Beyond Dense Futures: World Models as Structured Planners for Robotic Manipulation

Minghao Jin[1*], Mozheng Liao[1*], Mingfei Han[1,2*], Zhihui Li[1], and Xiaojun Chang[1,2(✉)]

[1] University of Science and Technology of China, Hefei, China
[2] Division of CMS, MBZUAI, Abu Dhabi, United Arab Emirates

cxj273@gmail.com

https://wm-planner.github.io/structvla/

*Equal contribution.

Abstract. Recent world-model-based Vision-Language-Action (VLA) architectures have improved robotic manipulation through predictive visual foresight. However, dense future prediction introduces visual redundancy and accumulates errors, causing long-horizon plan drift. Meanwhile, recent sparse methods typically represent visual foresight using high-level semantic subtasks or implicit latent states. These representations often lack explicit kinematic grounding, weakening the alignment between planning and low-level execution. To address this, we propose StructVLA, which reformulates a generative world model into an explicit structured planner for reliable control. Instead of dense rollouts or semantic goals, StructVLA predicts sparse, physically meaningful structured frames. Derived from intrinsic kinematic cues (*e.g.*, gripper transitions and kinematic turning points), these frames capture spatiotemporal milestones closely aligned with task progress. We implement this approach through a two-stage training paradigm with a unified discrete token vocabulary: the world model is first trained to predict structured frames and subsequently optimized to map the structured foresight into low-level actions. This approach provides clear physical guidance and bridges visual planning and motion control. In our experiments, StructVLA achieves strong average success rates of 75.0% on SimplerEnv-WidowX and 94.8% on LIBERO. Real-world deployments further demonstrate reliable task completion and robust generalization across both basic pick-and-place and complex long-horizon tasks.

Keywords: Vision language action model · World model · Robotic manipulation

1 Introduction

Robotic manipulation typically requires coupling coherent task planning with reliable physical control. Vision-Language-Action (VLA) models [2–4, 6, 9, 16, 19, 20, 22, 35–37, 42, 47, 50, 52] address this challenge by grounding natural language

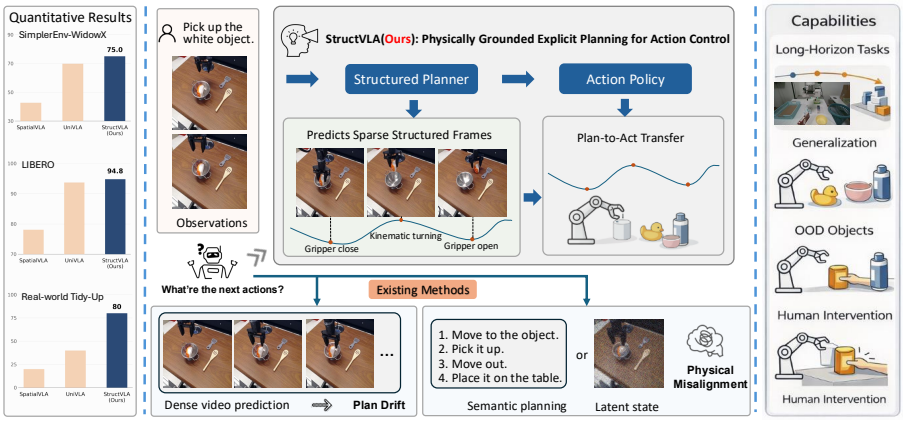

Fig. 1: Illustration of StructVLA (Structured Planner for Vision-Language-Action). Existing generative VLA methods either predict dense future observations *(bottom left)*, which accumulate errors and cause long-horizon plan drift, or rely on semantic or latent planning *(bottom right)* that lacks explicit geometric grounding and leads to physical misalignment. Our StructVLA *(top)* learns a physically grounded structured planner by training a world model to predict sparse structured frames, and transfers this representation to action control. This design enables stable long-horizon manipulation and demonstrates strong generalization and robustness across simulation and real-world benchmarks.

instructions in visual observations to directly output low-level actions. Recently, the integration of generative world models into VLA architectures [10,39,42] has emerged as a powerful paradigm for long-horizon manipulation. By internalizing environmental dynamics, these generative architectures provide predictive visual foresight, offering spatiotemporal context that enables the policy to anticipate future states beyond instantaneous perception.

Despite these advantages, effectively representing this visual foresight remains a fundamental challenge. Many approaches represent foresight as dense or step-wise future predictions (pixel or latent) [5,19,36,42,47], which can amplify long-horizon errors when used as direct guidance. Moreover, dense representations contain massive visual redundancy, often obscuring essential task events within high-frequency, task-irrelevant noise. In addition, generating continuous temporal rollouts tends to accumulate prediction errors. Over extended horizons, local visual artifacts or physical inconsistencies compound and can result in plan drift.

To mitigate the visual redundancy and error accumulation from dense rollouts, recent methods have explored sparse visual foresight [16,28,48]. While these paradigms effectively compress the prediction horizon, existing sparse frameworks also introduce structural limitations. Some approaches define subgoals in semantic space, where high-level abstractions may not specify the precise geometric grounding required for manipulation. Other approaches rely on implicit

latent representations [45,47]; while these methods facilitate policy learning, they lack explicit, verifiable geometric targets for closed-loop control. Without clear physical grounding, both representations struggle to provide actionable guidance for low-level kinematic execution, motivating a planning representation that is both physically grounded and control-effective.

To address these fundamental limitations, we propose **StructVLA**, a framework that reformulates the generative world model into an explicit, physically grounded structured planner. StructVLA predicts sparse structured frames as mid-level visual subgoals that mark essential manipulation milestones. Rather than using semantic subtask boundaries to define subgoals, we derive these frames from intrinsic kinematic cues, such as gripper state transitions and kinematic turning points. Gripper transitions capture contact-state changes, while kinematic turning points reflect meaningful shifts in motion; together they provide progress anchors that are mechanically meaningful across tasks. As a result, the generated visual foresight provides actionable spatiotemporal cues and is tightly coupled to executable interaction phases.

As illustrated in Fig. 2, StructVLA employs a two-stage training paradigm. In the first stage, the model autoregressively predicts explicit, discrete structured frames, which serve as explicit geometric targets and help reduce long-horizon drift. In the second stage, the model is optimized to generate low-level action tokens, leveraging the structured-planning capability learned in Stage 1. By representing vision, language, and action in a shared discrete vocabulary, StructVLA strengthens the coupling between visual planning and motion control.

Our main contributions are summarized as follows:

- **Physics-grounded structured planning targets.** We introduce an automated data curation pipeline that extracts structured frames from robot-centric kinematic cues, providing sparse, mechanically meaningful anchors without semantic subtask labeling.
- **Unified StructVLA framework.** We present StructVLA, a unified framework that reformulates a world model into an explicit structured planner for control. It is trained in two stages: first autoregressively predicting structured frames as mid-level visual subgoals, then generating low-level action tokens by leveraging the structured-planning capability learned in the first stage, within a shared discrete token space.
- **Strong empirical results.** StructVLA achieves average success rates of 75.0% on SimplerEnv-WidowX [23] and 94.8% on LIBERO [26], and demonstrates reliable task completion and robust generalization in real-world evaluations on both basic pick-and-place and complex long-horizon tasks.

2 Related Work

Vision-Language-Action Models. Recent vision-language-action (VLA) models [2–4,6,8,9,16,19,20,22,35–37,42,47,50,52] have demonstrated strong performance for robotic manipulation. A common line of work builds on large vision-language models (VLMs) and learns to map multimodal representations to robot

actions. For instance, OpenVLA [20] directly predicts action tokens from VLM, while π_0 [2] and CogACT [22] augment the backbone with a flow-matching or diffusion-based action expert to improve continuous control. Villa-X [6] further introduces an intermediate latent-action expert before generating final actions. Recent extensions also incorporate 3D cues to strengthen spatial grounding, such as SpatialVLA [35] and GeoVLA [37]. Another line of work explores generative action modeling and video-diffusion-based control. Diffusion Policy [7] formulates action generation directly as a diffusion process, while VPP [15] exploits representations from video diffusion models for control. VIDAR [9] integrates action reasoning into a video diffusion model, and VideoVLA [36] further studies generalization in video-diffusion-based VLA frameworks. FlowVLA [50] uses optical flow as a motion-foresight cue for action prediction. DreamVLA [47] uses world-model forecasting to guide long-horizon action generation. In contrast, our planner predicts kinematically grounded structured frames as explicit progress anchors, making foresight directly actionable for long-horizon manipulation.

Generative models for robotic manipulation. Many studies have applied world models [11, 12, 21, 46] to robotic manipulation across both simulation and real-robot settings. DayDreamer [44] introduces the Dreamer algorithm into real-world robot learning, improving sample efficiency. CASCADE [46] expands world-model training coverage via collaborative exploration to improve generalization. SWIM [29] trains world models on human demonstration videos and uses them to guide manipulation. UWM [51] further unifies video diffusion and action diffusion for joint modeling of dynamics and policy learning. UniVLA [42] unifies vision, language, and action as an autoregressive discrete sequence and incorporates world-model post-training to capture temporal and causal structure for downstream policy learning. For joint forecasting, VideoVLA [36] employs a multimodal DiT [33] to jointly forecast an action sequence and future visual outcomes. Large Video Planner [5] leverages foundation-scale video pretraining to generate dense video plans, which are then post-processed into executable robot actions. Similarly, Cosmos Policy [19] adapts a pretrained video diffusion model via latent frame injection, and can optionally perform value-guided best-of-N planning to evaluate candidate action trajectories, incurring higher test-time sampling cost. In contrast, we explore world models as explicit sparse structured planners, producing physically grounded progress anchors that are directly actionable for closed-loop control.

Task planning in robot learning. Task planning is crucial for long-horizon manipulation. Many studies have explored decomposing language instructions into sub-tasks, such as utilizing code-generation models [24] to translate natural language into executable policies. Text2Motion [25] converts both long and short natural language instructions into symbolic task goals and feasible action sequences, while ThinkAct [17] leverages VLM reasoning for trajectory-level planning. Complementary to language-centric planning, other approaches derive planning signals from visual dynamics. Track2Act [1] predicts pixel-level trajectories from internet videos to assist manipulation, and LDP [45] models latent actions between adjacent frames for trajectory planning. UniVLA (Bu et al.) [4]

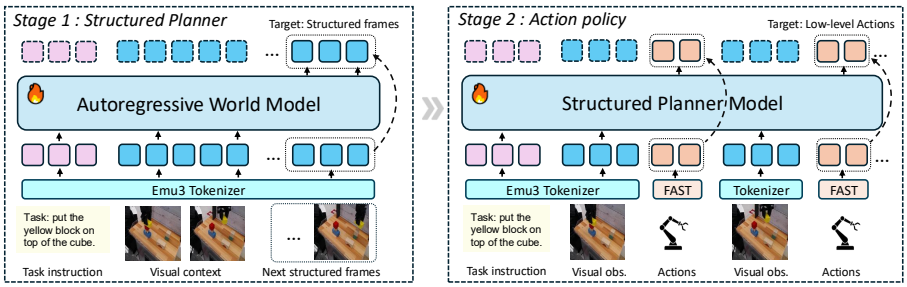

Fig. 2: Overview of StructVLA. StructVLA is trained in two stages. **Stage 1 (Structured Planner):** an autoregressive world model predicts sparse structured frames that capture physically grounded progress anchors, conditioned on the instruction and visual context. **Stage 2 (Action Policy):** we fine-tune the structured planner for control by conditioning on the instruction together with interleaved observation and actions, transferring structured planning into low-level control.

extracts task-centric latent actions from unannotated videos to enable cross-embodiment policy learning. More recently, generative foresight has been used to instantiate planners in different forms. ATM [43] / NoTVLA [18] sparsify the trajectory/action side via point tracks or end-effector trajectory compression, while StructVLA uses progress-aware kinematic milestones for sparse world-model planning and planner-to-policy transfer, showing gains beyond simple trajectory sparsification. ForeAct [48] and VISTA [28] employ generative models as high-level planners that provide subgoal guidance to separate low-level VLAs, while BagelVLA [16], mimic-video [32] and LatentPilot [13] incorporate planning signals through implicit latent foresight. In particular, BagelVLA [16] and ForeAct [48] adopt subtask-anchored foresight, where subgoals are specified or sampled within semantic subtask segments. In contrast, StructVLA derives structured subgoals from kinematic interaction cues and reformulates the world model as a kinematically grounded planner, providing control-effective visual anchors for action generation.

3 Method

Fig. 2 summarizes the overall design of StructVLA. We reformulate a single generative world model as a structured planner by introducing sparse structured frames as physically grounded visual subgoals. Sec. 3.1 describes an automated data curation pipeline for extracting these structured frames. We train StructVLA with a two-stage paradigm: Sec. 3.2 details Stage 1 for structured frame prediction, and Sec. 3.3 presents Stage 2 for transferring the structured foresight to action policy learning.

3.1 Physics-Grounded Data Curation

At the core of our structured planner are structured frames, which serve as physically grounded visual subgoals. During offline data curation, an automated pipeline extracts these frames from raw robotic trajectories to exclusively supervise the generative world model. Instead of representing foresight as dense continuous rollouts, structured frames provide compact progress anchors that connect task intent to executable motion phases. By deriving them from robot-centric kinematic cues, our curation produces subgoals that are consistently aligned with interaction transitions in manipulation.

Formally, consider a robotic manipulation trajectory $\mathcal{T} = \big(l, \{(o_t, a_t)\}_{t=0}^{t_n}\big)$, where l denotes the language instruction, o_t is the visual observation, and a_t represents the executed action at timestep t. Within a trajectory, the most intuitive anchors are explicit milestones (*e.g.*, object grasped or placed), which define clear subtask boundaries. However, relying only on these sparse events omits critical task-progression details. In practice, many informative intermediate states are implicitly reflected in the motion trajectory, such as the kinematic turning point where the gripper aligns above a target before shifting direction to descend. To capture such physically meaningful states without manual annotation, we introduce an automated extraction procedure based on kinematic cues. Our offline pipeline first proposes candidate frames from robot-centric kinematic cues, and then applies filtering to remove redundancy and refine the candidates.

Kinematic Structured Frame Extraction. To ensure robustness across diverse manipulation speeds and tasks, we compute adaptive kinematic thresholds for each episode from the distribution of end-effector motion magnitudes $\|v_t\|$, which is derived from the actions a_t. Combined with Exponential Moving Average (EMA) smoothing to filter out high-frequency control noise, we extract structured frames using two primary kinematic cues:

Gripper State Transitions: We extract structured frames around gripper state changes (open or close), which often mark contact state transitions. Since raw gripper signals can be noisy and the physical actuation may lag behind the signal flip in practice, we apply a stabilization window δ_{settle} after each flip time t_c. We then select the last quasi static observation o_{t^*} immediately before the robot resumes a major motion, identified when the smoothed speed exceeds an adaptive threshold τ_{high}. Formally, for each flip time t_c, we define

$$S_{\text{grip}} = \Big\{ o_{t^*} \;\Big|\; t^* = \min\big\{ t > t_c + \delta_{\text{settle}} \mid \text{EMA}(\|v_t\|) > \tau_{\text{high}} \big\} - 1 \Big\} \tag{1}$$

Kinematic Turning Points: Even with a fixed gripper state, manipulation often involves brief slowdowns during motion regime transitions. For example, the end-effector may complete lateral alignment above the target and stabilize before initiating the downward approach. We identify such moments by applying a sliding window of length W and selecting frames whose smoothed speed remains below an adaptive threshold τ_{low} throughout the window ending at timestep t:

$$S_{\text{turn}} = \big\{ o_t \;\big|\; \max_{k \in [t-W+1, t]} \text{EMA}(\|v_k\|) \le \tau_{\text{low}} \big\} \tag{2}$$

Temporal Coverage: To avoid extracting redundant frames from a prolonged low-speed segment, we enforce a minimum temporal gap between selected frames. For unusually long segments where distinct kinematic events are sparse, we apply a gap filling step to ensure adequate subgoal coverage. Specifically, we sample additional observations S_{fill} by selecting frames with the smallest L_2 change in consecutive action commands, $\Delta a_t = \|a_t - a_{t-1}\|_2$, which serves as a proxy for low-motion, stable states within the segment. The union of these sets forms our initial candidate structured frame set $\mathcal{S} = S_{\text{grip}} \cup S_{\text{turn}} \cup S_{\text{fill}}$.

Offline Trajectory Filtering. While kinematic cues reliably detect interaction transitions, they provide limited task-level context and may occasionally propose false positives under hardware perturbations. To refine the final structured frames, we use a VLM, *i.e.*, GPT-5.2 [30], strictly for offline refinement. Specifically, we provide the global instruction l together with the candidate observations $\mathcal{S}$, and use a fixed prompt (detailed in the Appendix) with filtering rules to remove visually redundant or task-irrelevant frames. This verification refines the preliminary structured frames obtained from kinematic cues, so that the final sequence better reflects coherent and executable milestones. Crucially, the filtering is applied only offline during dataset construction. We train StructVLA solely on the curated data. During inference, we deploy the action policy with no additional curation or VLM assistance.

3.2 Structured Planner (Stage 1: Visual Foresight)

To achieve this structured planning, we train the generative world model to predict the physically grounded structured frames $s \in \mathcal{S}$ identified during data curation. Formally, given a language instruction l and a visual context window $\mathcal{O}_t = \{o_{t-H+1}, \ldots, o_t\}$, the model learns the conditional distribution $P_\theta(s \mid l, \mathcal{O}_t)$ of a future structured frame s.

We employ the Emu3 [41] tokenizer with Vector Quantization (VQ) to discretize the language instructions and image frames into a shared vocabulary. Specifically, the language instruction l is tokenized into T_l, while the visual context window $\mathcal{O}_t$ is transformed into a sequence of tokens $T_{\mathcal{O}_t} = \{T_{o_{t-H+1}}, \ldots, T_{o_t}\}$, where each T_{o_i} represents the discretized features of observation o_i. Similarly, the target structured frame s is represented by a discrete token sequence $T_s = \{z_1, z_2, \ldots, z_K\}$.

Following the standard autoregressive generation paradigm, the model is trained to minimize the cross-entropy loss over the predicted visual tokens T_s, conditioned on the tokenized instruction and historical observations:

$$\mathcal{L}_{\text{planner}} = -\sum_{k=1}^{K} \log P_\theta(z_k \mid z_{<k}, T_l, T_{\mathcal{O}_t}) \tag{3}$$

Through sequential token prediction, the model learns to forecast kinematically grounded structured frames, capturing key transitions in manipulation and developing an implicit awareness of task progress in its visual foresight. This provides stable visual guidance for subsequent action control.

Table 1: Results on SimplerEnv [23]. Each model is evaluated for task success rate across four manipulation tasks. StructVLA yields the strongest average performance.

Model	Spoon	Carrot	Stack	Eggplant	Avg.
RT-1-X [31]	0.0%	4.2%	0.0%	0.0%	1.1%
Octo-Base [38]	12.5%	31.9%	0.0%	43.1%	16.0%
Octo-Small [38]	47.2%	9.7%	4.2%	56.9%	29.5%
OpenVLA [20]	0.0%	0.0%	0.0%	4.1%	1.0%
RoboVLMs [27]	45.8%	20.8%	4.2%	79.2%	37.5%
SpatialVLA [35]	16.7%	25.0%	29.2%	**100%**	42.7%
VideoVLA [36]	75.0%	20.8%	**45.8%**	70.8%	53.1%
mimic-video [32]	41.7%	54.2%	29.2%	**100%**	56.3%
UniVLA [42]	83.3%	66.7%	33.3%	95.8%	69.8%
StructVLA (Ours)	**87.5%**	**75.0%**	**45.8%**	91.7%	**75.0%**

Table 2: Results on LIBERO [26]. Comparison of task success rates across four suites. StructVLA outperforms prior baselines, especially on the LIBERO-Long suite. * denotes reproduced results.

Method	SPATIAL	OBJECT	GOAL	LONG	Avg.
DP [7]	78.3%	92.5%	68.3%	50.5%	72.4%
Octo [38]	78.9%	85.7%	84.6%	51.1%	75.1%
OpenVLA [20]	84.9%	88.4%	79.2%	53.7%	76.5%
SpatialVLA [35]	88.2%	89.9%	78.6%	55.5%	78.1%
CoT-VLA [49]	87.5%	91.6%	87.6%	69.0%	81.1%
π_0-FAST [34]	96.4%	96.8%	88.6%	60.2%	85.5%
FlowVLA [50]	93.2%	95.0%	91.6%	72.6%	88.1%
DreamVLA [47]	**97.5%**	94.0%	89.5%	89.5%	92.6%
UniVLA* [42]	96.4%	**98.0%**	90.8%	89.6%	93.7%
StructVLA (Ours)	95.4%	**98.0%**	**93.4%**	**92.2%**	**94.8%**

3.3 Action Policy (Stage 2: Plan-to-Act Transfer)

In Stage 2, the pretrained structured planner is directly formulated as an executable action policy by shifting the optimization objective to predicting action tokens. We discretize continuous actions into discrete tokens using the FAST tokenizer [34]. Crucially, these action tokens are integrated into a unified vocabulary shared with the visual and language modalities, enabling the model to directly output control signals through standard autoregressive generation.

During training, the conditioning context is explicitly constructed by concatenating the tokenized global language instruction T_l with interleaved visual and action tokens. Specifically, for a given timestep t and a context window of length H, this multimodal history sequence is defined as $\mathcal{H}_t = \{T_{o_{t-H+1}}, T_{a_{t-H+1}}, \ldots, T_{o_t}\}$, where T_{o_i} and T_{a_i} represent the discretized tokens of the observation and action at timestep i, respectively. Given this explicitly interleaved sequence, the model is optimized to autoregressively predict the sequence of discretized action tokens $T_{a_t} = \{a_1, a_2, \ldots, a_M\}$ required to execute the current subtask.

Following the standard next-token prediction paradigm, we restrict the supervision signal strictly to the action domain. The cross-entropy loss is computed exclusively over these sequentially generated action tokens:

$$\mathcal{L}_{\text{act}} = -\sum_{m=1}^{M} \log P_\theta(a_m \mid a_{<m}, T_l, \mathcal{H}_t) \tag{4}$$

By dynamically shifting the prediction target within a unified vocabulary, this formulation allows the generative structured planner to directly translate its internalized visual foresight into reliable, low-level action control.

4 Experiments

In this section, we comprehensively evaluate StructVLA across simulation benchmarks and real-world deployments. Sec. 4.1 introduces the simulation and real-

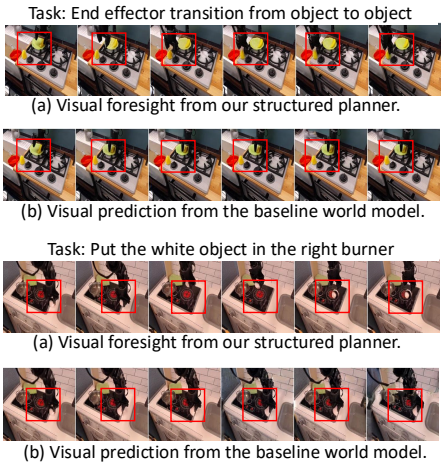

Fig. 3: Qualitative comparison of visual predictions. Our structured planner generates coherent, long-horizon visual foresight, supporting robust planning, whereas the baseline world model struggles with long-range comprehension, producing short-horizon predictions with degraded image quality. Red boxes highlight key differences.

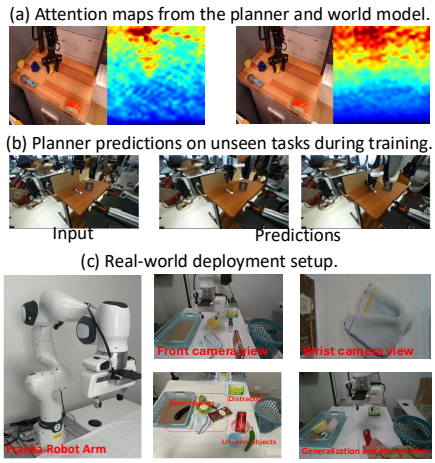

Fig. 4: More visualizations. (a) **Attention maps:** The planner (left) localizes task-critical interactions (*e.g.*, gripper–object contact), while the baseline (right) is diffuse. (b) **OOD predictions:** StructVLA retains world-model priors, enabling zero-shot visual planning on unseen scenes and tasks. (c) **Real-world deployment setup.**

world deployment settings, together with the curated datasets used for training. Sec. 4.2 details the model architecture and training configurations. Sec. 4.3 presents comparative results against representative baselines. Finally, Sec. 4.4 presents in-depth analyses and extended studies of the overall VLA pipeline.

4.1 Experimental Setup

SimplerEnv benchmark. SimplerEnv [23] evaluates transfer and generalization of policies trained on real-world robot trajectories under controlled distribution shifts (*e.g.*, lighting, texture, and camera pose) across WidowX and Google Robot platforms. We mainly report results on WidowX, which is commonly used. We train StructVLA on the standard BridgeData V2 [40] dataset.

LIBERO benchmark. The LIBERO benchmark [26] evaluates multi-task manipulation and long-horizon consistency within a simulated Franka Emika Panda robot arm. There are four task suites including LIBERO-Spatial, LIBERO-Object, LIBERO-Goal, and LIBERO-Long (LIBERO-10). Specifically, each suite comprises 10 distinct tasks with 50 expert demonstrations per task, and we jointly train a single model across the combined datasets of all suites.

Real-world evaluation. As shown in Fig. 4 (c), we deploy StructVLA on a physical Franka arm to evaluate robustness under real-world noise and perturba-

tions. Our setup includes four pick-and-place tasks (yellow bowl, biscuit box, toy, eggplant) and a long-horizon tidy-up task with distractors. We train a multi-task policy for the four tasks using 1,152 trajectories (288 per task) and a dedicated policy for tidy-up using 648 trajectories. We compare against UniVLA [42] under identical conditions and evaluate zero-shot generalization to unseen objects (novel colors and geometries) and robustness under human interventions.

4.2 Implementation Details

Model structure. Our world model adopts a purely autoregressive Transformer architecture with 8.5B parameters, initialized from the world model of UniVLA [42]. Images are tokenized using the Emu3 [41] VQ-based encoder with a spatial compression factor of 8. For action tokenization, we first apply 1st and 99th percentile normalization, and then utilize the FAST [34] tokenizer with a vocabulary size of 1024. We integrate actions by mapping them to the final 1024 token IDs of the language vocabulary. Consequently, vision, language, and action share a single discrete token space, enabling joint representation learning and cross-modal alignment. Visual observation setups vary by evaluation domain: (1) SimplerEnv: A single-view RGB image (256×256) is used across both stages; (2) LIBERO: Stage 1 uses a third-person view (200×200), supplemented by an additional wrist-view camera (200×200) for Stage 2; (3) Real-world: Stage 1 relies on the front-camera view (256×144), while Stage 2 incorporates a wrist-view image (240×135).

Structured planner training. We train the world model to predict the next structured frame foresight from the instruction and visual context. We initialize the model with pretrained weights of world model from UniVLA [42]. Supervision is applied only to vision tokens, guiding the model to generate physically grounded visual foresight. Because training solely on structured frames is temporally sparse, we apply a sliding-window augmentation that slightly shifts the target timestep and resamples its context, densifying supervision and exposing the model to richer spatiotemporal patterns. The structured planner is trained with a universal batch size of 32 for 30K, 10K, and 2K steps on the SimplerEnv, LIBERO, and real-world setups, respectively.

Action policy training. We fine-tune the planner to shift from visual foresight prediction to action prediction. The input sequence begins with the language instruction, followed by interleaved visual and action tokens. Supervision is applied solely to action tokens, allowing the model to learn an explicit action policy grounded in visual planning. Subsequently, the action policy is trained across the three domains with varying configurations: 20K steps (batch size 128) for SimplerEnv, 4K steps (batch size 192) for LIBERO, and 2K steps (batch size 32) for real-world deployments.

Efficient adaptation. To specialize the pre-trained world model for generating task-semantic structured frames without costly full-model retraining, we employ LoRA [14] (rank=32) for parameter-efficient fine-tuning. This approach preserves the model's foundational visual prediction capabilities while achieving

high-quality foresight at a fraction of the computational cost, validating it as a highly effective adaptation strategy.

4.3 Main Results

SimplerEnv benchmark. As shown in Tab. 1, our StructVLA achieves an average success rate of 75.0% on the SimplerEnv-WidowX benchmark, outperforming all prior baselines. Beyond the overall score, StructVLA achieves consistently strong performance across tasks. In particular, on the challenging *Stack Green on Yellow Block* task, which requires sustained progress tracking and precise spatial execution, StructVLA matches VideoVLA and clearly outperforms the other baselines. We attribute these gains to structured frame foresight, which provides sparse, kinematically grounded progress anchors and helps stabilize closed-loop control over longer horizons.

LIBERO benchmark. As shown in Tab. 2, our StructVLA achieves 94.8% average success on LIBERO and shows strong performance overall, especially on LIBERO-Object, LIBERO-Goal, and LIBERO-Long. In particular, on the challenging LIBERO-Long suite, StructVLA performs notably better than prior methods, suggesting more reliable execution over extended horizons. We attribute this to the structured planner, which predicts structured frames at key interaction transitions and provides sparse mid-level visual subgoals for control. These subgoals act as progress anchors that guide the policy, reduce drift from accumulated errors, and improve consistency in multi-stage manipulation.

Real-world deployments. As shown in Fig. 5, StructVLA outperforms the baselines in real-world deployments. On the four pick-and-place tasks, StructVLA achieves an 87.5% average success rate, which is 20.0% higher than UniVLA, and reaches 10/10 on the eggplant task. On the long-horizon *tidy-up* task, StructVLA completes 8/10 trials, compared with 4/10 for UniVLA and 2/10 for SpatialVLA, indicating stronger stability over extended execution. These results suggest that incorporating structured planning improves real-world robustness, especially for long-horizon execution where small deviations and environmental noise can otherwise compound over time.

Real-world generalization and robustness. Beyond standard instruction following, we evaluate zero-shot generalization and robustness under dynamic perturbations (Fig. 7). On out-of-distribution pick-and-place with unseen colors (*e.g.*, a pink bowl) and novel geometries (*e.g.*, a blue cup), StructVLA achieves 80.0% success, compared to 30.0% for UniVLA. On the long-horizon tidy-up task, StructVLA generalizes to unseen objects and maintains 70.0% success. Under human interventions such as adding objects during execution, StructVLA retains 80.0% success versus 40.0% for UniVLA. We attribute these gains to structured planning with kinematically grounded milestones, which provides stable progress anchors and helps the policy remain consistent under appearance shifts and dynamic disturbances.

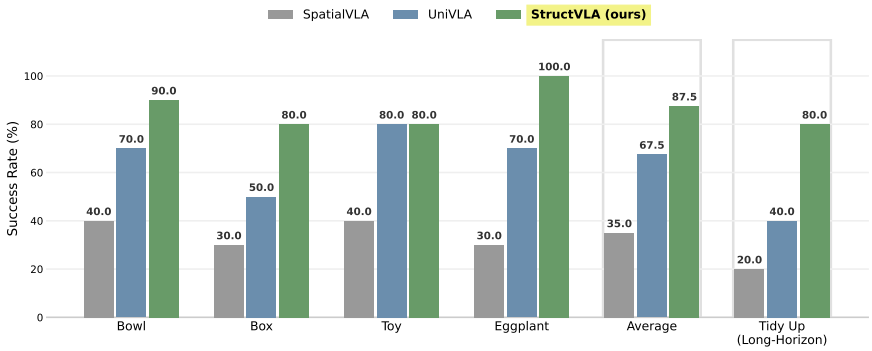

Fig. 5: Quantitative Results on Real-World Deployments. Success rates on diverse physical manipulation tasks (10 trials per task). **StructVLA** matches or exceeds prior baselines on the foundation tasks and remains strong on the challenging long-horizon tidy-up task.

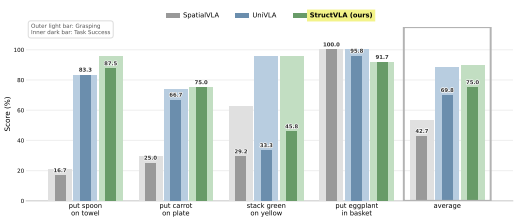

Fig. 6: Grasp Success vs. Task Success. We analyze how grasp success translates to task completion. **StructVLA** attains the highest mean grasp success and a higher grasp-to-task conversion rate than the baseline UniVLA and SpatialVLA, indicating improved long-horizon stability with reduced drift and error accumulation.

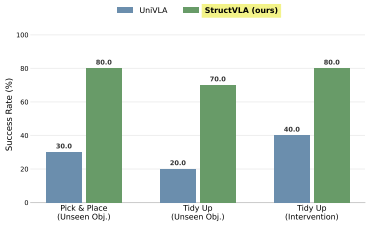

Fig. 7: Real-World Generalization and Robustness. Success rates for zero-shot generalization (novel object geometries) and dynamic human interventions. **StructVLA** outperforms the baseline in all settings.

4.4 Extended Analysis

Beyond the main performance results, this section provides a deeper analysis of StructVLA's core properties through a series of empirical investigations and ablation studies. Specifically, we first evaluate the performance and mechanistic insights of the proposed structured planner. We then examine the training efficiency of our approach and investigate the correlation between grasp quality and overall task success. Finally, we analyze the role of historical context in ensuring accurate predictions, and present an extended comparative evaluation of real-world deployments.

Structured planner performance. As shown in Fig. 3, our structured planner predicts the next structured frame from the current observation and language instruction, providing explicit visual guidance for subsequent task execution.

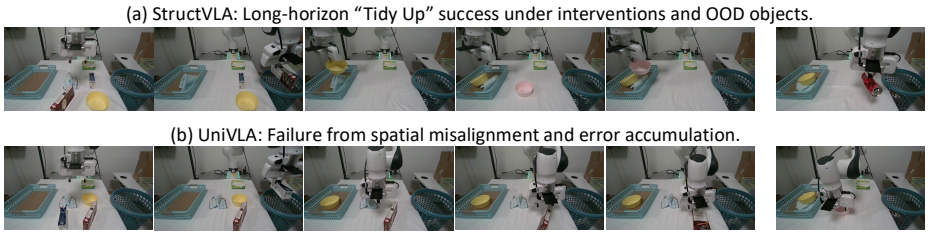

Fig. 8: Qualitative comparison of real-world deployments. (a) StructVLA (Ours): Robust long-horizon execution with zero-shot generalization to OOD objects (*e.g.*, pink bowl, Coke can). **(b) UniVLA (Baseline):** Prone to error accumulation and grasp failures, consistent with dense-prediction drift and limited generalization.

Across multi-step rollouts, the predicted frames preserve coherent task progression and exhibit substantially less drift than those generated by a conventional world model. Notably, even when conditioned on a single observation, the planner can generate a five-step rollout that remains consistent with the intended manipulation trajectory and captures meaningful intermediate task states. In contrast, the baseline world model often produces nearly static future predictions with limited progress information, or introduces visual distortions when larger state transitions are required. These results indicate that structured-frame supervision encourages the model to produce more stable, task-relevant, and informative visual foresight. Such structured predictions provide reliable intermediate visual targets that can be more effectively followed by the downstream action policy during long-horizon execution.

Mechanistic insights of the structured planner. We visualize attention maps from the final self-attention layer for both the structured planner and the baseline world model in Fig. 4 (a). During kinematic transitions, the structured planner assigns higher attention to motion-relevant regions, particularly around the gripper and its interaction with nearby objects. This focused attention pattern suggests that the planner better captures task-critical interaction cues. In contrast, the baseline world model exhibits a more diffuse attention distribution, without a clear focus on the regions most relevant to manipulation progress. We further evaluate the structured planner on out-of-distribution scenes and tasks that are excluded from Stage 1 training in Fig. 4 (b). Despite the distribution shift, the predicted structured frames remain visually coherent and consistent with the task context. These results suggest that structured-frame fine-tuning preserves useful visual and dynamic priors from pretraining, while enabling the model to generalize its structured planning capability to novel environments and unseen task configurations.

Training efficiency of StructVLA. As shown in Tab. 3, StructVLA demonstrates strong training efficiency, reaching competitive success rates with substantially fewer optimization steps than prior methods under the same batch-size setting. This improvement suggests that the proposed structured-planning formulation provides a more effective learning signal than directly modeling dense

Table 3: Learning Efficiency and Convergence Speed. Comparison of the training steps required for different models to reach peak success. All methods are trained under identical environments and hyperparameters (*e.g.*, batch size 128).

Method	Batch	Steps	Accuracy
RoboVLMs	128	50k	37.5%
UniVLA	128	12k	64.8%
StructVLA	128	**10k**	**66.7%**

Table 4: Historical Context Length. Unlike UniVLA, StructVLA improves with longer context, effectively leveraging extended history for long-horizon reasoning and distant goal prediction.

Method	History	Accuracy
StructVLA	2	70.8%
	3	71.9% ↑1.55%
	4	**75.0%** ↑**5.93%**
UniVLA	2	**69.8%**
	3	69.8% −0.0
	4	66.7% ↓**4.44%**

future observations or relying solely on low-level action supervision. By focusing the model on sparse, kinematically meaningful progress anchors, StructVLA encourages representations that explicitly capture task progression and interaction-relevant state changes. This structured representation reduces the effective search space faced by the downstream action policy, allowing it to focus on translating informative intermediate visual targets into executable control actions. As a result, the action policy has a lower learning burden and can improve more rapidly with fewer training samples and optimization steps.

Analysis of Grasp Quality vs. Task Success. Beyond overall task success, Fig. 6 examines how grasp success translates into task completion. StructVLA achieves the highest mean grasp success (89.6%), although on *Put Eggplant* it is slightly below two baselines (91.7% vs. 100%). More importantly, StructVLA attains a higher grasp-to-task conversion rate than the baseline UniVLA, indicating more consistent post-grasp execution. For instance, on *Stack Green on Yellow Block*, StructVLA matches UniVLA in grasp success (95.8%) but achieves higher task success (45.8% vs. 33.3%), suggesting that structured planning improves multi-step, long-horizon execution beyond grasping alone and may reduce plan drift and error accumulation.

History context for accurate prediction. On SimplerEnv, we evaluate history windows of 2, 3, and 4 in Tab. 4 and compare StructVLA with UniVLA under identical computation budgets. StructVLA improves as the history window increases, rising from 70.8% to 75.0%, whereas UniVLA drops from 69.8% to 66.7%. This ablation supports our claim that the structured planner benefits from longer temporal context because it uses the additional observations to forecast the next structured frame and capture long-range task progress, rather than being dominated by short-term visual details. Increasing the history window provides richer temporal evidence, helping StructVLA better disambiguate the current state and form more stable long-horizon plans. As a result, larger history windows provide useful semantic and spatiotemporal evidence for planning and lead to better performance, instead of introducing distracting noise.

Comparative Evaluation of Real-World Deployments. As shown in Fig. 8, StructVLA demonstrates reliable execution on the long-horizon *tidy-up* task, successfully manipulating four objects and remaining stable under dynamic interventions with unseen objects such as a pink bowl and a coke can. In contrast, UniVLA shows weaker long-horizon stability and can suffer from error accumulation when task progress is not explicitly constrained. In our qualitative examples, UniVLA fails to grasp the towel and later knocks over the biscuit box due to compounding positional errors, and it also struggles to handle the novel pink bowl. Overall, these observations highlight a clear gap in deployment behavior, where StructVLA provides more consistent task completion and stronger zero-shot robust generalization in real-world settings.

5 Conclusion

We presented **StructVLA**, a novel Vision-Language-Action framework that reformulates generative world models into explicit, physically grounded structured planners for action control. To achieve robust long-horizon planning and strict physical alignment, StructVLA predicts discrete structured frames grounded in kinematic cues. Through a two-stage training paradigm within a unified token vocabulary, StructVLA bridges the gap between high-level planning signals and low-level action generation. Extensive empirical evaluations demonstrate that StructVLA achieves highly competitive performance, delivering a 75.0% average success rate on the SimplerEnv-WidowX benchmark and 94.8% on LIBERO. Real-world deployments further validate its exceptional cross-category generalization and robustness in executing complex, long-horizon tasks. Ultimately, StructVLA moves beyond dense video generation and semantic planning, providing physically grounded visual foresight for reliable robot control.

Limitations and Discussion. While our results are promising, the primary limitation of this work is its experimental scale, constrained by computational resources. Future evaluation on larger, multi-domain datasets is thus essential to validate broader generalization. Promising research directions also include exploring advanced training strategies and architectural refinements to enhance model stability, and incorporating online reinforcement learning to improve the policy's robustness for real-world deployment.

Acknowledgements. This work was partially supported by New Generation Artificial Intelligence-National Science and Technology Major Project (2025ZD0123100) and by The National Natural Science Foundation of China (NSFC) under no. 62573399 and U25A20530.

References

1. Bharadhwaj, H., Mottaghi, R., Gupta, A., Tulsiani, S.: Track2act: Predicting point tracks from internet videos enables generalizable robot manipulation. In: European Conference on Computer Vision. pp. 306–324. Springer (2024)
2. Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., et al.: π_0: A vision-language-action flow model for general robot control. arXiv preprint arXiv:2410.24164 (2024)
3. Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al.: Rt-1: Robotics transformer for real-world control at scale. arXiv preprint arXiv:2212.06817 (2022)
4. Bu, Q., Yang, Y., Cai, J., Gao, S., Ren, G., Yao, M., Luo, P., Li, H.: Univla: Learning to act anywhere with task-centric latent actions. arXiv preprint arXiv:2505.06111 (2025)
5. Chen, B., Zhang, T., Geng, H., Song, K., Zhang, C., Li, P., Freeman, W.T., Malik, J., Abbeel, P., Tedrake, R., et al.: Large video planner enables generalizable robot control. arXiv preprint arXiv:2512.15840 (2025)
6. Chen, X., Wei, H., Zhang, P., Zhang, C., Wang, K., Guo, Y., Yang, R., Wang, Y., Xiao, X., Zhao, L., et al.: Villa-x: enhancing latent action modeling in vision-language-action models. arXiv preprint arXiv:2507.23682 (2025)
7. Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S.: Diffusion policy: Visuomotor policy learning via action diffusion. The International Journal of Robotics Research **44**(10-11), 1684–1704 (2025)
8. Dai, T., Han, M., Du, T., Liu, Z., Zhang, Z., Li, Z., Khan, S., Yu, J., Chang, X.: See, plan, rewind: Progress-aware vision-language-action models for robust robotic manipulation. arXiv preprint arXiv:2603.09292 (2026)
9. Feng, Y., Tan, H., Mao, X., Liu, G., Huang, S., Xiang, C., Su, H., Zhu, J.: Generalist bimanual manipulation via foundation video diffusion models. arXiv e-prints pp. arXiv–2507 (2025)
10. Gao, S., Zhou, S., Du, Y., Zhang, J., Gan, C.: Adaworld: Learning adaptable world models with latent actions. arXiv preprint arXiv:2503.18938 (2025)
11. Ha, D., Schmidhuber, J.: World models. arXiv preprint arXiv:1803.10122 **2**(3), 440 (2018)
12. Hafner, D., Lillicrap, T., Ba, J., Norouzi, M.: Dream to control: Learning behaviors by latent imagination (2019)
13. Hao, H., Chen, L., Han, M., Li, C., An, D., Yang, Y., Li, Z., Chang, X.: Latentpilot: Scene-aware vision-and-language navigation by dreaming ahead with latent visual reasoning. arXiv preprint arXiv:2603.29165 (2026)
14. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. ICLR **1**(2), 3 (2022)
15. Hu, Y., Guo, Y., Wang, P., Chen, X., Wang, Y.J., Zhang, J., Sreenath, K., Lu, C., Chen, J.: Video prediction policy: A generalist robot policy with predictive visual representations. arXiv preprint arXiv:2412.14803 (2024)
16. Hu, Y., Zhang, J., Luo, Y., Guo, Y., Chen, X., Sun, X., Feng, K., Lu, Q., Chen, S., Zhang, Y., et al.: Bagelvla: Enhancing long-horizon manipulation via interleaved vision-language-action generation. arXiv preprint arXiv:2602.09849 (2026)
17. Huang, C.P., Wu, Y.H., Chen, M.H., Wang, F., Yang, F.E.: Thinkact: Vision-language-action reasoning via reinforced visual latent planning. Advances in Neural Information Processing Systems **38**, 82782–82802 (2026)

18. Huang, Z., Liu, M., Lin, X., Zhu, M., Zhao, C., Du, Z., Lin, Y., Li, X., Jia, Y., Zhong, H., et al.: Notvla: Semantics-preserving robot adaptation via narrative action interfaces. arXiv preprint arXiv:2510.03895 (2025)
19. Kim, M.J., Gao, Y., Lin, T.Y., Lin, Y.C., Ge, Y., Lam, G., Liang, P., Song, S., Liu, M.Y., Finn, C., et al.: Cosmos policy: Fine-tuning video models for visuomotor control and planning. arXiv preprint arXiv:2601.16163 (2026)
20. Kim, M.J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., et al.: Openvla: An open-source vision-language-action model. arXiv preprint arXiv:2406.09246 (2024)
21. LeCun, Y., et al.: A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. Open Review **62**(1), 1–62 (2022)
22. Li, Q., Liang, Y., Wang, Z., Luo, L., Chen, X., Liao, M., Wei, F., Deng, Y., Xu, S., Zhang, Y., et al.: Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. arXiv preprint arXiv:2411.19650 (2024)
23. Li, X., Hsu, K., Gu, J., Pertsch, K., Mees, O., Walke, H.R., Fu, C., Lunawat, I., Sieh, I., Kirmani, S., et al.: Evaluating real-world robot manipulation policies in simulation. arXiv preprint arXiv:2405.05941 (2024)
24. Liang, J., Huang, W., Xia, F., Xu, P., Hausman, K., Ichter, B., Florence, P., Zeng, A.: Code as policies: Language model programs for embodied control. In: 2023 IEEE International conference on robotics and automation (ICRA). pp. 9493–9500. IEEE (2023)
25. Lin, K., Agia, C., Migimatsu, T., Pavone, M., Bohg, J.: Text2motion: From natural language instructions to feasible plans. Autonomous Robots **47**(8), 1345–1365 (2023)
26. Liu, B., Zhu, Y., Gao, C., Feng, Y., Liu, Q., Zhu, Y., Stone, P.: Libero: Benchmarking knowledge transfer for lifelong robot learning. Advances in Neural Information Processing Systems **36**, 44776–44791 (2023)
27. Liu, H., Li, X., Li, P., Liu, M., Wang, D., Liu, J., Kang, B., Ma, X., Kong, T., Zhang, H.: Towards generalist robot policies: What matters in building vision-language-action models (2025)
28. Long, Q., Wang, Y., Song, J., Zhang, J., Li, P., Wang, W., Wang, Y., Li, H., Xie, S., Yao, G., et al.: Scaling world model for hierarchical manipulation policies. arXiv preprint arXiv:2602.10983 (2026)
29. Mendonca, R., Bahl, S., Pathak, D.: Structured world models from human videos. In: Robotics: Science and Systems (RSS) (2023)
30. OpenAI: Introducing gpt-5.2. `https://openai.com/index/introducing-gpt-5-2/` (2025), accessed: 2026-03-06
31. O'Neill, A., Rehman, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., Mandlekar, A., Jain, A., 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). pp. 6892–6903. IEEE (2024)
32. Pai, J., Achenbach, L., Montesinos, V., Forrai, B., Mees, O., Nava, E.: mimic-video: Video-action models for generalizable robot control beyond vlas. arXiv preprint arXiv:2512.15692 (2025)
33. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4195–4205 (2023)
34. Pertsch, K., Stachowicz, K., Ichter, B., Driess, D., Nair, S., Vuong, Q., Mees, O., Finn, C., Levine, S.: Fast: Efficient action tokenization for vision-language-action models. arXiv preprint arXiv:2501.09747 (2025)

35. Qu, D., Song, H., Chen, Q., Yao, Y., Ye, X., Ding, Y., Wang, Z., Gu, J., Zhao, B., Wang, D., et al.: Spatialvla: Exploring spatial representations for visual-language-action model. arXiv preprint arXiv:2501.15830 (2025)
36. Shen, Y., Wei, F., Du, Z., Liang, Y., Lu, Y., Yang, J., Zheng, N., Guo, B.: Videovla: Video generators can be generalizable robot manipulators. Advances in neural information processing systems **38**, 95597–95621 (2026)
37. Sun, L., Xie, B., Liu, Y., Shi, H., Wang, T., Cao, J.: Geovla: Empowering 3d representations in vision-language-action models. arXiv preprint arXiv:2508.09071 (2025)
38. Team, O.M., Ghosh, D., Walke, H., Pertsch, K., Black, K., Mees, O., Dasari, S., Hejna, J., Kreiman, T., Xu, C., et al.: Octo: An open-source generalist robot policy. arXiv preprint arXiv:2405.12213 (2024)
39. Tharwat, B., Nasser, Y., Abouzeid, A., Reid, I.: Latent action pretraining through world modeling. arXiv preprint arXiv:2509.18428 (2025)
40. Walke, H.R., Black, K., Zhao, T.Z., Vuong, Q., Zheng, C., Hansen-Estruch, P., He, A.W., Myers, V., Kim, M.J., Du, M., et al.: Bridgedata v2: A dataset for robot learning at scale. In: Conference on Robot Learning. pp. 1723–1736. PMLR (2023)
41. Wang, X., Zhang, X., Luo, Z., Sun, Q., Cui, Y., Wang, J., Zhang, F., Wang, Y., Li, Z., Yu, Q., et al.: Emu3: Next-token prediction is all you need. arXiv preprint arXiv:2409.18869 (2024)
42. Wang, Y., Li, X., Wang, W., Zhang, J., Li, Y., Chen, Y., Wang, X., Zhang, Z.: Unified vision-language-action model (2025), `https://arxiv.org/abs/2506.19850`
43. Wen, C., Lin, X., So, J., Chen, K., Dou, Q., Gao, Y., Abbeel, P.: Any-point trajectory modeling for policy learning. arXiv preprint arXiv:2401.00025 (2023)
44. Wu, P., Escontrela, A., Hafner, D., Abbeel, P., Goldberg, K.: Daydreamer: World models for physical robot learning. In: Conference on robot learning. pp. 2226–2240. PMLR (2023)
45. Xie, A., Rybkin, O., Sadigh, D., Finn, C.: Latent diffusion planning for imitation learning. arXiv preprint arXiv:2504.16925 (2025)
46. Xu, Y., Parker-Holder, J., Pacchiano, A., Ball, P.J., Rybkin, O., Roberts, S.J., Rocktäschel, T., Grefenstette, E.: Learning general world models in a handful of reward-free deployments. In: Advances in Neural Information Processing Systems (NeurIPS) 35 (2022)
47. Zhang, W., Liu, H., Qi, Z., Wang, Y., Yu, X., Zhang, J., Dong, R., He, J., Wang, H., Zhang, Z., et al.: Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge. Advances in Neural Information Processing Systems **38**, 24195–24228 (2026)
48. Zhang, Z., Yang, S., Hu, Q., Huang, L.J., Hou, J., Sun, Y., Lu, Y., Han, S.: Foreact: Steering your vla with efficient visual foresight planning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 37195–37205 (2026)
49. Zhao, Q., Lu, Y., Kim, M.J., Fu, Z., Zhang, Z., Wu, Y., Li, Z., Ma, Q., Han, S., Finn, C., et al.: Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 1702–1713 (2025)
50. Zhong, Z., Yan, H., Li, J., Liu, X., Gong, X., Zhang, T., Song, W., Chen, J., Zheng, X., Wang, H., et al.: Flowvla: Visual chain of thought-based motion reasoning for vision-language-action models. arXiv preprint arXiv:2508.18269 (2025)
51. Zhu, C., Yu, R., Feng, S., Burchfiel, B., Shah, P., Gupta, A.: Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. In: Proceedings of Robotics: Science and Systems (RSS) (2025)

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