VIVA: A VIDEO-GENERATIVE VALUE MODEL FOR ROBOT REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

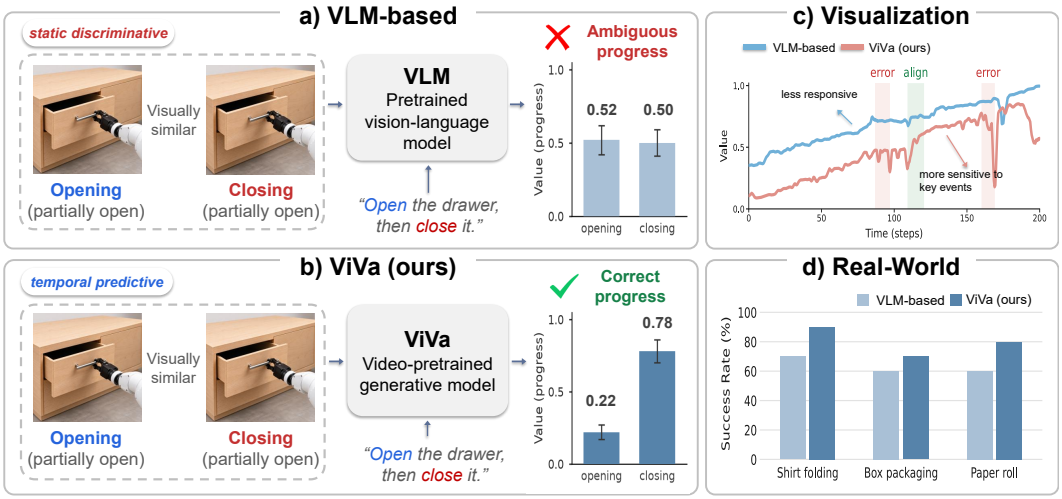


Figure 1: Existing VLM-based value models rely on static discriminative representations, making it difficult to distinguish visually similar states at different stages of task execution (e.g., opening versus closing a drawer) and respond to key events. By leveraging spatiotemporal priors from video generative pretraining, ViVa better captures task progression and execution errors, providing more informative value signals for policy refinement and improving real-world manipulation performance.

ABSTRACT

Vision-language-action (VLA) models have advanced robot manipulation through large-scale pretraining, but real-world deployment remains challenging due to partial observability and delayed feedback. Reinforcement learning addresses this challenge through value functions, which assess task progress and guide policy improvement. However, existing value models built on vision-language models (VLMs) primarily rely on discriminative representations and struggle to capture how interactions evolve over time, limiting reliable value estimation in long-horizon tasks. In this paper, we propose ViVa, a video-generative value model that repurposes a pretrained video generator for robotic value estimation. ViVa jointly models the empirical return and future proprioception, combining direct value supervision with a compact temporal prediction objective. By leveraging spatiotemporal priors from video generative pretraining, ViVa enables event-sensitive value estimation beyond static visual semantics. Across three long-horizon manipulation tasks, ViVa outperforms representative VLM-based value models in progress estimation and sensitivity to execution errors. Integrated into RECAP, ViVa further improves the average real-world success rate from 63.3% to 80.0%, translating improved value estimation into stronger policy performance.

1 INTRODUCTION

Building robots that can perceive, reason, and act in the physical world remains a central challenge in embodied artificial intelligence (Sapkota et al., 2025; Li et al., 2026). Vision-language-action (VLA) models (Zitkovich et al., 2023; Kim et al., 2024; Intelligence et al., 2025b; Li et al., 2025; Team

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.

et al., 2026) have made significant strides by leveraging large-scale pretraining to enable general-purpose manipulation across diverse tasks. Yet success in real-world settings requires reasoning beyond immediate observations: robotic interaction unfolds under partial observability and delayed feedback, where the consequences of decisions may only manifest over extended horizons (Huang et al., 2022; Zitkovich et al., 2023). Learning to connect present behavior with future outcomes thus remains a fundamental challenge for real-world robotics.

This challenge calls for an ability that assesses whether ongoing interaction is progressing toward successful task completion. Such progress awareness allows robots to distinguish beneficial behaviors from undesirable ones and improve through experience. In reinforcement learning (RL) (Sutton et al., 1998), this capability is formalized by the value function, which estimates expected future outcomes and provides a learning signal for policy improvement. Recent VLA frameworks such as $\pi^*_{0.6}$ (Intelligence et al., 2025a) further demonstrate this need: their RL with Experience and Corrections via Advantage-conditioned Policies (RECAP) pipeline relies on a value function for advantage estimation, making reliable value estimation critical for effective policy refinement.

Motivated by this importance, recent works have explored leveraging vision-language models (VLMs) (Chen et al., 2024; Comanici et al., 2025; Bai et al., 2025a; Zhu et al., 2025; Li et al., 2024a; Marafioti et al., 2025) for value estimation, framing value prediction as classification (Intelligence et al., 2025a) or temporal ordering problems (Ma et al., 2024). While promising, these approaches primarily rely on discriminative representations optimized for semantic understanding rather than modeling how interactions evolve over time. As illustrated in Figure 1, visually similar observations, such as opening and closing a drawer, can correspond to different stages of task progress yet remain difficult to distinguish from static visual semantics alone. This limitation hinders VLM-based value models from reliably capturing task progression and execution events.

These limitations motivate a key insight: *value estimation can benefit from representations that anticipate how interactions evolve over time*. Unlike discriminative models primarily trained on static data, video generative models learn spatiotemporal representations by modeling how scenes evolve. Such predictive representations provide a natural foundation for value estimation, which requires relating current observations to future outcomes. Guided by this observation, we explore video generative pretraining as a new foundation for robotic value learning.

In this paper, we propose **Vi**deo-generative **Va**lue model (**ViVa**), a novel approach that repurposes a pretrained video generator for value estimation in robotic reinforcement learning. By leveraging spatiotemporal priors learned from large-scale video corpora, ViVa brings predictive representations into value learning. Given current multi-view observations, robot proprioception, and a task instruction, ViVa jointly predicts the empirical return and future proprioceptive state. The return provides direct value supervision, while future proprioception serves as a compact auxiliary target for modeling robot-centric temporal evolution. Jointly modeling these targets within a video-pretrained backbone incorporates robot-centric temporal supervision into value learning, supporting event-sensitive value estimation for downstream policy refinement.

We conduct extensive experiments across three long-horizon manipulation tasks to evaluate both value estimation quality and downstream policy improvement. ViVa accurately captures task progression and responds sensitively to milestones and execution errors, while generalizing to novel objects beyond the training distribution. Across offline evaluations, ViVa shows strong overall performance, particularly in event-sensitive value estimation. When integrated into RECAP, these improved value estimates further lead to consistent gains in real-world policy performance, demonstrating the effectiveness of ViVa for downstream policy refinement.

We highlight the main contributions of this paper below:

- We identify video generative pretraining as a promising foundation for robotic value estimation, providing spatiotemporal priors beyond static discriminative representations.
- We introduce ViVa, a video-generative value model that jointly models empirical return and future proprioception within a shared video-pretrained representation.
- ViVa achieves strong performance across three long-horizon manipulation tasks, with high sensitivity to execution errors, generalization to novel objects, and consistent gains in real-world policy refinement.

2 Related Work

2.1 Value Functions in Robot Learning

Value functions are central to reinforcement learning for robotics, providing learning signals under delayed and sparse feedback (Sutton et al., 1998; Ross et al., 2011). Prior works have applied reinforcement learning to robotic manipulation, spanning offline Q-learning (Kalashnikov et al., 2018; Levine et al., 2020; Mandlekar et al., 2020; Huang et al., 2025), autonomous real-world interaction (Sharma et al., 2023; Mendonca et al., 2023; Luo et al., 2024; Lampe et al., 2024), and end-to-end value-guided policy optimization (Zhai et al., 2025; Ghasemipour et al., 2025).

To enable reinforcement learning for VLA models (Team et al., 2024; O'Neill et al., 2024; Kim et al., 2024; Li et al., 2024b; Cheang et al., 2024; Liu et al., 2024), recent works have explored pretrained vision-language models and representations for value or reward estimation (Frans et al., 2025; Ma et al., 2024; 2023). For instance, GVL (Ma et al., 2024) directly queries VLMs to assess task progress by framing value prediction as temporal ordering over shuffled video frames. TopReward (Chen et al., 2026) instead leverages VLM token probabilities as zero-shot reward signals. $\pi^*_{0.6}$ (Intelligence et al., 2025a) trains a dedicated VLM-based value model with dense supervision as a core component of the RECAP pipeline for advantage-conditioned policy refinement.

Despite their formulations, these approaches primarily rely on discriminative vision-language representations optimized for semantic understanding rather than explicitly modeling how interactions evolve over time. In contrast, we explore video-generative representations as a foundation for robotic value estimation, leveraging spatiotemporal priors learned from large-scale video pretraining.

2.2 Video Generation Models for Robot Manipulation

Video generation models (Zheng et al., 2024; Yang et al., 2024; Kong et al., 2024; Blattmann et al., 2023) are designed to model temporal evolution, commonly using diffusion Transformers (Peebles & Xie, 2023; Bao et al., 2023) conditioned on language (Singer et al., 2022; Villegas et al., 2022; Blattmann et al., 2023) or visual inputs (Ceylan et al., 2023; Qi et al., 2023). Their ability to capture spatiotemporal dependencies has motivated increasing interest in video-generative models for robotics.

In robot manipulation, video generation has been used for planning with world models (Zhou et al., 2024), where predicted futures support decision making. It has also been incorporated into policy learning through inverse dynamics (Du et al., 2023b; Yang et al., 2023), goal-conditioned policies (Du et al., 2023a; Zhang et al., 2025), and joint video-action generation (Cheang et al., 2024; Wu et al., 2023; Ye et al., 2026b;a). More recently, video generation has also been explored for human-to-robot transfer by synthesizing human-object interaction videos (Bharadhwaj et al., 2024; Zhao et al., 2025; Kareer et al., 2025).

Despite these advances, existing works primarily use video generation to support planning, action generation, or policy learning. We investigate a complementary role in *value estimation*. ViVa repurposes a pretrained video-generative model to jointly model the empirical return and future proprioception, bringing video-pretrained spatiotemporal representations into robotic value learning.

3 Method

ViVa is a video-generative value model for robotic reinforcement learning. It combines trajectory-return supervision with future-proprioception prediction in a shared video-pretrained model. We first formulate the value-modeling problem, then introduce the joint prediction design, and finally describe its generative training objective.

3.1 Value Modeling Problem

At each time step t, the robot receives multi-view RGB observations $\mathbf{o}_t = \{\mathbf{o}_t^i\}_{i=1}^M$, proprioception $\mathbf{q}_t$, and a task instruction ℓ. We denote the robot observation by $\mathbf{x}_t = (\mathbf{o}_t, \mathbf{q}_t)$. For a policy π, the

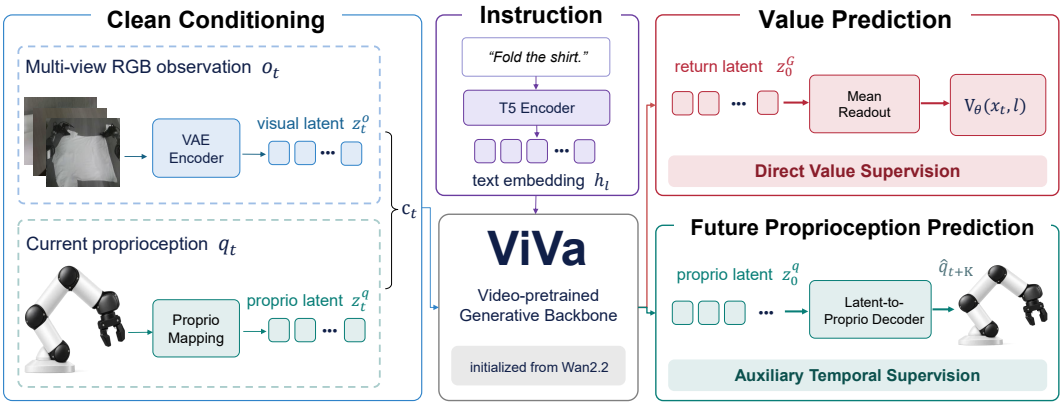


Figure 2: Overview of ViVa. Current multi-view RGB observations and robot proprioception are encoded as clean conditioning, while the task instruction provides text conditioning. A shared video-pretrained generative backbone, initialized from Wan2.2, jointly models the empirical return and future proprioception. The return latent is averaged to obtain the value estimate, while future proprioception provides auxiliary temporal supervision.

task-conditioned value and empirical return are defined as

$$V^\pi(\mathbf{x}_t, \ell) = \mathbb{E}_\pi[G_t \mid \mathbf{x}_t, \ell], \qquad G_t = \sum_{k=t}^{T} r_k, \tag{1}$$

where r_k denotes the reward at step k and T is the episode termination time. We learn a task-conditioned value estimator from robot trajectories, using empirical returns G_t as supervision.

To construct this supervision signal, we define the reward to encode both temporal progress and final task outcome. For an episode of length T, we define

$$r_t = \begin{cases} \frac{1}{T}, & t < T, \\ 0, & t = T \text{ and success}, \\ 1, & t = T \text{ and failure}. \end{cases} \tag{2}$$

The resulting empirical return G_t jointly captures temporal progress and final task outcome, while the length-normalized step-wise reward provides consistent supervision across episodes of varying durations.

3.2 Joint Value and Future-Proprioception Modeling

ViVa builds on Wan2.2 (Wan et al., 2025), a pretrained video generative model, as the foundation for value modeling. Video generative pretraining provides representations shaped by temporal evolution and spatiotemporal dependencies, offering a useful prior for modeling the consequences of ongoing robot interactions.

We formulate ViVa as a joint conditional generative model over the empirical return and the robot's future proprioceptive state:

$$p_\theta\left(G_t, \mathbf{q}_{t+K} \mid \mathbf{x}_t, \ell\right), \tag{3}$$

where K denotes a fixed prediction horizon.

We use future proprioception as a compact predictive target describing the robot's configuration at a future horizon. Conditioned on the current observation, predicting $\mathbf{q}_{t+K}$ encourages the shared model to capture information about subsequent changes in robot configuration. This predictive objective complements return supervision within the video-pretrained model, while current multi-view observations provide the visual context for joint prediction. The two targets share model parameters: G_t supervises value learning, while $\mathbf{q}_{t+K}$ provides an auxiliary temporal prediction supervision.



Figure 3: Detection of subtle manipulation errors by ViVa. The predicted progress score drops clearly at execution errors, illustrating ViVa's sensitivity to fine-grained manipulation failures.

3.3 Joint Learning Objective

We instantiate the joint generative model through latent flow matching. Each camera view is independently encoded by the pretrained VAE, while the current proprioceptive state is mapped to a latent frame and the instruction is encoded by T5:

$$\mathbf{z}_t^{o_i} = E_{\mathrm{VAE}}(\mathbf{o}_t^i), \qquad \mathbf{z}_t^q = \phi_q(\mathbf{q}_t), \qquad \mathbf{h}_\ell = E_{\mathrm{T5}}(\ell). \tag{4}$$

The visual and proprioceptive latents form the clean observation context $\mathbf{c}_t$, while $\mathbf{h}_\ell$ provides text conditioning. Details of the latent sequence organization are provided in Appendix A.

We represent the two prediction targets in the latent format of the video backbone:

$$\mathbf{z}_0^q = \phi_q(\mathbf{q}_{t+K}), \qquad \mathbf{z}_0^G = \phi_G(G_t), \qquad \mathbf{z}_0 = \left[\mathbf{z}_0^q, \mathbf{z}_0^G\right], \tag{5}$$

where ϕ_q repeat-pads the proprioceptive vector to a latent frame and ϕ_G broadcasts the scalar return to the same latent shape. No separate prediction heads are introduced. During training, the conditioning context remains clean while the joint target is corrupted. Given $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\tau \sim \mathcal{U}[0, 1]$,

$$\mathbf{z}_\tau = (1 - \tau)\mathbf{z}_0 + \tau\boldsymbol{\epsilon}, \tag{6}$$

with target velocity $\boldsymbol{\epsilon} - \mathbf{z}_0$. Conditioned on $\mathbf{c}_t$ and $\mathbf{h}_\ell$, the shared video backbone predicts the velocity field. Let u_θ^m denote the predicted velocity component for target $m \in \{G, q\}$. We define

$$\mathcal{L}_{\mathrm{FM}}^m = \mathbb{E}_{\mathcal{D},\boldsymbol{\epsilon},\tau}\left[\left\|u_\theta^m(\mathbf{z}_\tau, \tau; \mathbf{c}_t, \mathbf{h}_\ell) - (\boldsymbol{\epsilon}^m - \mathbf{z}_0^m)\right\|_2^2\right], \tag{7}$$

where the superscript m denotes the corresponding target modality. The joint objective is

$$\mathcal{L}_{\mathrm{joint}} = \lambda_{\mathrm{val}}\mathcal{L}_{\mathrm{FM}}^G + \lambda_{\mathrm{prop}}\mathcal{L}_{\mathrm{FM}}^q. \tag{8}$$

The weights λ_{val} and λ_{prop} control the contributions of return and future-proprioception supervision. Both target components are jointly processed by the same backbone and share model parameters, enabling return and future-state supervision to contribute to a shared predictive representation.

At inference, the generated return latent is converted to a scalar value estimate by averaging all its entries:

$$V_\theta(\mathbf{x}_t, \ell) = \mathrm{Mean}\left(\hat{\mathbf{z}}_0^G\right). \tag{9}$$

The resulting value estimate provides a dense value signal for the current robot state.

4 Experiments

We evaluate ViVa on three real-world manipulation tasks: shirt folding, box packaging, and paper roll organization. Our experiments evaluate both the quality of the learned value function and its effectiveness for downstream policy refinement. Task details, evaluation datasets, and implementation details are provided in Appendix B.1, B.2, and B.3. Additional results are provided in Appendix C.

4.1 Evaluation Metrics

For consistent evaluation, we convert each method's predictions to a normalized progress score S_t. Details of the score normalization are provided in Appendix B.3.

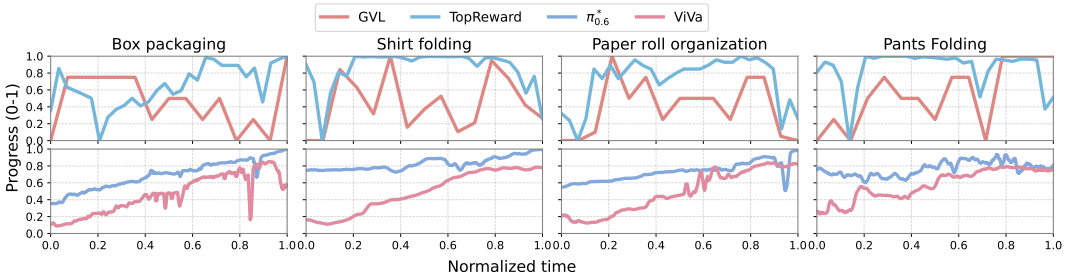


Figure 4: Value trajectory visualization across all tasks. Trajectories for $\pi^*_{0.6}$ and ViVa are smoothed by 20%. ViVa tracks task progress with clear responses at key events and maintains sensitivity on the unseen object, demonstrating stronger generalization.

Event Response Score (ERS). For each evaluation episode, we annotate a set of key event frames $\mathcal{K}$, where each frame k is assigned a direction $d_k \in \{+1, -1\}$: $+1$ for milestones and -1 for execution errors. Let $S^{(k)}_{\text{before}}$ and $S^{(k)}_{\text{after}}$ denote the mean scores within windows of w frames immediately before and after event k. We define

$$\text{ERS}_k = d_k \cdot \frac{S^{(k)}_{\text{after}} - S^{(k)}_{\text{before}}}{S^{(k)}_{\text{before}} + \epsilon}, \tag{10}$$

where ϵ is a small constant for numerical stability. We report *Milestone Sensitivity (MS)* and *Error Sensitivity (ES)* as the average ERS over milestone and error frames, respectively. A positive ERS indicates a response in the expected direction, while its magnitude reflects response strength.

Event Detection Rate (EDR). We consider an event detected if $\text{ERS}_k > 0$ and report *Milestone Detection Rate (MDR)* and *Error Detection Rate (ErrDR)* as the fractions of detected milestones and execution errors:

$$\text{MDR} = \frac{N^+_{\text{milestone}}}{N_{\text{milestone}}}, \qquad \text{ErrDR} = \frac{N^+_{\text{error}}}{N_{\text{error}}}, \tag{11}$$

where N^+ and N denote the numbers of detected and total events of the corresponding type.

4.2 Value Model Evaluation

Evaluation setup. We benchmark ViVa against recent state-of-the-art value models on three tasks: shirt folding, box packaging, and paper roll organization. GVL (Ma et al., 2025) and TopReward (Chen et al., 2026) leverage VLMs for sparse-sampled value estimation, while $\pi^*_{0.6}$ (Intelligence et al., 2025a) and ViVa are trained on $\sim$20 hours of mixed demonstration data from all three tasks. We use 50 held-out episodes per task and manually annotate milestone and error frames for evaluation. See Sec. B.2 for dataset details.

Quantitative comparison. Table 1 first evaluates global value quality using Pearson and Spearman correlations. The densely supervised $\pi^*_{0.6}$ and ViVa substantially outperform GVL and TopReward, with ViVa leading on shirt folding and roll organization and $\pi^*_{0.6}$ on box packaging.

Table 1: Comparison with representative value models. Dense-supervised methods achieve stronger correlations, with ViVa leading on two of three tasks.

Method	Shirt		Box		Paper	
	Pearson	Spearman	Pearson	Spearman	Pearson	Spearman
Sparse-sampled						
GVL	0.276	0.263	-0.081	-0.114	-0.099	-0.096
TopReward	0.118	0.090	0.503	0.499	0.318	0.241
Dense-supervised						
$\pi^*_{0.6}$	0.740	0.720	**0.946**	**0.945**	0.926	0.936
ViVa	**0.984**	**0.982**	0.868	0.850	**0.952**	**0.948**

Global correlation, however, does not capture how value predictions respond to fine-grained local task events, which is important for distinguishing trajectories with different progress and outcomes. We therefore further evaluate event-level sensitivity and detection. As shown in Tables 2 and 3, ViVa shows substantially stronger responses to milestones and execution errors across different tasks. On shirt folding, for example, $\pi^*_{0.6}$ produces negative MS and ES, whereas ViVa achieves positive responses and improves MDR from 0.407 to 0.990 and ErrDR from 0.500 to 1.000. Figure 3 further illustrates ViVa's sensitivity to subtle execution errors during manipulation.

Table 2: Event-sensitivity comparison. MS and ES measure directional response quality. ViVa achieves substantially higher sensitivity than the VLM baseline across all three tasks.

Method	Backbone	Shirt		Box		Paper	
		MS	ES	MS	ES	MS	ES
$\pi^*_{0.6}$	VLM	-0.266	-0.299	0.013	0.006	0.023	0.028
ViVa	Video-gen	**0.095**	**0.012**	**0.034**	**0.026**	**0.039**	**0.055**

Table 3: Event-detection comparison. MDR and ErrDR measure event detection rates. ViVa improves most metrics, with particularly strong gains in error detection.

Method	Backbone	Shirt		Box		Paper	
		MDR	ErrDR	MDR	ErrDR	MDR	ErrDR
$\pi^*_{0.6}$	VLM	0.407	0.500	**0.765**	0.506	0.704	0.667
ViVa	Video-gen	**0.990**	**1.000**	0.733	**0.612**	**0.762**	**0.778**

Trajectory visualization. Figure 4 visualizes the predicted progress-score trajectories on the three in-domain tasks. GVL and TopReward produce noisy or relatively flat trajectories that often fail to reflect task progression. $\pi^*_{0.6}$ captures the overall trend, but its responses to fine-grained events are often weak or delayed. In contrast, ViVa exhibits clearer score increases at milestones and decreases at execution errors, consistent with its stronger event-level performance. Additional trajectory visualizations are provided in Appendix C.1.

4.3 Object Generalization

Evaluation setup. We further evaluate object generalization on an out-of-domain pants folding task, where the manipulated object is unseen during training but shares similar manipulation dynamics with shirt folding. We use 50 held-out episodes annotated with milestone frames to evaluate whether the learned value function can transfer progress estimation beyond the training objects.

Quantitative comparison. As shown in Table 4, ViVa consistently outperforms $\pi^*_{0.6}$ on the unseen object. Beyond higher Pearson and Spearman correlations, ViVa improves MS from 0.016 to 0.037 and MDR from 0.397 to 0.643. These results indicate that ViVa better transfers both global progress estimation and milestone-level discrimination to unseen objects.

Table 4: Object generalization on the out-of-domain pants folding task. ViVa outperforms $\pi^*_{0.6}$ across all metrics.

Method	Pearson	Spearman	MS	MDR
$\pi^*_{0.6}$	0.740	0.721	0.016	0.397
ViVa	**0.819**	**0.799**	**0.037**	**0.643**

One possible explanation is that video-pretrained representations capture spatiotemporal interaction patterns that are shared across visually different but dynamically related objects. For example, although shirts and pants differ in appearance, their folding processes involve similar deformation and manipulation patterns, providing transferable temporal structure for value estimation.

Trajectory visualization. The pants folding results in Figure 4 provide a complementary trajectory-level view. $\pi^*_{0.6}$ captures the overall progress trend but responds weakly to several milestones, whereas ViVa exhibits clearer score changes at these events. This is consistent with the quantitative results, showing that ViVa's event-sensitive value estimation transfers to unseen objects. Additional trajectory visualizations are provided in Appendix C.1.

4.4 Real-Robot Experiments

Evaluation setup. We evaluate whether the learned value function can improve downstream policy refinement on the three real-world tasks. We compare the imitation learning baselines Gigabrain-0 (Team et al., 2025) and $\pi_{0.5}$ (Intelligence et al., 2025b) with two RECAP (Intelligence et al., 2025a) variants initialized from the same Gigabrain-0 policy: one using the VLM-based value model and the other using ViVa. Both variants undergo two rounds of policy rollouts under the same RECAP pipeline. Implementation details are provided in Appendix B.3.

Main results. As shown in Table 5, both RECAP variants improve over the imitation learning baselines, demonstrating the benefit of policy refinement with learned value signals. More importantly, under the same initial policy and RECAP pipeline, replacing the VLM-based value model with ViVa improves the average success rate from 63.3% to 80.0%, with

Table 5: Real-robot policy performance. RECAP (ViVa) outperforms both imitation learning baselines and RECAP (VLM) across all three tasks.

Method	Shirt (%)	Box (%)	Paper (%)	Avg (%)
Imitation Learning				
$\pi_{0.5}$	60.0	40.0	40.0	46.7
Gigabrain-0	50.0	50.0	20.0	40.0
Reinforcement Learning				
RECAP (VLM)	70.0	60.0	60.0	63.3
RECAP (ViVa)	**90.0**	**70.0**	**80.0**	**80.0**

Table 6: Backbone ablation. Both video-generative backbones outperform the VLM, with Wan2.2 achieving the strongest results.

Backbone	Family	Params.	MS ↑	ES ↑	MDR ↑	ErrDR ↑
Qwen2.5-VL	VLM	3B	-0.266	-0.299	0.407	0.500
CogVideoX	Video Gen.	2B	0.101	-0.071	0.662	0.500
Wan2.2	Video Gen.	5B	**0.137**	**-0.026**	**1.000**	0.500

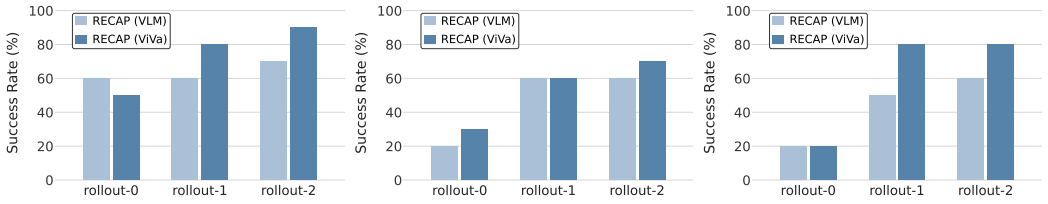
Table 7: Video pretraining ablation. Video-pretrained initialization improves milestone and error detection, supporting the benefit of spatiotemporal priors.

Initialization	MS ↑	ES ↑	MDR ↑	ErrDR ↑
Random	**0.117**	-0.524	0.770	0.500
Video Pretrained	0.095	**0.012**	**0.990**	**1.000**

consistent gains across all three tasks. This controlled comparison shows that ViVa's improved value estimation translates into stronger downstream policy performance.

Rollout-wise performance evolution. Figure 5 further compares policy performance across rollout rounds. ViVa yields larger gains after the first round, with success rates increasing from 50% to 80% on shirt folding and from 20% to 80% on paper roll organization. In the second round, the VLM-based variant largely plateaus, while the ViVa-based variant continues to improve.

This faster improvement suggests that ViVa provides more informative value signals for policy refinement. Its stronger sensitivity to milestones and execution errors better distinguishes rollout trajectories with different progress and outcomes, providing more discriminative supervision for RECAP. Together with the offline evaluation, these results connect ViVa's event-sensitive value estimation to more effective real-world policy refinement.



(a) Shirt folding. (b) Box packaging. (c) Paper roll organization.

Figure 5: Rollout-wise performance evolution across three manipulation tasks. RECAP (ViVa) achieves larger performance gains across rollout rounds and stronger final performance.

4.5 Ablation Studies

We ablate the key design choices of ViVa, including the backbone family, video pretraining, prediction targets, prediction horizon, and loss weight. Unless otherwise specified, all experiments are conducted on shirt folding under the same evaluation protocol.

Effect of Backbone Family. We first compare a VLM backbone with two video-generative backbones under the same value-modeling formulation. As shown in Table 6, replacing Qwen2.5-VL (Bai et al., 2025b) with CogVideoX (Yang et al., 2025) improves MS from -0.266 to 0.101, ES from -0.299 to -0.071, and MDR from 0.407 to 0.662. Notably, CogVideoX achieves these gains with fewer parameters (2B vs. 3B), suggesting that the improvement is not simply due to model scale. Wan2.2 further improves MS to 0.137, ES to -0.026, and MDR to 1.000.

The consistent gains across two video-generative backbones suggest that the benefit is not specific to a particular model. This supports video-generative representations as a more effective foundation for event-sensitive value estimation.

Effect of Video Pretraining. We isolate the effect of video pretraining by comparing identical ViVa architectures with random and pretrained initialization. As shown in Table 7, pretraining improves ES from -0.524 to 0.012, MDR from 0.770 to 0.990, and ErrDR from 0.500 to 1.000, while MS slightly decreases from 0.117 to 0.095.

These results show that ViVa benefits not only from the video-generative architecture, but also from video pretraining. The particularly large gains in error-sensitive metrics suggest that pretrained spatiotemporal representations provide useful priors for capturing fine-grained interaction changes.

Effect of Prediction Targets. We examine how different prediction targets affect value estimation. As shown in Table 8, adding future-image prediction improves MS from 0.047 to 0.121 and MDR

from 0.778 to 0.982, but decreases ES from 0.008 to -0.031. Further adding future proprioception achieves the best MS (0.134), but provides limited gains in error-sensitive metrics.

In contrast, predicting value with future proprioception achieves the best ES, MDR, and ErrDR of 0.012, 0.990, and 1.000, respectively. Future images are high-dimensional targets that contain visual variation weakly related to task value, which may dilute supervision from task-critical state changes. Future proprioception instead provides a compact, behavior-related target focused on robot state evolution, providing more targeted temporal supervision for value learning. This leads to stronger event-sensitive value estimation while avoiding the additional computation required for future-image generation.

Table 8: Ablation on prediction targets for value modeling. Future proprioception provides the most effective auxiliary supervision for event-sensitive value estimation.

Value	Future Image	Future Proprio.	MS ↑	ES ↑	MDR ↑	ErrDR ↑
✓	✗	✗	0.047	0.008	0.778	0.500
✓	✓	✗	0.121	-0.031	0.982	0.500
✓	✓	✓	**0.134**	-0.018	0.946	0.500
✓	✗	✓	0.095	**0.012**	**0.990**	**1.000**

This design remains complementary to video pretraining. The pretrained backbone provides rich spatiotemporal representations learned from video, while future-proprioception prediction adapts them toward robot-centric temporal evolution. ViVa thus leverages video generation for its pretrained spatiotemporal prior without requiring future-image generation as the downstream objective.

Effect of Prediction Horizon. We further study the prediction horizon K for future proprioception. As shown in Table 9, a short horizon of $K = 10$ achieves high MDR but fails to detect execution errors, with ErrDR dropping to 0.000. Increasing the horizon to $K = 50$ yields the best ES of 0.014 and ErrDR of 1.000, while maintaining a high MDR of 0.990.

Further increasing the horizon to $K = 75$ slightly improves MS but degrades ES, MDR, and ErrDR. These results suggest that an intermediate horizon better balances meaningful future state changes with prediction reliability. We therefore use $K = 50$ as the default setting.

Table 9: Ablation on the future-proprioception prediction horizon K. An intermediate horizon provides the best balance between milestone and error-sensitive value estimation.

Horizon K	MS ↑	ES ↑	MDR ↑	ErrDR ↑
10	0.089	-0.002	**0.995**	0.000
25	0.109	0.012	0.990	0.500
50	0.095	**0.014**	0.990	**1.000**
75	**0.113**	-0.002	0.968	0.500

Table 10: Ablation on the future-proprioception loss weight λ_{prop}. Moderate future-proprioception supervision yields the strongest error-sensitive value estimation.

λ_{prop}	MS ↑	ES ↑	MDR ↑	ErrDR ↑
0.10	**0.107**	-0.007	**0.998**	0.000
0.25	0.088	-0.010	0.988	0.500
0.50	0.095	**0.013**	0.990	**1.000**
0.75	0.102	0.012	0.995	1.000

Effect of Loss Weight. We study the effect of the future-proprioception loss weight λ_{prop}, while fixing $\lambda_{\text{val}} = 1$. As shown in Table 10, MS and MDR remain relatively stable across different loss weights, whereas error-sensitive metrics are more sensitive to λ_{prop}. With a small weight of 0.10, ErrDR drops to 0.000, indicating that weak future-proprioception supervision provides limited benefit for error detection.

Increasing λ_{prop} to 0.50 yields the best ES of 0.013 and ErrDR of 1.000. Further increasing the weight to 0.75 provides no additional improvement. These results suggest that sufficient future-proprioception supervision is important for capturing fine-grained interaction events, while overly increasing its contribution is unnecessary. We therefore use $\lambda_{\text{prop}} = 0.50$ by default.

5 Conclusion

In this work, we introduced ViVa, a video-generative value model that leverages video-pretrained spatiotemporal representations for robotic value learning. ViVa jointly models empirical return and future proprioception, combining direct value supervision with auxiliary temporal prediction. Across three real-world manipulation tasks, ViVa provides more reliable value estimates and improves downstream policy performance when integrated into RECAP.

Limitation. Due to resource constraints, we have not yet pretrained the value model at scale. Exploring larger-scale value pretraining and broader cross-task generalization remains an important direction for future work.

AI USE STATEMENT

AI assistance was used for language editing, consistency checks, and manuscript presentation. The authors reviewed and verified all generated content and retain full responsibility for the technical claims, experimental results, and conclusions presented in this work.

ETHICS STATEMENT

This work studies value modeling and reinforcement learning for robotic manipulation. All experiments are conducted on robot manipulation tasks in controlled laboratory settings and do not involve human subjects or personal data. The collected robot trajectories are used solely for training and evaluating the proposed models. We are not aware of any conflicts of interest in this work.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the model formulation, training objective, evaluation metrics, baselines, and experimental protocol in the main paper. Appendix A describes the latent sequence organization and multimodal conditioning used by ViVa, while Appendix B.3 provides implementation details including training settings, prediction horizon, loss weights, and inference configuration. Appendix B.2 describes the training and evaluation datasets, and Appendix B.1 details the real-world manipulation tasks and evaluation setup. We plan to release the code and evaluation resources upon acceptance.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025b. URL https://arxiv.org/abs/2502.13923.

Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22669–22679, 2023.

Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation. *arXiv preprint arXiv:2409.16283*, 2024.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Duygu Ceylan, Chun-Hao P Huang, and Niloy J Mitra. Pix2video: Video editing using image diffusion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 23206–23217, 2023.

Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, et al. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14455–14465, 2024.

Shirui Chen, Cole Harrison, Ying-Chun Lee, Angela Jin Yang, Zhongzheng Ren, Lillian J Ratliff, Jiafei Duan, Dieter Fox, and Ranjay Krishna. Topreward: Token probabilities as hidden zero-shot rewards for robotics. *arXiv preprint arXiv:2602.19313*, 2026.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Yilun Du, Mengjiao Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Brian Ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B Tenenbaum, et al. Video language planning. *arXiv preprint arXiv:2310.10625*, 2023a.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *Advances in neural information processing systems*, 36:9156–9172, 2023b.

Kevin Frans, Seohong Park, Pieter Abbeel, and Sergey Levine. Diffusion guidance is a controllable policy improvement operator. *arXiv preprint arXiv:2505.23458*, 2025.

Seyed Kamyar Seyed Ghasemipour, Ayzaan Wahid, Jonathan Tompson, Pannag Sanketi, and Igor Mordatch. Self-improving embodied foundation models. *arXiv preprint arXiv:2509.15155*, 2025.

Dongchi Huang, Zhirui Fang, Tianle Zhang, Yihang Li, Lin Zhao, and Chunhe Xia. Co-rft: Efficient fine-tuning of vision-language-action models through chunked offline reinforcement learning. *arXiv preprint arXiv:2508.02219*, 2025.

Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International conference on machine learning*, pp. 9118–9147. PMLR, 2022.

Physical Intelligence, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Kevin Black, Ken Conley, Grace Connors, James Darpinian, Karan Dhabalia, Jared DiCarlo, et al. $\pi^*_{0.6}$: a vla that learns from experience. *arXiv preprint arXiv:2511.14759*, 2025a.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025b.

Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, pp. 651–673. PMLR, 2018.

Simar Kareer, Karl Pertsch, James Darpinian, Judy Hoffman, Danfei Xu, Sergey Levine, Chelsea Finn, and Suraj Nair. Emergence of human to robot transfer in vision-language-action models. *arXiv preprint arXiv:2512.22414*, 2025.

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.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Thomas Lampe, Abbas Abdolmaleki, Sarah Bechtle, Sandy H Huang, Jost Tobias Springenberg, Michael Bloesch, Oliver Groth, Roland Hafner, Tim Hertweck, Michael Neunert, et al. Mastering stacking of diverse shapes with large-scale iterative reinforcement learning on real robots. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7772–7779. IEEE, 2024.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024a.

Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024b.

Wei Li, Renshan Zhang, Rui Shao, Jie He, and Liqiang Nie. Cogvla: Cognition-aligned vision-language-action model via instruction-driven routing & sparsification. *arXiv preprint arXiv:2508.21046*, 2025.

Xinghang Li, Peiyan Li, Long Qian, Minghuan Liu, Dong Wang, Jirong Liu, Bingyi Kang, Xiao Ma, Xinlong Wang, Di Guo, et al. What matters in building vision–language–action models for generalist robots. *Nature Machine Intelligence*, pp. 1–15, 2026.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.

Jianlan Luo, Zheyuan Hu, Charles Xu, You Liang Tan, Jacob Berg, Archit Sharma, Stefan Schaal, Chelsea Finn, Abhishek Gupta, and Sergey Levine. Serl: A software suite for sample-efficient robotic reinforcement learning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 16961–16969. IEEE, 2024.

Yecheng Jason Ma, Vikash Kumar, Amy Zhang, Osbert Bastani, and Dinesh Jayaraman. Liv: Language-image representations and rewards for robotic control. In *International Conference on Machine Learning*, pp. 23301–23320. PMLR, 2023.

Yecheng Jason Ma, Joey Hejna, Chuyuan Fu, Dhruv Shah, Jacky Liang, Zhuo Xu, Sean Kirmani, Peng Xu, Danny Driess, Ted Xiao, et al. Vision language models are in-context value learners. In *The Thirteenth International Conference on Learning Representations*, 2024.

Yecheng Jason Ma, Joey Hejna, Chuyuan Fu, Dhruv Shah, Jacky Liang, Zhuo Xu, Sean Kirmani, Peng Xu, Danny Driess, Ted Xiao, et al. Vision language models are in-context value learners. In *International Conference on Learning Representations*, volume 2025, pp. 33984–34009, 2025.

Ajay Mandlekar, Fabio Ramos, Byron Boots, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Dieter Fox. Iris: Implicit reinforcement without interaction at scale for learning control from offline robot manipulation data. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4414–4420. IEEE, 2020.

Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, et al. Smolvlm: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025.

Russell Mendonca, Shikhar Bahl, and Deepak Pathak. Alan: Autonomously exploring robotic agents in the real world. *arXiv preprint arXiv:2302.06604*, 2023.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903. IEEE, 2024.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Qiaosong Qi, Le Zhuo, Aixi Zhang, Yue Liao, Fei Fang, Si Liu, and Shuicheng Yan. Diffdance: Cascaded human motion diffusion model for dance generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 1374–1382, 2023.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Ranjan Sapkota, Yang Cao, Konstantinos I Roumeliotis, and Manoj Karkee. Vision-language-action (vla) models: Concepts, progress, applications and challenges. *arXiv preprint arXiv:2505.04769*, 2025.

Archit Sharma, Ahmed M Ahmed, Rehaan Ahmad, and Chelsea Finn. Self-improving robots: End-to-end autonomous visuomotor reinforcement learning. *arXiv preprint arXiv:2303.01488*, 2023.

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

GigaBrain Team, Angen Ye, Boyuan Wang, Chaojun Ni, Guan Huang, Guosheng Zhao, Haoyun Li, Jie Li, Jiagang Zhu, Lv Feng, et al. Gigabrain-0: A world model-powered vision-language-action model. *arXiv preprint arXiv:2510.19430*, 2025.

GigaBrain Team, Boyuan Wang, Chaojun Ni, Guan Huang, Guosheng Zhao, Hao Li, Jie Li, Jindi Lv, Jingyu Liu, Lv Feng, et al. Gigabrain-0.5 m*: a vla that learns from world model-based reinforcement learning. *arXiv preprint arXiv:2602.12099*, 2026.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description. *arXiv preprint arXiv:2210.02399*, 2022.

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.

Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. *arXiv preprint arXiv:2312.13139*, 2023.

Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 1(2):6, 2023.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Angen Ye, Boyuan Wang, Chaojun Ni, Guan Huang, Guosheng Zhao, Hao Li, Hengtao Li, Jie Li, Jindi Lv, Jingyu Liu, et al. Gigaworld-policy: An efficient action-centered world–action model. *arXiv preprint arXiv:2603.17240*, 2026a.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026b.

Shaopeng Zhai, Qi Zhang, Tianyi Zhang, Fuxian Huang, Haoran Zhang, Ming Zhou, Shengzhe Zhang, Litao Liu, Sixu Lin, and Jiangmiao Pang. A vision-language-action-critic model for robotic real-world reinforcement learning. *arXiv preprint arXiv:2509.15937*, 2025.

Hongyin Zhang, Pengxiang Ding, Shangke Lyu, Ying Peng, and Donglin Wang. Gevrm: Goal-expressive video generation model for robust visual manipulation. *arXiv preprint arXiv:2502.09268*, 2025.

Hongxiang Zhao, Xingchen Liu, Mutian Xu, Yiming Hao, Weikai Chen, and Xiaoguang Han. Taste-rob: Advancing video generation of task-oriented hand-object interaction for generalizable robotic manipulation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 27683–27693, 2025.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

Siyuan Zhou, Yilun Du, Jiaben Chen, Yandong Li, Dit-Yan Yeung, and Chuang Gan. Robodreamer: Learning compositional world models for robot imagination. *arXiv preprint arXiv:2404.12377*, 2024.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pp. 2165–2183. PMLR, 2023.

A Latent Sequence Organization

ViVa organizes heterogeneous robot observations and prediction targets in the latent-frame format of the pretrained video backbone. Let each latent frame have the spatial shape expected by the video model. For multi-view observations, each camera view is independently encoded by the pretrained VAE, producing $\{\mathbf{z}_t^{o_i}\}_{i=1}^{M}$. The current proprioceptive state $\mathbf{q}_t$ is mapped to a latent frame through ϕ_q, which repeat-pads the proprioceptive vector and reshapes it to the same latent-frame format.

The clean observation context is organized as

$$\mathbf{c}_t = \left[\mathbf{z}_{\text{blank}}, \mathbf{z}_t^q, \mathbf{z}_t^{o_1}, \ldots, \mathbf{z}_t^{o_M}\right], \tag{12}$$

where $\mathbf{z}_{\text{blank}}$ denotes a blank placeholder frame. All frames in $\mathbf{c}_t$ remain clean during training and provide the observation conditioning for joint prediction. The task instruction is encoded separately by T5 as $\mathbf{h}_\ell$ and is provided to the video backbone through text conditioning rather than being inserted as an additional latent frame.

The two prediction targets are represented using the same latent-frame format:

$$\mathbf{z}_0^q = \phi_q(\mathbf{q}_{t+K}), \qquad \mathbf{z}_0^G = \phi_G(G_t), \tag{13}$$

where ϕ_q applies the same repeat-padding operation used for proprioception, and ϕ_G broadcasts the scalar empirical return G_t over a latent frame. The joint target sequence is then

$$\mathbf{z}_0 = \left[\mathbf{z}_0^q, \mathbf{z}_0^G\right]. \tag{14}$$

Thus, each prediction target occupies one latent frame, and no separate prediction heads are introduced.

During flow-matching training, only the target sequence is corrupted, while the observation context remains clean. Specifically, the corrupted target $\mathbf{z}_\tau$ is concatenated with the clean conditioning prefix,

$$\left[\mathbf{c}_t, \mathbf{z}_\tau\right], \tag{15}$$

and processed by the shared video-pretrained backbone conditioned on $\mathbf{h}_\ell$. This organization allows visual observations, proprioception, empirical return, and future proprioception to be modeled within a unified latent-frame representation.

B Experimental Setup

B.1 Evaluation Tasks

Our quantitative evaluations and real-robot experiments are conducted on three long-horizon manipulation tasks: shirt folding, box packaging, and paper roll organization. We summarize each task below, with illustrations in Figure 6.

Shirt folding. This task evaluates dual-arm coordination for manipulating highly deformable textiles. The robot must first take a garment from a basket, flatten it on the table, execute a folding sequence (sleeves and sides inward, then longitudinal and cross folds), and return the folded shirt to the basket. Success requires neat folding and placement in the basket within 300 seconds. Failure occurs if entanglement damages the garment, fold collapses, or return is improper.

Box packaging. This task evaluates long-horizon dual-arm coordination through a multi-stage manipulation sequence. The robot picks a target item, places it into a partially formed cardboard box, then folds the side flaps and closes the lid. Success requires the item to be fully enclosed in a structurally sound box with all tabs interlocked within 300 seconds. Failure occurs if the item is dropped, the box is damaged, or the box cannot be fully sealed.

Paper roll organization. This task evaluates precise, multi-stage manipulation of flexible paper. The robot must grasp and tear off a single sheet, discarding it into a receptacle, then collaboratively rewind the remaining loose end until it is flush with the roll. Finally, a sealing sticker is applied to secure the end. Success requires completing the tear, disposal, and sealing within 240 seconds. Excessive tearing or failure to secure the sticker is recorded as a failure.



Figure 6: Illustration of the three real-world tasks. For each task, we show the initial state (left), key intermediate stages (middle), and the final successful state (right).

B.2 Evaluation Dataset

We use 50 held-out episodes per task for evaluation. Milestone and error frames are pre-defined per task and manually annotated for each episode. Milestone frames mark key task stages and appear in every episode; error frames, by contrast, are annotated only for episodes in which the corresponding failure mode actually occurs. Figure 7 reports the error type distribution for each task.

Shirt folding. Failures are concentrated in the flattening stage, with Failed to spread flat (45.5%) and Poor spreading quality (27.3%) being the most frequent, followed by Grasp failure and Basket placement failure. The key challenge lies in reliably flattening a highly deformable textile without residual wrinkles or misalignment.

Box packaging. Failures span multiple sub-tasks, dominated by Side panel operation (44.5%) and Box operation (26.7%), with additional errors in Ear operation, Object A/B operation, and Lay-down. The difficulty stems from the need to coordinate a sequential, multi-stage assembly where each step must be executed precisely before the next can begin.

Paper roll organization. The majority of failures involve tearing and retrieval: Label and tear operation (50.0%) and Roll retrieval deviation (35.7%), along with Roll retrieval slip and Roll displacement. The core challenge is fine-grained manipulation of flexible paper, where even small deviations in tearing angle or retrieval trajectory can ruin the product.

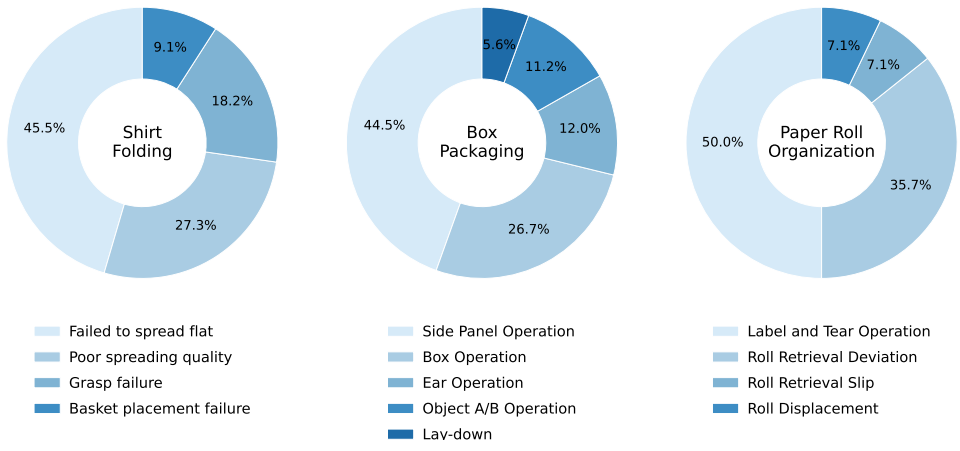


Figure 7: Error type distribution for shirt folding, box packaging, and paper roll organization. Each pie chart shows the relative frequency of failure modes, based on 50 held-out episodes per task.

B.3 Implementation Details

For the VLM-based value model, we follow the design of $\pi^*_{0.6}$ (Intelligence et al., 2025a) and formulate value estimation as a 201-way classification problem over discretized return bins. The VLM-based baseline and our ViVa-based variant are trained within the same RECAP pipeline for controlled comparison.

We conduct two rounds of policy rollouts. For each task, the first round collects 150 successful and 50 failed episodes, while the second round collects 100 successful and 50 failed episodes. The collected rollout data are combined with demonstration data from all three tasks for value-model training. For real-robot evaluation, we conduct 10 trials per task for each policy. All models are trained for one epoch with a batch size of 192.

For ViVa, we set the future-proprioception prediction horizon to $K = 50$ for all tasks. The loss weights are set to $\lambda_{\text{prop}} = 0.5$ and $\lambda_{\text{val}} = 1.0$. During inference, we use one denoising step with DDIM sampling.

For evaluation, we normalize the predicted values and reverse their direction to obtain the progress score used throughout the experiments:

$$S_t = 1 - \text{Norm}(V_\theta(\mathbf{x}_t, \ell)) \,. \tag{16}$$

This transformation only standardizes the score convention used for evaluation and visualization; model training is performed directly on the original return supervision G_t.

All experiments are conducted on 8 NVIDIA A800 GPUs.

C Additional Experimental

C.1 Value Curve Visualization

Figures 8, 9, and 10 provide qualitative comparisons of the predicted value trajectories on the three in-domain manipulation tasks. Across tasks, ViVa exhibits clearer local responses to key execution events. Its predicted score typically increases around successfully completed milestones and decreases when execution errors occur, producing value trajectories that more closely reflect the underlying task progress. In contrast, the VLM-based value model often shows weaker, delayed, or even directionally incorrect responses to local events. For example, on shirt folding, the VLM-based prediction decreases around several correctly completed milestones, while ViVa maintains more consistent progress-aware responses.

Figure 11 further evaluates this behavior on the out-of-domain pants folding task. Although the manipulated object is unseen during training, ViVa continues to respond clearly to successive folding milestones, whereas the VLM-based model exhibits weaker and less stable local variations. This qualitative evidence is consistent with the quantitative object-generalization results, suggesting that ViVa's event-sensitive value estimation transfers beyond the training objects.

C.2 Efficiency Comparison

Table 11 compares the computational cost of the VLM-based baseline and different video-generative value-model variants. The VLM-based baseline incurs the highest training cost of 6 GPU·days and an inference latency of 0.32 seconds per frame.

For the video-generative variants, all models use the same Wan2.2 backbone and differ only in their prediction targets. The value-only variant achieves the lowest cost, requiring 3 GPU·days for training and 0.11 seconds per frame at inference, but performs worse than the full ViVa model in value estimation. Adding future-image prediction increases the computational cost to 5 GPU·days and 0.25 seconds per frame. In comparison, ViVa with future-proprioception prediction requires 4 GPU·days and 0.18 seconds per frame.

Overall, future proprioception provides a more computationally efficient auxiliary target than future-image prediction while retaining the benefits of temporal supervision.

(a) Box packaging

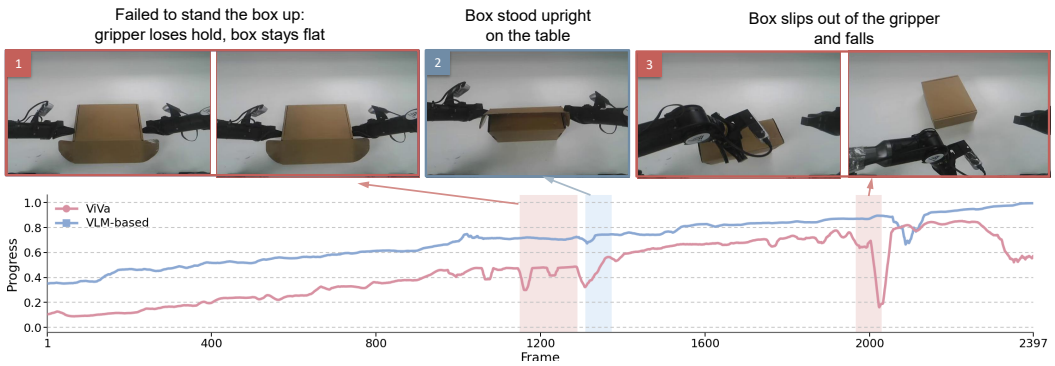


Figure 8: Value visualization on the box packaging task. ViVa responds clearly to both task milestones and execution errors, including successful upright placement and box slippage, whereas the VLM-based value model shows weaker response

(b) Paper roll organization

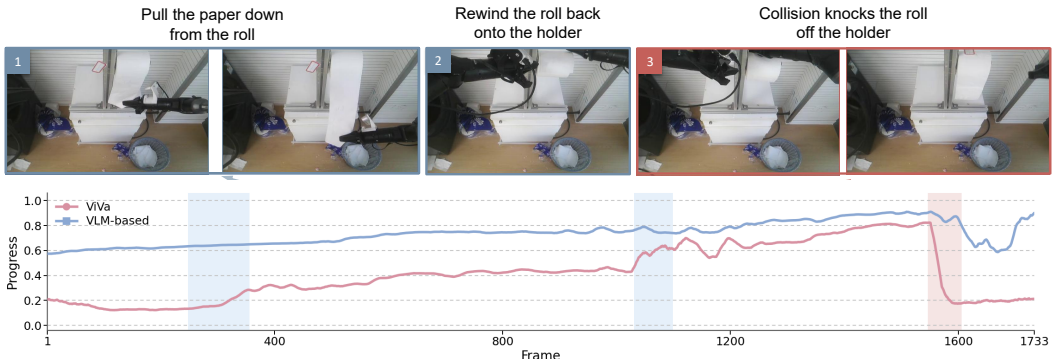


Figure 9: Value visualization on the paper roll organization task. ViVa responds clearly to both task milestones and execution errors, including successful rewinding of the roll and collision-induced failure, while the VLM-based value model exhibits weaker responses to these local events.

(c) Shirt folding

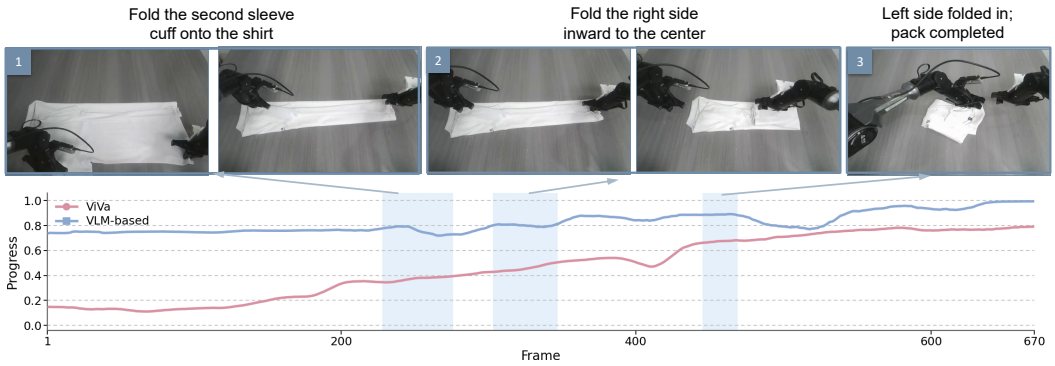


Figure 10: Value visualization on the shirt folding task. ViVa responds consistently to successive task milestones, whereas the VLM-based value model shows value decreases at several correctly completed key frames, revealing less reliable event-level progress estimation.

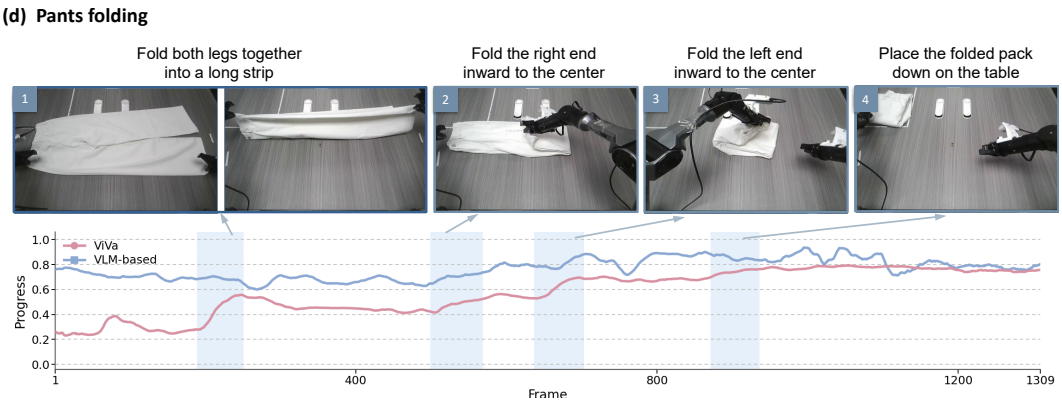


Figure 11: Value visualization on the out-of-domain pants folding task. ViVa responds consistently to successive folding milestones on the unseen object, whereas the VLM-based value model exhibits weaker and less stable local responses. This illustrates the transfer of ViVa's event-sensitive value estimation beyond the training objects.

Table 11: Computational cost comparison. Training cost is reported in GPU·days and inference cost in seconds per frame.

Method	Backbone	Training Cost (GPU·days)	Inference Cost (s/frame)
$\pi^*_{0.6}$	VLM	6	0.32
Value only	VG	3	0.11
Value + Future Proprio.	VG	4	0.18
Value + Future Image	VG	5	0.25