WARM: A WORLD ACTION RECTIFICATION MODEL UNIFYING VLA AND WAM THROUGH A PLANNING-RECTIFICATION ARCHITECTURE

Anonymous authors
Paper under double-blind review

ABSTRACT

Embodied manipulation policies have recently evolved into two dominant families: Vision-Language-Action (VLA) models, which directly map visual observations and language instructions to robot actions, and World-Action Models (WAMs), which model the video dynamics induced by these actions. We show that these families are complementary rather than competing: across 50 bimanual tasks, each exhibits distinct signature strengths. Inspired by the complementary roles of the cerebrum and cerebellum, we propose **WARM** (**W**orld **A**ction **R**ectification **M**odel), which uses a VLA to propose actions and a WAM to evaluate and refine them at inference time. Systematic ablations show that WAM-based video prediction provides a transferable source of physical reasoning, improving a pretrained VLA without replacing its semantic plan. Among four coupling schemes, `clamp` performs best by first predicting task motion under a fixed VLA proposal and then refining the action based on the resulting video–action context. On RoboTwin 2.0, WARM, instantiated with LingBot-VLA and Motus, achieves 90.3% under the Clean setting and 90.7% under the Random setting, outperforming both baselines.

1 INTRODUCTION

Robot manipulation has increasingly centered around two dominant modeling paradigms. On one side, *Vision-Language-Action (VLA)* models learn an end-to-end mapping from language instructions and egocentric visual observations to low-level actions Black et al. (2025); Intelligence et al. (2025); Wu et al. (2026c;a). Their strength lies in *semantic planning*: they can reliably translate an instruction into a sequence of high-level intentions, such as reaching, grasping, placing, and transporting, while generalizing across diverse language descriptions. On the other side, *World Action Models (WAMs)* jointly model future video observations and action chunks, thereby internalizing a physics-grounded forward model of how the world responds to actions (Bi et al., 2026; Ye et al., 2026b; Yuan et al., 2026; Team et al., 2026b; Luo et al., 2026b; Robotics, 2026). Their strength lies in *physical planning*: they can constrain motions to maintain valid contact relationships, account for occlusions, and obey physical dynamics.

A natural question is whether one family simply dominates the other. Prior work has often implicitly framed the two paradigms as competitors. However, the fact that different studies report advantages for VLAs and WAMs on different types of tasks suggests that their strengths may be complementary. We conduct a quantitative analysis of both families on a common 50-task bimanual manipulation benchmark, comparing a group of VLAs with a group of WAMs rather than a single pair (Section 4.1). Our results show that the two families are indeed complementary rather than competing: although their performances are moderately correlated, each family exhibits distinct task-level advantages.

Relative to a VLA, a WAM introduces an additional video-prediction module. This raises a central question: can the physical reasoning enabled by video prediction be transferred to a VLA without explicitly training an action–video alignment objective? If so, this would indicate that video prediction provides a distinct and transferable capability, rather than merely improving action generation through joint training. We draw inspiration from the complementary roles of cortical and cerebel-

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.

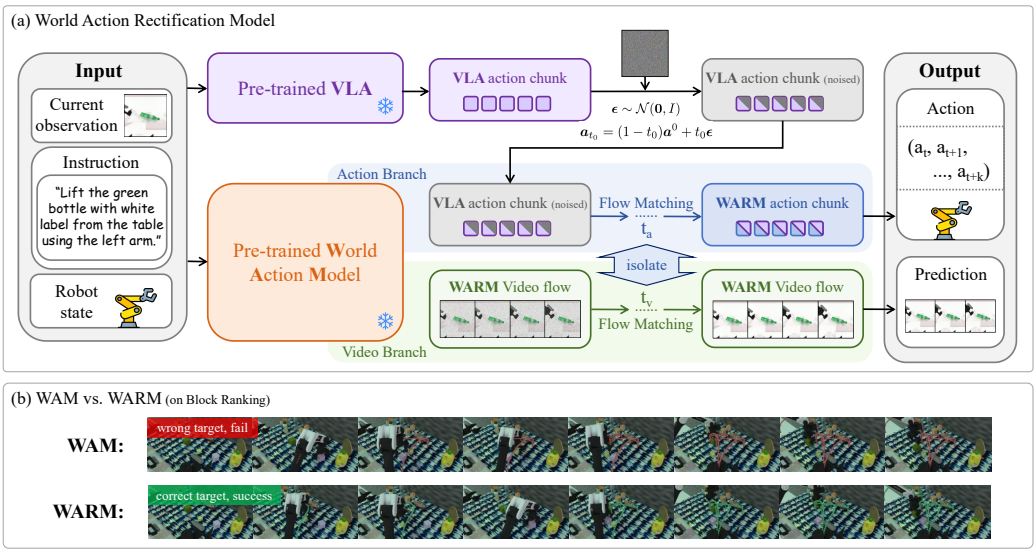


Figure 1: **WARM: world action rectification.** **(a)** At each control step, a frozen VLA (marked by the snowflake icon) maps the language instruction, the current observation, and the robot state into a nominal action chunk $\boldsymbol{a}^0$, which is perturbed to an intermediate flow-matching time t_0 via $\boldsymbol{a}_{t_0} = (1 - t_0)\boldsymbol{a}^0 + t_0\boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, I)$. A frozen pre-trained World Action Model (also snowflake-marked) refines this noised prior through coupled video–action flow matching: the video branch predicts the future task motion while the action branch produces the rectified chunk $\boldsymbol{a}^\star$, which is executed and fed back in a closed loop. **(b)** On the `blocks_ranking_size` task (Blocks Ranking Size), WAM commits to the wrong target block and fails, whereas WARM grounds the correct block and succeeds on the same seed.

lar systems within recurrent cerebrum–cerebellum circuits, where the cerebral cortex (cerebrum) is broadly involved in goal formation, task planning, action selection, and language-mediated behavior, and the cerebellum is associated with predictive sensorimotor control: it learns internal forward models that estimate the sensory consequences of motor commands and supports rapid error-based corrections (Wolpert et al., 1998; Ito, 2008; Middleton & Strick, 2000; Buckner, 2013). This provides an analogy for our framework: the VLA provides language-conditioned task planning, while the WAM uses its video-conditioned forward model to predict the consequences of candidate actions and refine their physical execution. And thus, we propose **WARM (World Action Refinement Model)**, a simple and effective framework that combines the complementary strengths of VLAs and WAMs through *plan-then-rectify*. Given an action chunk proposed by a VLA, WARM uses the WAM as a physics-aware action refiner: it preserves the VLA's semantic plan while correcting the proposed motion to better agree with the predicted visual consequences and physical dynamics. This design allows the WAM to contribute its physical reasoning capability without replacing the VLA's language-conditioned planning.

Realizing this plan-then-rectify principle requires deciding how the WAM's video and action branches should interact during refinement. Because the WAM is trained with independently sampled video and action timesteps, different relative denoising schedules remain within the training distribution while inducing distinct refinement behaviors. We therefore formulate and systematically evaluate a family of video–action coupling schemes, in which the action branch follows, lags behind, or is temporarily held relative to the video branch. Our best-performing strategy, `clamp`, uses an asymmetric *plan-then-rectify* procedure with initialization by a forward stochastic differential equation (SDE). The VLA proposal is first perturbed to an intermediate noise level using this forward SDE. During the early denoising stage, the video branch generates the predicted task motion while the action branch remains fixed at this SDE-perturbed proposal. This allows the WAM to predict the visual consequences of the proposed plan without prematurely overriding its semantic intent. In the later denoising stage, the action branch is released and refined jointly with the video branch

toward a motion consistent with the predicted object contacts, occlusions, trajectories, and physical dynamics. Thus, the WAM corrects the physical execution of the VLA plan while preserving its language-conditioned task semantics. Our contributions are threefold:

- We present a systematic statistical analysis showing that VLAs and WAMs are *complementary* rather than competing: they exhibit distinct task-level strengths, and a per-task oracle combining the two strictly outperforms either family alone (Section 4.1). The comparison is X-VLA and LingBot-VLA against Motus, Fast-WAM, and LingBot-VA.
- We propose **WARM**, a cerebrum–cerebellum-inspired *plan-then-rectify* framework that combines VLA planning with WAM-based physical refinement. By evaluating several video–action coupling schemes, we identify the asymmetric `clamp` strategy, which first predicts the consequences of the VLA plan and then refines its execution (Section 3).
- We analyze WAM-based refinement and show that video prediction provides a transferable capability for correcting the physical execution of VLA-generated plans. Semantic planning must remain with the VLA, while WARM combines the VLA plan with WAM-based refinement to outperform both standalone models (Section 4).

2 Related Work

VLAs & WAMs. Vision-Language-Action (VLA) models Black et al. (2025); Intelligence et al. (2025); Bjorck et al. (2025); Team et al. (2025); Zhai et al. (2025); Jiang et al. (2025); Yang et al. (2025); Cheang et al. (2025) have emerged as a promising paradigm for generalist robot control. Recent efforts improve their generalization and embodiment coverage through heterogeneous robot data, human-centric priors, embodiment-aware architectures, unified action spaces, and predictive or reasoning–action objectives Wang et al. (2026a); Zhai et al. (2025); Ye et al. (2026a); Luo et al. (2025a; 2026a; 2025b); Lin et al. (2026a); Liu et al. (2026); MoE designs further support specialization across skills, sensors, motion phases, and embodiments Yu et al. (2026); Li et al. (2026b); Zhang et al. (2026a); You et al. (2026); Du et al. (2025). In parallel, World-Action Models (WAMs) jointly model future videos and action chunks, enabling robot policies to account for the predicted physical consequences of their actions (Bi et al., 2026; Ye et al., 2026b; Yuan et al., 2026; Team et al., 2026b; Luo et al., 2026b; Robotics, 2026). They differ from action-conditioned world models, which primarily predict future observations for externally specified actions, and from value or reward models that evaluate predicted outcomes (Gao et al., 2026; Guo et al., 2026b; Team et al., 2026a; Wang et al., 2026b).

WM for VLA model training. Recent post-training methods use world models to provide supervision beyond expert action imitation. WMPO optimizes VLA policies with world-model-based objectives (Zhu et al., 2026); RISE enables policy self-improvement through a compositional world model (Yang et al., 2026); Reinforcing Action Policies by Prophesying learns from future-conditioned feedback (Zhang et al., 2025); and NORA-1.5 derives preference rewards from world-model and action predictions (Hung et al., 2025). These methods use world models primarily as auxiliary sources of supervision for training VLA policies. In contrast, WARM investigates whether a pretrained world model can directly improve the policy itself by refining its action outputs at inference time.

WM for VLA model policies. Another line of work integrates world modeling more directly into VLA policies. Unified approaches such as 3D-VLA, DreamVLA, and UniVLA augment action prediction with generative or latent world modeling to improve spatial, temporal, and causal reasoning (Zhen et al., 2024; Zhang et al., 2026b; Bu et al., 2025b). Other methods jointly train or iteratively improve VLA policies and action-conditioned video models (Cen et al., 2025; Guo et al., 2026a), while World Pilot uses WAM-derived priors to steer the visual representations and action generation of a VLA (Lin et al., 2026b). These studies show the benefits of combining language-conditioned planning with predictive world knowledge. However, they primarily focus on unified modeling, training-time co-improvement, or feature-level guidance. In contrast, WARM investigates whether video prediction provides a distinct and transferable capability beyond improving action generation through joint training.

3 METHOD

WARM combines a VLA planner with a WAM-based physical rectifier through a *plan-then-rectify* pipeline (Figure 1). Given a language instruction ℓ and the current observation o_t, the VLA first generates a semantic action proposal

$$\boldsymbol{a}^0 = f_{\mathrm{VLA}}(\ell, o_t) \in \mathbb{R}^{T\times d}. \tag{1}$$

The proposal is then perturbed to an intermediate noise level using the forward stochastic differential equation (SDE) and refined by Motus through its coupled video–action flow-matching process, producing the final action chunk $\boldsymbol{a}^\star$.

Section 3.1 describes how the VLA proposal is perturbed by the forward SDE to initialize the refinement process, while Section 3.2 presents several strategies for coupling the video and action branches of the WAM during denoising to obtain the refined action.

3.1 CEREBRAL PLANNING: VLA-GUIDED ACTION INITIALIZATION

In WARM, the VLA plays the role of a semantic planner. Given the current visual–language context, it produces a nominal action chunk that specifies the task-level intention of the robot. We treat this action chunk as a cerebral-like plan: it determines *what* to do, but need not be physically optimal in its low-level execution. Let $\boldsymbol{a}^0$ denote the action chunk generated by the VLA and $\boldsymbol{c}$ denote the current visual–language context. We perturb the VLA proposal to an intermediate noise level $t_0 \in (0, 1]$ using the Gaussian transition induced by the forward SDE:

$$\boldsymbol{a}_{t_0} = (1 - t_0)\boldsymbol{a}^0 + t_0\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, I). \tag{2}$$

This initialization provides the WAM with a noisy neighborhood around the VLA plan. A small t_0 preserves the semantic plan more faithfully, whereas a large t_0 gives the WAM greater freedom to modify its physical execution. The WAM provides two coupled velocity fields,

$$\frac{d\boldsymbol{z}_t}{dt} = \boldsymbol{f}_\theta^v\left(\boldsymbol{z}_t, \boldsymbol{a}_t, \boldsymbol{c}; t_v, t_a\right), \frac{d\boldsymbol{a}_t}{dt} = \boldsymbol{f}_\theta^a\left(\boldsymbol{a}_t, \boldsymbol{z}_t, \boldsymbol{c}; t_a, t_v\right), \tag{3}$$

where $\boldsymbol{z}_t$ and $\boldsymbol{a}_t$ are the video and action latents, respectively, $\boldsymbol{c}$ is language condition, t_v and t_a are their corresponding flow-matching timesteps. In the shared-time regime, where $t_a = t_v = t \le t_0$, the refined action latent is defined by

$$\boldsymbol{a}_t = \boldsymbol{a}_{t_0} + \int_{t_0}^{t} \boldsymbol{f}_\theta^a\left(\boldsymbol{a}_\tau, \boldsymbol{z}_\tau, \boldsymbol{c}; \tau, \tau\right) d\tau, \qquad 0 \le t \le t_0. \tag{4}$$

Thus, $\boldsymbol{a}_{t_v}$ denotes the action latent obtained by integrating the learned action velocity field from the SDE-initialized state $\boldsymbol{a}_{t_0}$ to time t. The final refined action is $\boldsymbol{a}^\star = \boldsymbol{a}_0$.

3.2 CEREBELLAR RECTIFICATION: HOW VIDEO–ACTION COUPLING

In WARM, the WAM refines the VLA proposal through its coupled video and action branches. Because the WAM is trained with independently sampled video and action timesteps, different relative denoising schedules remain within the training distribution while inducing distinct refinement behaviors. We therefore formulate and systematically evaluate a family of video–action coupling schemes, in which the action branch may follow the video branch (`joint` and `original`), lag behind it (`parallel`), or remain temporarily fixed during early video denoising (`clamp` and `self_clamp`). The illustration is displayed in Figure. 2.

`joint`. The two branches use a shared timestep, $t_a = t_v = t$, throughout denoising. For $t > t_0$, however, the action latent is not updated by the learned action velocity field. Instead, it follows the prescribed VLA-conditioned path

$$\boldsymbol{a}_t = (1 - t)\boldsymbol{a}^0 + t\boldsymbol{\epsilon}. \tag{5}$$

Once $t \le t_0$, the action latent is released and both branches are freely denoised on the shared schedule. Thus, this scheme preserves synchronized (t, t) conditioning while following a predefined action path before refinement.

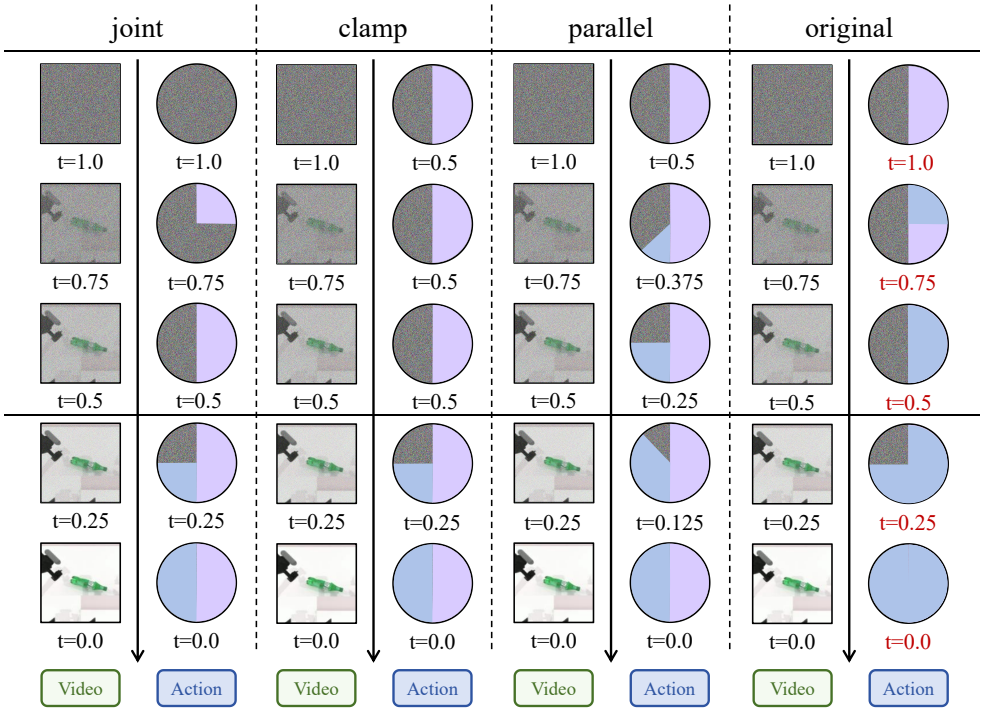


Figure 2: **Illustration of the video–action coupling schemes.** Each column shows the denoising trajectories of the video branch (green) and action branch (blue), where gray denotes noise and colored regions denote the corresponding denoised content.

`original`. The two branches use the shared schedule $t_v = t_a = t$ over the full $t = 1 \to 0$ trajectory. Unlike `joint`, the action latent is never pinned to the VLA-conditioned path: $\boldsymbol{a}_{t_a}$ is updated by the learned action velocity field at every step with Eq. 4. The action branch is initialized from the SDE-perturbed VLA proposal $\boldsymbol{a}_{t_0}$.

`parallel`. The video branch follows its full schedule from 1 to 0, while the action branch follows its own schedule from t_0 to 0 using the same number of denoising steps. Both branches are updated at every step, but generally $t_a \neq t_v$.

`clamp`. The action branch is initialized with $\boldsymbol{a}_{t_0}$. During video denoising, the input to the action branch is

$$\tilde{\boldsymbol{a}}(t_v) = \begin{cases} \boldsymbol{a}_{t_0}, & t_v > t_0, \\ \boldsymbol{a}_{t_v}, & t_v \leq t_0, \end{cases} \tag{6}$$

where $\boldsymbol{a}_{t_v}$ is the jointly denoised action latent defined in Eq. 4. Hence, while $t_v > t_0$, the video branch denoises from 1 to t_0 while the action branch always receives the fixed SDE-perturbed VLA proposal $\boldsymbol{a}_{t_0}$. Once $t_v \leq t_0$, the action branch is released and follows the learned velocity field jointly with the video branch until $t = 0$.

`self_clamp`. This scheme uses the same coupling schedule as `clamp`, but removes the VLA proposal. The action latent at t_0 is initialized from the WAM's own action prior starting from pure noise. This variant isolates the contribution of the VLA as the semantic planner.

Our experiments in Section 4.3 show that `clamp` is the most effective coupling strategy. Its asymmetric schedule provides a useful functional analogy to cerebellar predictive motor control:

Key takeaway. The video branch first predicts the consequences of the VLA plan, after which the action branch uses this prediction for physical refinement.

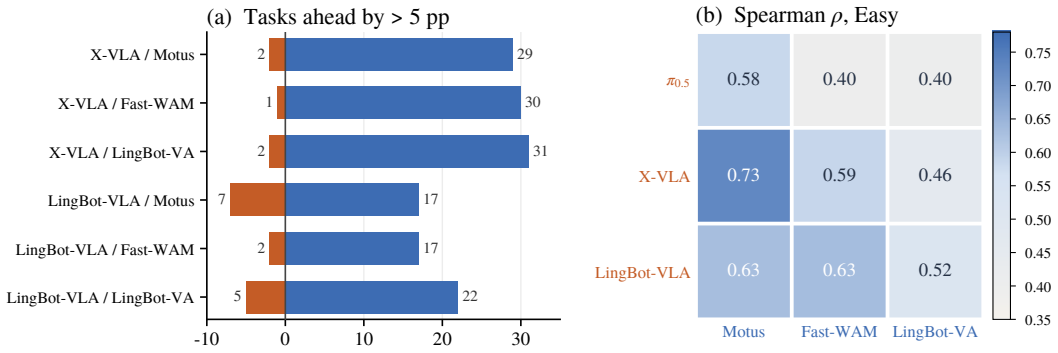


Figure 3: **Two VLAs against three WAMs.** (a) Tasks on which one model leads the other by more than 5% (Clean). Left of zero is the VLA; right of zero is the WAM. X-VLA leads on at most two tasks. LingBot-VLA keeps a minority against each WAM, and the largest of these is against Motus (7 tasks). (b) Spearman ρ on Clean. Rows are VLAs, including Motus's $\pi_{0.5}$; columns are WAMs. The highest cell is X-VLA vs. Motus (ρ=0.73).

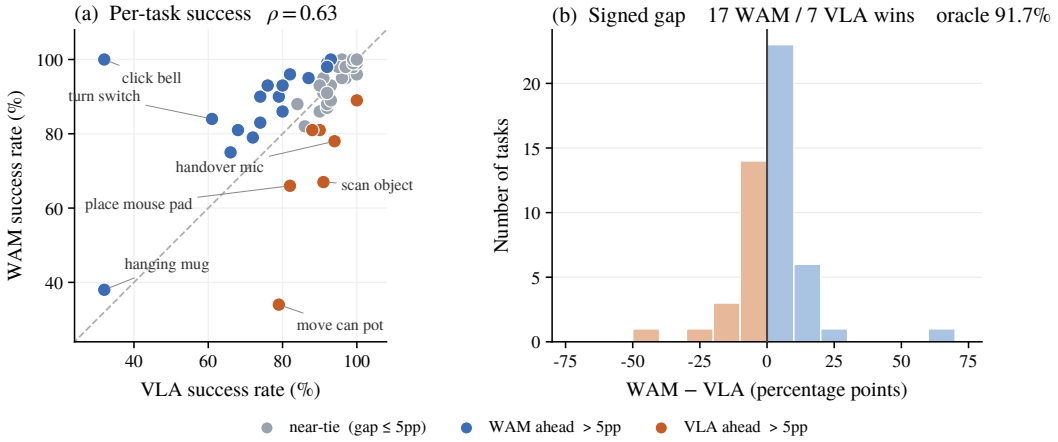


Figure 4: **Per-task view of LingBot-VLA vs. Motus,** the closest pair inside the group of Figure 3. Left: success on 50 tasks. The moderate Spearman correlation (ρ=0.63) rules out dominance of this pair, and the off-diagonal tasks (`click_bell`, `move_can_pot`, ...) are the successes each model does not share. Right: histogram of per-task gaps. Motus leads on more tasks, but LingBot-VLA's wins keep the per-task oracle at 91.7%.

4 Experiments

Benchmarks. We evaluate on RoboTwin (Chen et al., 2025), a dual-arm manipulation benchmark featuring a broad range of procedurally generated environments. We consider 50 tasks under both clean and randomized settings, and conduct 100 closed-loop evaluation episodes for each task–setting combination.

Implementation Details. We instantiate the planner with the publicly released post-trained LingBot-VLA w/o depth(Wu et al., 2026b) and the rectifier with Motus (Bi et al., 2026). The t_0 is set to 0.6 by default.

4.1 VLA and WAM are complementary, not competing

We compare two VLAs with three WAMs on the same 50 RoboTwin 2.0 tasks, considering all cross-family pairs. The VLAs are X-VLA (Zheng et al., 2026) and LingBot-VLA (Wu et al., 2026b), while the WAMs are Motus (Bi et al., 2026), Fast-WAM (Yuan et al., 2026), and LingBot-VA (Li et al.,

Table 1: Per-task success rates (%) averaged within seven task groups on RoboTwin 2.0. The number in parentheses denotes the number of tasks in each group.

	GO-1		$\pi_{0.5}$		X-VLA		LingBot-VLA		Motus		WARM (t_0=0.6)	
Group	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.
Acquisition & lifting (5)	76.4	76.6	36.8	36.2	80.8	74.2	92.2	**95.8**	**94.2**	**94.6**	86.0	86.8
Handover (2)	10.5	10.0	23.0	18.5	36.5	18.5	87.0	90.5	82.0	68.0	**89.0**	**91.5**
Targeted placement (13)	25.0	23.2	42.5	46.0	78.8	79.8	86.0	83.0	82.6	85.1	**88.8**	**87.9**
Container packing (8)	36.9	40.3	34.8	37.0	74.4	73.3	87.5	**90.3**	88.6	87.3	**89.3**	89.9
Arrangement & stacking (8)	17.6	17.3	43.6	42.4	63.0	64.8	84.3	87.6	90.1	88.0	**92.8**	**96.1**
Articulated/device interaction (7)	61.4	54.7	47.9	47.4	78.4	82.4	79.3	65.0	**93.6**	89.9	89.9	**91.4**
Tool use & dynamic manipulation (7)	42.7	37.7	57.9	58.4	70.6	73.7	91.9	**92.9**	91.3	86.4	**95.6**	92.4
Average (50)	37.86	36.24	42.98	43.84	72.88	72.84	86.50	85.34	88.66	87.02	**90.32**	**90.70**

2026a). X-VLA and LingBot-VA use the checkpoints reported by LingBot-VA (X-VLA: 72.9% Clean and 72.8% Random; LingBot-VA: 92.9% Clean and 91.6% Random), with X-VLA's per-task rates adopted from Motus. Fast-WAM reports 91.88% Clean and 91.78% Random, while LingBot-VLA denotes the post-trained checkpoint with 86.5% Clean success.

VLA and WAM exhibit non-redundant task-level strengths. Figure 3 compares every VLA–WAM pair, while Figure 4 provides a detailed comparison between LingBot-VLA and Motus, the closest pair in overall success rate and the pair used by WARM. Across the 3×3 cross-family comparisons, neither VLA is redundant with the WAMs: each pair exhibits task-level wins on both sides, with only moderate correlations in most cases. X-VLA correlates with Motus at $\rho = 0.73$ but leads on only 2 tasks, whereas Motus leads on 29; against Fast-WAM and LingBot-VA, the correlations are 0.59 and 0.46, and X-VLA leads on only 1 and 2 tasks, respectively.

The closest VLA–WAM pair reveals a semantic–physical division of strengths. LingBot-VLA and Motus have only moderate task-level agreement (Spearman's $\rho = 0.63$, Pearson's $r = 0.49$). LingBot-VLA leads Motus by more than 5% on 7 tasks, typically involving long-horizon rearrangement and transportation, including `move_can_pot`, `scan_object`, `place_mouse_pad`, and `handover_mic`. Motus leads by more than 5% on 17 tasks, which are typically shorter-horizon and more contact-sensitive, including `click_bell`, `turn_switch`, and `stamp_seal`. A per-task oracle reaches 91.7%, 3.2% above Motus, showing that the VLA's strengths can complement those of the WAM.

Complementarity depends on the specific model pairing. LingBot-VLA also remains non-redundant with Fast-WAM and LingBot-VA, with correlations of 0.63 and 0.52, respectively. The corresponding per-task oracles reach 92.8% and 94.1%, although their gains over the stronger individual models are smaller, at 0.9% and 1.2%. Moreover, a VLA advantage over one WAM does not imply an advantage over the others: on `move_can_pot`, LingBot-VLA outperforms Motus 79% to 34%, but trails Fast-WAM and LingBot-VA, which achieve 90% and 94%, respectively. Thus, complementarity is not a fixed ranking between model families; it is a pair-dependent pattern of non-redundant task-level capabilities. A system that can preserve the gains of both components should therefore outperform either component alone.

4.2 Main Results

Competitive Performance. To contextualize WARM against representative policy families, we compare it with two VLA baselines, GO-1 (Bu et al., 2025a) and $\pi_{0.5}$ (Intelligence et al., 2025), as well as X-VLA (Zheng et al., 2026), LingBot-VLA (Wu et al., 2026b), and the WAM baseline Motus (Bi et al., 2026). Table 1 reports their success rates across seven task groups under both clean and randomized configurations, with the number of tasks shown for each group.

Comparison against baselines. Because WARM is built directly on LingBot-VLA and Motus, the most relevant question is whether refinement can improve over both of its constituent models. Table 1 shows that it does: WARM achieves 90.3% success in the clean setting and 90.7% under randomization, outperforming LingBot-VLA by 3.8% and 5.4%, and Motus by 1.7% and 3.7%, respectively.

Table 2: Per-task success (%) on RoboTwin 2.0, where `clamp` at t_0=0.6 exceeds both LingBot-VLA and the Motus, on Clean or on Rand.

	X-VLA		LingBot-VLA		Motus		WARM (t_0=0.6)	
Task	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.
Blocks Ranking Size	67	74	66	73	75	63	**76**	**89**
Hanging Mug	23	27	32	27	38	38	**40**	**43**
Stack Bowls Three	76	86	72	83	79	87	**86**	**94**
Blocks Ranking Rgb	83	83	92	91	**99**	97	**99**	**100**
Stack Bowls Two	96	93	92	95	**98**	**98**	97	**98**
Place Object Scale	52	74	84	90	88	85	**95**	**91**
Place A2b Left	48	49	86	83	88	79	**96**	**95**
Stamp Seal	76	82	76	86	93	92	**96**	**100**
Move Pillbottle Pad	73	71	93	94	93	**96**	**94**	**96**
Place A2b Right	36	36	74	77	91	87	**92**	**95**
Place Burger Fries	94	94	95	96	**98**	**98**	97	**98**
Open Laptop	93	**100**	96	96	95	91	**97**	97
Place Fan	80	75	91	79	91	87	**97**	**93**
Place Dual Shoes	79	88	80	83	**93**	87	89	**93**
Beat Block Hammer	92	88	87	91	95	88	**97**	**99**
Open Microwave	79	71	91	75	95	91	**99**	**100**
Handover Mic	0	0	94	**98**	78	63	**98**	**98**
Place Object Stand	86	88	97	93	**98**	97	**98**	**99**
Turn Switch	40	61	61	65	**84**	**78**	73	**78**
Handover Block	73	37	80	83	**86**	73	80	**85**
Place Shoe	96	95	**99**	94	**99**	97	**99**	**100**
Dump Bin Bigbin	79	77	**97**	92	95	91	95	**97**
Place Bread Skillet	77	67	90	**89**	86	83	**91**	86
Place Mouse Pad	70	70	82	**78**	66	68	**83**	77
Stack Blocks Three	6	10	92	**99**	91	95	**96**	98
· · · (50 tasks)								
Average	72.88	72.84	86.50	85.34	88.66	87.02	**90.32**	**90.70**

Table 3: Coupling-scheme ablation (clean).

Coupling scheme	t_0	Success rate (%)
joint	0.6	48–64
original	0.6	59.2
parallel	0.6	87.5
clamp	0.6	**90.3**

Table 4: Ablantion on VLA-SDE init (clean).

Method	Success rate (%)
clamp	90.3
clamp w/o SDE	74.5
self_clamp	89.1
self_clamp w/o SDE	56.4

The gains are selective but meaningful. Table 1 shows that WARM is strongest on tasks requiring both semantic planning and precise execution, achieving the best performance in five of seven groups on Clean and four on Rand. It improves over both LingBot-VLA and Motus on diverse tasks such as handover, placement, stacking, and device interaction (Table 2), while Motus remains stronger on acquisition and articulated interaction. This pattern suggests that WARM does not simply average the two models: the VLA specifies the task objective, and the WAM corrects the physical details of execution.

4.3 Ablations and Analysis

Ablation on SDE. We evaluate the contribution of SDE by comparing each method with its corresponding variant without SDE in Table 4. Removing SDE decreases the success rate of *clamp* from 90.3% to 74.5%,. The effect is even more pronounced for *self_clamp*, whose success rate falls from 89.1% to 56.4%. These results show that SDE improves the success rate of both methods, with a particularly substantial benefit for *self_clamp*.

Ablation on Coupling scheme. We next fix the VLA as the planner and compare the coupling schemes in Table 3. The `clamp` scheme, which holds the action prior fixed while the video branch predicts the task motion and then refines the action in the final denoising stage, achieves the best success rate of 90.3%. The synchronized `joint` and `original` schemes perform substantially worse, while `parallel` achieves an intermediate success rate of 87.5%. These results indicate

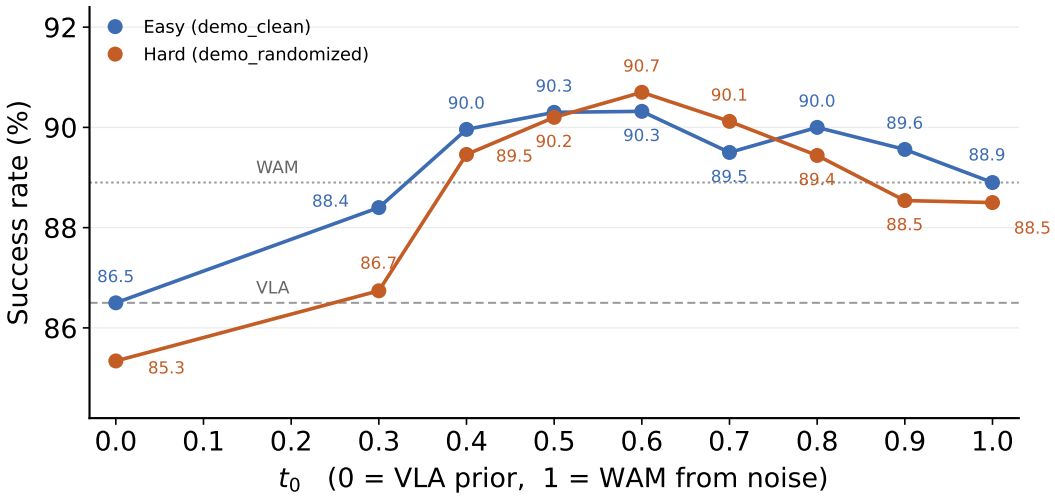


Figure 5: **Success versus t_0 on RoboTwin 2.0.** Easy is `demo_clean` and Hard is `demo_randomized`, both completed over 50×100 episodes.

that the video branch should first develop a prediction around the planned action before the action branch is released for refinement.

Clamp v.s. Self-Clamp. With SDE, *clamp* achieves a success rate of 90.3%, slightly higher than the 89.1% obtained by *self_clamp*. Without SDE, the gap is considerably larger: *clamp* reaches 74.5%, compared with 56.4% for *self_clamp*. Since *clamp* refines the VLA whereas *self_clamp* uses the WAM to refine itself, these results suggest that refining the VLA is more effective than self-refining the WAM. The stronger performance of *clamp* supports the complementary nature of the VLA and WAM: leveraging the WAM to refine the VLA can yield better performance than relying on the WAM alone.

Sensitivity to t_0. The initialization time t_0 exhibits a clear intermediate optimum rather than a monotonic trade-off. Increasing t_0 from 0 to 0.4–0.6 yields substantial gains on both domains, improving from 86.5% to 90.3% on Clean and from 85.3% to 90.7% on Random. Performance then remains competitive but gradually declines as the VLA proposal is increasingly discarded, reaching 88.9% and 88.5% at $t_0 = 1$. The consistent advantage of intermediate noise levels indicates that WARM benefits from retaining the VLA's semantic plan while allowing the WAM sufficient freedom to perform physical refinement.

5 Conclusion

We presented WARM, a plan-then-rectify framework built on the central finding that Vision-Language-Action models and World-Action Models are *complementary rather than competing*: VLAs provide semantic planning, whereas WAMs contribute physics-aware action refinement through video-based forward modeling. On RoboTwin 2.0, we systematically evaluated several video–action coupling strategies and found that the asymmetric `clamp` schedule performs best. It first allows the video branch to predict the consequences of the VLA plan and then releases the action branch for physical refinement. Using LingBot-VLA and Motus, WARM achieves 90.3% on Clean and 90.7% on Random, outperforming both constituent models. These results show that video prediction can provide a transferable capability for improving a pretrained policy at inference time while preserving its language-conditioned planning ability.

References

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35101–35113, 2026.

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, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. 2025.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025a.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*, 2025b.

Randy L Buckner. The cerebellum and cognitive function: 25 years of insight from anatomy and neuroimaging. *Neuron*, 80(3):807–815, 2013.

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

Chilam Cheang, Sijin Chen, Zhongren Cui, Yingdong Hu, Liqun Huang, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Xiao Ma, et al. GR-3 technical report. *arXiv preprint arXiv:2507.15493*, 2025.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

Zhiying Du, Bei Liu, Yaobo Liang, Yichao Shen, Haidong Cao, Xiangyu Zheng, Zhiyuan Feng, Zuxuan Wu, Jiaolong Yang, and Yu-Gang Jiang. Himoe-vla: Hierarchical mixture-of-experts for generalist vision-language-action policies. *arXiv preprint arXiv:2512.05693*, 2025.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, et al. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

Yanjiang Guo, Tony Lee, Lucy Xiaoyang Shi, Jianyu Chen, Percy Liang, and Chelsea Finn. Vlaw: Iterative co-improvement of vision-language-action policy and world model. *arXiv preprint arXiv:2602.12063*, 2026a.

Yanjiang Guo, Lucy Shi, Jianyu Chen, and Chelsea Finn. Ctrl-world: A controllable generative world model for robot manipulation. In *International Conference on Learning Representations*, volume 2026, pp. 6121–6138, 2026b.

Chia-Yu Hung, Navonil Majumder, Haoyuan Deng, Liu Renhang, Yankang Ang, Amir Zadeh, Chuan Li, Dorien Herremans, Ziwei Wang, and Soujanya Poria. Nora-1.5: A vision-language-action model trained using world model-and action-based preference rewards. *arXiv preprint arXiv:2511.14659*, 2025.

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*, 2025.

Masao Ito. Control of mental activities by internal models in the cerebellum. *Nature Reviews Neuroscience*, 9(4):304–313, 2008.

Tao Jiang, Tianyuan Yuan, Yicheng Liu, Chenhao Lu, Jianning Cui, Xiao Liu, Shuiqi Cheng, Jiyang Gao, Huazhe Xu, and Hang Zhao. Galaxea open-world dataset and g0 dual-system vla model. *arXiv preprint arXiv:2509.00576*, 2025.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026a.

Yang Li, Zhaxizhuoma Zhaxizhuoma, Hongru Jiang, Junjie Xia, Hongquan Zhang, Jinda Du, Yunsong Zhou, Jia Zeng, Ce Hao, Jieji Ren, et al. Forcevla2: Unleashing hybrid force-position control with force awareness for contact-rich manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8911–8920, 2026b.

Xuewu Lin, Tianwei Lin, Yun Du, Hongyu Xie, Yiwei Jin, Jiawei Li, Shijie Wu, Qingze Wang, Mengdi Li, Mengao Zhao, et al. Holobrain-0 technical report. *arXiv preprint arXiv:2602.12062*, 2026a.

Zefu Lin, Rongxu Cui, Junjia Xu, Xiaojuan Jin, Wenling Li, Lue Fan, and Zhaoxiang Zhang. World pilot: Steering vision-language-action models with world-action priors. *arXiv preprint arXiv:2606.12403*, 2026b.

Yicheng Liu, Zibin Dong, Baijun Ye, Tianyuan Yuan, Tao Jiang, Anqi Yang, Shicheng Cao, Haonan Liu, Yue Sun, Zihan Guo, et al. G0. 5: One autoregressive stream for robot reasoning and action. *arXiv preprint arXiv:2608.11739*, 2026.

Hao Luo, Yicheng Feng, Wanpeng Zhang, Sipeng Zheng, Ye Wang, Haoqi Yuan, Jiazheng Liu, Chaoyi Xu, Qin Jin, and Zongqing Lu. Being-h0: vision-language-action pretraining from large-scale human videos. *arXiv preprint arXiv:2507.15597*, 2025a.

Hao Luo, Yicheng Feng, Wanpeng Zhang, Sipeng Zheng, Ye Wang, Haoqi Yuan, Jiazheng Liu, Chaoyi Xu, Qin Jin, and Zongqing Lu. Being-h0: vision-language-action pretraining from large-scale human videos. *arXiv preprint arXiv:2507.15597*, 2025b.

Hao Luo, Ye Wang, Wanpeng Zhang, Sipeng Zheng, Ziheng Xi, Chaoyi Xu, Haiweng Xu, Haoqi Yuan, Chi Zhang, Yiqing Wang, et al. Being-h0.5: Scaling human-centric robot learning for cross-embodiment generalization. *arXiv preprint arXiv:2601.12993*, 2026a.

Hao Luo, Wanpeng Zhang, Yicheng Feng, Sipeng Zheng, Haiweng Xu, Chaoyi Xu, Ziheng Xi, Yuhui Fu, and Zongqing Lu. Being-h0.7: A latent world-action model from egocentric videos. *arXiv preprint arXiv:2605.00078*, 2026b.

Frank A Middleton and Peter L Strick. Basal ganglia and cerebellar loops: motor and cognitive circuits. *Brain research reviews*, 31(2-3):236–250, 2000.

Dyna Robotics. Dyna-2: A 1-million-hour scaling law for world-action models. *Dyna Robotics, Official Research Report*, 2026.

Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

GigaWorld Team, Angyuan Ma, Boyuan Wang, Bohan Li, Chaojun Ni, Guo Li, Guan Huang, Guosheng Zhao, Hao Li, Hengtao Li, et al. Gigaworld-1: A roadmap to build world models for robot policy evaluation. *arXiv preprint arXiv:2607.02642*, 2026a.

MotuBrain Team, Chendong Xiang, Fan Bao, Haitian Liu, Hengkai Tan, Hongzhe Bi, James Li, Jiabao Liu, Jingrui Pang, Kiro Jing, et al. Motubrain: An advanced world action model for robot control. *arXiv preprint arXiv:2604.27792*, 2026b.

Qiuyue Wang, Mingsheng Li, Jian Guan, Jinhui Ye, Sicheng Xie, Yitao Liu, Junhao Chen, Zhixuan Liang, Jie Zhang, Xintong Hu, et al. Qwen-vla: Unifying vision-language-action modeling across tasks, environments, and robot embodiments. *arXiv preprint arXiv:2605.30280*, 2026a.

Zhihao Wang, Jianxiong Li, Yu Cui, Yuan Gao, Xianyuan Zhan, Junzhi Yu, and Xiao Ma. World value models for robotic manipulation, 2026b.

Daniel M Wolpert, R Chris Miall, and Mitsuo Kawato. Internal models in the cerebellum. *Trends in cognitive sciences*, 2(9):338–347, 1998.

Wei Wu, Fan Lu, Yunnan Wang, Shuai Yang, Shi Liu, Fangjing Wang, Shuailei Ma, He Sun, Yong Wang, Zhenqi Qiu, Houlong Xiong, Ziyu Wang, Shuai Zhou, Yiyu Ren, Kejia Zhang, Hui Yu, Jingmei Zhao, Qian Zhu, Ran Cheng, Yong-Lu Li, Yongtao Huang, Xing Zhu, Yujun Shen, and Kecheng Zheng. LingBot-VLA: A pragmatic vla foundation model. *arXiv preprint arXiv:2601.18692*, 2026a. URL https://arxiv.org/abs/2601.18692.

Wei Wu, Fan Lu, Yunnan Wang, Shuai Yang, Shi Liu, Fangjing Wang, Qian Zhu, He Sun, Yong Wang, Shuailei Ma, et al. A pragmatic vla foundation model. *arXiv preprint arXiv:2601.18692*, 2026b.

Wei Wu, Fangjing Wang, Fan Lu, He Sun, Shi Liu, Yunnan Wang, Yibin Yan, Yong Wang, Shuailei Ma, Xinyang Wang, et al. From foundation to application: Improving vla models in practice. *arXiv preprint arXiv:2607.06403*, 2026c.

Jianwei Yang, Reuben Tan, Qianhui Wu, Ruijie Zheng, Baolin Peng, Yongyuan Liang, Yu Gu, Mu Cai, Seonghyeon Ye, Joel Jang, et al. Magma: A foundation model for multimodal ai agents. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14203–14214. IEEE, 2025.

Jiazhi Yang, Kunyang Lin, Jinwei Li, Wencong Zhang, Tianwei Lin, Longyan Wu, Zhizhong Su, Hao Zhao, Ya-Qin Zhang, Li Chen, et al. Rise: Self-improving robot policy with compositional world model. *arXiv preprint arXiv:2602.11075*, 2026.

Jinhui Ye, Ning Gao, Senqiao Yang, Jinliang Zheng, Zixuan Wang, Yuxin Chen, Pengguang Chen, Yilun Chen, Shu Liu, and Jiaya Jia. Starvla-$alpha$: Reducing complexity in vision-language-action systems. *arXiv preprint arXiv:2604.11757*, 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.

Zihan You, Hongwei Liu, Chenxu Dang, Zhe Wang, Sining Ang, Aoqi Wang, and Yan Wang. Samoe-vla: A scene adaptive mixture-of-experts vision-language-action model for autonomous driving. *arXiv preprint arXiv:2603.08113*, 2026.

Jiawen Yu, Hairuo Liu, Qiaojun Yu, Jieji Ren, Ce Hao, Haitong Ding, Guangyu Huang, Guofan Huang, Yan Song, Panpan Cai, et al. Forcevla: Enhancing vla models with a force-aware moe for contact-rich manipulation. *Advances in Neural Information Processing Systems*, 38:93409–93439, 2026.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

Andy Zhai, Brae Liu, Bruno Fang, Chalse Cai, Ellie Ma, Ethan Yin, Hao Wang, Hugo Zhou, James Wang, Lights Shi, et al. Igniting vlms toward the embodied space. *arXiv preprint arXiv:2509.11766*, 2025.

Jiahui Zhang, Ze Huang, Chun Gu, Zipei Ma, and Li Zhang. Reinforcing action policies by prophesying. *arXiv preprint arXiv:2511.20633*, 2025.

Likui Zhang, Tao Tang, Zhihao Zhan, Xiuwei Chen, Zisheng Chen, Jianhua Han, Jiangtong Zhu, Pei Xu, Hang Xu, Hefeng Wu, et al. Atomicvla: Unlocking the potential of atomic skill learning in robots. *arXiv preprint arXiv:2603.07648*, 2026a.

Wenyao Zhang, Hongsi Liu, Zekun Qi, Yunnan Wang, Xinqiang Yu, Jiazhao Zhang, Runpei Dong, Jiawei He, He Wang, Zhizheng Zhang, et al. Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge. *Advances in Neural Information Processing Systems*, 38: 24195–24228, 2026b.

Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, et al. X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. In *International Conference on Learning Representations*, volume 2026, pp. 60580–60606, 2026.

Fangqi Zhu, Zhengyang Yan, Zicong Hong, Quanxin Shou, Xiao Ma, and Song Guo. Wmpo: World model-based policy optimization for vision-language-action models. In *International Conference on Learning Representations*, volume 2026, pp. 62486–62502, 2026.