BUILDING ROBUST WORLD MODELS FOR VLA RL: VISUAL GENERALIZATION AND LONG-HORIZON CONSISTENCY

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language-action (VLA) policies can be post-trained in learned simulators, but current world models often fail under visual shifts and accumulate errors during long-horizon rollouts. We present *Sword*, a style-robust world model for policy post-training (Figure 1). *Sword* addresses these failure modes with two complementary components. Structure-Guided Style Augmentation (SGSA) uses depth, segmentation, and task-conditioned style transfer to diversify visual appearance while preserving task-relevant geometry and semantics. Dynamic Latent Bootstrapping (DLB) reuses cached model-predicted latents as context during fixed-window training, exposing the model to self-generated inputs without requiring full-sequence rollouts. We further introduce *LIBERO-Mixed*, an evaluation set combining original episodes with style-transferred episodes generated using prompts held out from training. Experiments on LIBERO show that *Sword* improves prediction quality and temporal consistency over representative action-conditioned world models, including under held-out style shifts. In GRPO post-training, using *Sword* as the simulator also yields higher VLA task success rates than using WoVR.

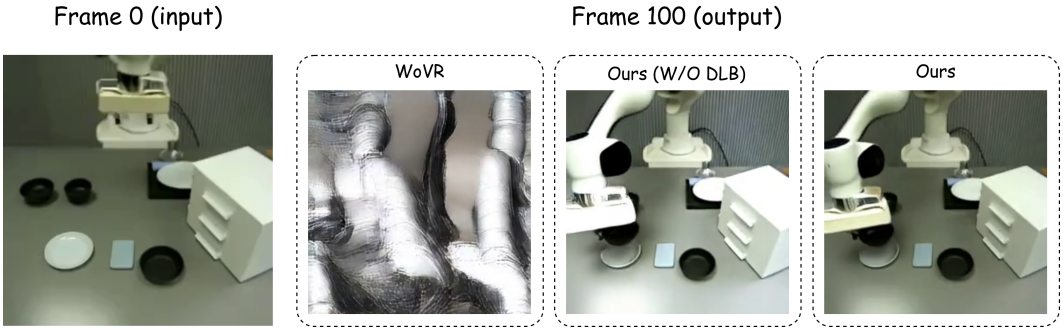


Figure 1: We propose a new world model, Sword, and compare its predicted video frames with those of its variant without Dynamic Latent Bootstrapping (Ours w/o DLB) and WoVR Jiang et al. (2026).

1 INTRODUCTION

The development of Vision-Language-Action (VLA) models Kim et al. (2025); Bjorck et al. (2025) has marked a critical milestone in robotic manipulation, enabling end-to-end action generation conditioned on multimodal inputs Ma et al. (2024); Zhong et al. (2025); Zhang et al. (2025). While imitation learning has established a strong foundation for these models, the paradigm fundamentally limits the performance ceiling to the quality of human demonstrations. Reinforcement learning (RL) offers a principled mechanism to surpass this bottleneck Guan et al. (2026), yet the prohibitive cost of physical robot interaction necessitates the use of simulated environments. Consequently, World Models have emerged as a compelling alternative, acting as generative simulators that approximate transition dynamics. Recent literature explores the synergistic relationship between these domains, primarily diverging into two paradigms: the first emphasizes the co-evolution of world models and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

VLA policies within a unified architecture, as demonstrated by frameworks such as RynnVLA Cen et al. (2025a) and WorldVLA Cen et al. (2025b), where action understanding and visual generation mutually enhance one another; the second paradigm, exemplified by WoVR Jiang et al. (2026), treats the world model strictly as a surrogate simulator for post-training policies via reinforcement learning, explicitly regulating imagined rollouts to mitigate hallucination.

Despite these structural advancements, using world models as simulators still exposes two key limitations. The first is an environment-level generalization problem, where the model fails under OOD visual or state perturbations. Current world models are prone to severe overfitting when trained on narrow data distributions, adapting primarily to the specific textures and static configurations of a single environment. When evaluated on the LIBERO benchmark Liu et al. (2023a), these models exhibit clear fragility under out-of-distribution (OOD) perturbations in the initial state, such as changes in background color or lighting conditions. Such discrepancies can rapidly trigger catastrophic cascading hallucinations, manifested as severe blurriness or abnormal exposure, thereby depriving the simulator of its ability to model physical dynamics. This failure mode suggests that existing models often merely capture static statistical correlations, rather than learning the underlying dynamic laws.

The second is an autoregressive rollout problem, where teacher forcing creates a mismatch between ground-truth contexts during training and self-generated contexts during inference. A world model is typically formulated as a state transition function, taking context frames and actions as input and producing predicted observation frames of future states. However, current world models commonly adopt the Teacher Forcing paradigm Fangqi et al. (2025); Jiang et al. (2026), where ground-truth frames are used as context frames during training to predict future observations. In contrast, during inference, the model must condition on its own predicted frames for autoregressive generation. This discrepancy introduces a distribution shift between training and inference, leading to severe *exposure bias*, where small errors in the initial predictions accumulate and are amplified over long-horizon rollouts.

To overcome the above limitations, we propose *Sword*, which contains two complementary modules for constructing a robust generative simulator by disentangling environmental visual styles and aligning the training and inference distributions. First, to suppress overfitting to superficial visual features, we introduce Structure-Guided Style Augmentation based on the Cosmos architecture NVIDIA et al. (2026). During training, it applies diverse perturbations to the background appearance, tabletop color, and illumination while preserving task semantics and geometric structure, encouraging the model to focus more on physical dynamics. Second, to mitigate the train–inference gap, we propose Dynamic Latent Bootstrapping (DLB). DLB continuously uses the model's own latent predictions as context under fixed-window training and employs a latent-space cache to avoid the high cost of complete rollouts, thereby suppressing long-horizon error accumulation. The two modules complement each other, allowing the model to maintain stable long-horizon predictions under style shifts. To further evaluate this capability, we construct *LIBERO-Mixed* from raw samples and samples with unseen styles, ensuring that its evaluation style prompts are completely disjoint from the training prompts. The main contributions of this paper are summarized as follows:

- **Structure-Guided Style Augmentation.** We introduce a style augmentation pipeline that diversifies visual conditions while preserving geometric structure and task semantics, encouraging the world model to learn transition-relevant dynamics rather than overfitting to superficial textures and thereby improving generalization to visual distribution shifts.

- **Dynamic Latent Bootstrapping.** We propose a latent bootstrapping mechanism tailored to fixed-window training. By caching predicted latents across iterations and progressively using them as context, DLB mitigates the train–inference mismatch caused by teacher forcing and improves long-horizon consistency in autoregressive generation.

- **Out-of-Distribution Evaluation and Comprehensive Validation.** We construct *LIBERO-Mixed*, an out-of-distribution style evaluation dataset whose evaluation prompts are completely disjoint from the training prompts. We then systematically evaluate out-of-distribution generalization, prediction quality, simulator fidelity, and VLA reinforcement-learning post-training.

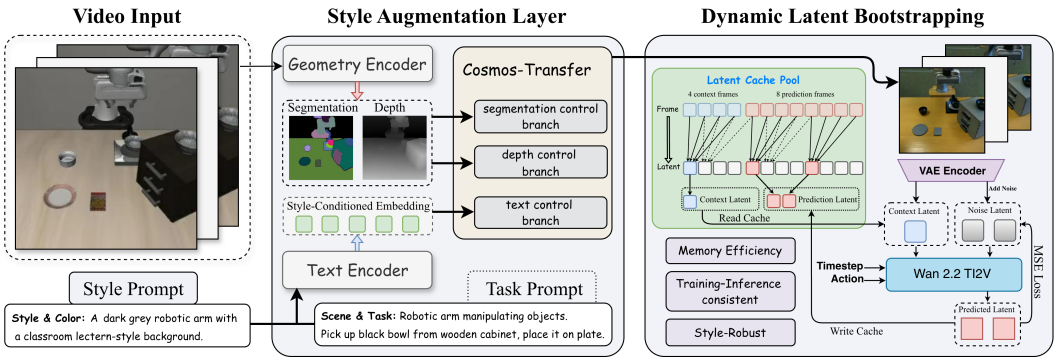


Figure 2: Pipeline of *Sword*. Input observations first undergo Structure-Guided Style Augmentation to diversify visual appearance while preserving task semantics and geometry. The augmented sequences are then used to train the world model, where Dynamic Latent Bootstrapping reuses cached predictions as context to mitigate exposure bias and maintain long-horizon consistency.

2 Related Work

The intersection of visual generation and robotic control has catalyzed the evolution of Vision-Language-Action models Zitkovich et al. (2023); Black et al. (2024); Gemini Robotics Team (2025); Kim et al. (2024). Early frameworks relied predominantly on behavior cloning Cadene et al. (2026); Zhou et al. (2026), mapping visual and linguistic inputs directly to continuous or discrete action spaces. To transcend the limitations of static datasets, on-policy reinforcement learning has been adapted for post-training Guan et al. (2026); Shukor et al. (2025), though it is historically bottlenecked by the sample inefficiency of real-world environments. This constraint has motivated the integration of World Models Wan et al. (2025); Esser et al. (2024), which learn to approximate the environment's transition function $P(o_{t+1}|o_t, a_t)$. Recent architectures like RynnVLA Cen et al. (2025a) and WorldVLA Cen et al. (2025b) advocate for a unified model where policy derivation and future state prediction share a latent space, demonstrating that action conditioning improves visual fidelity and vice versa. Conversely, frameworks like WoVR Jiang et al. (2026) decouple the components, utilizing an action-conditioned diffusion transformer strictly as a reliable simulator to execute algorithms like PPO Schulman et al. (2017) or GRPO Shao et al. (2024) entirely in imagination.

While these world-model-based simulators demonstrate promise, they inherit the systemic flaws of autoregressive video generation, most notably exposure bias. Traditional training methodologies rely on teacher forcing Fangqi et al. (2025); Jiang et al. (2026), evaluating the step-wise loss using ground-truth historical frames. During closed-loop inference, the model must condition on its own imperfect outputs, leading to rapid degradation—a phenomenon exacerbated when the policy explores out-of-distribution states. Recent literature addresses this through techniques like Distribution Matching Distillation Yin et al. (2024) or Diffusion Forcing Chen et al. (2024), which inject varying levels of noise into the context frames to simulate inference conditions. The Self-Forcing paradigm Huang et al. (2025) further advances this by explicitly unrolling the autoregressive generation during training, forcing the model to learn to recover from its own errors. To mitigate the train–inference distribution shift in long-horizon embodied tasks, we propose Dynamic Latent Bootstrapping (DLB). Instead of unrolling a complete sequence within a single training iteration, DLB caches generated latents across iterations and reuses them as historical context for fixed-window training, improving long-horizon consistency at a lower cost. In addition, we construct *LIBERO-Mixed* and use style prompts that are completely disjoint from the training prompts to evaluate generalization to visual distribution shifts.

3 Methodology

We define the world model as a learned transition function that predicts a future observation window conditioned on recent observations, the corresponding action sequence, and the first frame of the

episode:

$$\hat{o}_{t+1:t+H} \sim \hat{P}_\phi\big(o_{t+1:t+H} \mid o_{t-C+1:t}, a_{t:t+H-1}, o_0\big), \tag{1}$$

where $C = 4$ is the number of context frames, $H = 8$ is the prediction horizon, and o_0 provides the global episode condition. The transition distribution $\hat{P}_\phi$ is parameterized by a diffusion-based generative Transformer. By training a reliable world model, we enable the VLA policy to optimize its actions within a learned simulation environment, thereby reducing reliance on real-environment interactions.

3.1 Structure-Guided Style Augmentation

Style Transfer with Cosmos-Transfer. Cosmos-Transfer 2.5 NVIDIA et al. (2025) is a conditional style-transfer model. In general, the current observation o_t and a target style prompt style_t can be provided to the model to obtain a new observation with the desired style:

$$\tilde{o}_t = \text{Cosmos-Transfer}(o_t, \mathrm{style}_t). \tag{2}$$

By changing the style prompt, this process can modify brightness, saturation, robot-arm color, tabletop texture, and background appearance, allowing the same robot-interaction trajectory to be rendered under diverse visual conditions. However, style transfer conditioned only on the target prompt lacks explicit constraints from the original observation. It may alter object shapes and positions, discard task-relevant semantics, or even disrupt the geometry and interaction relationships between the robot and objects. Thus, visually diverse augmented samples do not necessarily preserve the physical consistency of the original task.

Structure-Guided Style Augmentation. To improve the controllability of style transfer, we introduce both geometric structure and task semantics into Cosmos-Transfer. We first use DepthAnything Yang et al. (2024) to extract a depth map d_t from the current observation and use GroundingDINO Liu et al. (2023b) together with SAM2 Ravi et al. (2024) to extract a semantic segmentation map s_t. The depth map preserves the three-dimensional geometry of the scene, while the segmentation map constrains the shapes, positions, and spatial boundaries of the robot and task-relevant objects. Both are injected through the modality-control branches of Cosmos-Transfer as explicit geometric guidance.

Meanwhile, we concatenate the current task description task_t with the target style prompt style_t. We first tokenize the joint text into token IDs and then obtain its text embedding through the T5 encoder:

$$q_t = \text{Tokenizer}([\mathrm{task}_t; \mathrm{style}_t]), \qquad e_t = \mathrm{Enc}_{\mathrm{T5}}(q_t). \tag{3}$$

The style prompt specifies the desired visual appearance, while the task description preserves the semantics of the current robot–object interaction. The resulting embedding e_t is injected into the generation process through the T5 text branch of Cosmos-Transfer Raffel et al. (2020), providing a unified semantic condition that identifies the task-relevant content that should remain unchanged during style transfer.

The resulting Structure-Guided Style Augmentation process is written as

$$\tilde{o}_t = \text{Cosmos-Transfer}(o_t \mid d_t, s_t, e_t; \eta)\,, \tag{4}$$

where e_t is the joint text embedding of the task description and target style prompt, and η controls the strength of conditional guidance. By jointly constraining textual semantics, scene geometry, and object structure, the proposed method diversifies visual appearance while preserving object positions, geometric structure, and robot–object dynamics. The augmented data therefore remain consistent with the original task semantics and encourage the world model to learn stable physical regularities rather than superficial visual textures. Figure 2 illustrates the style-augmentation process, with additional qualitative examples provided in the supplementary material.

3.2 Dynamic Latent Bootstrapping for Mitigating Exposure Bias

Video world models are commonly trained with teacher forcing, using latent representations of ground-truth historical observations to predict future observations. Although this provides stable

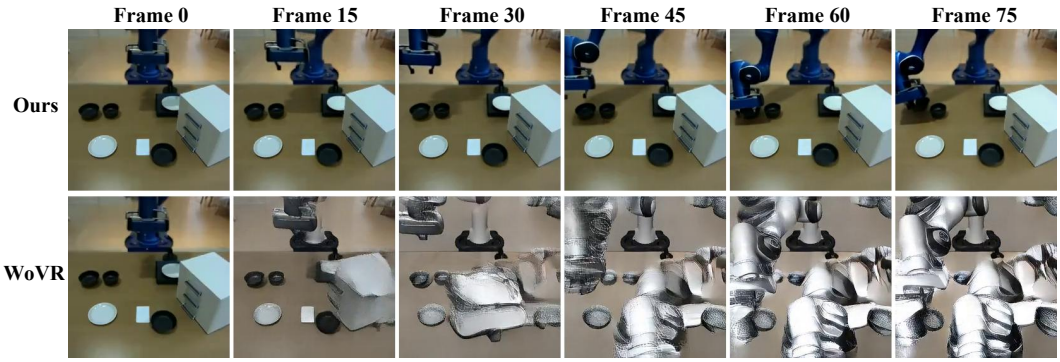


Figure 3: Qualitative comparison under held-out style prompts. WoVR fails to preserve the initial visual style, while our model maintains stable and accurate predictions.

	LIBERO-Raw				*LIBERO-Mixed (raw + ood)*			
Model	LPIPS ↓	FID ↓	FVD ↓	FloLPIPS ↓	LPIPS ↓	FID ↓	FVD ↓	FloLPIPS ↓
EVAC	0.27	66.69	94.82	0.41	0.40	103.72	196.66	0.49
WMPO	0.15	35.03	39.06	**0.22**	0.39	79.54	137.21	0.44
WoVR	0.13	22.01	61.26	0.26	0.39	119.62	198.84	0.46
Cosmos-Predict2	0.39	133.23	209.34	0.28	0.50	180.04	310.48	0.56
Ours w/o SGSA & DLB	0.16	23.64	61.57	0.29	0.47	111.66	213.33	0.52
Ours w/o DLB	0.13	20.80	48.46	0.25	0.22	37.84	110.71	0.32
Ours w/o SGSA	0.14	18.90	34.64	0.27	0.34	77.19	95.58	0.44
Ours	**0.12**	**18.51**	**32.17**	0.23	**0.17**	**28.09**	**86.84**	**0.26**

Table 1: Quantitative comparison with baselines and module-level ablations on LIBERO-Raw and LIBERO-Mixed. "w/o" denotes that the corresponding module(s) are removed.

supervision, it differs fundamentally from closed-loop inference: historical conditions always come from the data distribution during training, whereas inference recursively conditions on the model's own outputs. The model therefore receives little training experience in recovering from its own prediction errors, allowing small early deviations to accumulate over long-horizon generation and resulting in exposure bias.

One feasible way to mitigate this problem is to use the model's own predictions as historical context during training, exposing the model to a conditioning distribution closer to inference. The most direct implementation recursively rolls out a complete sequence from the initial observation and feeds every prediction back into the model. However, a complete rollout requires multiple rounds of sequential sampling. If the computation graph is retained across windows, activation memory also grows substantially with rollout length. Even with gradient truncation, repeated sampling remains computationally expensive. Moreover, low-quality predictions early in training can rapidly shift the subsequent conditioning distribution and make optimization more difficult.

To control the training cost of large video diffusion models, we sample fixed-length local windows from long trajectories. Each training sample contains 12 consecutive frames: the first 4 serve as historical context and the remaining 8 as prediction targets. This strategy confines forward and backward propagation to a single window, but makes different windows independent and therefore unable to directly use predictions from preceding windows as the current context.

Our key observation is that mitigating the train–inference mismatch requires exposing the model to contexts formed by its own predictions, but these predictions need not be produced through a complete rollout within the current iteration. Based on this observation, we propose *Dynamic Latent Bootstrapping* (DLB), which reuses predicted latents across iterations to construct self-generated historical conditions while retaining fixed-window training.

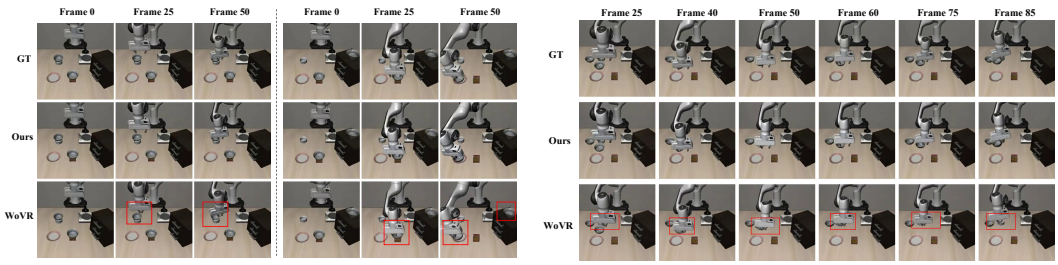


Figure 4: Combined evaluation of our model. **Left:** Our model produces sharper and more stable long-horizon predictions. **Right:** Evaluation of action-following ability and physical interaction consistency.

Dynamic Latent Cache At each training step, we first sample an episode index e and then sample a starting temporal position k within that episode, yielding a 12-frame training window. To reuse predictions produced at different training steps, we maintain a dynamic cache $\mathcal{C}$ indexed jointly by the episode index and latent temporal position. For the current window, the model queries $\mathcal{C}$ for predicted latents corresponding to its historical frames; if an entry is unavailable, it falls back to the ground-truth historical latents. After predicting the current window, all output target latent chunks are detached and written to their corresponding episode and temporal positions for use by subsequently sampled windows. This indexing preserves temporal alignment among cached observations and action sequences while decoupling cross-window dependencies from the computation graph of a single iteration.

To prevent low-quality predictions early in training from abruptly replacing all ground-truth conditions, we schedule cache usage with a linear probability. At training step s, the sampling probability of ground-truth context, $p_{\mathrm{GT}}(s)$, decreases linearly; equivalently, the probability of reading predicted latents from the cache, $p_{\mathcal{C}}(s) = 1 - p_{\mathrm{GT}}(s)$, increases linearly. The model therefore relies primarily on ground-truth context while the cache is being populated early in training, and then smoothly adapts to the increasing proportion of cached context. After each window prediction, the detached predicted latents directly replace the corresponding episode and temporal positions in the cache, allowing the cache to evolve with training while backward propagation remains confined to the current 12-frame window. This avoids the GPU-memory growth caused by retaining a multi-window rollout graph. We maintain the cache directly in the VAE-compressed latent space rather than decoding predictions into pixel space. This avoids repeatedly decoding and re-encoding cached data, improves training efficiency, and reduces the additional storage cost by approximately $60\times$ compared with pixel-space caching.

Training Objective DLB changes only the historical condition used by the model and leaves the base Flow Matching objective unchanged. At training step s, the model uses cached predicted latents as context with the linearly increasing probability $p_{\mathcal{C}}(s)$ and otherwise uses ground-truth historical latents; if the corresponding cache entry is unavailable, it falls back to ground-truth context. This simple schedule smoothly transitions the training condition from ground-truth latents to the model's own predicted latents.

For a target latent z, we sample Gaussian noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ and construct the intermediate state $z_\tau = (1 - \sigma_\tau)z + \sigma_\tau \epsilon$, whose Flow Matching velocity target is $v_\tau = \epsilon - z$. Given action condition a and historical latent context c, the model predicts this velocity using an MSE loss:

$$\mathcal{L}_{\mathrm{DLB}}(\theta) = \mathbb{E}_{z,\epsilon,\tau,a,c}\left[w(\tau)\,\|v_\theta(z_\tau, \tau, a, c) - (\epsilon - z)\|_2^2\right], \tag{5}$$

where $w(\tau)$ is the training weight assigned by the Flow Matching scheduler at timestep τ, a denotes the action condition temporally aligned with the current 12-frame window, and c denotes the historical latent context selected from either ground-truth latents or cached predicted latents according to the linear schedule. Under the 4-context-frame and 8-prediction-frame protocol, the MSE loss is computed only over the two target latent chunks. By conditioning on its own predicted latents, the model learns to recover correct state transitions under prediction-induced deviations.

3.3 Model Architecture

We adopt Wan 2.2 TI2V as the backbone of our world model, and inject the action sequence into the model in a textual form to guide future frame prediction. The backbone follows a standard DiT architecture, where actions are incorporated into each Transformer block through two MLP branches via AdaLN and Cross-Attention, respectively.

At each inference step, the model takes 4 frames as context and predicts the next 8 frames. Since the Wan VAE applies a temporal compression factor of 4, this corresponds to using a single context latent in the latent space to predict two future latents, which are then decoded by the Wan VAE into 8 frames. Furthermore, based on empirical findings from prior work Shin et al. (2026); Yang et al. (2025); Jiang et al. (2026), we concatenate the first frame of each episode as a global condition with the context at every timestep to enhance temporal consistency across generated frames.

4 Experiments

In this section, we conduct extensive experiments to evaluate the performance of *Sword*. We focus on assessing whether our world model can serve as a high-fidelity simulator for VLA tasks, particularly under held-out visual styles.

We evaluate our world model from three perspectives: generalization to held-out style prompts, prediction quality, and robustness and fidelity as a simulator.

4.1 Experimental Setup

Computational Resources. All experiments are primarily conducted on NVIDIA A100 GPUs. We conservatively estimate the total computational cost for training and inference to be approximately 13,000 A100 GPU hours, which corresponds to about 1.5 A100 GPU years.

Dataset and Tasks. We collect 1,600 rollout episodes on LIBERO-Spatial Liu et al. (2023a) using OpenVLA-OFT, with each episode containing 512 frames. We use 1,500 episodes for training and the remaining 100 for evaluation. In addition to the standard *LIBERO-Raw* evaluation set, we use the proposed Structure-Guided Style Augmentation method to construct *LIBERO-Mixed*, which contains visually out-of-distribution samples. This mixed evaluation set combines untransformed raw episodes and style-augmented episodes in a 1:1 ratio, allowing us to evaluate performance jointly under the original distribution and style shifts.

To prevent evaluation styles from leaking into training, we split the style prompts in advance into a training pool $\mathcal{P}_{\text{train}}$ and an evaluation pool $\mathcal{P}_{\text{test}}$, enforcing $\mathcal{P}_{\text{train}} \cap \mathcal{P}_{\text{test}} = \emptyset$. Training-time augmentation uses only $\mathcal{P}_{\text{train}}$, whereas the style-shifted samples in *LIBERO-Mixed* are constructed exclusively with prompts from $\mathcal{P}_{\text{test}}$ that never appear during training. The resulting evaluation therefore measures generalization to visual distribution shifts induced by unseen style prompts rather than memorization of training augmentation prompts.

Baselines. We compare our approach with four baselines: EVAC Jiang et al. (2025), WMPO Fangqi et al. (2025), WoVR Jiang et al. (2026), and Cosmos-Predict2 NVIDIA et al. (2026). All methods use the same 12-frame window with 4 context frames and 8 prediction frames, and are trained on the same training split for the same number of steps.

Evaluation Metrics. We use FID Heusel et al. (2017) and LPIPS Zhang et al. (2018) to measure image fidelity, FVD Unterthiner et al. (2019) and FloLPIPS Danier et al. (2022) to measure video consistency, and task success rate to evaluate downstream utility.

4.2 Generalization to Out-of-Distribution Styles

We test generalization to unseen styles on the *LIBERO-Mixed* dataset constructed in this work. As shown in Fig. 3, WoVR Jiang et al. (2026) begins to exhibit style drift and geometric artifacts from Frame 15 and produces pronounced blurring and structural distortion by Frame 30. In contrast, *Sword* consistently preserves the visual style, object structure, and prediction clarity of the initial

Step	10	65	120	172
WoVR	41.02%	45.70%	49.22%	53.12%
Ours	**43.36%**	**49.22%**	**58.59%**	**61.72%**

Table 2: Average task success rates of OpenVLA-OFT after GRPO post-training with Sword and WoVR as virtual environments.

observation, demonstrating stronger generalization to out-of-distribution styles. Additional qualitative results are provided in the supplementary material.

4.3 Quality of Generated Predictions

Quantitative Analysis. As shown in Table 1, *Sword* achieves the strongest overall performance, with particularly clear advantages on LIBERO-Mixed. This demonstrates its improved prediction quality and temporal consistency under visual distribution shifts.

Qualitative Comparison. We qualitatively compare *Sword* with WoVR on randomly sampled LIBERO sequences. As shown in the left part of Fig. 4, WoVR accumulates errors over time, producing blurring, structural artifacts, and incorrect gripper dynamics. In contrast, *Sword* remains sharp and semantically coherent, demonstrating stronger long-horizon generation quality.

4.4 Robustness and Fidelity of the World Model

Fidelity is essential for a world model to serve as a reliable simulator for VLA policy optimization. We therefore evaluate whether the model follows long-horizon action inputs while preserving physically consistent interaction semantics on LIBERO.

The right part of Fig. 4 shows a challenging example in which the first grasp fails and displaces the object, followed by corrective actions and a successful second grasp. WoVR closes the gripper prematurely at Frames 25, 40, and 50, then diverges from the interaction and eventually "forgets" the object. In contrast, *Sword* keeps the gripper open until the arm is properly aligned and closes it at the correct moment, preserving both the action sequence and object state. This demonstrates stronger action fidelity and long-horizon physical consistency. Additional results are provided in the supplementary material.

4.5 Policy Success Rate

To evaluate downstream utility for reinforcement learning, we conduct GRPO post-training of OpenVLA-OFT within the RLinf framework Yu et al. (2025) on LIBERO-Spatial. During policy rollouts, we replace the original LIBERO simulator with either *Sword* or WoVR Jiang et al. (2026), using each world model as a learned virtual environment. Both experiments start from the same policy checkpoint and use the same training configuration. We average results over multiple random seeds. As shown in Table 2, training with *Sword* consistently achieves higher task success rates, demonstrating its stronger utility as a simulator.

5 Ablation Study

5.1 Module-Level Ablation Study

We conduct a 2×2 ablation on LIBERO-Raw and LIBERO-Mixed, comparing the full model with variants that remove SGSA, DLB, or both modules.

As shown in Table 1, each module independently improves all four metrics, while their combination achieves the best overall performance. Figure 5 further shows that adding DLB to SGSA alleviates late-stage overexposure and brightness drift, demonstrating that the two modules provide complementary gains in out-of-distribution robustness and long-horizon consistency.

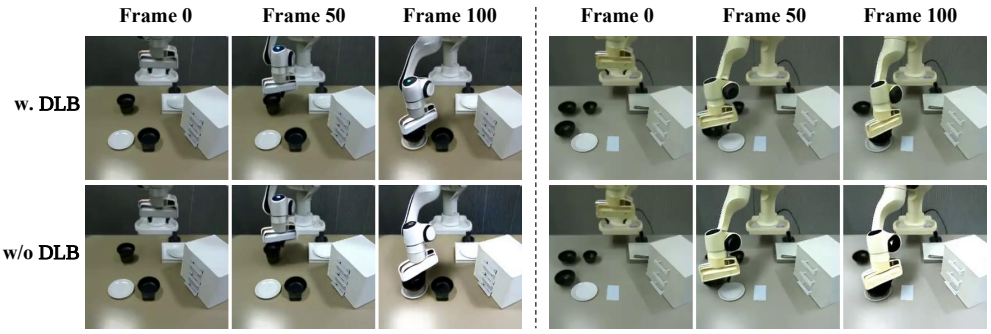


Figure 5: Qualitative DLB ablation comparing the full model with the w/o DLB variant. Without DLB, prediction quality deteriorates at later timesteps, with noticeable inconsistencies in lighting and brightness.

Rank	Read Schedule	Replacement	Write Scope	LPIPS ↓	FID ↓	FVD ↓	FloLPIPS ↓
1	Linear Decay	Direct Replacement	All Chunks	**0.17**	**28.09**	**86.84**	**0.26**
2	Linear Decay	EMA Replacement	Single Chunk	0.19	30.16	86.97	0.28
3	Warmup + Cosine Decay	Direct Replacement	Single Chunk	0.19	29.25	88.44	0.28
4	Linear Decay	EMA Replacement	All Chunks	0.20	31.31	89.51	0.30
5	Warmup + Cosine Decay	EMA Replacement	All Chunks	0.18	29.49	92.01	0.28
6	Linear Decay	Direct Replacement	Single Chunk	0.20	34.25	95.98	0.29
7	Warmup + Cosine Decay	Direct Replacement	All Chunks	0.20	33.89	101.20	0.29
8	Warmup + Cosine Decay	EMA Replacement	Single Chunk	0.21	35.06	103.69	0.31

Table 3: Full-factorial comparison of different cache design strategies. We evaluate two read-cache schedules and four write-cache designs. Results are sorted by FVD in ascending order.

5.2 Comparison of Different Cache Design Strategies

We compare all eight combinations of two read-cache schedules, two replacement rules, and two write-back scopes. As shown in Table 3, **Linear Decay + Direct Replacement + All Chunks** performs best, achieving an FVD of 86.84. Linear decay also occupies the top two positions ranked by FVD, indicating that the read schedule has a larger effect than the write-cache design.

This is likely because the read schedule directly controls the model's conditioning distribution, whereas frequent cache refreshes reduce the differences among write strategies later in training.

6 Conclusion and Limitations

World models used for VLA policy optimization remain vulnerable to visual distribution shifts and exposure bias, leading to hallucinations and error accumulation during long-horizon rollouts. We address these problems with two complementary modules: Structure-Guided Style Augmentation preserves task semantics and geometry while diversifying visual appearance, and Dynamic Latent Bootstrapping reduces the train–inference gap by reusing cached model predictions as context. We further construct LIBERO-Mixed with evaluation style prompts disjoint from those used in training. Experiments on LIBERO show that the resulting world model improves out-of-distribution robustness, long-horizon consistency, prediction quality, and the effectiveness of VLA policy post-training.

Due to limited computational resources, our current evaluation is restricted to LIBERO and does not include other simulated environments or real-world experiments. We leave these broader evaluations, together with VLA post-training in style-transferred environments, to future work.

AI USE STATEMENT

In this work, we used generative AI tools for grammar checking. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Remi Cadene, Simon Aliberts, Francesco Capuano, Michel Aractingi, Adil Zouitine, Pepijn Kooijmans, Jade Choghari, Martino Russi, Caroline Pascal, Steven Palma, et al. Lerobot: An open-source library for end-to-end robot learning. *arXiv preprint arXiv:2602.22818*, 2026.

Jun Cen, Siteng Huang, Yuqian Yuan, Kehan Li, Hangjie Yuan, Chaohui Yu, Yuming Jiang, Jiayan Guo, Xin Li, Hao Luo, et al. Rynnvla-002: A unified vision-language-action and world model. *arXiv preprint arXiv:2511.17502*, 2025a.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, et al. Worldvla: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*, 2025b.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

Duolikun Danier, Fan Zhang, and David Bull. Flolpips: A bespoke video quality metric for frame interpolation. In *2022 Picture Coding Symposium (PCS)*, pp. 283–287. IEEE, 2022.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Zhu Fangqi, Yan Zhengyang, Hong Zicong, Shou Quanxin, Ma Xiao, and Guo Song. Wmpo: World model-based policy optimization for vision-language-action models. *arXiv preprint arXiv:2511.09515*, 2025. URL https://arxiv.org/abs/2511.09515.

Gemini Robotics Team. Gemini Robotics 1.5: Pushing the Frontier of Generalist Robots with Advanced Embodied Reasoning, Thinking, and Motion Transfer. *arXiv e-prints*, art. arXiv:2510.03342, October 2025. doi: 10.48550/arXiv.2510.03342.

Zhong Guan, Haoran Sun, Yongjian Guo, Shuai Di, Xiaodong Bai, Jing Long, Tianyun Zhao, Mingxi Luo, Chen Zhou, Yucheng Guo, et al. Rl-vla 3: Reinforcement learning vla accelerating via full asynchronism. *arXiv preprint arXiv:2602.05765*, 2026.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025.

Yuxin Jiang, Shengcong Chen, Siyuan Huang, Liliang Chen, Pengfei Zhou, Yue Liao, Xindong He, Chiming Liu, Hongsheng Li, Maoqing Yao, and Guanghui Ren. Enerverse-ac: Envisioning embodied environments with action condition. *arXiv preprint arXiv:2505.09723*, 2025.

Zhennan Jiang, Shangqing Zhou, Yutong Jiang, Zefang Huang, Mingjie Wei, Yuhui Chen, Tianxing Zhou, Zhen Guo, Hao Lin, Quanlu Zhang, et al. Wovr: World models as reliable simulators for post-training vla policies with rl. *arXiv preprint arXiv:2602.13977*, 2026.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning*, pp. 2679–2713. PMLR, 2025.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023a.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023b.

Yueen Ma, Zixing Song, Yuzheng Zhuang, Jianye Hao, and Irwin King. A survey on vision-language-action models for embodied ai. *arXiv preprint arXiv:2405.14093*, 2024.

NVIDIA, :, Hassan Abu Alhaija, Jose Alvarez, Maciej Bala, Tiffany Cai, Tianshi Cao, Liz Cha, Joshua Chen, Mike Chen, Francesco Ferroni, Sanja Fidler, Dieter Fox, Yunhao Ge, Jinwei Gu, Ali Hassani, Michael Isaev, Pooya Jannaty, Shiyi Lan, Tobias Lasser, Huan Ling, Ming-Yu Liu, Xian Liu, Yifan Lu, Alice Luo, Qianli Ma, Hanzi Mao, Fabio Ramos, Xuanchi Ren, Tianchang Shen, Xinglong Sun, Shitao Tang, Ting-Chun Wang, Jay Wu, Jiashu Xu, Stella Xu, Kevin Xie, Yuchong Ye, Xiaodong Yang, Xiaohui Zeng, and Yu Zeng. Cosmos-transfer1: Conditional world generation with adaptive multimodal control, 2025. URL https://arxiv.org/abs/2503.14492.

NVIDIA, :, Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, Prithvijit Chattopadhyay, Mike Chen, Yongxin Chen, Yu Chen, Shuai Cheng, Yin Cui, Jenna Diamond, Yifan Ding, Jiaojiao Fan, Linxi Fan, Liang Feng, Francesco Ferroni, Sanja Fidler, Xiao Fu, Ruiyuan Gao, Yunhao Ge, Jinwei Gu, Aryaman Gupta, Siddharth Gururani, Imad El Hanafi, Ali Hassani, Zekun Hao, Jacob Huffman, Joel Jang, Pooya Jannaty, Jan Kautz, Grace Lam, Xuan Li, Zhaoshuo Li, Maosheng Liao, Chen-Hsuan Lin, Tsung-Yi Lin, Yen-Chen Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Yifan Lu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Seungjun Nah, Yashraj Narang, Abhijeet Panaskar, Lindsey Pavao, Trung Pham, Morteza Ramezanali, Fitsum Reda, Scott Reed, Xuanchi Ren, Haonan Shao, Yue Shen, Stella Shi, Shuran Song, Bartosz Stefaniak, Shangkun Sun, Shitao Tang, Sameena Tasmeen, Lyne Tchapmi, Wei-Cheng Tseng, Jibin Varghese, Andrew Z. Wang, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Jiashu Xu, Dinghao Yang, Xiaodong Yang, Haotian Ye, Seonghyeon Ye, Xiaohui Zeng, Jing Zhang, Qinsheng Zhang, Kaiwen Zheng, Andrew Zhu, and Yuke Zhu. World simulation with video foundation models for physical ai, 2026. URL https://arxiv.org/abs/2511.00062.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. URL https://arxiv.org/abs/2408.00714.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Joonghyuk Shin, Zhengqi Li, Richard Zhang, Jun-Yan Zhu, Jaesik Park, Eli Shechtman, and Xun Huang. Motionstream: Real-time video generation with interactive motion controls, 2026. URL https://arxiv.org/abs/2511.01266.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, Simon Alibert, Matthieu Cord, Thomas Wolf, and Remi Cadene. Smolvla: A vision-language-action model for affordable and efficient robotics, 2025. URL https://arxiv.org/abs/2506.01844.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. FVD: A new metric for video generation, 2019. URL https://openreview.net/forum?id=rylgEULtdN.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. Longlive: Real-time interactive long video generation, 2025. URL https://arxiv.org/abs/2509.22622.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024.

Chao Yu, Yuanqing Wang, Zhen Guo, Hao Lin, Si Xu, Hongzhi Zang, Quanlu Zhang, Yongji Wu, Chunyang Zhu, Junhao Hu, et al. Rlinf: Flexible and efficient large-scale reinforcement learning via macro-to-micro flow transformation. *arXiv preprint arXiv:2509.15965*, 2025.

Dapeng Zhang, Jing Sun, Chenghui Hu, Xiaoyan Wu, Zhenlong Yuan, Rui Zhou, Fei Shen, and Qingguo Zhou. Pure vision language action (vla) models: A comprehensive survey. *arXiv preprint arXiv:2509.19012*, 2025.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

Yifan Zhong, Fengshuo Bai, Shaofei Cai, Xuchuan Huang, Zhang Chen, Xiaowei Zhang, Yuanfei Wang, Shaoyang Guo, Tianrui Guan, Ka Nam Lui, et al. A survey on vision-language-action models: An action tokenization perspective. *arXiv preprint arXiv:2507.01925*, 2025.

Chen Zhou, Haoran Sun, Hedan Yang, Jing Long, Junwu Xiong, Luqiao Wang, Mingxi Luo, Qiming Yang, Shuai Di, Song Wang, et al. Thousand-gpu large-scale training and optimization recipe for ai-native cloud embodied intelligence infrastructure. *arXiv preprint arXiv:2603.11101*, 2026.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of The 7th Conference on Robot*

Algorithm 1 Dynamic Latent Bootstrapping (DLB)

Input: Episode dataset $\mathcal{D}$, VAE encoder E, world model f_θ, training steps S
Parameter: Window size $K = 12$, VAE temporal compression ratio $r = 4$

$N \leftarrow |\mathcal{D}|$ {number of episodes}
$L \leftarrow |\mathcal{D}[0]|$ {frames per episode}
$M \leftarrow L - r + 1$ {valid chunk start positions}
$\mathcal{C} \leftarrow \mathrm{InitializeLatentCache}(N, M)$
$p_{\mathcal{C}} \leftarrow \mathrm{InitializeLinearSchedule}(S)$
for $s = 1$ to S **do**
  Sample $e \sim \mathrm{Uniform}\{0, \ldots, N-1\}$
  Sample $k \sim \mathrm{Uniform}\{0, \ldots, L-K\}$
  $(\mathrm{Obs}, \mathrm{Action}) \leftarrow \mathrm{GetWindow}(\mathcal{D}[e], k, K)$
  $(z^{\mathrm{ctx}}, z^{\mathrm{tar}}_{1:2}) \leftarrow E(\mathrm{Obs})$
  Sample $u \sim \mathcal{U}(0, 1)$
  if $u < p_{\mathcal{C}}(s)$ and $\mathcal{C}[e, k]$ is available **then**
    $c \leftarrow \mathcal{C}[e, k]$
  else
    $c \leftarrow z^{\mathrm{ctx}}$
  end if
  $(\hat{z}^{\mathrm{tar}}_{1:2}, \mathcal{L}) \leftarrow \mathrm{ModelStep}(f_\theta, c, z^{\mathrm{tar}}_{1:2}, \mathrm{Action})$
  $\theta \leftarrow \mathrm{UpdateModel}(\theta, \mathcal{L})$
  $\mathcal{C}[e, k + r] \leftarrow \hat{z}^{\mathrm{tar}}_1$
  $\mathcal{C}[e, k + 2r] \leftarrow \hat{z}^{\mathrm{tar}}_2$
end for
return f_θ

Learning, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 06–09 Nov 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

A Dynamic Latent Bootstrapping

Algorithm 1 summarizes the DLB training procedure in latent space. Each sampled 12-frame window is encoded by the VAE into one context latent and two target latents. The cache is indexed by episode and the starting frame of each four-frame chunk: for example, $\mathcal{C}[e, 5]$ stores the latent encoding of Frames 5–8, whereas $\mathcal{C}[e, 6]$ corresponds to Frames 6–9. This latent-space design avoids repeated VAE decoding and re-encoding. A linear scheduler gradually increases the probability of replacing the ground-truth context latent with a cached prediction, and the newly predicted target latents are written back to their corresponding start-frame positions. Here, Action denotes the action sequence aligned with the eight prediction frames. The function $\mathrm{ModelStep}$ abstracts model prediction and training-loss computation.

B Visualization of Style Augmentation Effects

Figures 6 and 7 present four examples that visualize the effects of our structure-guided style augmentation. For each example, six temporally aligned frames are sampled uniformly from the original and transferred videos. The upper row shows the original LIBERO observations, while the lower row shows the corresponding style-augmented observations. In addition to these image examples, we also provide video versions of the examples in the OpenReview Media Supplement.

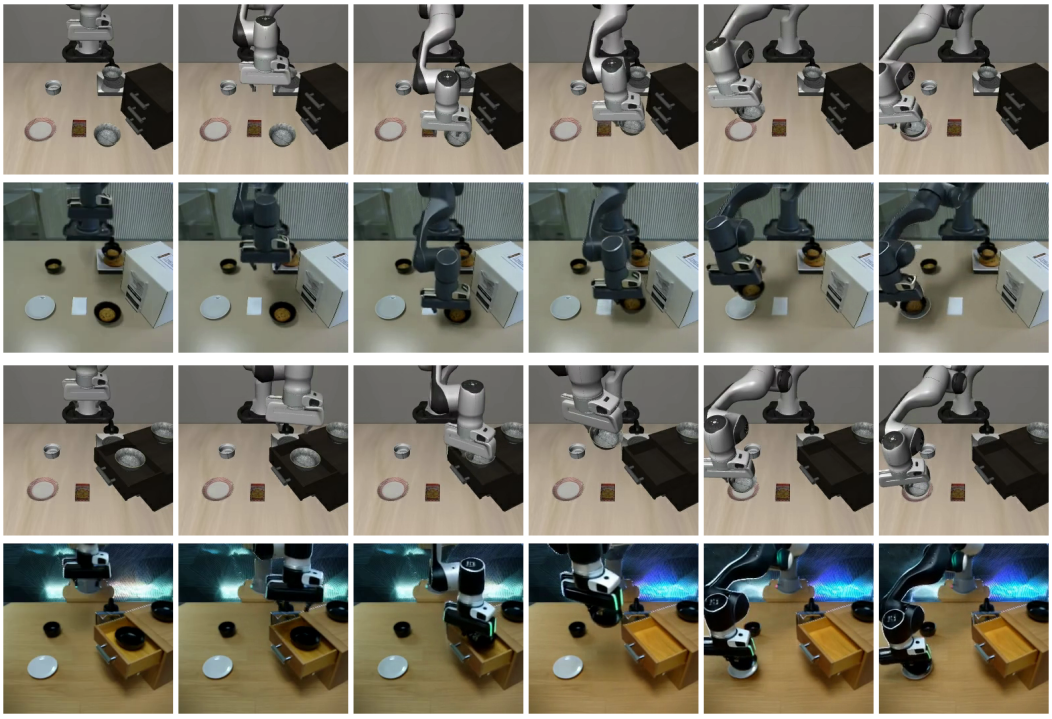


Figure 6: **Effects of structure-guided style augmentation for Examples 0 and 1.** In each comparison, the upper row contains six frames from the original video and the lower row contains their temporally aligned style-transferred counterparts. The augmentation changes visual appearance while preserving the task-relevant scene layout, object states, and robot motion.

C Additional Results

C.1 Additional Qualitative Results under Held-Out Style Prompts

This section supplements the generalization evaluation in the main paper. These comparisons further demonstrate the generalization performance of Sword (Ours) under held-out style prompts.

C.2 Additional Qualitative Results for Physical Fidelity

This section supplements the robustness and fidelity evaluation in the main paper. As shown in Fig. 9, Sword (Ours) better preserves object states, robot–object interactions, and physically plausible temporal dynamics, further demonstrating its ability to model accurate physical evolution during long-horizon prediction.

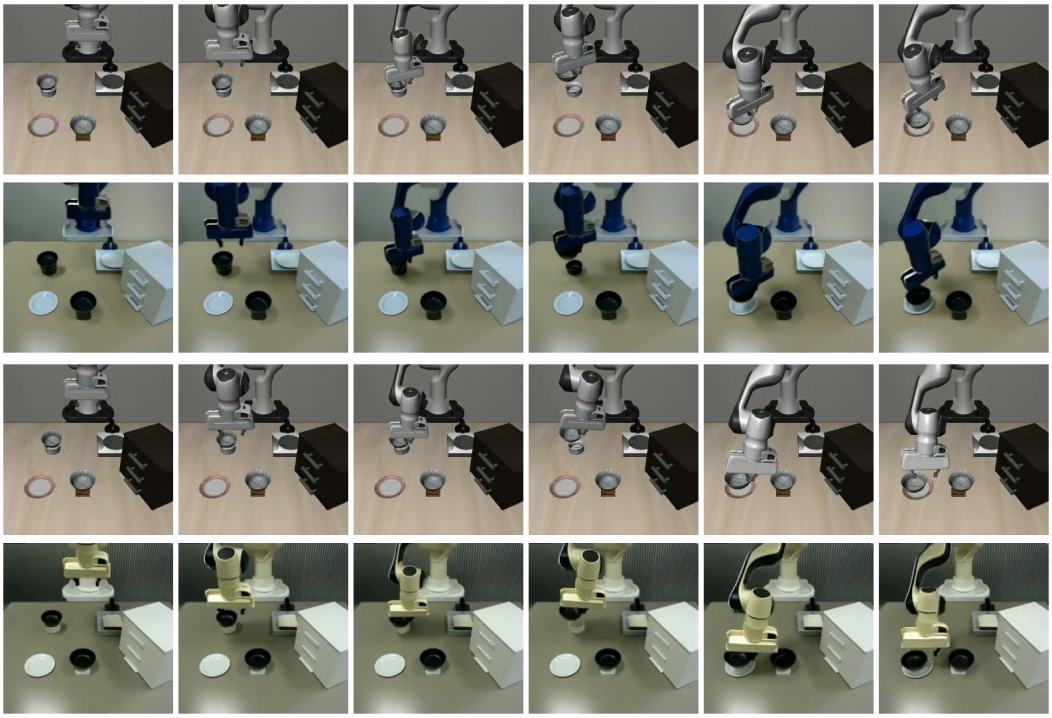


Figure 7: **Effects of structure-guided style augmentation for Examples 2 and 3.** The upper and lower rows show the original and style-transferred frames, respectively. Across different examples, the augmented videos retain the temporal correspondence and task-relevant geometric structure of the original observations.

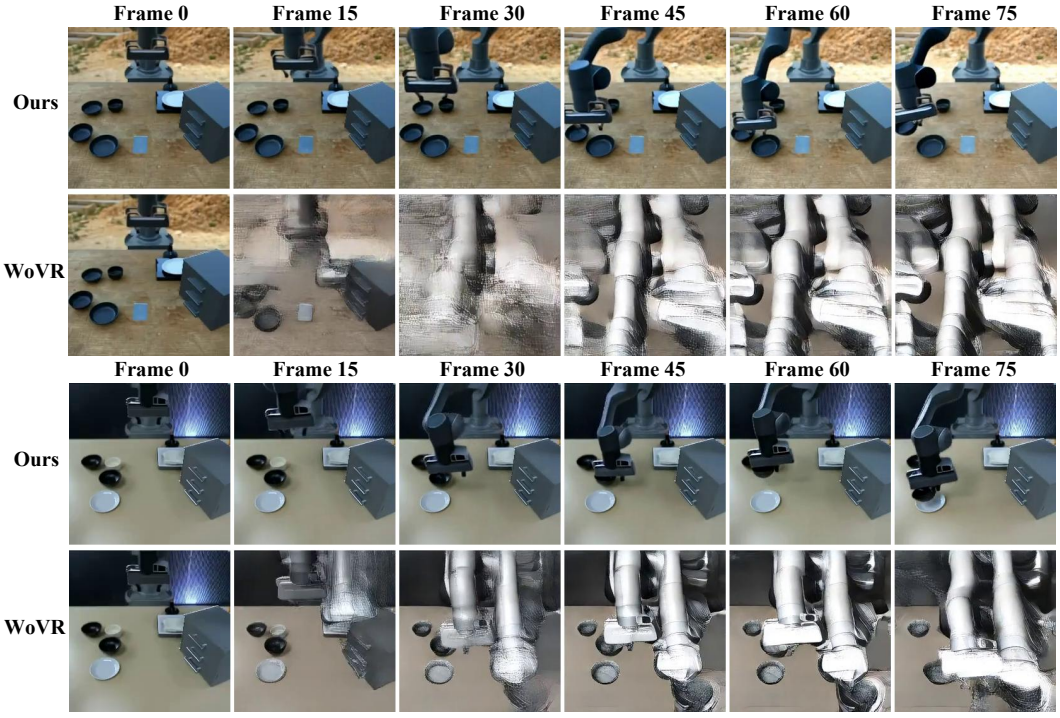


Figure 8: **Additional qualitative comparisons under held-out style prompts.** Sword produces more accurate and temporally consistent predictions than WoVR for evaluation prompts that are disjoint from the training prompt pool.

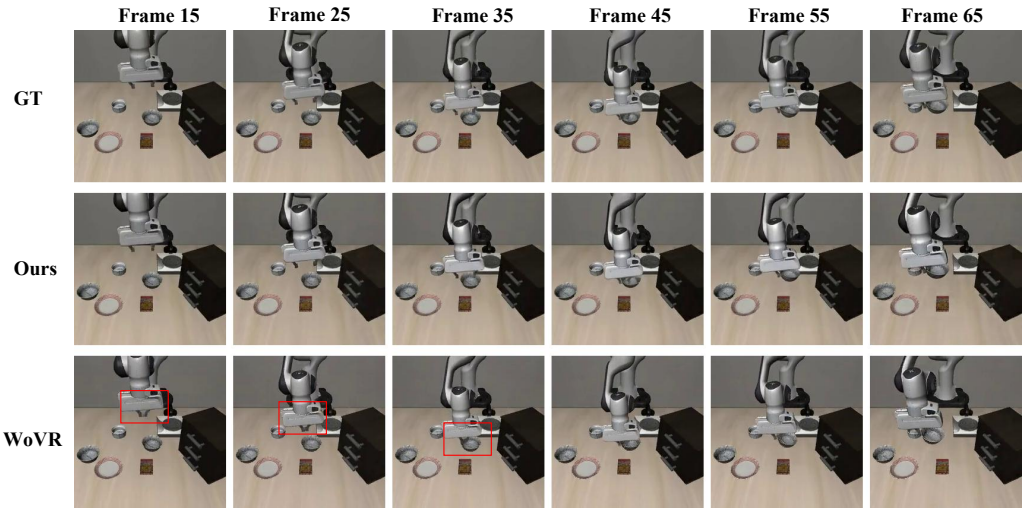


Figure 9: **Additional qualitative comparisons of robustness and fidelity.** We provide supplementary results to compare the robustness and fidelity of predicted video frames. Sword (Ours) better follows the input actions, whereas WoVR produces incorrect gripper states of the robotic arm.