UniviewVLA: A Unified Multiview Vision-Language-Action Model with World Modeling

Tao Xu[1,2], **Runhao Zhang**[2], **Zhijian Huang**[4], **Jiayi Guan**[4], **Jiaxin Wang**[2], **Yifan Ding**[2], **Yong-Lu Li**[2,3], **Long Chen**[4], **Guang Chen**[1,2,†], **Jinghui Lu**[4,†]

[1]Tongji University, [2]Shanghai Innovation Institute, [3]Shanghai Jiao Tong University, [4]Xiaomi EV

Abstract: Occluded tasks remain a bottleneck in robot manipulation. Existing solutions either deploy additional physical cameras requiring training-inference camera parity, or rely on explicit 3D reconstruction with high computational cost. Moreover, both approaches rely on standard agent-view and wrist-view observations, while failing to capture occlusion information and future scene evolution. To this end, we propose **UniviewVLA**, a **uni**fied multi**view** **V**ision-**L**anguage-**A**ction model with world modeling, which infers multiview scene evolution for action prediction from only standard two-camera observations. We demonstrate that by leveraging generated multiview future views from the world model, UniviewVLA reveals occluded cues and models future scene evolution, improving action prediction and removing the need for extra hardware or explicit reconstruction. Besides, to accelerate inference while preserving prediction accuracy, UniviewVLA develops Motion-Informative Token Compression, which compresses each generated view from 625 to 16 tokens and reduces per-view latency from 6–7s to 0.2–0.3s. UniviewVLA also proposes training-free Action-Entropy View Selection, which dynamically identifies the most action-informative view at different inference stages. Extensive experiments show that UniviewVLA achieves 95.8% on LIBERO and 4.60 on CALVIN ABCD→D, both standard occlusion-free benchmarks. On customized occlusion-focused tasks, it improves success rate from 40.0% to 73.3%, and average real-robot success rate by 33.4 points, demonstrating stronger occlusion-focused performance without sacrificing standard occlusion-free benchmarks. Project website: This URL.

1 Introduction

Vision-Language-Action (VLA) models have made substantial progress in robot manipulation by mapping visual observations and language instructions to robot actions [1, 2, 3, 4, 5]. However, their performance is fundamentally constrained by the visual information available at deployment time. In manipulation, action-critical cues, including gripper-object relations, object poses, and contact regions, are highly view-dependent. When these cues are occluded in the deployed camera views, the model may fail even if the underlying task knowledge has been learned.

A natural solution is to expand the visual input by deploying additional physical cameras, and prior multiview methods have shown that richer visual coverage improves manipulation performance [6, 7, 8]. However, these approaches require strict camera parity between training and inference, *i.e.,* the number, placement, and calibration of cameras must be consistently replicated across deployment sites, which is brittle and difficult to scale. Another line of work resorts to explicit 3D reconstruction to infer scene geometry from partial observations [9, 10, 11, 12], but it is limited to current scene states and incurs substantial computational cost. This motivates a central question: **Can a world model conditioned on only standard observations infer multiview information and future scene evolution for downstream action prediction, and thereby improve occlusion-focused manipulation without high-cost sensing setups?**

To answer this question, we propose **UniviewVLA**, a **uni**fied multi**view** **VLA** model with world modeling capabilities, that infers multiview scene evolution for action prediction. The model learns

[†]Corresponding authors.

to predict multiview future scene evolution from only standard two-camera observations. By conditioning action prediction on the generated auxiliary future-view representations, UniviewVLA injects multiview information and future scene evolution knowledge into robot manipulation, and produces occluded action-critical evidence without requiring costly additional camera deployment or explicit 3D reconstruction [13, 14, 15, 16, 17, 18]. We first demonstrate that introducing additional views, whether derived from ground-truth images quantized using visual quantization (VQ) techniques [4] or generated by a world model, consistently improves prediction accuracy, as shown in Section 3. Further, we discovered that such formulation introduces two practical challenges: (1) redundant auxiliary-view tokens increase latency and may harm action prediction, and (2) the most informative viewpoint varies across different inference stages. UniviewVLA addresses token redundancy with *Motion-Informative Token Compression* (Section 4.3), which compresses each generated view from 625 to 16 tokens and reduces per-view latency from 6–7s to 0.2–0.3s. To enable dynamic view selection, UniviewVLA also proposes *Action-Entropy View Selection* (Section 4.4), which identifies the most action-informative view for action prediction. We evaluate UniviewVLA on standard occlusion-free benchmarks, where it achieves 95.8% success on LIBERO and 4.60 on CALVIN ABCD→D. To further evaluate its occlusion performance, we construct six occlusion-focused simulation tasks with customized occlusion scenarios and collect multiview demonstrations through 3D-SpaceMouse teleoperation. UniviewVLA achieves an average improvement from 40.0% to 73.3% success on these tasks. In two real-world occlusion tasks, it yields a 33.4 points gain in average success rate.

Our contributions are summarized as follows:

- We propose **UniviewVLA**, a unified multiview VLA framework with world modeling that infers multiview future scene evolution before predicting action, without requiring additional physical cameras or explicit 3D reconstruction. Experimental results show that UniviewVLA achieves an average improvement of 33.3 percentage points on six occlusion-focused simulation tasks and 33.4 percentage points gain on two real-world occlusion-focused tasks.
- We develop Motion-Informative Token Compression and Action-Entropy View Selection to reduce redundant auxiliary-view tokens and enable dynamic multiview selection across inference stages without additional training, significantly decreasing the inference latency.
- We introduce six customized occlusion-focused manipulation tasks and an open-source pipeline for multiview data collection, processing, and evaluation, enabling assessment of occlusion-focused performance.

2 Related Work

Vision-Language-Action Models. Modern Vision-Language-Action (VLA) models incorporate language conditioning and vision-language pretraining [19, 20, 21, 22, 23, 24, 25], leading to generalist policies that map language-conditioned visual observations directly to robot actions [26, 27, 1, 28, 29, 2, 30, 31]. Recent work further improves action prediction performance through flow matching, action tokenization, diffusion experts, video-based policy learning, and world modeling [3, 32, 9, 6, 4, 5]. Yet most models still perceive the scene through a fixed set of deployed cameras. When action-critical cues are occluded beyond these fixed views, policy-side architectural improvements alone cannot overcome the resulting observability bottleneck.

Multiview Robot Learning. Multiview perception has been widely adopted in robot manipulation to provide complementary visual evidence for robot learning [33, 12, 11, 10, 8]. These methods exploit multiview observations through voxelized representations, virtual re-rendering, multiview transformers, view-scaled demonstrations, or generated views and demonstrations [34, 35, 36, 37, 16, 38, 7]. However, their gains are often tied to hardware constraints: policies trained with additional views typically require similar camera poses, calibration, and synchronization at deployment.

3 Preliminary Experiments

At deployment, VLA models usually use only two standard physical observations, the agent-view $\mathbf{x}_a$ and wrist-view $\mathbf{x}_w$, which may miss critical information under occlusions. As shown in Fig. 1 and Table 4, the agent-view cannot observe the occluded switch in Task 6, whereas the 120° third-camera view exposes it, improving success from 4% to 16% and highlighting the value of multiview information for occlusion-focused manipulation. Nevertheless, physical multiview policies require matched camera configurations between training and inference, and only observe the current scene without predicting future scene evolution. UniviewVLA addresses these constraints by generating auxiliary future views from the two standard observations.

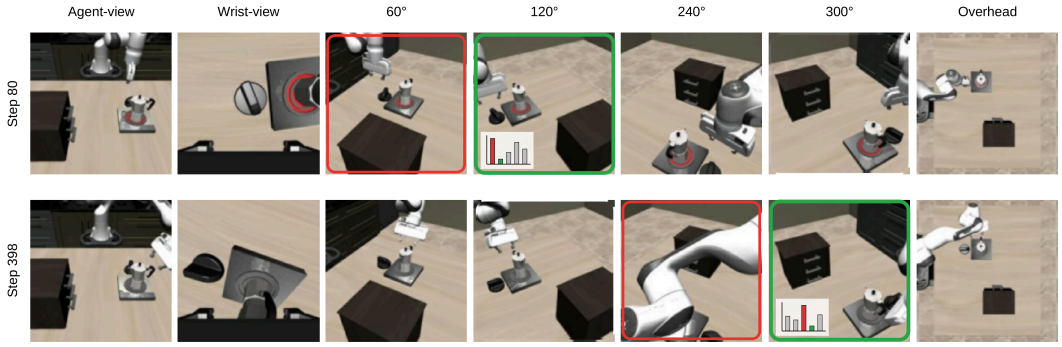

Figure 1: **Multiview observations under occlusion.** Green boxes mark the selected best views with the lowest action entropy, while red boxes mark higher-entropy views. Bar charts show action entropy, with lower entropy indicating a more action-informative view.

However, as shown in Figure 2, each auxiliary-view generation incurs excessive visual tokens from dense future-view prediction, leading to high inference latency and redundant information that may impair action prediction. Therefore, compact auxiliary views that preserve motion-informative evidence are important for downstream action prediction.

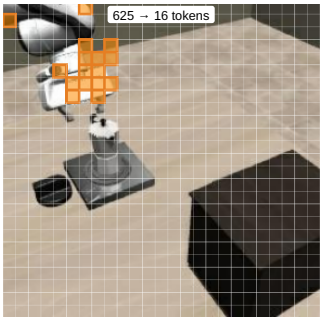

Figure 2: **Motion-informative token compression.**

Moreover, the best viewpoint may change across different inference stages. As shown in Fig. 1, the bar charts report action entropy across auxiliary views, where lower entropy indicates a more action-informative view. For example, the 120° view reveals the occluded switch more clearly than the 300° view at step 80, whereas the 300° view better exposes the mug handle at step 398. The 240° view further shows this temporal shift: it is informative early but becomes self-occluded later. These observations motivate dynamic action-informative view selection.

4 Method

4.1 Overview

As illustrated in Figure 3, UniviewVLA follows the discrete-token autoregressive modeling paradigm of UniVLA [4], representing language, visual observations, auxiliary views, and actions as token sequences modeled by an autoregressive transformer $\mathcal{M}_\theta$. Given the standard agent-view $\mathbf{x}_a$, wrist-view $\mathbf{x}_w$, and instruction L, we denote by $o_{t-1:t} = \{\mathbf{x}_{a,t-1:t}, \mathbf{x}_{w,t-1:t}\}$ the two-frame standard observation history, and by $\mathcal{V}_{\text{sel}}$ the candidate auxiliary-view set. UniviewVLA is trained in two stages:

$$\begin{aligned} &\text{(Stage 1) World model post-training:} \quad \hat{o}^{v}_{t+1} \sim p_\theta(\cdot \mid o_{t-1:t},\, L), \quad v \in \mathcal{V}_{\text{sel}}, \\ &\text{(Stage 2) Action fine-tuning:} \quad \hat{o}'^{v}_{t+1},\, a_t \sim \pi_\theta(\cdot \mid o_{t-1:t},\, L), \quad v \in \mathcal{V}_{\text{sel}}. \end{aligned} \tag{1}$$

Here, $\hat{o}^{v}_{t+1}$ denotes the generated future auxiliary-view representation, $\hat{o}'^{v}_{t+1}$ denotes its compact motion-informative form, and a_t denotes the action token predicted from this representation.

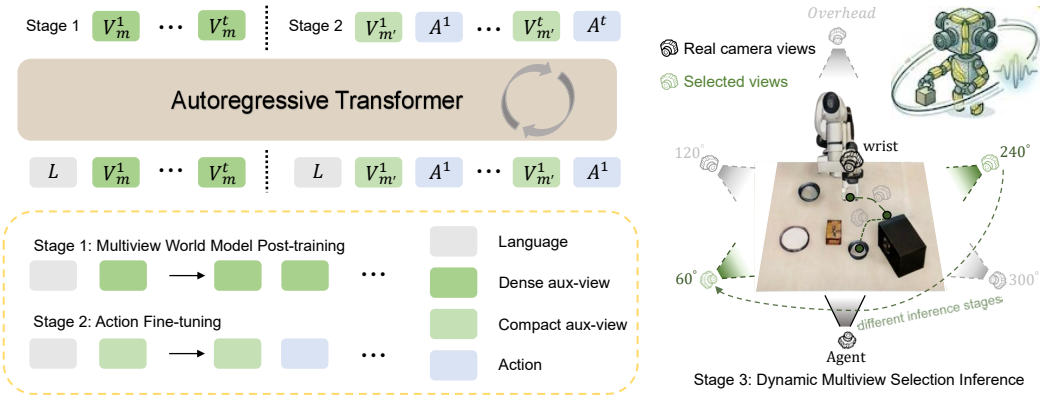

Figure 3: **UniviewVLA pipeline.** UniviewVLA models language instructions, multiview observations, and actions with discrete tokens that can be autoregressively predicted by a unified Transformer model [4], using two training stages and dynamic inference. **(1) Multiview world model post-training.** UniviewVLA takes language instructions, standard agent-view, and wrist-view inputs, and autoregressively generates future multiview images that incorporate multiview and world evolution information. **(2) Action fine-tuning.** UniviewVLA first predicts compact motion-informative tokens to avoid the high latency of full auxiliary-view tokens, and then predicts FAST action tokens. **(3) Dynamic inference.** During inference, transparent cameras denote generated auxiliary views, and the green camera denotes the selected auxiliary view. UniviewVLA periodically selects the best auxiliary view for action prediction across different inference stages instead of using a fixed viewpoint.

4.2 Generative Multiview World Model Post-training

In Stage 1, the world model learns to generate future auxiliary views from the two-frame standard physical observations, providing both multiview information and predictive scene evolution knowledge for downstream action prediction. Specifically, given $o_{t-1:t} = \{\mathbf{x}_{a,t-1:t}, \mathbf{x}_{w,t-1:t}\}$ and instruction L, it predicts the future VQ-token sequence $\mathbf{c}^v_{t+1} = (c^v_{t+1,1}, \ldots, c^v_{t+1,N_{\text{tok}}})$ for each auxiliary view $v \in \mathcal{V}_{\text{sel}}$, where N_{tok} denotes the number of VQ tokens per view. The training objective is autoregressive visual prediction:

$$\mathcal{L}_{\text{world}} = -\sum_{v \in \mathcal{V}_{\text{sel}}} \sum_{i=1}^{N_{\text{tok}}} \log p_\theta \left(c^v_{t+1,i} \mid c^v_{t+1,<i},\, o_{t-1:t},\, L \right), \tag{2}$$

where $c^v_{t+1,<i}$ denotes the previously generated tokens of the future auxiliary view from viewpoint v. By supervising future auxiliary-view prediction, this objective equips the model with future multiview prediction ability and injects scene evolution knowledge into the generated view tokens.

4.3 Joint Motion-Informative Token Compression and Action Fine-tuning

As shown in Figure 2, each full auxiliary view contains $N_{\text{tok}} = 25 \times 25 = 625$ VQ tokens, and generating five views requires 3125 tokens per step. To reduce this cost, we retain motion-relevant tokens from each predicted view. Given the VQ-token sequences $\mathbf{c}^v_t$ and $\mathbf{c}^v_{t+1}$ of auxiliary view v at consecutive timesteps t and $t+1$, we compute a cosine-distance score for each spatial token j:

$$\delta^v_{t,j} = 1 - \cos\left(E(c^v_{t,j}),\, E(c^v_{t+1,j})\right), \quad j = 1, \ldots, N_{\text{tok}}, \tag{3}$$

where $E(\cdot)$ denotes the VQ embedding table. A higher $\delta^v_{t,j}$ indicates stronger visual change and therefore higher motion relevance. We retain the top $K = 16$ tokens ranked by $\delta^v_{t,j}$, yielding the compact motion-informative representation $\hat{o}'^v_{t+1}$ for each view. The model is trained to predict these compact tokens and directly generates them at inference. This reduces each auxiliary view from 625 to 16 tokens and the total auxiliary-view token budget from 3125 to 80.

Given the compact representation, action fine-tuning concatenates the $K = 16$ tokens of each candidate view with the action-token sequence $\mathbf{a}_t = (a_{t,1}, \ldots, a_{t,N_a})$:

$$\mathbf{s}_t^v = \left(\hat{o}_{t+1,1}'^v, \ldots, \hat{o}_{t+1,K}'^v, a_{t,1}, \ldots, a_{t,N_a}\right),$$

where N_a is the number of action tokens. The training objective applies autoregressive cross-entropy supervision over the concatenated sequence:

$$\mathcal{L}_{\text{act}} = -\sum_{v \in \mathcal{V}_{\text{sel}}} \sum_{m=1}^{K+N_a} \log p_\theta\left(s_{t,m}^v \mid s_{t,<m}^v,\, o_{t-1:t},\, L\right). \tag{4}$$

Each candidate view is trained as a view-prefixed sequence, enabling action prediction to condition on compact multiview evidence and the scene evolution knowledge introduced by the world model.

4.4 Dynamic Test-Time View Selection via Action Entropy

As shown in Fig. 3 and motivated by Section 3, UniviewVLA dynamically selects the most action-informative auxiliary view during inference instead of relying on a fixed generated viewpoint. For each view-selection round, it specifies a candidate view $v \in \mathcal{V}_{\text{sel}}$ to the model and generates the corresponding compact representation $\hat{o}_{t+1}'^v$ from $(o_{t-1:t}, L)$. We then compute the mean action-token entropy for this candidate view:

$$H_v = \frac{1}{N_a} \sum_{n=1}^{N_a} \mathcal{H}\left[p_\theta(a_{t,n} \mid a_{t,<n}, \hat{o}_{t+1}'^v, o_{t-1:t}, L)\right]. \tag{5}$$

where $a_{t,n}$ denotes the n-th token in the action-token sequence $\mathbf{a}_t$, and $\mathcal{H}[p] = -\sum_a p(a) \log p(a)$ denotes Shannon entropy over the action-token distribution [39]. UniviewVLA identifies the most action-informative view as the one with the lowest action entropy:

$$v^* = \arg\min_{v \in \mathcal{V}_{\text{sel}}} H_v, \qquad \hat{o}_{t+1}'^* = \hat{o}_{t+1}'^{v^*}. \tag{6}$$

The final action-token sequence is predicted conditioned on $(o_{t-1:t}, L, \hat{o}_{t+1}'^*)$, leveraging compact multiview evidence and future scene evolution knowledge. To adapt to stage-dependent viewpoint changes, UniviewVLA dynamically updates the selected view every 30 timesteps, requiring no view-selection labels or additional training objective.

5 Experiments

5.1 Experimental Setup

We first evaluate UniviewVLA on standard occlusion-free simulation benchmarks, including LIBERO [40] and CALVIN ABCD→D [41]. To further evaluate occlusion-focused manipulation, we introduce six LIBERO-style occlusion tasks and two real-robot occlusion tasks on an ALOHA platform. For the simulation occlusion tasks, we collect demonstrations through 3D-SpaceMouse teleoperation. In both simulation and real-robot occlusion tasks, action-critical cues are occluded from the agent-view and wrist-view cameras, allowing us to evaluate whether generated auxiliary views improve occlusion-aware robot manipulation. At inference time, UniviewVLA uses only the agent-view and wrist-view as physical camera inputs. Implementation details are provided in Appendix A, and task details are provided in Appendix B.

5.2 Comparison with State-of-the-Art Methods

CALVIN. Table 1 reports CALVIN ABCD→D results. UniviewVLA achieves the highest average chain length among the compared methods. This demonstrates its effectiveness in long-horizon, multi-task manipulation, where the world model's generated future views provide additional visual evidence and predictive dynamics knowledge across extended action sequences.

Table 1: **Long-horizon manipulation performance on CALVIN ABCD→D.**

Method	1	2	3	4	5	Avg.
RT-1 [24]	0.844	0.617	0.438	0.323	0.227	2.45
Robo-Flamingo [28]	0.964	0.896	0.824	0.740	0.660	4.09
GR-1 [42]	0.949	0.896	0.844	0.789	0.731	4.21
MDT [19]	**0.986**	**0.958**	0.916	0.862	0.801	4.52
RoboVLMs [43]	0.967	0.930	0.899	0.865	0.826	4.49
Fast-dVLA [21]	0.984	0.952	0.922	0.870	0.812	4.54
MINT [44]	0.974	0.942	0.917	0.882	**0.861**	4.57
UniviewVLA	0.983	**0.958**	**0.928**	**0.893**	0.838	**4.60**

Table 2: **Manipulation success rates on the LIBERO benchmark.**

Method	Long	Goal	Spatial	Object	Avg.
OpenVLA [2]	53.7	79.2	84.9	88.4	76.6
π_0-FAST [32]	60.2	88.6	96.4	96.8	85.5
CoT-VLA [45]	69.0	87.6	87.5	91.6	83.9
VLA-0 [46]	87.6	**96.2**	**97.0**	97.8	94.7
GR00T N1 [47]	90.6	93.0	94.4	97.6	93.9
UniVLA [4]	91.4	93.2	96.0	**99.2**	95.0
OpenVLA-OFT [48]	90.7	96.2	96.2	98.3	95.4
UniviewVLA	**94.2**	93.4	96.4	**99.2**	**95.8**

LIBERO. Table 2 compares UniviewVLA with state-of-the-art methods on LIBERO. UniviewVLA achieves the best average success rate of 95.8% over the four standard suites, evaluated over 500 episodes per suite. It outperforms GR00T N1 and π_0-FAST by 1.9 and 10.3 percentage points, respectively, showing that generated multiview evidence from the world model also benefits standard occlusion-free manipulation beyond the targeted occlusion benchmarks.

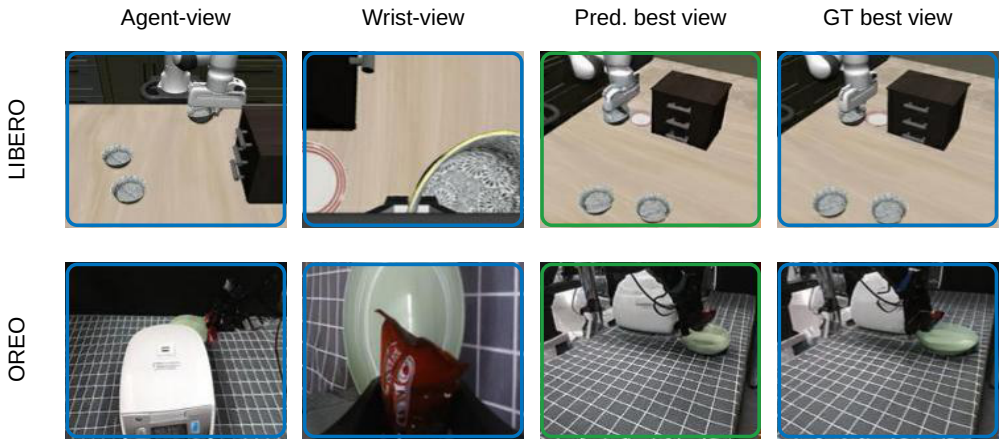

Figure 4: **Full future auxiliary-view token generation.** The first two blue boxed columns denote the standard physical camera views (agent-view and wrist-view). Green boxes denote the generated future auxiliary view selected from these inputs. The last blue boxed column shows the ground-truth physical camera observation from the same auxiliary viewpoint for comparison.

5.3 Analysis of Generated Views without Camera Coupling

This section evaluates whether world model generated multiview evidence can replace additional deployment-time cameras. Figure 4 visualizes full future auxiliary-view generation from standard agent-view and wrist-view inputs, where the world model predicts future dynamics based on stan-

Table 3: **Camera-configuration evaluation on standard occlusion-free simulation benchmarks.** Success rates are reported on LIBERO and average chain length on CALVIN ABCD→D. **A**: agent-view, **W**: wrist-view, **S**: physical side-view.

Method	Cameras	LIBERO Long	LIBERO Goal	LIBERO Spatial	LIBERO Object	LIBERO Avg.	CALVIN Avg.
Two-Camera Policy	A+W	90.2	93.4	92.8	95.6	93.0	4.42
Three-Camera Policy	A+W+S	**94.2**	**95.4**	93.6	97.4	95.2	4.54
UniviewVLA	A+W	**94.2**	93.4	**96.4**	**99.2**	**95.8**	**4.60**

dard observations. Full auxiliary-view generation requires $25 \times 25 = 625$ tokens per viewpoint and incurs 6–7 seconds of latency, while UniviewVLA reduces each generated view to $4 \times 4 = 16$ tokens through Motion-Informative Token Compression, lowering per-view latency to 0.2–0.3 seconds. Table 3 compares three view-input configurations. During both training and inference, the *Two-Camera Policy* uses only the agent-view and wrist-view, while the *Three-Camera Policy* additionally uses a physical camera at a manually selected viewpoint. Both camera baselines follow the two-stage training design with future-view prediction and action fine-tuning, controlling for world-evolution knowledge and focusing on the effect of compact generated auxiliary-view tokens.

As shown in Table 3, additional visual coverage improves standard benchmark performance. The physical third camera raises the LIBERO average from 93.0% to 95.2%, while UniviewVLA reaches 95.8% with only two physical cameras. On CALVIN, UniviewVLA improves over both the two-camera policy (4.42) and the three-camera policy (4.54). These results suggest that compact world model generated workspace views provide the performance gains of additional viewpoint coverage without the deployment cost of an extra camera, while entropy-based dynamic selection further avoids the limitation of relying on a single fixed side viewpoint.

5.4 Occlusion-Focused Evaluation and Dynamic View Selection

Table 4: **Occlusion-focused evaluation with camera and view-selection variants.** Success rates (%) are reported on six customized LIBERO-style occlusion tasks.

Method	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Avg.
Two-Camera Policy	24	68	34	26	84	4	40.0
Three-Camera Policy	52	74	68	**88**	**100**	16	66.3
UniviewVLA (Manual)	48	78	66	86	96	28	67.0
UniviewVLA (Entropy)	**58**	**88**	**72**	**88**	98	**36**	**73.3**

The occlusion tasks further evaluate whether world model generated views help when critical state information is hidden from the deployed agent-view and wrist-view cameras. As shown in Table 4, the two-camera policy achieves only 40.0% average success, and achieves only 4% on Task 6 where the switch is heavily occluded. This confirms that the default deployed views are insufficient when key task information is visually inaccessible. Adding a physical third camera improves the average to 66.3%, while UniviewVLA reaches 67.0% with a manually selected best fixed view, matching the fixed viewpoint of the Three-Camera Policy, and 73.3% with entropy-based dynamic selection.

These results demonstrate that compact world model generated views that predict future dynamics based on standard observations can outperform adding a physical camera by efficiently predicting additional viewpoint evidence without the cost of extra camera deployment. Entropy-based selection further adapts this evidence to stage-dependent occlusions.

5.5 Real-Robot Deployment

In addition, UniviewVLA is deployed on an ALOHA platform for real-world occlusion evaluation. The setup includes two right-arm-only manipulation tasks, each evaluated over 15 trials. To enable a practical multiview comparison with minimal hardware modification, the unused left-wrist camera

is detached and extended to provide an additional physical viewpoint, allowing comparison among two-camera, three-camera, and world model generated-view settings. As shown in Figure 5, the default agent-view misses action-critical object cues in both tasks. In *Oreo-to-Plate*, the robot grasps an Oreo and places it on a plate hidden behind a rice cooker. In *Occluded-Doll Move*, the robot grasps a doll partially blocked by a box and moves it into another container.

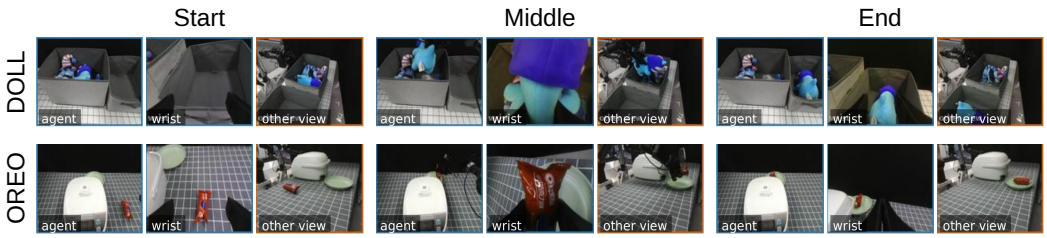

Figure 5: **Real-robot occlusion tasks.** The target plate or manipulated object is partially hidden from the default agent-view, requiring additional spatial evidence for reliable execution.

We compare three deployment configurations: two-camera, three-camera, and UniviewVLA. As shown in Figure 6, the two-camera policy succeeds in only 13.3% of Oreo-to-Plate trials and 20.0% of Occluded-Doll Move trials. Adding a physical third camera improves success to 40.0% and 53.3%, respectively, confirming the importance of additional visual coverage in real-world occlusion tasks. UniviewVLA achieves 53.3% and 46.7% with only the two default cameras, substantially outperforming the two-camera policy and approaching the three-camera setting without the hardware cost of an extra calibrated camera. These results demonstrate that the world model's compact multiview token prediction and action-entropy view selection transfer to real-world deployment, providing a favorable return on investment (ROI) between occlusion robustness and deployment cost.

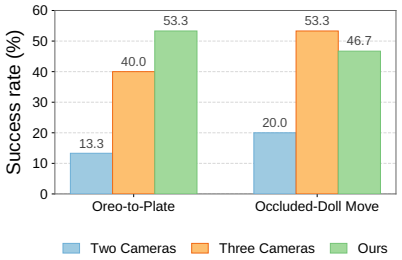

Figure 6: **Real-robot performance over 15 trials per task.**

6 Conclusion

We presented UniviewVLA, a unified multiview VLA framework with world-modeling capabilities that predicts multiview future scene evolution for action prediction from standard agent-view and wrist-view observations, decoupling auxiliary-view reasoning from deployment-time camera requirements. To make world model generated views efficient for closed-loop robot manipulation, UniviewVLA compresses each generated view from 625 to 16 tokens and selects the most action-informative view via action entropy. Across simulation and real-robot evaluations, UniviewVLA maintains excellent occlusion-free manipulation performance while improving occlusion-task performance with only two standard deployed cameras. In addition, this work contributes six occlusion-focused tasks and a complete multiview data collection, processing, and evaluation pipeline for robot learning. These results demonstrate that UniviewVLA provides a practical framework for improving occluded robot manipulation without adding deployment-time cameras by leveraging world model capabilities to predict future scene dynamics.

7 Limitations

UniviewVLA shares common constraints of generative robot policies. Auxiliary-view generation adds modest inference overhead, and performance depends on the coverage of multiview training data. Our evaluation focuses on tabletop and occlusion-focused tasks; extending to more diverse scenes, mobile viewpoints, and longer-horizon manipulation remains future work.

References

[1] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

[2] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi, Q. Vuong, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning*, pages 2679–2713. PMLR, 2025.

[3] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[4] Y. Wang, X. Li, W. Wang, J. Zhang, Y. Li, Y. Chen, X. Wang, and Z. Zhang. Unified vision-language-action model. *arXiv preprint arXiv:2506.19850*, 2025.

[5] Y. Fan, P. Ding, S. Bai, X. Tong, Y. Zhu, H. Lu, F. Dai, W. Zhao, Y. Liu, S. Huang, et al. Long-vla: Unleashing long-horizon capability of vision language action model for robot manipulation. *arXiv preprint arXiv:2508.19958*, 2025.

[6] C.-L. Cheang, G. Chen, Y. Jing, T. Kong, H. Li, Y. Li, Y. Liu, H. Wu, J. Xu, Y. Yang, et al. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

[7] P. Li, Y. Chen, Y. Xu, J. Yang, X. Wu, J. Guo, N. Sun, L. Qian, X. Li, X. Xiao, et al. Multi-view video diffusion policy: A 3d spatio-temporal-aware video action model. *arXiv preprint arXiv:2604.03181*, 2026.

[8] Y. Xie, Y. Wang, S. Zhao, C.-E. Wu, M. Tomizuka, J. Xie, and H.-S. Fang. Multi-camera view scaling for data-efficient robot imitation learning. *arXiv preprint arXiv:2604.00557*, 2026.

[9] 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. In *Conference on Robot Learning*, pages 3094–3114. PMLR, 2025.

[10] A. Goyal, V. Blukis, J. Xu, Y. Guo, Y.-W. Chao, and D. Fox. Rvt-2: Learning precise manipulation from few demonstrations. *arXiv preprint arXiv:2406.08545*, 2024.

[11] A. Goyal, J. Xu, Y. Guo, V. Blukis, Y.-W. Chao, and D. Fox. Rvt: Robotic view transformer for 3d object manipulation. In *Conference on Robot Learning*, pages 694–710. PMLR, 2023.

[12] M. Shridhar, L. Manuelli, and D. Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pages 785–799. PMLR, 2023.

[13] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

[14] P. Wu, A. Escontrela, D. Hafner, P. Abbeel, and K. Goldberg. Daydreamer: World models for physical robot learning. In *Conference on robot learning*, pages 2226–2240. PMLR, 2023.

[15] F. Yang, D. Di, L. Tang, X. Zhang, L. Fan, H. Li, W. Chen, T. Su, and B. Ma. Chain of world: World model thinking in latent motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6675–6684, 2026.

[16] B. Cai, Q. Liang, J. Li, S. Weng, Z. Zhang, T. Lin, X. Chen, W. Zhang, J. Mao, W. Xu, et al. Beyond viewpoint generalization: What multi-view demonstrations offer and how to synthesize them for robot manipulation? *arXiv preprint arXiv:2603.26757*, 2026.

[17] S. Tian, B. Wulfe, K. Sargent, K. Liu, S. Zakharov, V. Guizilini, and J. Wu. View-invariant policy learning via zero-shot novel view synthesis. *arXiv preprint arXiv:2409.03685*, 2024.

[18] S. Wang, H. Dong, J. Tian, J. Li, Z. Yang, T. Cao, A. Chen, S. Wu, L. Wang, and S. Zhou. Efficient camera pose augmentation for view generalization in robotic policy learning. *arXiv preprint arXiv:2603.29192*, 2026.

[19] 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.

[20] Z. Huang, C. Feng, F. Yan, B. Xiao, Z. Jie, Y. Zhong, X. Liang, and L. Ma. Robotron-drive: All-in-one large multimodal model for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8011–8021, 2025.

[21] W. Song, J. Chen, S. Chen, J. Wang, P. Ding, H. Zhao, Y. Qin, X. Zheng, D. Wang, Y. Wang, et al. Fast-dvla: Accelerating discrete diffusion vla to real-time performance. *arXiv preprint arXiv:2603.25661*, 2026.

[22] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*.

[23] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[24] 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. *Robotics: Science and Systems XIX*, 2023.

[25] Z. Huang, T. Tang, S. Chen, S. Lin, Z. Jie, L. Ma, G. Wang, and X. Liang. Making large language models better planners with reasoning-decision alignment. In *European Conference on Computer Vision*, pages 73–90. Springer, 2024.

[26] 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.

[27] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, et al. Palm-e: an embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, pages 8469–8488, 2023.

[28] X. Li, M. Liu, H. Zhang, C. Yu, J. Xu, H. Wu, C. Cheang, Y. Jing, W. Zhang, H. Liu, et al. Vision-language foundation models as effective robot imitators. In *International Conference on Learning Representations*, volume 2024, pages 26703–26721, 2024.

[29] O. Mees, D. Ghosh, K. Pertsch, K. Black, H. R. Walke, S. Dasari, J. Hejna, T. Kreiman, C. Xu, J. Luo, et al. Octo: An open-source generalist robot policy. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024.

[30] K. Bousmalis, G. Vezzani, D. Rao, C. Devin, A. X. Lee, M. Bauzá, T. Davchev, Y. Zhou, A. Gupta, A. Raju, et al. Robocat: A self-improving generalist agent for robotic manipulation. *arXiv preprint arXiv:2306.11706*, 2023.

[31] 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)*, pages 6892–6903. IEEE, 2024.

[32] 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.

[33] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020.

[34] A. Zeng, P. Florence, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin, D. Duong, V. Sindhwani, et al. Transporter networks: Rearranging the visual world for robotic manipulation. In *Conference on Robot Learning*, pages 726–747. PMLR, 2021.

[35] M. Shridhar, L. Manuelli, and D. Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on robot learning*, pages 894–906. PMLR, 2022.

[36] 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.

[37] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki. 3d diffuser actor: Policy diffusion with 3d scene representations. In *Conference on Robot Learning*, pages 1949–1974. PMLR, 2025.

[38] S. Yang, W. Yu, J. Zeng, J. Lv, K. Ren, C. Lu, D. Lin, and J. Pang. Novel demonstration generation with gaussian splatting enables robust one-shot manipulation. *arXiv preprint arXiv:2504.13175*, 2025.

[39] C. E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[40] 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*, 36:44776–44791, 2023.

[41] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

[42] H. Wu, Y. Jing, C. Cheang, G. Chen, J. Xu, X. Li, M. Liu, H. Li, and T. Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *International Conference on Learning Representations*, volume 2024, pages 10641–10662, 2024.

[43] H. Liu, X. Li, P. Li, M. Liu, D. Wang, J. Liu, B. Kang, X. Ma, T. Kong, and H. Zhang. Towards generalist robot policies: What matters in building vision-language-action models. 2025.

[44] R. Huang, C. Zeng, W. Tang, J. Cai, C. Lu, and P. Cai. Mimic intent, not just trajectories. *arXiv preprint arXiv:2602.08602*, 2026.

[45] Q. Zhao, Y. Lu, M. J. Kim, Z. Fu, Z. Zhang, Y. Wu, Z. Li, Q. Ma, S. Han, C. Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1702–1713, 2025.

[46] A. Goyal, H. Hadfield, X. Yang, V. Blukis, and F. Ramos. Vla-0: Building state-of-the-art vlas with zero modification. *arXiv preprint arXiv:2510.13054*, 2025.

[47] 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.

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

A Implementation details

Stage 1: multiview world-model post-training. All UniviewVLA experiments use models trained on 8 NVIDIA H-series GPUs with bf16 and DeepSpeed ZeRO-3. We use the same Emu3-based autoregressive backbone as UniVLA, together with a VQ visual tokenizer for full-view prediction. In Stage 1, the world model is trained to autoregressively predict next-frame VQ tokens of auxiliary workspace views from the agent view, wrist view, and language instruction, learning to predict multiview future scene evolution from standard observations. Each full view is represented as 25×25 VQ tokens. Training runs for 30K steps with global batch size 8 and a cosine learning rate schedule from 8×10^{-5} to 5×10^{-6}.

Stage 2: action fine-tuning. Stage-2 fine-tuning uses a FAST action tokenizer for action prediction and runs for 24K steps on LIBERO and CALVIN and 30K steps on real-robot tasks. To prevent the model from collapsing different auxiliary views into the same action-conditioned supervision and to enable action-entropy view selection at inference, each demonstration is paired with a view-specific prompt prefix of the form "predict {view} view and {instruction}". The same prompt format is used for training and entropy-based inference-time view selection. For example, the instruction "open the bottom drawer of the cabinet and put the bowl in it" is rewritten as "predict 60deg view and open the bottom drawer of the cabinet and put the bowl in it".

Inference: entropy-based dynamic view selection. UniviewVLA uses the same view-specific prompt format as in Stage 2, selects among the candidate auxiliary views by minimizing action entropy, and re-selects every 30 policy steps, enabling adaptive view selection during execution.

B Multiview Tasks and Data Details

We provide additional details for the multiview data collection and customized occlusion-focused tasks used in Section 5.4. For standard simulation benchmarks, we obtain multiview supervision by replaying recorded demonstrations with additional workspace cameras in the simulator. LIBERO uses $60°$, $120°$, $240°$, $300°$, and overhead views, as shown in Fig. 7. Since the rear tabletop side in CALVIN is mostly redundant or weakly informative, we use more widely spaced views at $30°$, $216°$, $288°$, and overhead, as shown in Fig. 8.

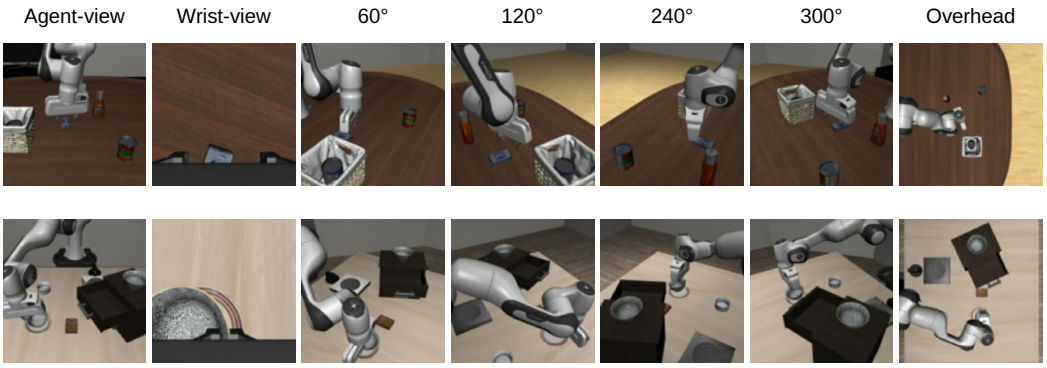

Figure 7: **LIBERO multiview.**

To further evaluate the importance of multiview information, we construct six customized occlusion-focused tasks that hide action-critical cues from the standard viewpoints. Specifically, we follow the LIBERO BDDL format to design occluded manipulation scenes, as shown in Fig. 9, and collect multiview demonstrations with a hardware-in-the-loop teleoperation setup based on a 3D-SpaceMouse. Each task contains 60 demonstrations. Table 5 lists the corresponding language instructions. Task 1 requires the robot to open a drawer that is partially hidden by the occluder and place the bowl inside it. Task 2 requires placing a bowl onto a plate occluded by the drawer. Task 3 requires grasping a

wine bottle and placing it on a wine rack hidden by a box. Task 4 requires placing an occluded bowl into the drawer and closing it. Task 5 requires placing a mug onto a plate partially occluded by a microwave door. Task 6 requires turning off a switch occluded by the moka pot and then placing the moka pot on top of the cabinet.

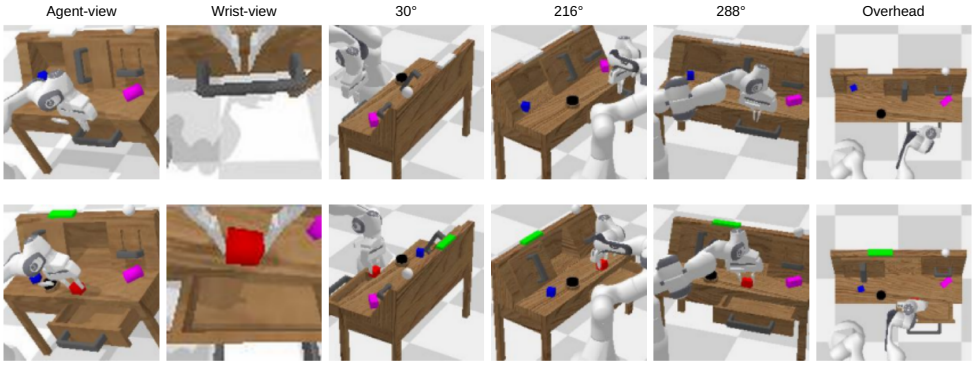

Figure 8: **CALVIN multiview.**

Table 5: **Language instructions for six customized occlusion tasks.**

ID	Task	Language Instruction
1	Scene1 Bowl	open the bottom drawer of the cabinet and put the bowl in it
2	Scene2 Bowl	put the black bowl on the left plate
3	Scene4 Wine	pick up the wine bottle at the back and put it on the wine rack
4	Scene4 Drawer	put the black bowl at the left in the bottom drawer of the cabinet and close it
5	Scene7 Mug	put the yellow and white mug on the plate
6	Scene8 Moka	turn off the stove and put the moka pot on top of the cabinet

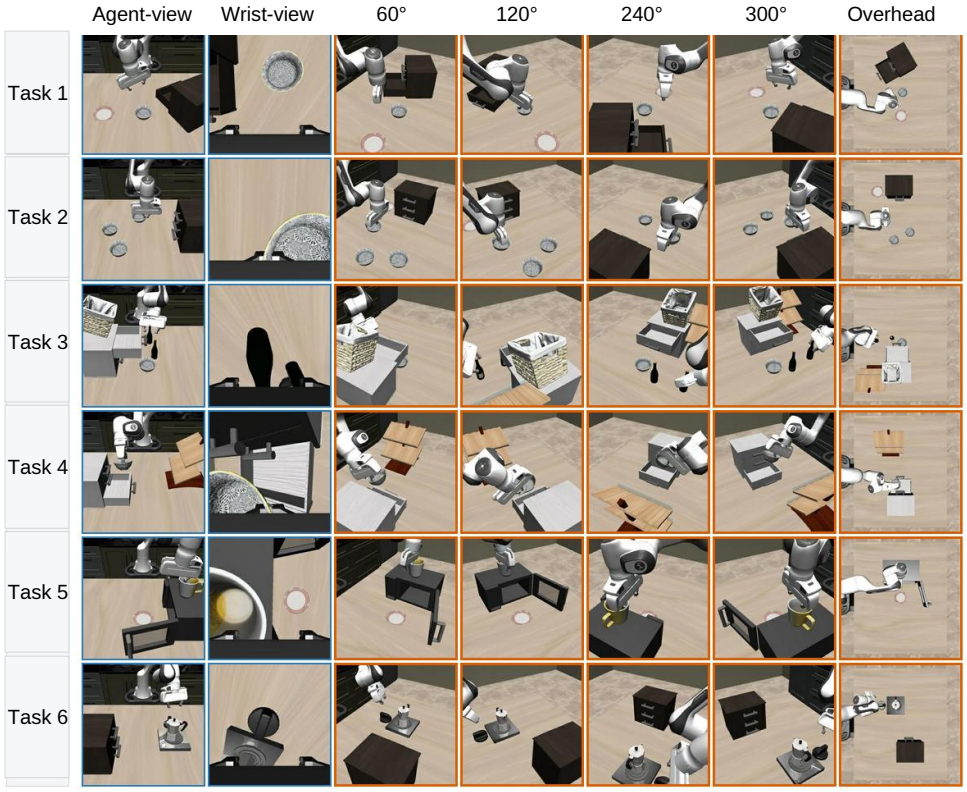

Figure 9: **Six customized occlusion-focused tasks.** Each task hides action-critical state cues from the default agent-view camera while preserving the same two deployed physical observations.

For real-robot evaluation, to avoid costly hardware and data-interface modifications, we design right-arm-only occlusion tasks on a Mobile ALOHA dual-arm platform. Accordingly, the unused left-wrist camera is repurposed as an additional side-view camera by placing it at a task-specific viewpoint using an extension cable. We collect 100 demonstrations for each of the Oreo-to-Plate and Occluded-Doll Move tasks. The real-robot multiview setup is shown in Fig. 10.

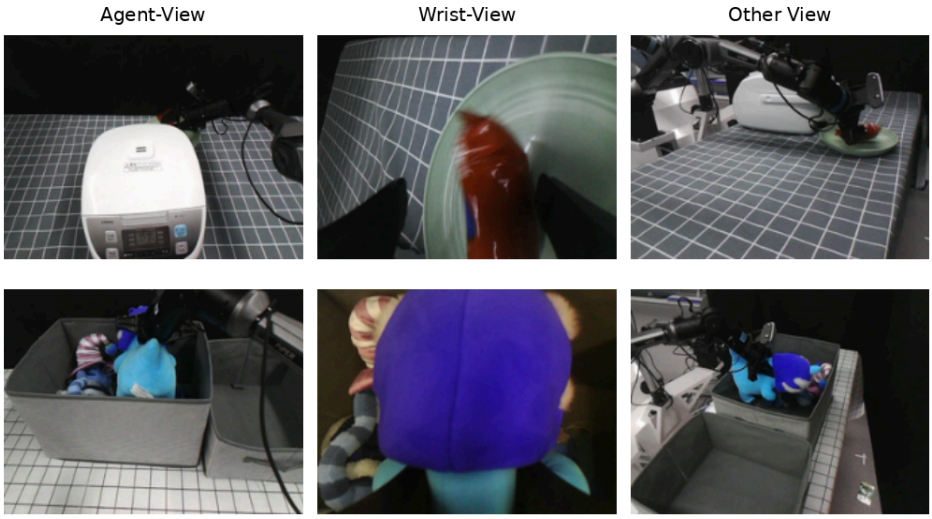

Figure 10: **Real-robot multiview.**