes. The performance scales up as we introduce more contextual data.

Method	Goal	Spatial	Object	Long	Average
OpenVLA Kim et al. [2024]	76.2	84.7	87.0	51.8	74.9
SFT-4LIBERO	77.8	84.8	87.4	54.7	76.2
MetaVLA($\mathbf{b}_C = 4$)	75.0	82.2	85.0	50.4	73.2
MetaVLA($\mathbf{b}_C = 8$)	75.4	85.5	86.8	51.4	74.8
MetaVLA($\mathbf{b}_C = 16$)	76.8	87.8	88.0	54.3	76.7
MetaVLA($\mathbf{b}_C = 32$)	**78.9**	**88.5**	**88.5**	**55.3**	**77.8**

Table 4: Effect of different context batch sizes across different LIBERO task suites.

A.5 Auxiliary Task Comparisons

We show an example from each of the three data types in Figure 8

A.6 Success Cases in LIBERO Simulation

Figures 9, 10, 11, and 12 demonstrate example execution sequences of MetaVLA successfully completing one task from each LIBERO suite in its simulation: Goal, Spatial, Object, and Long.

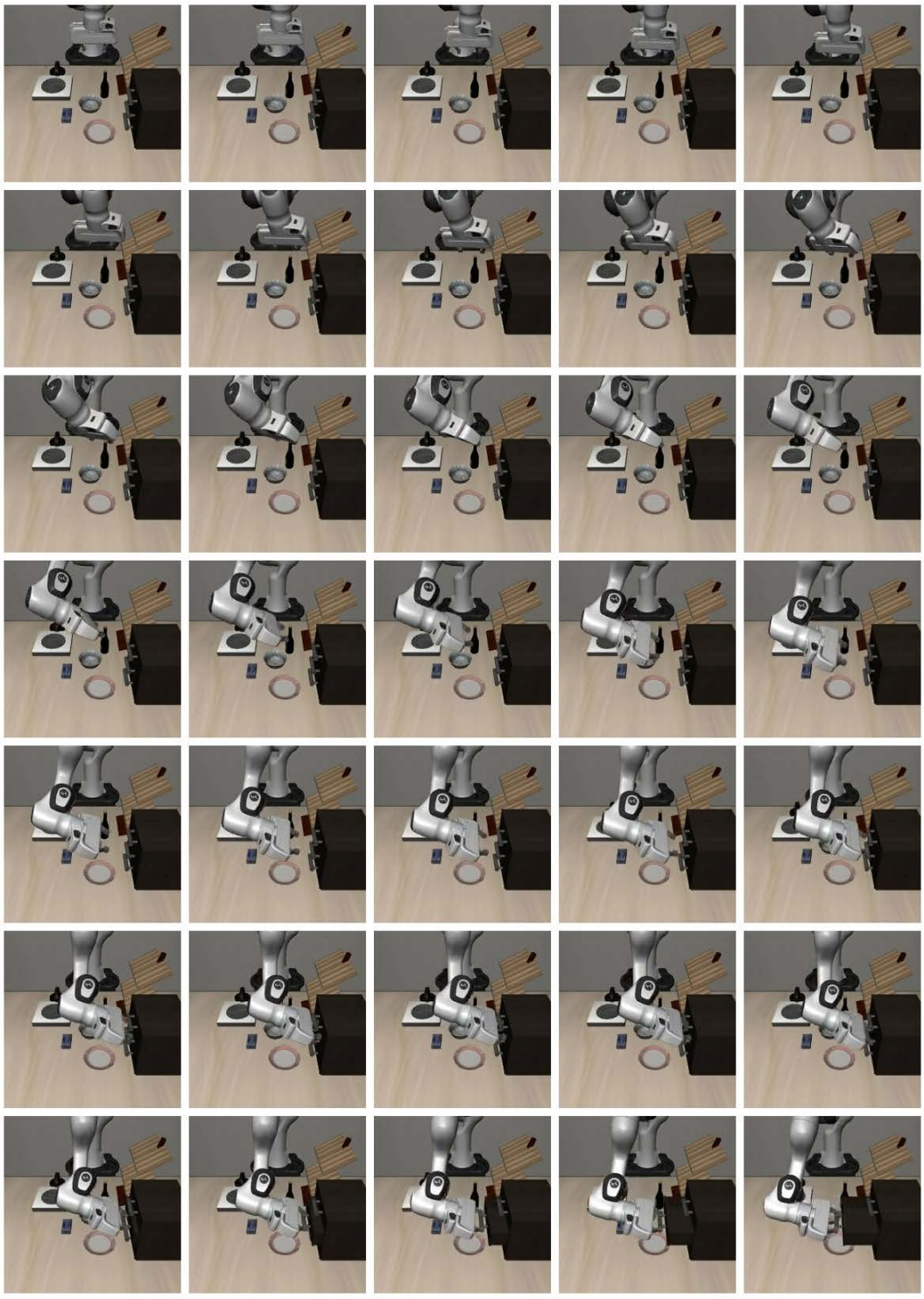

Figure 9: **MetaVLA Execution Sequence Example on LIBERO-Goal.** Instruction: *Open the middle drawer of the cabinet*

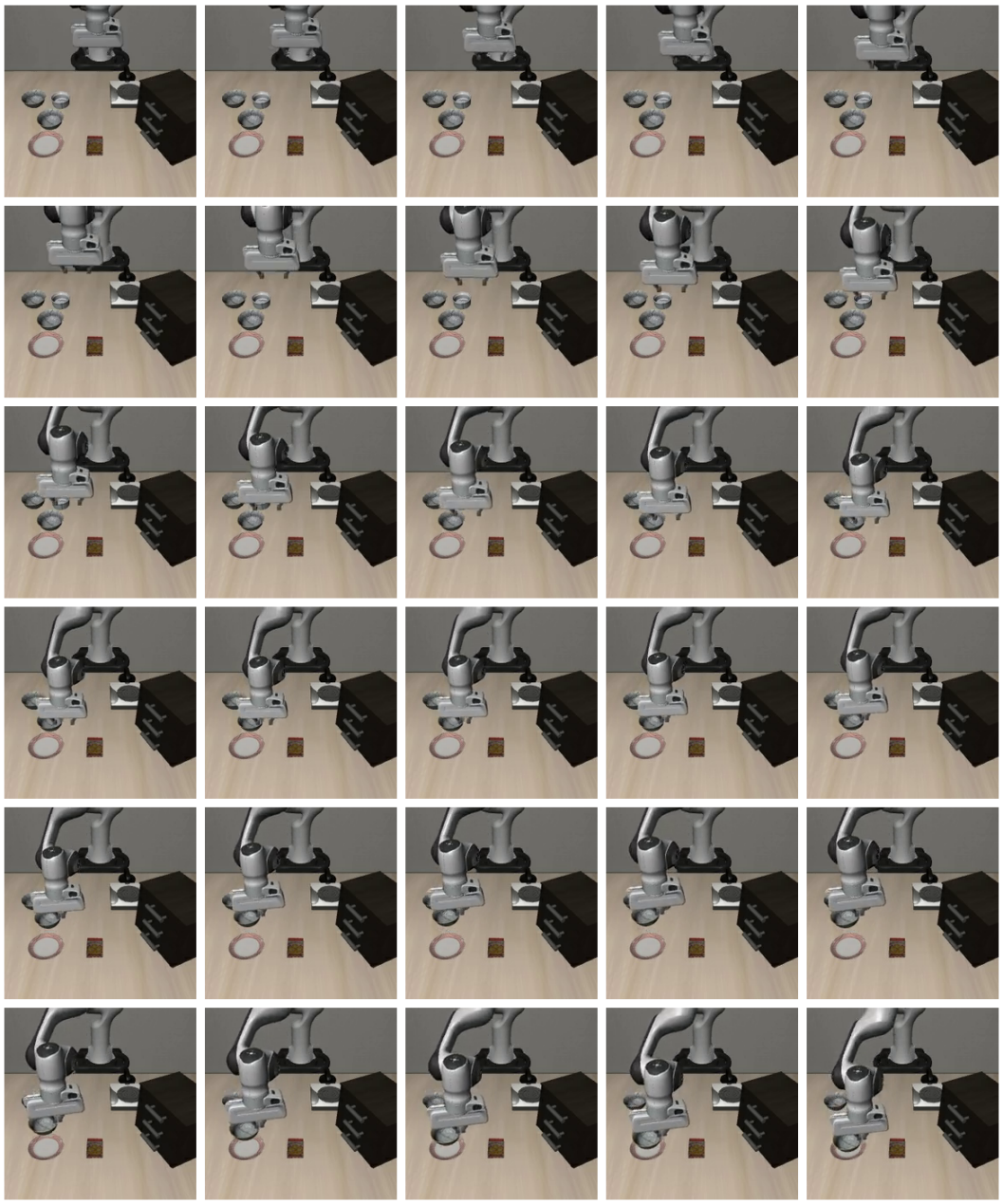

Figure 10: **MetaVLA Execution Sequence Example on LIBERO-Spatial.** Instruction: *Pick up the black bowl between the plate and the ramekin and place it on the plate*

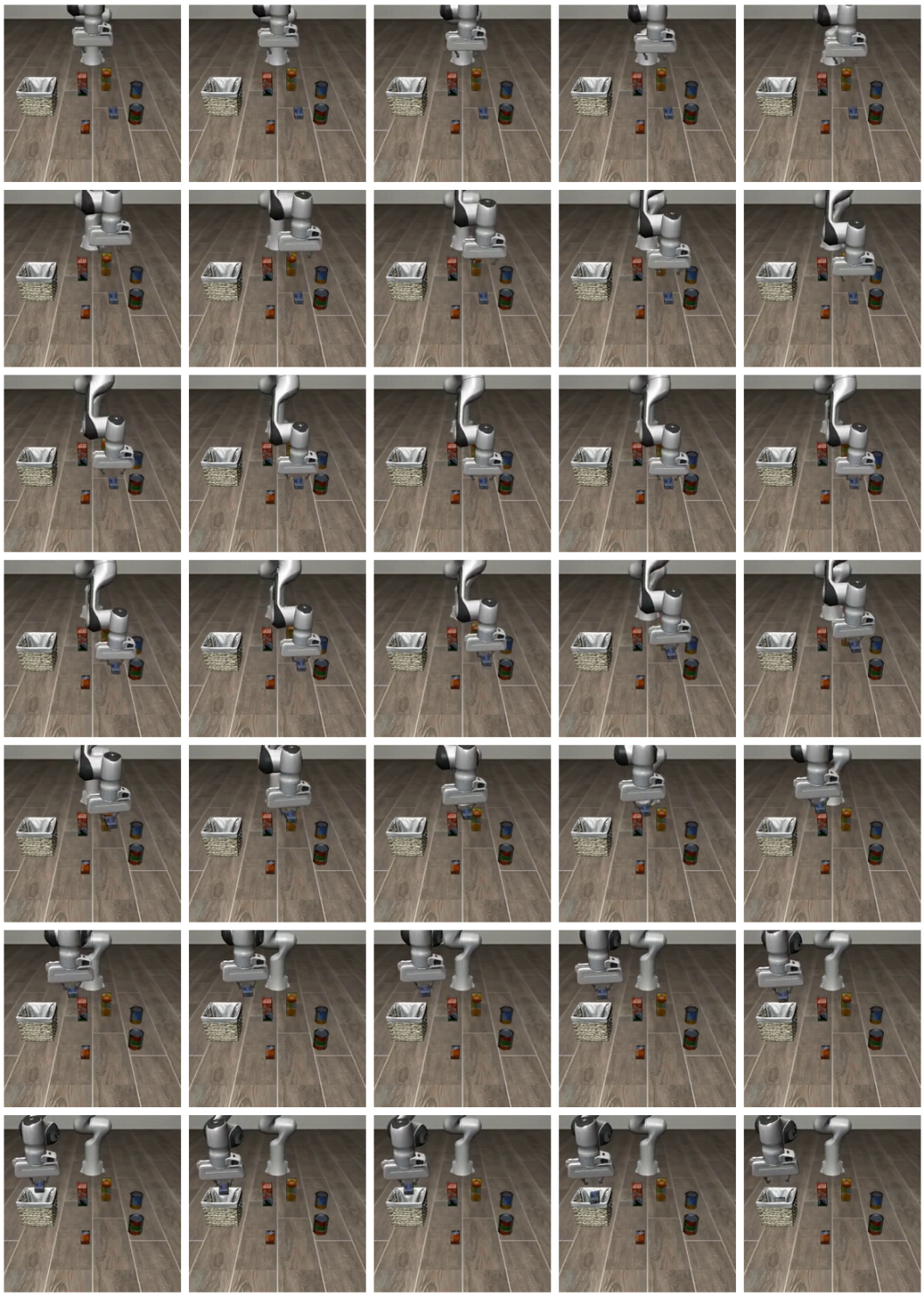

Figure 11: **MetaVLA Execution Sequence Example on LIBERO-Object.** Instruction: *Pick up the cream cheese and place it in the basket*

Figure 12: **MetaVLA Execution Sequence Example on LIBERO-Long.** Instruction: *Put the black bowl in the bottom drawer of the cabinet and close it*

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our main claims are summarized in Figure 1 and Table 1, and detailed in Section 3 and Section 4.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have discussed the limitations in the final part of Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This is not a theoretical paper.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have provided our experiment settings in Section 3, 4 and Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: While our codebase is not yet ready for public release at the time of submission, we provide detailed implementation and hyperparameter descriptions to support reproducibility.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We have provided our experiment settings in Section 3, 4 and Appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We provide standard errors in the results in Appendix A.3 and Table 2, with an average of over three random seeds. All of the MetaVLA series models fall within the provided error bound. We haven't shown errors for all training curves due to high computation cost.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provided them in Section A.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We are convinced that we comply with NeurIPS Code of Ethics

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Our work focuses on small-scale vision language action model training. It does not pose broader societal impact beyond advancing our understanding of specific aspects of deep learning.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our paper does not have such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We've credited and cited the references and codebases appropriately in the paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We don't release any new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our research does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: Our paper only uses the LLM for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.