WORLD-CALIBRATED PROPOSAL-TO-ACTION FLOW FOR VISION-LANGUAGE-ACTION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Flow-based Vision-Language-Action (VLA) policies generate action chunks by transporting samples from a task-agnostic isotropic Gaussian source. As this source is conditioned on neither recent execution nor predicted future evolution, ***(i)*** it discards the local continuity established by recently executed motion. ***(ii)*** Even when predictive world representations are introduced, they often only condition the transport dynamics rather than determine where generation starts, how far it may deviate, or along which action directions it may expand. Building on this observation, we introduce **PROACT**, a world-calibrated proposal-to-action framework that makes the generative source itself predictable. ***(i)*** To preserve motion continuity, a lightweight Proposal Expert converts recent actions into a scene-aware hypothesis via one motion-anchored endpoint flow-matching step, initializing generation near the demonstrated action manifold. ***(ii)*** To jointly capture intended scene evolution and proposal–future compatibility, a prospective World Expert treats the hypothesis as a soft motion prior while predicting the task-consistent latent future. ***(iii)*** From this compatibility, the model calibrates a proposal-centered anisotropic source, where a bounded per-step extent controls the allowed deviation and a trace-normalized low-rank geometry under a condition-number budget allocates refinement over coupled translation, rotation, and gripper directions. Compared with $\pi_{0.5}$, **PROACT** improves performance across simulation and real-world tasks while reducing denoising steps by 50%, inference latency by up to 25.8%, and increasing throughput by up to 34.8%.

1 INTRODUCTION

Vision-language-action (VLA) models Black et al. (2024); Bjorck et al. (2025); Li et al. (2026b); Luo et al. (2026) increasingly use diffusion Chi et al. (2025) or flow matching Lipman et al. (2022) to generate continuous robot actions. However, these generative decoders typically initialize each action chunk from a generic source distribution Intelligence et al. (2025); Zheng et al. (2026), forcing the policy to repeatedly transport uninformed samples toward executable trajectories, as illustrated in the upper panel of Fig. 1. Such a generic source ignores readily available context that could inform where action generation should begin. By making the source distribution itself contextually predictable, action generation can begin from a more informed region of the action space, yielding a policy that is both more accurate and substantially cheaper to sample.

A generic isotropic source leaves two useful sources of information unexploited. The first is ***recently executed motion***. Although task-level behavior can be highly multimodal, physical motion often exhibits strong local regularity over short horizons, and recently executed, reliable actions are already available during closed-loop control Black et al. (2026); Wang et al. (2026a). A naive strategy is to extrapolate this motion and reuse it for the next action chunk Jia et al. (2026); Zhan et al. (2026). Yet such continuation becomes unreliable around contact, reorientation, gripper actuation, or task-phase transitions, where appropriate motion may deviate sharply from recent behavior Yang et al. (2026c). Local continuity therefore indicates what is *plausible* from the recent past, but not necessarily what remains *appropriate* for the current scene and task. Recent motion should thus be treated as a proposal to be evaluated against the intended future, rather than as the action itself.

The second is ***predicted future evolution***, which provides prospective evidence for calibrating such a proposal. Prospective world modeling Zhao et al. (2025); Zhang et al. (2025); Tian et al. (2025); Zhang et al. (2026); Li et al. (2026c) has increasingly been incorporated into VLA policies, but

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.

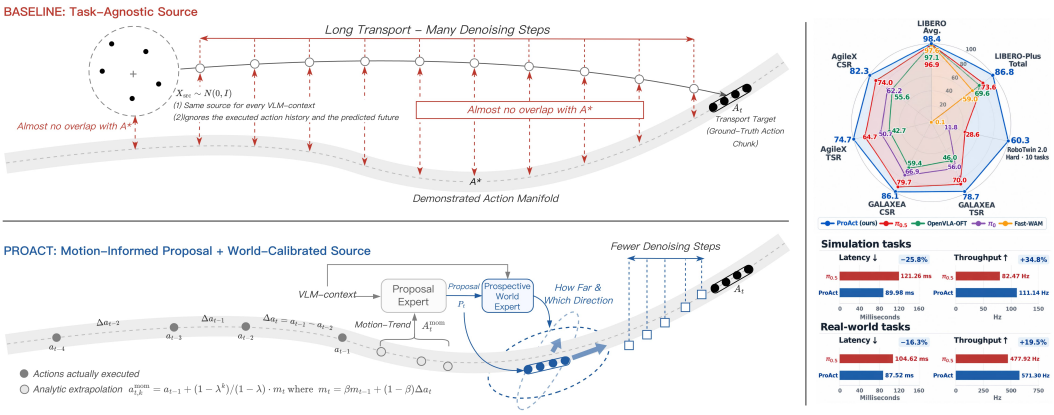


Figure 1: **Motivation of PROACT**. PROACT uses recent motion to form a scene-aware proposal and prospective world reasoning to calibrate its source distribution. Starting from this informed, proposal-centered source shortens action transport while improving performance and efficiency.

typically as auxiliary supervision or additional conditioning. In these formulations, prospective information helps predict what should happen next, but is not explicitly used to calibrate an existing action hypothesis according to whether it remains appropriate for the anticipated task evolution. The missing element is therefore an explicit interface between a candidate motion, prospective world reasoning, and generative refinement.

We build this missing interface around a *propose–calibrate–refine* principle and instantiate it as **PROACT**, a World-Calibrated Proposal-to-Action Flow framework. PROACT first constructs a scene-aware motion hypothesis, then uses prospective world reasoning to estimate the correction required to realize the intended future, and finally converts this calibration into the source distribution for flow-based action generation, as illustrated in the lower panel of Fig. 1.

Specifically, ***(i)*** a lightweight ***Proposal Expert*** first converts recently executed actions into a scene-aware motion hypothesis. It extrapolates short-horizon motion and uses it to initialize the motion-anchored flow module, which grounds the proposal in the current VLM context through a single endpoint flow step. Rather than producing an executable action, this stage places generation near a plausible region of the demonstrated action manifold and provides a ***soft motion prior*** for subsequent reasoning. ***(ii)*** A prospective ***World Expert*** calibrates this proposal against the task-consistent future. The motion hypothesis is provided as contextual evidence for prospective reasoning, while the prediction target remains the demonstrated future encoded in the latent space of a frozen V-JEPA 2 encoder Assran et al. (2025). Calibration readouts are further supervised by the residual between the demonstrated action and the proposal, forcing the shared world representation to encode both the task-consistent latent future and the proposal–future compatibility. ***(iii)*** The resulting calibration is finally expressed as a ***proposal-centered anisotropic source*** for action generation. A bounded per-step ***extent*** limits how far generation may deviate from the proposal, while trace-normalized low-rank ***geometry*** determines along which coupled action directions that correction should occur. An Action Expert then transports samples from this calibrated source into executable action chunks using the base VLA's original flow parameterization. Thus, **PROACT** changes where and how action generation begins while preserving the flexibility to depart from the proposal when needed.

Our primary contributions are summarized as follows:

- We formulate generative action prediction from a propose–calibrate–refine perspective, where recent motion provides a soft local hypothesis calibrated by task-consistent prospective reasoning, providing an interface between local motion continuity and future-aware action generation.
- We introduce an asymmetric proposal-aware World Expert that uses motion to assist future prediction while anchoring the desired future independently of proposal correctness, and jointly predicts proposal-specific refinement extent and geometry at every horizon step.
- We replace the isotropic Gaussian of flow-based VLAs with a world-calibrated, proposal-centered source while leaving their transport parameterization unchanged.

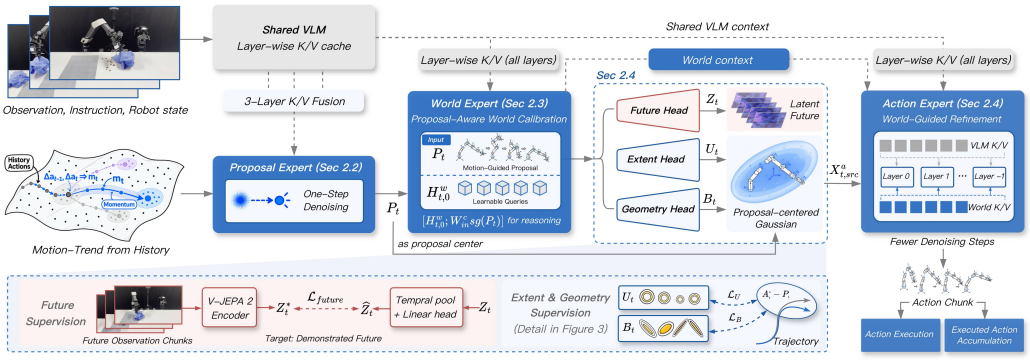


Figure 2: **Overview of PROACT.** A Proposal Expert forms a scene-aware motion hypothesis, which the World Expert calibrates against the task-consistent future to predict refinement extent and geometry. The resulting source is then refined into executable actions by the Action Expert.

- We validate PROACT across extensive simulation benchmarks and real-world robotic manipulation tasks, where it consistently outperforms strong baselines while reducing denoising steps by up to 50%, demonstrating both superior effectiveness and inference efficiency.

2 PROACT

As shown in Fig. 2, PROACT follows a *propose–calibrate–refine* pipeline. A ***Proposal Expert*** grounds recent motion into a scene-aware hypothesis, a prospective ***World Expert*** models latent future and proposal–future compatibility, and an ***Action Expert*** transports the calibrated source into executable actions. This calibration determines where refinement starts, how far it may travel, and along which coupled directions it should proceed at each horizon step.

2.1 PROBLEM FORMULATION

Given demonstrations $\mathcal{D} = \{(\tau_i, \ell_i)\}_{i=1}^{N}$ with $\tau_i = \{(O_t, q_t, a_t)\}_{t=1}^{T_i}$, we predict an action chunk $A_t = (a_{t,1}, \ldots, a_{t,H}) \in \mathbb{R}^{H \times d_a}$ from $\mathcal{C}_t = (O_{\le t}, \ell, q_{\le t})$ and recent action history $\mathcal{H}_t^a = \{a_{t-M}, \ldots, a_{t-1}\}$ containing only executed actions. We use a shared horizon $M = H$. A VLM encodes $\mathcal{C}_t$ and exposes layer-wise keys and values $\mathcal{K}_t^{\mathrm{VLM}} = \{(K_{t,l}^v, V_{t,l}^v)\}_{l=1}^{L_v}$, which provide common perception–language context to all experts throughout generation.

Suppressing time indices and treating $\mathcal{K}_t^{\mathrm{VLM}}$ as implicit context, for notational clarity we let $C = \mathcal{C}_t$, $H^a = \mathcal{H}_t^a$, and $X = X_{\mathrm{src}}$ throughout this formulation. Generation then factorizes as

$$p(A \mid C, H^a) = \iiint \underbrace{p_\theta(A \mid X, H^w)}_{\text{Action}} \underbrace{p_\eta(X \mid P, H^w)}_{\text{Source}} \underbrace{p_\omega(H^w \mid P)}_{\text{World}} \underbrace{p_\phi(P \mid H^a)}_{\text{Proposal}} \, dH^w \, dX \, dP. \tag{1}$$

Read right-to-left, the factors propose motion, model its desired future, calibrate the source, and generate actions; p_ω may condition on P when beneficial, whereas p_η must explicitly preserve P as the source anchor and calibrate deviations around it.

2.2 MOTION-INFORMED ACTION PROPOSAL

Robot motion is locally continuous despite task-level multimodality. Let $\Delta a_t = a_{t-1} - a_{t-2}$ and maintain the smoothed motion state $m_t = \beta m_{t-1} + (1-\beta)\Delta a_t$. We extrapolate

$$a_{t,k}^{\mathrm{mom}} = a_{t-1} + \frac{1-\lambda^k}{1-\lambda} m_t, \quad k = 1, \ldots, H, \tag{2}$$

where β suppresses fluctuations and λ attenuates long-horizon extrapolation; gripper dimensions retain their last executed value. The chunk A_t^{mom} captures local motion continuity but is not scene-aware. It initializes smooth motion while leaving contact and target changes to visual grounding.

A shallow Proposal Expert grounds this analytic prior in VLM context. It uses L_p layers of the Action Expert and a single flow time $t_p = 1$, treating A_t^{mom} as a source rather than noise. During

training, we perturb it as $X_t^p = A_t^{\text{mom}} + \sigma_p \epsilon_p$, $\epsilon_p \sim \mathcal{N}(0, I)$, and initialize $H_{t,0}^p = W_{\text{in}}^p X_t^p$. Each layer jointly attends to VLM context and the proposal state:

$$\widehat{H}_{t,j}^p = \text{Attn}\Big(Q_j^p(H_{t,j-1}^p), [\widetilde{K}_{t,j}^v; K_j^p], [\widetilde{V}_{t,j}^v; V_j^p]\Big), \tag{3}$$

where K_j^p, V_j^p project $H_{t,j-1}^p$. Because the proposer is shallower than the VLM ($L_p = 6$ versus $L_v = 18$), its layer j receives a compressed cache from $\mathcal{S}_j = \{3j-2, 3j-1, 3j\}$. It averages jointly RMS-normalized caches using parameter-free reliability weights $w_{t,j,l} \propto \widehat{\sigma}_{t,j,l}^{-2}$, where $\widehat{\sigma}_{t,j,l}^2$ estimates layer noise from within-window dispersion. Appendix E.1 derives the fusion rule. This favors locally consistent VLM depths without learned fusion parameters while matching the proposer's smaller depth. Residual and feed-forward updates yield $H_t^p = H_{t,L_p}^p$; we read out $v_\phi^p = W_{\text{out}}^p H_t^p$ and take one Euler step from $t_p = 1$ to 0:

$$P_t = X_t^p - v_\phi^p, \qquad \mathcal{L}_{\text{prop}} = \frac{1}{H} \sum_k \|P_{t,k} - a_{t,k}^*\|_2^2. \tag{4}$$

This endpoint objective requires no additional parameterization or iterative proposal refinement. At inference, we simply remove the training perturbation and set $X_t^p = A_t^{\text{mom}}$.

2.3 Asymmetric Desired-Future Modeling and Proposal Calibration

A single World Expert jointly reasons over the proposal and VLM semantics to capture proposal-conditioned prospective information. It is a Transformer without flow-time conditioning whose shared queries support asymmetric future-prediction and proposal-calibration readouts.

V-JEPA future latent target. Rather than reconstructing pixels or imposing object slots, we supervise prospective reasoning in video representation space. We form the future clip $\mathcal{V}_t^{\rightarrow} = (O_{t+1}, O_{t+2}, \ldots, O_{t+H})$ and use the frozen V-JEPA 2 encoder from V-JEPA 2-AC Assran et al. (2025) as the target space. With temporal tubelet size 2, $N^f = H/2$ is its temporal resolution. We spatially pool the encoder tokens into N_z groups within each temporal tubelet:

$$Z_t^* = \mathcal{P}_s(\text{sg}(E_{\text{VJ}}(\mathcal{V}_t^{\rightarrow}))) = \{z_{t,\tau,j}^*\}_{\tau,j} \in \mathbb{R}^{N^f \times N_z \times d_f}, \tag{5}$$

where $\mathcal{P}_s$ performs spatial pooling and feature-wise normalization. This removes the frozen encoder's arbitrary scale and balances future prediction against the remaining objectives. The target captures physical evolution without reconstructing task-irrelevant texture.

Proposal-aware world reasoning. The World Expert initializes learnable spatiotemporal queries $H_{t,0}^w = \{q_j^w + e_k^{\text{time}}\}_{k,j} \in \mathbb{R}^{H \times N_z \times d_w}$ at the full action-horizon resolution and embeds the detached proposal as H condition tokens, $H_{t,0}^c = W_{\text{in}}^c \,\text{sg}(P_t)$. Only the final P_t is shared. Writing $S_{t,l} = [H_{t,l}^w; H_{t,l}^c]$, each layer jointly attends to the VLM cache and its own state:

$$\overline{S}_{t,l} = \text{Attn}\big(Q_l^w(S_{t,l-1}), \big[K_{t,l}^v; K_{t,l}^s\big], \big[V_{t,l}^v; V_{t,l}^s\big]\big), \tag{6}$$

where $K_{t,l}^s, V_{t,l}^s$ project $S_{t,l-1}$, followed by standard Transformer updates. VLM features provide task and scene semantics, while proposal tokens inject a concrete motion tendency. Retaining only the $H \times N_z$ world-query outputs gives $H_t^w = \{h_{t,k,j}^w\}_{k,j}$, a shared *proposal-conditioned prospective representation* for calibration. Joint attention exposes proposal–scene compatibility per horizon step rather than compressing the motion hypothesis into one global token.

Asymmetric future and calibration readouts. For future prediction, neighboring world-token groups are pooled to match the V-JEPA resolution, $\overline{h}_{t,\tau,j}^w = \text{Pool}_t(h_{t,2\tau-1,j}^w, h_{t,2\tau,j}^w)$, and a linear head maps $\overline{h}_{t,\tau,j}^w$ to $\widehat{z}_{t,\tau,j}$. Fixed token ordering avoids matching, giving $\mathcal{L}_{\text{future}} = \frac{1}{N^f N_z d_f} \sum_{\tau,j} \|\widehat{z}_{t,\tau,j} - z_{t,\tau,j}^*\|_2^2$. The target is the demonstrated future, not the counterfactual outcome of executing P_t, so the proposal remains a *soft motion prior*.

For calibration, we pool latent tokens into one horizon-aligned feature, $g_{t,k}^w = \frac{1}{N_z} \sum_j h_{t,k,j}^w$, and denote $G_t^w = (g_{t,1}^w, \ldots, g_{t,H}^w)$. The future head decodes *what task-consistent evolution should look like*; calibration decodes *how the proposal must be loosened and reshaped* from the same prospective representation. Both calibration targets use residual $A_t^* - P_t$, making proposal information necessary even though the future target itself remains task-consistent. This asymmetry also prevents an inaccurate proposal from redefining the desired future merely to justify itself.

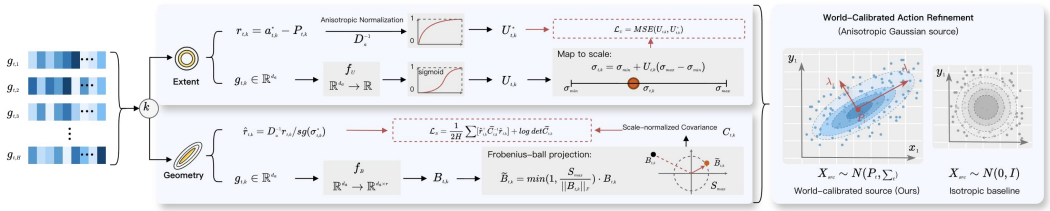


Figure 3: **World-calibrated source construction.** Proposal-conditioned future features yield per-step refinement extent and low-rank geometry, defining an anisotropic source for action refinement.

2.4 World-Calibrated Action Refinement

The proposal-aligned G_t^w defines a source with scalar *extent* and low-rank *geometry*.

Refinement extent. Let $r_{t,k} = a^*_{t,k} - P_{t,k}$, with the diagonal action-scale matrix D_a (See Appendix E). We infer correction extent via proposal–future compatibility:

$$\begin{aligned} e_{t,k} &= \frac{\|D_a^{-1} r_{t,k}\|_2}{\sqrt{d_a}}, \qquad & U^*_{t,k} &= 1 - \exp(-e_{t,k}/\tau_u), \\ U_{t,k} &= \mathrm{sigmoid}(f_u(g^w_{t,k})), \qquad & \sigma^{(*)}_{t,k} &= \sigma_{\min} + U^{(*)}_{t,k}(\sigma_{\max} - \sigma_{\min}). \end{aligned} \tag{7}$$

Here $\sigma^*_{t,k}$ uses $U^*_{t,k}$, while $\sigma_{t,k}$ uses $U_{t,k}$. MSE $\mathcal{L}_U$ supervises $U_{t,k}$ against $U^*_{t,k}$. Compatible proposals induce concentrated sources; incompatible ones open more room for refinement.

Refinement geometry. Magnitude alone misses dependencies among translation, rotation, and gripper dimensions. From the same world feature, we predict a low-rank factor $B_{t,k} = f_B(g^w_{t,k}) \in \mathbb{R}^{d_a \times r}$, with $r \ll d_a$. It defines the scale-normalized covariance at each proposal horizon step:

$$C_{t,k} = \sigma^2_{t,k} d_a \left(I + B_{t,k} B_{t,k}^\top\right) / \operatorname{tr}\left(I + B_{t,k} B_{t,k}^\top\right). \tag{8}$$

To reduce magnitude dependence when learning refinement geometry, we normalize residuals by the detached target extent and score them under the scale-normalized $\widetilde{C}_{t,k} = C_{t,k}/\sigma^2_{t,k}$:

$$\widehat{r}_{t,k} = \frac{D_a^{-1} r_{t,k}}{\mathrm{sg}(\sigma^*_{t,k})}, \qquad \mathcal{L}_B = \frac{1}{2H} \sum_k \left[\widehat{r}_{t,k}^\top \widetilde{C}^{-1}_{t,k} \widehat{r}_{t,k} + \log\det \widetilde{C}_{t,k}\right]. \tag{9}$$

Target normalization reduces residual-scale variation, while trace normalization prevents $B_{t,k}$ from changing the average variance across action dimensions. Thus, $\mathcal{L}_U$ primarily controls refinement magnitude and $\mathcal{L}_B$ its geometry. Trace normalization removes scale but not single-sample shape degeneracy, so we project $B_{t,k}$ onto $\|B_{t,k}\|_F^2 \le S_{\max}$. This enforces $\kappa(\widetilde{C}_{t,k}) \le 1 + S_{\max}$ while preserving $\operatorname{tr} C_{t,k} = d_a \sigma^2_{t,k}$. We use $S_{\max} = 5$; Appendix E.2 gives the derivation and $O(d_a r^2)$ evaluation. Consequently, $\sigma_{t,k}$ fixes average variance while $B_{t,k}$ redistributes it toward coupled residual directions without silently changing the overall refinement extent.

With action-space covariance $C^a_{t,k} = D_a C_{t,k} D_a^\top$, we sample the source as

$$X^a_{t,\mathrm{src}} = P_t + L_t \epsilon_a, \quad \epsilon_a \sim \mathcal{N}(0, I), \quad L_t L_t^\top = \mathrm{BlockDiag}\left(C^a_{t,1}, \ldots, C^a_{t,H}\right). \tag{10}$$

Appendix E.2 details sampling and covariance construction.

Action flow refinement. The calibrated covariance defines preferred correction axes but not the signed transport direction, which the Action Expert supplies. We preserve the base VLA's flow parameterization, where $s = 1$ is the source and $s = 0$ the data, replacing only its Gaussian source by $X^a_{t,\mathrm{src}}$. For base-model flow time s and velocity $v^A_\theta = W_v H^a_{t,s,L_a}$, the path and objective are

$$X^a_{t,s} = s X^a_{t,\mathrm{src}} + (1-s) A^*_t, \quad v^*_t = X^a_{t,\mathrm{src}} - A^*_t, \quad \mathcal{L}_{\mathrm{flow}} = \mathbb{E}_{t,s,\epsilon_p,\epsilon_a}\left[\left\| v^A_\theta - v^*_t \right\|_2^2\right]. \tag{11}$$

The Action Expert uses $X^a_{t,s}$ as its action state, reads shared VLM context, and attends to H^w_t for proposal-aware prospective guidance. Its velocity therefore receives both desired future evolution and proposal compatibility, with attention and integration detailed in Appendix A.1.3.

Table 1: **Results on LIBERO and LIBERO-Plus.** We report performance on the four LIBERO task suites and generalization under seven perturbation dimensions in LIBERO-Plus.

Method	LIBERO					LIBERO-Plus							
	Spatial	Object	Goal	Long	Avg.	Camera	Robot	Language	Light	Background	Noise	Layout	Total
OpenVLA-OFT Kim et al. (2025)	97.6	98.4	97.9	94.5	97.1	56.4	31.9	79.5	88.7	93.3	75.8	74.2	69.6
$\pi_{0.5}$ Intelligence et al. (2025)	98.8	98.2	98.0	92.4	96.9	59.7	65.5	75.3	87.0	82.4	72.1	80.3	73.6
WorldVLA Cen et al. (2025)	87.6	96.2	83.4	60.0	81.8	0.1	27.9	41.6	43.7	17.1	10.9	38.0	25.0
UniVLA Bu et al. (2025)	95.4	98.8	93.6	94.0	95.4	1.8	46.2	69.6	69.0	81.0	21.2	31.9	42.9
VLA-Adapter Wang et al. (2026b)	97.8	99.2	97.2	95.0	97.3	36.2	37.9	74.6	70.6	76.1	58.0	69.7	59.1
DreamVLA Zhang et al. (2026)	97.5	94.0	89.5	89.5	92.6	65.0	40.8	63.5	85.7	82.6	84.9	74.0	69.9
VLA-JEPA Sun et al. (2026)	96.2	99.6	97.2	95.8	97.2	63.3	67.1	85.4	95.6	93.6	66.3	85.1	79.5
Fast-WAM Yuan et al. (2026)	98.2	**100.0**	98.2	95.2	97.6	34.0	55.1	**88.9**	90.0	44.9	33.3	73.4	59.0
HoloBrain Lin et al. (2026)	–	–	–	–	–	66.3	49.0	65.9	94.9	93.3	73.1	78.2	72.6
PROACT	**99.4**	99.6	**98.4**	**96.0**	**98.4**	**76.1**	**80.3**	87.4	**97.8**	**95.5**	**90.2**	**86.2**	**86.8**

Table 2: **Results on RoboTwin 2.0 benchmark.** All policies are trained on the Clean setting and evaluated under the Randomized (hard) zero-shot setting, with 100 rollouts per task.

Method	Put Bottles Dustbin	Place Dual Shoes	Pick Diverse Bottles	Place Burger Fries	Open Laptop	Blocks Ranking RGB	Pick Dual Bottles	Dump Bin Bigbin	Stamp Seal	Place Bread Basket	Avg.
π_0 Black et al. (2024)	13%	0%	6%	4%	46%	5%	12%	24%	4%	4%	11.8%
$\pi_{0.5}$ Intelligence et al. (2025)	9%	7%	3%	70%	35%	35%	6%	42%	23%	56%	28.6%
X-VLA Zheng et al. (2026)	20%	12%	30%	35%	16%	4%	51%	41%	2%	52%	26.3%
Spatial Forcing Li et al. (2026a)	41%	2%	38%	71%	15%	7%	47%	70%	2%	53%	34.6%
X-WAM Guo et al. (2026)	21%	7%	26%	36%	6%	36%	55%	40%	16%	40%	28.3%
Abot-M0 Yang et al. (2026b)	26%	15%	7%	55%	64%	23%	19%	68%	19%	47%	34.3%
EventVLA Yang et al. (2026a)	31%	1%	9%	7%	23%	6%	8%	64%	3%	15%	16.7%
Xiaomi Robotics-0 Cai et al. (2026b)	21%	9%	12%	27%	24%	8%	28%	49%	3%	23%	20.4%
FastWAM Yuan et al. (2026)	0%	0%	0%	0%	0%	0%	1%	0%	0%	0%	0.1%
GalaxeaVLA Liu et al. (2026)	3%	9%	11%	32%	13%	1%	21%	40%	2%	32%	16.4%
starVLA Community (2026)	0%	2%	15%	11%	2%	2%	20%	4%	0%	5%	6.1%
PROACT	**69%**	**39%**	**57%**	**84%**	**76%**	**46%**	**63%**	**78%**	**29%**	**62%**	**60.3%**

At inference, we construct P_t and H_t^w, predict U_t and B_t, sample $X_{t,\mathrm{src}}^a$, and integrate from $s = 1$ to 0. Starting closer to the action manifold shortens the required transport, allowing a coarser integration grid while reducing both sampling cost and accumulated integration error.

Training objective. The complete model is trained end-to-end with

$$\mathcal{L} = \lambda_p \mathcal{L}_{\mathrm{prop}} + \lambda_f \mathcal{L}_{\mathrm{future}} + \lambda_u \mathcal{L}_U + \lambda_b \mathcal{L}_B + \lambda_a \mathcal{L}_{\mathrm{flow}}. \tag{12}$$

These terms learn a useful proposal, anchor world reasoning in V-JEPA space, calibrate proposal-specific extent and geometry, and transport samples toward executable actions.

3 EXPERIMENTS

We evaluate PROACT through four questions: **(Q1)** Does the world-calibrated, proposal-centered source outperform strong VLA baselines? **(Q2)** Are gains larger when long-horizon or perturbed conditions weaken motion continuity? **(Q3)** Which propose–calibrate–refine components drive the gains? **(Q4)** Does calibration improve proposal refinement and generation efficiency?

3.1 COMPARISON ON SIMULATION BENCHMARKS

This section addresses **Q1** and **Q2** on three complementary benchmarks, testing whether a calibrated source improves strong VLA baselines under saturation, distribution shift, and long-horizon bimanual control across increasingly demanding manipulation conditions.

LIBERO Benchmark. LIBERO Liu et al. (2023) spans spatial, object, goal, and long-horizon variation; following $\pi_{0.5}$ Intelligence et al. (2025), we report 500 rollouts per suite. Tab. 1 shows that PROACT achieves 98.4% average success, 1.5 points above its backbone. While single-stage suites are near saturation, the margin grows to 3.6 points on Long, supporting calibrated source expansion when stage transitions sharply break local motion continuity.

LIBERO-Plus Benchmark. Without retraining, we test seven LIBERO-Plus perturbations Fei et al. (2025) under the Total protocol. Tab. 1 shows that PROACT reaches 86.8%, 13.2 points above $\pi_{0.5}$ and 7.3 above the strongest competing average. It improves every category, with the largest gains on Noise, Camera, and Robot shifts (18.1/16.4/14.8 points), showing that calibration re-anchors motion proposal to altered visual and geometric conditions.

RoboTwin 2.0 Benchmark. We train one policy on all 50 Aloha-AgileX tasks Chen et al. (2025) and evaluate 100 zero-shot *Randomized (Hard)* rollouts per task. Tab. 2 reports the ten largest

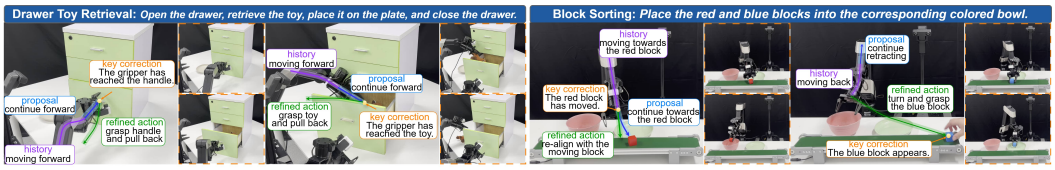


Figure 4: **Closed-loop proposal refinement on real robots.** Drawer Toy Retrieval (left) and Block Sorting (right). Enlarged frames overlay history (purple), proposal (blue), and refined action (green); orange callouts mark changes requiring departure from local motion continuity.

Table 3: **Real-world manipulation results.** Results are reported over the same six tasks on both on GALAXEA R1 Lite Galaxea (2025) and AgileX Cobot Magic Fu et al. (2024) platforms. The superscript † denotes our reproduced baselines.

Galaxea R1 Lite Platform

Method	Block Sorting					Pot Lid Retrieval			Drawer Manipulation				
	Pick Red Block	+Place Red in Bowl	+Pick Blue Block	+Place Blue in Bowl	CSR	Remove Lid	+Retrieve Object to Plate	CSR	Open Drawer	+Grasp Toy	+Place Toy Inside	+Close Drawer	CSR
OpenVLA-OFT Kim et al. (2025)†	19/25	17/25	13/25	12/25	61.0%	21/25	17/25	76.0%	17/25	15/25	12/25	12/25	56.0%
π_0 Black et al. (2024)†	20/25	18/25	14/25	14/25	66.0%	21/25	16/25	74.0%	18/25	17/25	13/25	13/25	61.0%
$\pi_{0.5}$ Intelligence et al. (2025)†	23/25	22/25	19/25	17/25	81.0%	22/25	20/25	84.0%	21/25	21/25	18/25	16/25	76.0%
ProAct	**23/25**	**23/25**	**21/25**	**20/25**	**87.0%**	**24/25**	**21/25**	**90.0%**	**23/25**	**22/25**	**20/25**	**19/25**	**84.0%**

Method	Drawer Toy Retrieval					Table Wiping			Color Sorting				Overall	
	Open Drawer	+Retrieve Toy	+Place Toy on Plate	+Close Drawer	CSR	Grasp Cloth	+Wipe Table	CSR	Red Block→Bin	+Blue Block→Bin	+Yellow Block→Bin	CSR	TSR	CSR
OpenVLA-OFT Kim et al. (2025)†	15/25	9/25	8/25	7/25	39.0%	19/25	12/25	62.0%	22/25	16/25	9/25	62.7%	46.0%	59.4%
π_0 Black et al. (2024)†	19/25	14/25	14/25	12/25	59.0%	22/25	16/25	76.0%	20/25	16/25	13/25	65.3%	56.0%	66.9%
$\pi_{0.5}$ Intelligence et al. (2025)†	22/25	18/25	17/25	16/25	73.0%	24/25	20/25	88.0%	23/25	18/25	16/25	76.0%	70.0%	79.7%
ProAct	**22/25**	**20/25**	**19/25**	**19/25**	**80.0%**	**24/25**	**21/25**	**90.0%**	**24/25**	**22/25**	**18/25**	**85.3%**	**78.7%**	**86.1%**

AgileX Cobot Magic Platform

Method	Block Sorting		Pot Lid Retrieval		Drawer Manipulation		Drawer Toy Retrieval		Table Wiping		Color Sorting		Average	
	TSR	CSR	TSR	CSR	TSR	CSR	TSR	CSR	TSR	CSR	TSR	CSR	TSR	CSR
OpenVLA-OFT Kim et al. (2025)†	44.0%	55.0%	60.0%	70.0%	40.0%	52.0%	28.0%	37.0%	48.0%	62.0%	36.0%	57.3%	42.7%	55.6%
π_0 Black et al. (2024)†	52.0%	63.0%	56.0%	68.0%	48.0%	59.0%	44.0%	54.0%	56.0%	68.0%	48.0%	61.3%	50.7%	62.2%
$\pi_{0.5}$ Intelligence et al. (2025)†	64.0%	73.0%	72.0%	78.0%	60.0%	70.0%	56.0%	65.0%	72.0%	82.0%	64.0%	76.0%	64.7%	74.0%
ProAct	**76.0%**	**82.0%**	**80.0%**	**86.0%**	**72.0%**	**80.0%**	**68.0%**	**75.0%**	**80.0%**	**88.0%**	**72.0%**	**82.7%**	**74.7%**	**82.3%**

margins; full results appear in Appendix B. ProAct averages 60.3%, 31.7 points above $\pi_{0.5}$ and 25.7 above the strongest competing average. Gains across placement, bimanual coordination, and object-dependent grasping show robustness across randomized scenes with unreliable local motion.

3.2 Comparison on Real-World Tasks

This section further addresses **Q1** and **Q2** on six tasks across GALAXEA R1 Lite and AgileX Cobot Magic, with 25 trials each. TSR requires full completion, while CSR retains checkpoint progress; Appendix C.1 provides metric details. All baselines use the same setup, enabling direct comparison across methods and embodiments under identical protocols.

Closed-loop correction on real robots. Fig. 4 shows selective refinement over complete rollouts. Forward proposals become grasp-and-pull motions at contact in Drawer Toy Retrieval, while Block Sorting realigns to displaced or newly relevant blocks. Proposals preserve stable phases, and calibration handles contact and target transitions throughout execution.

Results on both robot platforms. Tab. 3 shows consistent gains over $\pi_{0.5}$: overall TSR/CSR rises by 8.7/6.4 points on GALAXEA and 10.0/8.3 on AgileX, with 12-point TSR gains on three AgileX tasks. The CSR–TSR gap narrows by 2.3/1.7 points, showing that intermediate progress more often survives task completion across both physical embodiments and task settings. The consistent gains across tasks further support the robustness of the refinement mechanism.

3.3 Ablation Study

This section addresses **Q3** through component ablations in simulation and the real world.

Motion-informed source. Tab. 4 separates motion continuity from scene grounding. Reusing the last chunk is slightly worse than Gaussian noise, whereas motion extrapolation gains 0.6/3.3 points on LIBERO/real-world tasks. Proposal Expert grounding adds another 0.6/3.4 points, showing that recent motion is a useful soft prior but needs scene grounding when local continuity is unreliable.

Table 4: **Motion proposal ablation.** Success rates (%) under different source initializations for action generation, from generic noise to the scene-aware motion proposal.

Initialization	LIBERO					Real-World
	Spa.	Obj.	Goal	Long	Avg.	SR
Gaussian $\mathcal{N}(0, I)$	98.8	98.6	97.8	93.6	97.2	72.0
Last Action Chunk	98.4	98.2	97.4	92.8	96.7	70.7
Motion Extrapolation	99.0	99.2	98.0	94.8	97.8	75.3
Scene-aware Proposal	99.4	99.6	98.4	96.0	**98.4**	**78.7**

Table 5: **World calibration ablation.** We report success rates (%) while ablating the World Expert (W), proposal conditioning (P), future supervision (F), and source calibration (C).

Components				LIBERO					Real-World
W	P	F	C	Spa.	Obj.	Goal	Long	Avg.	SR
✓	✓	×	✓	99.0	99.2	98.0	93.6	97.5	74.7
✓	×	✓	✓	98.8	99.0	97.6	92.8	97.1	73.3
✓	✓	✓	×	99.2	99.4	98.2	94.4	97.8	76.0
×	–	–	–	98.6	98.4	97.2	91.6	96.5	70.7
✓	✓	✓	✓	99.4	99.6	98.4	96.0	**98.4**	**78.7**

Table 6: **Source calibration ablation.** Success rates (%) across source variants; U and B denote refinement extent and geometry.

Calibration			LIBERO					Real-World
Source	U	B	Spa.	Obj.	Goal	Long	Avg.	SR
Fixed Isotropic	×	×	98.6	98.8	97.4	92.8	96.9	72.7
Extent Only	✓	×	99.0	99.2	98.0	94.8	97.8	76.0
Geometry Only	×	✓	98.8	99.0	97.6	93.6	97.3	74.7
ProAct	✓	✓	99.4	99.6	98.4	96.0	**98.4**	**78.7**

Table 7: **Inference efficiency.** Latency, throughput, and success rate under default inference on LIBERO (A800, upper rows) and real-world tasks (RTX 5090, lower rows).

Model	Latency↓	Throughput↑	Success Rate↑
$\pi_{0.5}$	121.26 ms	82.47 Hz	96.9%
ProAct	**89.98 ms**	**111.14 Hz**	**98.4%**
$\pi_{0.5}$	104.62 ms	477.92 Hz	70.0%
ProAct	**87.52 ms**	**571.30 Hz**	**78.7%**

World calibration. Tab. 5 validates the asymmetric calibration interface. Removing the World Expert causes the largest drop, by 1.9 points on LIBERO and 8.0 points in the real world. Removing proposal conditioning, future supervision, or source calibration costs 1.3/5.4, 0.9/4.0, and 0.6/2.7 points, respectively. Thus, proposal-aware future reasoning and source calibration are both necessary, especially when real-world motion continuity becomes unreliable. The larger real-world penalties highlight their importance under physical manipulation settings.

Refinement extent and geometry. Tab. 6 shows that the two calibration outputs are complementary. Relative to a fixed isotropic proposal-centered source, adaptive extent improves LIBERO/real-world success by 0.9/3.3 points, while low-rank geometry contributes 0.4/2.0 points. Combining both raises the gains to 1.5/6.0 points. Extent controls correction magnitude, while geometry allocates it across coupled action dimensions. Their complementary gains show that effective refinement needs both adaptive magnitude and structured directionality, especially across embodiments, real-world contact transitions, and long-horizon manipulation settings in practice.

3.4 Qualitative Analysis

This section addresses **Q4** by examining whether the learned proposals and calibrations are interpretable and whether proposal-centered transport improves efficiency during action generation.

Scene-aware proposal grounding. Fig. 5 shows that, despite compressed VLM KV caches, Proposal Expert attention remains focused on manipulated objects and interaction regions across LIBERO and RoboTwin 2.0 while suppressing background distractors. Thus, cache compression preserves task cues for reliable scene-aware proposals during execution.

Calibrated source geometry. Fig. 6 shows uncertainty over action dimensions. From left to right, nearly circular contours indicate no correlation, upward-tilted ellipses indicate positive correlation, and downward-tilted ellipses indicate negative correlation. Dense samples mark source actions, while solid and dashed ellipses denote approximately 1σ and 2σ contours. Together, the panels show how calibration adjusts the source distribution in extent, orientation, and cross-dimensional coupling according to action geometry.

World-calibrated refinement. Fig. 7 shows that the proposal preserves recent motion's coarse direction while refinement bends toward the demonstration within an anisotropic region. Stable dimensions remain close to the proposal, whereas incompatible ones move toward the desired future, consistent with the roles of extent and geometry in Tab. 6 across domains and robot embodiments.

Efficient proposal-to-action transport. Fig. 7 and Tab. 7 show that calibrated geometry shortens transport by starting near the demonstration and focusing corrections along task-relevant direc-

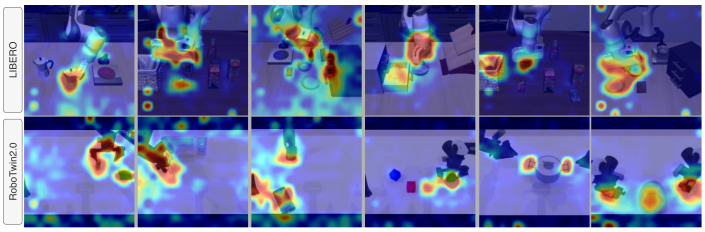


Figure 5: **Visual attention of the Proposal Expert.** Attention maps on LIBERO (top) and RoboTwin 2.0 (bottom); warmer colors highlight objects, interaction regions, and receptacles.

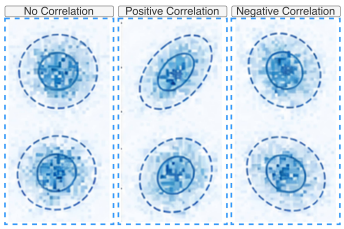


Figure 6: **Calibrated source correlation.** Action samples and correlation ellipses.

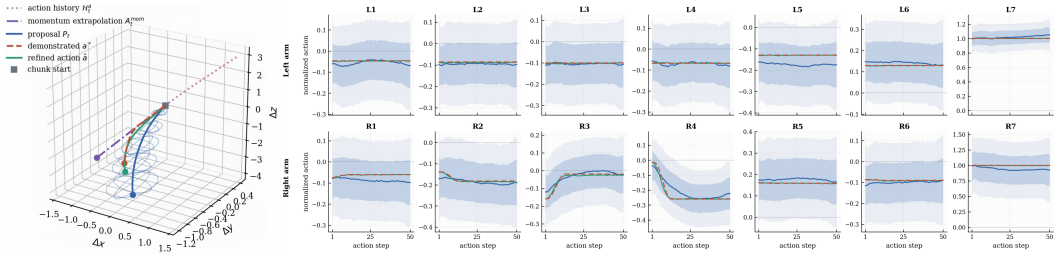


Figure 7: **World-calibrated proposal refinement.** Left: proposal refinement in 3D translation space. Right: normalized trajectories and calibrated support across seven action dimensions.

tions, removing unnecessary transport during generation. On LIBERO/real robots, latency falls by 25.8/16.3%, throughput rises by 34.8/19.5%, and success improves by 1.5/8.7 points.

4 Related Work

Structured Priors for Action Generation. Recent generative robot policies increasingly inject structured priors to simplify action synthesis. Diffusion and flow-based policies typically generate actions from a generic Gaussian source Chi et al. (2025); Black et al. (2024); Intelligence et al. (2025); Bjorck et al. (2025); Zheng et al. (2026); Luo et al. (2026). Recent work instead explores more informative starting points. Streaming Flow Policy Jiang et al. (2025) initializes generation around recently executed actions, while VITA Gao et al. (2026) directly transports visual latents toward action representations. A2A Jia et al. (2026) further encodes historical proprioceptive sequences as an informative flow source. Together, these studies establish source design as an important dimension of generative action policies. However, an informative source does not ensure that its implied motion remains suitable as the task evolves. ProAct therefore treats recent motion as a proposal and determines how strongly and along which directions it should be corrected.

World Models for Action Generation. A complementary line of work augments action policies with predictive world representations to account for future scene evolution. DreamZero Ye et al. (2026), LingBot-VA Li et al. (2026b), DreamVLA Zhang et al. (2026), and WorldVLA Cen et al. (2025) couple future prediction or world modeling with action learning, while X-WAM Guo et al. (2026) incorporates future spatial structure through RGB-D prediction. VLA-JEPA Sun et al. (2026) introduces latent future prediction into VLA learning without explicit pixel-space generation, with related predictive objectives explored in UP-VLA Zhang et al. (2025). Fast-WAM Yuan et al. (2026) further suggests that explicit test-time future generation is not always necessary. Together, these studies leave open how predictive information should most effectively influence action generation. In ProAct, the predicted future calibrates the proposal-centered source, controlling how far and along which directions the proposal is refined.

5 Conclusion

We presented ProAct, a world-calibrated proposal-to-action framework that makes the source of flow-based VLAs context dependent. A scene-aware motion proposal and task-consistent future calibrate refinement extent and geometry while preserving the original flow parameterization. Across simulation and real-world benchmarks, this design improves robustness under perturbations and long-horizon execution while reducing denoising and inference cost, motivating prospective source calibration for robust and efficient generative robot policies across manipulation settings.

6 AI USE DISCLOSURE

Generative AI tools were used solely to assist with English language polishing and to check formatting and presentation consistency in the manuscript. They were not used to develop research ideas, formulate hypotheses or mathematical claims, design the methodology or experiments, implement the proposed method, or analyze and interpret experimental results. All AI-assisted edits were reviewed by the authors, who take full responsibility for the final content of the paper.

REFERENCES

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

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

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

Kevin Black, Manuel Galliker, and Sergey Levine. Real-time execution of action chunking flow policies. *Advances in Neural Information Processing Systems*, 38:33383–33407, 2026.

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*, 2025.

Jisong Cai, Long Ling, Shiwei Chu, Zhongshan Liu, Jiayue Kang, Zhixuan Liang, Wenjie Xu, Yinan Mao, Weinan Zhang, Xiaokang Yang, et al. Aha-wam: Asynchronous horizon-adaptive world-action modeling with observation-guided context routing. *arXiv preprint arXiv:2606.09811*, 2026a.

Rui Cai, Jun Guo, Xinze He, Piaopiao Jin, Jie Li, Bingxuan Lin, Futeng Liu, Wei Liu, Fei Ma, Kun Ma, et al. Xiaomi-robotics-0: An open-sourced vision-language-action model with real-time execution. *arXiv preprint arXiv:2602.12684*, 2026b.

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

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

StarVLA Community. Starvla: A lego-like codebase for vision-language-action model developing. *arXiv preprint arXiv:2604.05014*, 2026.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, et al. Libero-plus: In-depth robustness analysis of vision-language-action models. *arXiv preprint arXiv:2510.13626*, 2025.

Zipeng Fu, Tony Z. Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. In *CoRL*, 2024.

Galaxea. Galaxea r1 lite. *https://galaxea-dynamics.com/*, 2025.

Dechen Gao, Boqi Zhao, Andrew Lee, Ian Chuang, Hanchu Zhou, Hang Wang, Zhe Zhao, Junshan Zhang, and Iman Soltani. Vita: Vision-to-action flow matching policy. In *International Conference on Learning Representations*, volume 2026, pp. 100903–100933, 2026.

Jun Guo, Qiwei Li, Peiyan Li, Zilong Chen, Nan Sun, Yifei Su, Heyun Wang, Yuan Zhang, Xinghang Li, and Huaping Liu. Unified 4d world action modeling from video priors with asynchronous denoising. *arXiv preprint arXiv:2604.26694*, 2026.

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.

Jindou Jia, Gen Li, Xiangyu Chen, Tuo An, Yuxuan Hu, Jingliang Li, Xinying Guo, and Jianfei Yang. Action-to-action flow matching. *arXiv preprint arXiv:2602.07322*, 2026.

Sunshine Jiang, Xiaolin Fang, Nicholas Roy, Tomás Lozano-Pérez, Leslie Pack Kaelbling, and Siddharth Ancha. Streaming flow policy: Simplifying diffusion/flow-matching policies by treating action trajectories as flow trajectories. *arXiv preprint arXiv:2505.21851*, 2025.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

Fuhao Li, Wenxuan Song, Han Zhao, Jingbo Wang, Pengxiang Ding, Donglin Wang, Long Zeng, and Haoang Li. Spatial forcing: Implicit spatial representation alignment for vision-language-action model. In *International Conference on Learning Representations*, volume 2026, pp. 132324–132345, 2026a.

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*, 2026b.

Yingyan Li, Shuyao Shang, Weisong Liu, Bing Zhan, Haochen Wang, Yuqi Wang, Yuntao Chen, Xiaoman Wang, Yasong An, Chufeng Tang, et al. Drivevla-w0: World models amplify data scaling law in autonomous driving. In *International Conference on Learning Representations*, volume 2026, pp. 7890–7911, 2026c.

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*, 2026.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

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

Yicheng Liu, Zibin Dong, Baijun Ye, Tianyuan Yuan, Tao Jiang, Anqi Yang, Shicheng Cao, Haonan Liu, Yue Sun, Zihan Guo, Xiao Liu, Ke Dong, Changxun Pan, Chenru Wu, Tailai Cheng, Xiaoshu Ren, Xinlei Zhang, Jianning Cui, Zijie Zhao, Haoyu Zhang, Kaiming Xu, Haodong Yang, Bowen Zhang, Jiahui Niu, Shaoting Zhu, Shiduo Zhang, and Hang Zhao. G0.5: One autoregressive stream for robot reasoning and action, 2026. URL `https://arxiv.org/abs/2608.11739`.

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*, 2026.

Jingwen Sun, Wenyao Zhang, Zekun Qi, Shaojie Ren, Zezhi Liu, Hanxin Zhu, Guangzhong Sun, Xin Jin, and Zhibo Chen. Vla-jepa: Enhancing vision-language-action model with latent world model. *arXiv preprint arXiv:2602.10098*, 2026.

Yang Tian, Sizhe Yang, Jia Zeng, Ping Wang, Dahua Lin, Hao Dong, and Jiangmiao Pang. Predictive inverse dynamics models are scalable learners for robotic manipulation. In *International Conference on Learning Representations*, volume 2025, pp. 92033–92052, 2025.

Haoxuan Wang, Gengyu Zhang, Yan Yan, Yuzhang Shang, Ramana Kompella, and Gaowen Liu. Real-time robot execution with masked action chunking. In *International Conference on Learning Representations*, volume 2026, pp. 120161–120179, 2026a.

Yihao Wang, Pengxiang Ding, Lingxiao Li, Can Cui, Zirui Ge, Xinyang Tong, Wenxuan Song, Han Zhao, Wei Zhao, Pengxu Hou, et al. Vla-adapter: An effective paradigm for tiny-scale vision-language-action model. In *Proceedings of the AAAI conference on artificial intelligence*, volume 40, pp. 18638–18646, 2026b.

Ganlin Yang, Zhangzheng Tu, Yuqiang Yang, Sitong Mao, Junyi Dong, Tianxing Chen, Jiaqi Peng, Jing Xiong, Jiafei Cao, Jifeng Dai, et al. Eventvla: Event-driven visual evidence memory for long-horizon vision-language-action policies. *arXiv preprint arXiv:2606.20092*, 2026a.

Yandan Yang, Shuang Zeng, Tong Lin, Xinyuan Chang, Dekang Qi, Junjin Xiao, Haoyun Liu, Ronghan Chen, Yuzhi Chen, Dongjie Huo, et al. Abot-m0: Vla foundation model for robotic manipulation with action manifold learning. *arXiv preprint arXiv:2602.11236*, 2026b.

Zhao Yang, Yinan Shi, Mingyuan Yao, Wenyao Xue, Yawei Jueluo, and Longjun Liu. Chunkflow: Towards continuity-consistent chunked policy learning. *arXiv preprint arXiv:2607.12992*, 2026c.

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*, 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.

Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. *arXiv preprint arXiv:2403.03954*, 2024.

Dijia Zhan, Xuemiao Xu, Jinyi Li, and Jie Tang. Seam: Smooth execution of action-chunked motion for vision-language-action policies. *arXiv preprint arXiv:2607.04609*, 2026.

Jianke Zhang, Yanjiang Guo, Yucheng Hu, Xiaoyu Chen, Xiang Zhu, and Jianyu Chen. Up-vla: A unified understanding and prediction model for embodied agent. *arXiv preprint arXiv:2501.18867*, 2025.

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, 2026.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1702–1713. IEEE, 2025.

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.

The appendix is organized as follows:

- Sec. A: Implementation Details.
- Sec. B: Supplementary RoboTwin 2.0 Results.
- Sec. C: Real-World Evaluation Details.
- Sec. D: Extended Qualitative Analysis.
- Sec. E: Derivations.
- Sec. F: Limitations and Future Work.

A Implementation Details

A.1 Model Details

A.1.1 Three-Layer KV Fusion: Instantiation

With $L_v = 18$ VLM layers and $L_p = 6$ Proposal layers we set $\rho_p(j) = 3j - 1$, so $\mathcal{S}_j = \{3j - 2, 3j - 1, 3j\}$ splits the depth into six disjoint centered windows of width $m^\star = 1$; no boundary clamping is required. Disjointness is a convenience, not a consequence of the analysis, which is stated per window. See Appendix E.1 for the derivation of the conditional risk characterization of $m^\star = 1$ and the inverse-variance weights $w_{t,j,l}$ used here.

A.1.2 Calibrated Source: Efficient Evaluation and Sampling

Role of the action scale D_a. The released ProAct data pipeline applies the same coordinate-wise normalization to the target action chunk and to the executed-action history. For the ProAct configurations, this is quantile normalization: with empirical quantiles $q_{.01,i}$ and $q_{.99,i}$ and $\varepsilon_{\mathrm{n}} = 10^{-6}$,

$$d_i = \frac{q_{.99,i} - q_{.01,i} + \varepsilon_{\mathrm{n}}}{2}, \qquad \mu_{a,i} = q_{.01,i} + d_i, \qquad \bar{a}_i = \frac{a_i^{\mathrm{raw}} - \mu_{a,i}}{d_i}. \tag{13}$$

The supported z-score alternative has $d_i = \mathrm{std}(a_i) + \varepsilon_{\mathrm{n}}$ and $\mu_{a,i} = \mathrm{mean}(a_i)$. Hence D_a is a fixed data-statistics scale, not a learned model output. It has two roles. First, it converts the raw residual to the dimensionless residual used by calibration: $D_a^{-1}(a_{t,k}^{*,\mathrm{raw}} - P_{t,k}^{\mathrm{raw}}) = \bar{a}_{t,k}^{*} - \bar{P}_{t,k}$. The additive center cancels, so it never enters the extent or geometry losses. Second, it maps a covariance learned in normalized coordinates back to original units by congruence, $C_{t,k}^{a} = D_a C_{t,k} D_a^\top$.

Importantly, normalization is performed before the ProAct model is called and unnormalization after it returns. Thus the implementation stores $\bar{a}^*$, $\bar{P}$, and $\bar{X}_{\mathrm{src}}$ and does not explicitly construct or apply D_a inside the model; internally, its role is already absorbed and is equivalent to setting $D_a = I$. In particular, the code expression `actions - proposal` is exactly the normalized residual above, not an unnormalized residual missing a factor of D_a^{-1}. Likewise, multiplying the sampled source by D_a inside the model would incorrectly apply the action scale twice.

The geometry objective of equation 9 uses the normalized shape covariance $\widetilde{C}_{t,k}$. The model samples the source using $C_{t,k}$ in normalized coordinates; $C_{t,k}^{a} = D_a C_{t,k} D_a^\top$ is its original-coordinate interpretation after output unnormalization. Neither requires forming a dense $d_a \times d_a$ matrix. Let $\alpha_{t,k} = d_a / \operatorname{tr}(I + B_{t,k} B_{t,k}^\top)$, so that $\widetilde{C}_{t,k} = \alpha_{t,k}(I + B_{t,k} B_{t,k}^\top)$. The Woodbury identity and the matrix determinant lemma give

$$\widetilde{C}_{t,k}^{-1} = \alpha_{t,k}^{-1}\big(I - B_{t,k}(I_r + B_{t,k}^\top B_{t,k})^{-1} B_{t,k}^\top\big)\,, \tag{14}$$

$$\log\det \widetilde{C}_{t,k} = d_a \log \alpha_{t,k} + \log\det(I_r + B_{t,k}^\top B_{t,k}), \tag{15}$$

so both terms cost $O(d_a r^2 + r^3) = O(d_a r^2)$ for $r \le d_a$ and never require a $d_a \times d_a$ inverse.

Sampling is equally cheap: for each k, draw mutually independent $\epsilon_{a,k} \sim \mathcal{N}(0, I_{d_a})$ and $\xi_{t,k} \sim \mathcal{N}(0, I_r)$,

$$\bar{X}_{t,\mathrm{src},k} = \bar{P}_{t,k} + \sigma_{t,k}\sqrt{\alpha_{t,k}}\,\left(\epsilon_{a,k} + B_{t,k}\xi_{t,k}\right), \tag{16}$$

which realizes $C_{t,k}$ without any Cholesky factorization. Applying the output transform gives

$$X^a_{t,\mathrm{src},k} = \mu_a + D_a \bar{X}_{t,\mathrm{src},k} = P^{\mathrm{raw}}_{t,k} + \sigma_{t,k}\sqrt{\alpha_{t,k}}\, D_a(\epsilon_{a,k} + B_{t,k}\xi_{t,k}), \tag{17}$$

and therefore $\mathrm{Cov}(X^a_{t,\mathrm{src},k} \mid P_t, U_t, B_t) = C^a_{t,k} = D_a C_{t,k} D_a^\top$. The block-diagonal Gaussian construction makes the H action steps conditionally independent given (P_t, U_t, B_t). Padding dimensions of the action tensor are excluded from $e_{t,k}$, $\mathcal{L}_U$, and $\mathcal{L}_B$, and d_a counts only the effective action dimensions. See Appendix E.2 for why the projection this evaluation relies on ($S \le S_{\max}$) is necessary.

A.1.3 Action Expert Architecture and Inference

The Action Expert keeps the layer count, width, and adaRMS time conditioning of the base VLA action expert. At layer l, its H action-state tokens attend jointly to three sources of keys and values,

$$\overline{H}^a_{t,s,l} = \mathrm{Attn}\big(Q^a_l(H^a_{t,s,l-1}), \left[K^v_{t,l}; K^w_l; K^a_{t,s,l}\right], \left[V^v_{t,l}; V^w_l; V^a_{t,s,l}\right]\big), \tag{18}$$

where $K^v_{t,l}, V^v_{t,l}$ is the VLM cache at the aligned depth, K^w_l, V^w_l are layer-wise projections of the $H \times N_z$ world tokens H^w_t, and $K^a_{t,s,l}, V^a_{t,s,l}$ are its own projections.

Training-time convention. In equation 11, a demonstration index and valid time index are sampled from the empirical training distribution, $s \sim \mathrm{Uniform}[0,1]$, and ϵ_p, ϵ_a are independent standard Gaussian noises independent of s. In the released configuration, the proposal, extent, and geometry are stop-gradient quantities when forming $X^a_{t,\mathrm{src}}$; consequently, both this source and $v^*_t = X^a_{t,\mathrm{src}} - a^*_t$ are held fixed with respect to their source-producing branches under $\mathcal{L}_{\mathrm{flow}}$. The flow loss still trains the Action Expert and can train the World Expert through the separate world-context conditioning path. The proposal, future, extent, and geometry objectives provide the corresponding direct supervision to the preceding modules. Thus equation 11 is standard conditional flow regression with a fixed sampled source, rather than an objective that can be reduced by moving the source itself. For completeness, the extent loss is $\mathcal{L}_U = H^{-1}\sum_{k=1}^{H}(U_{t,k} - U^*_{t,k})^2$.

Because the VLM prefix never attends to the action stream, the VLM cache and the world tokens are independent of the flow time s; both are computed once per observation and reused across all integration steps. Hence, the added inference cost over the base policy is one Proposal Expert pass and one World Expert pass. Integration follows the base policy exactly: starting from $X^a_{t,\mathrm{src}}$ at $s = 1$, we take uniform Euler steps

$$X^a \leftarrow X^a - \frac{1}{N_{\mathrm{step}}} v^A_\theta \tag{19}$$

down to $s = 0$.

A.2 Training Details

A.2.1 Episode Boundary Handling

Two boundary cases are handled by masking rather than by discarding samples. Near the end of an episode, the future clip is completed by repeating the last available frame. Because temporal attention would allow these padded frames to contaminate every latent token, such samples are excluded from $\mathcal{L}_{\mathrm{future}}$ while keeping all remaining losses.

Near the beginning of an episode, the action history is shorter than M. The missing entries and the motion state are initialized to zero, while gripper dimensions are initialized to the gripper value of the first executed action. At deployment, the motion state is recomputed from scratch over the last M executed actions at every call, ensuring that the analytical proposal is identical during training and inference for consistency across repeated evaluations.

A.2.2 Training Configuration

Tab. 8 summarizes the optimization, loss, architecture, and data configurations used for ProAct. We use the same optimizer and peak learning rate across LIBERO, RoboTwin 2.0, and real-world experiments, while adapting the training schedule and batch size to the scale of each setting. The

Table 8: **Training Configuration of PROACT.**

Setting	LIBERO	RoboTwin 2.0	Real-World
Optimization			
Training steps	60k	30k	30k
Batch size	128	64	64
Optimizer	AdamW, $\beta_1 = 0.9$, $\beta_2 = 0.95$		
Peak learning rate	5×10^{-5}	5×10^{-5}	5×10^{-5}
Warm-up steps	10k	10k	10k
GPUs	4	4	4
Loss Weights (Eq. 12)			
λ_p (proposal)	0.1	0.1	0.1
λ_f (future)	0.01	0.01	0.01
λ_u (extent)	0.1	0.1	0.1
λ_b (geometry)	0.001	0.001	0.001
λ_a (flow)	1.0	1.0	1.0
Architecture and Calibration			
Action horizon H	10	50	50
Action history length M	10	50	50
Proposal Expert depth L_p	6		
World queries N_z	16	4	4
Geometry rank r	4		
Geometry budget $S_{\max}$	5.0		
τ_u	0.35		
$\sigma_{\min}, \sigma_{\max}$	0.05, 1.0		
Momentum β, λ	0.9, 0.9		
Data			
Input cameras	2	3	3
Input resolution	224×224		
Number of arms	1	2	2
Robotic arm type	Franka Emika Panda	Aloha-AgileX	Galaxea R1 Lite & Aloha-AgileX

action history length is matched to the action horizon ($M = H$), with $H = 10$ for LIBERO and $H = 50$ for both RoboTwin 2.0 and real-world experiments. Most architectural and calibration hyperparameters, including the Proposal Expert depth, geometry rank, geometry budget, extent threshold, and momentum coefficients, are shared across all settings; only the number of world queries is adjusted, with $N_z = 16$ for LIBERO and $N_z = 4$ for RoboTwin 2.0 and real-world experiments. The real-world configuration otherwise follows the RoboTwin 2.0 setting.

B SUPPLEMENTARY ROBOTWIN 2.0 RESULTS

To complement the representative results in the main paper, we provide a task-level evaluation over all 50 RoboTwin 2.0 tasks under the Randomized (Hard) setting in Tab. 9. Each task is evaluated with 100 rollouts under the same protocol. The full benchmark covers a diverse range of manipulation behaviors, including interaction, placement, handover, articulated-object manipulation, and multi-object coordination, thereby providing a more comprehensive assessment of robustness under substantial visual and physical variation.

Across all 50 tasks, PROACT achieves the highest average success rate of **47.06%**, exceeding the strong $\pi_{0.5}$ baseline (43.84%) by **3.22 percentage points**. It also substantially outperforms Abot-M0 (30.4%), Spatial Forcing (26.7%), X-WAM (25.8%), and X-VLA (20.9%). At the task level, PROACT improves over $\pi_{0.5}$ on 22 tasks and ties on another four, while achieving the best or tied-best performance among all compared methods on 19 tasks. Notably, the improvements are partic-

Table 9: **Per-task success rates on 50 RoboTwin 2.0 tasks under Randomized (Hard).** All methods use the same protocol. The overall average is reported in the final block.

Method	Adjust Bottle	Beat Block Hammer	Blocks Rank. RGB	Blocks Rank. Size	Click Alarmclock	Click Bell	Dump Bin Bigbin	Grab Roller	Handover Block	Handover Mic
DP3 Ze et al. (2024)	3%	8%	0%	0%	14%	0%	53%	2%	0%	3%
π_0 Black et al. (2024)	56%	21%	5%	1%	11%	3%	24%	80%	8%	13%
$\pi_{0.5}$ Intelligence et al. (2025)	**83%**	**50%**	35%	14%	**93%**	**76%**	42%	89%	19%	18%
X-VLA Zheng et al. (2026)	33%	9%	4%	**25%**	43%	63%	41%	20%	1.3%	1%
Spatial Forcing Li et al. (2026a)	71%	13%	7%	3%	28%	28%	70%	**94%**	2%	2%
X-WAM Guo et al. (2026)	13%	21%	36%	0%	61%	52%	40%	16%	**21%**	8%
Abot-M0 Yang et al. (2026b)	49%	33%	23%	7%	21%	11%	68%	91%	0%	**22%**
EventVLA Yang et al. (2026a)	24%	1%	6%	1%	30%	48%	64%	48%	4%	5%
Xiaomi Robotics-0 Cai et al. (2026b)	9%	16%	8%	1%	58%	57%	49%	12%	1%	6%
FastWAM Yuan et al. (2026)	0%	0%	0%	0%	60%	7%	0%	2%	0%	0%
GalaxeaVLA Liu et al. (2026)	10%	31%	1%	0%	7%	27%	40%	33%	0%	2%
AHA-WAM Cai et al. (2026a)	0%	0%	1%	0%	17%	19%	27%	3%	1%	3%
starVLA Community (2026)	0%	2%	2%	0%	0%	7%	4%	44%	0%	1%
ProAct	75%	28%	**46%**	20%	61%	50%	**78%**	**94%**	11%	10%

Method	Hanging Mug	Lift Pot	Move Can Pot	Move Pillbottle Pad	Move Playingcard Away	Move Stapler Pad	Open Laptop	Open Microwave	Pick Diverse Bottles	Pick Dual Bottles
DP3 Ze et al. (2024)	1%	0%	6%	0%	3%	0%	7%	22%	1%	1%
π_0 Black et al. (2024)	3%	**36%**	21%	1%	22%	2%	46%	50%	6%	12%
$\pi_{0.5}$ Intelligence et al. (2025)	3%	0%	27%	**29%**	**67%**	**18%**	35%	37%	3%	6%
X-VLA Zheng et al. (2026)	2%	15%	0%	3%	5%	0%	16%	5%	30%	51%
Spatial Forcing Li et al. (2026a)	1%	0%	1%	14%	15%	1%	15%	18%	38%	47%
X-WAM Guo et al. (2026)	0%	0%	**34%**	26%	11%	2%	6%	24%	26%	55%
Abot-M0 Yang et al. (2026b)	4%	29%	4%	4%	50%	4%	64%	26%	7%	19%
EventVLA Yang et al. (2026a)	2%	1%	0%	3%	17%	2%	23%	44%	9%	8%
Xiaomi Robotics-0 Cai et al. (2026b)	6%	1%	7%	3%	12%	0%	24%	16%	12%	28%
FastWAM Yuan et al. (2026)	0%	0%	0%	0%	0%	0%	0%	0%	0%	1%
GalaxeaVLA Liu et al. (2026)	1%	0%	5%	2%	13%	1%	13%	13%	11%	21%
AHA-WAM Cai et al. (2026a)	0%	0%	0%	0%	0%	0%	9%	1%	0%	1%
starVLA Community (2026)	0%	0%	0%	0%	0%	0%	2%	1%	15%	20%
ProAct	**10%**	33%	27%	**29%**	55%	13%	**76%**	**54%**	**57%**	**63%**

Method	Place A2B Left	Place A2B Right	Place Bread Basket	Place Bread Skillet	Place Burger Fries	Place Can Basket	Place Cans Plasticbox	Place Container Plate	Place Dual Shoes	Place Empty Cup
DP3 Ze et al. (2024)	2%	0%	1%	0%	18%	2%	3%	1%	0%	1%
π_0 Black et al. (2024)	1%	6%	4%	1%	4%	5%	2%	45%	0%	11%
$\pi_{0.5}$ Intelligence et al. (2025)	**60%**	**57%**	56%	46%	70%	**25%**	47%	**78%**	7%	**86%**
X-VLA Zheng et al. (2026)	15%	13%	52%	17%	35%	9%	60%	39%	12%	16%
Spatial Forcing Li et al. (2026a)	11%	6%	53%	30%	71%	10%	11%	54%	2%	33%
X-WAM Guo et al. (2026)	18%	21%	40%	37%	36%	15%	36%	52%	7%	36%
Abot-M0 Yang et al. (2026b)	26%	22%	47%	25%	55%	13%	36%	62%	15%	43%
EventVLA Yang et al. (2026a)	20%	17%	15%	4%	7%	7%	14%	15%	1%	18%
Xiaomi Robotics-0 Cai et al. (2026b)	6%	2%	23%	11%	27%	7%	31%	59%	9%	12%
FastWAM Yuan et al. (2026)	0%	1%	0%	0%	0%	0%	0%	0%	0%	0%
GalaxeaVLA Liu et al. (2026)	5%	3%	32%	14%	32%	6%	22%	38%	9%	3%
AHA-WAM Cai et al. (2026a)	1%	0%	3%	0%	0%	2%	1%	11%	0%	2%
starVLA Community (2026)	0%	1%	5%	2%	11%	0%	2%	3%	2%	0%
ProAct	50%	44%	**62%**	**48%**	**84%**	11%	**63%**	76%	**39%**	82%

Method	Place Fan	Place Mouse Pad	Place Object Basket	Place Object Scale	Place Object Stand	Place Phone Stand	Place Shoe	Press Stapler	Put Bottles Dustbin	Put Object Cabinet
DP3 Ze et al. (2024)	1%	1%	0%	0%	0%	2%	2%	3%	21%	1%
π_0 Black et al. (2024)	10%	1%	2%	0%	11%	7%	6%	29%	13%	18%
$\pi_{0.5}$ Intelligence et al. (2025)	**36%**	**26%**	**36%**	**49%**	**65%**	**53%**	**61%**	70%	9%	15%
X-VLA Zheng et al. (2026)	1%	0%	6%	6%	43%	5%	39%	**73%**	20%	21%
Spatial Forcing Li et al. (2026a)	7%	9%	16%	4%	32%	1%	**61%**	55%	41%	1%
X-WAM Guo et al. (2026)	6%	6%	20%	14%	28%	4%	31%	20%	21%	16%
Abot-M0 Yang et al. (2026b)	10%	5%	33%	23%	25%	26%	24%	52%	26%	22%
EventVLA Yang et al. (2026a)	5%	6%	29%	24%	32%	3%	10%	32%	31%	**27%**
Xiaomi Robotics-0 Cai et al. (2026b)	3%	1%	15%	2%	24%	11%	29%	21%	21%	5%
FastWAM Yuan et al. (2026)	0%	0%	0%	0%	0%	0%	0%	3%	0%	0%
GalaxeaVLA Liu et al. (2026)	2%	1%	3%	2%	27%	8%	22%	17%	3%	5%
AHA-WAM Cai et al. (2026a)	0%	0%	0%	0%	0%	0%	5%	2%	1%	0%
starVLA Community (2026)	1%	0%	1%	1%	0%	0%	0%	10%	0%	0%
ProAct	28%	25%	35%	34%	61%	32%	53%	57%	**69%**	21%

Method	Rotate QRCode	Scan Object	Shake Bottle Horiz.	Shake Bottle	Stack Blocks Three	Stack Blocks Two	Stack Bowls Three	Stack Bowls Two	Stamp Seal	Turn Switch	**Overall Avg.**
DP3 Ze et al. (2024)	1%	1%	25%	19%	0%	0%	5%	6%	0%	8%	4.96%
π_0 Black et al. (2024)	15%	1%	51%	60%	0%	1%	24%	41%	4%	**23%**	16.34%
$\pi_{0.5}$ Intelligence et al. (2025)	**56%**	**38%**	**100%**	**100%**	16%	56%	35%	66%	23%	6%	43.84%
X-VLA Zheng et al. (2026)	1%	9%	78%	76%	0%	2%	9%	15%	2%	6%	20.9%
Spatial Forcing Li et al. (2026a)	22%	6%	87%	84%	0%	19%	**55%**	**79%**	2%	7%	26.7%
X-WAM Guo et al. (2026)	2%	14%	66%	67%	**26%**	**76%**	38%	67%	16%	4%	25.8%
Abot-M0 Yang et al. (2026b)	6%	17%	85%	84%	10%	42%	44%	65%	19%	20%	30.4%
EventVLA Yang et al. (2026a)	1%	3%	37%	34%	3%	13%	11%	17%	3%	8%	15.7%
Xiaomi Robotics-0 Cai et al. (2026b)	1%	8%	62%	61%	7%	22%	35%	53%	3%	14%	18.2%
FastWAM Yuan et al. (2026)	0%	0%	9%	10%	0%	0%	0%	0%	0%	3%	1.9%
GalaxeaVLA Liu et al. (2026)	2%	6%	46%	45%	1%	9%	5%	27%	2%	8%	12.7%
AHA-WAM Cai et al. (2026a)	0%	0%	15%	14%	1%	2%	0%	5%	0%	13%	3.2%
starVLA Community (2026)	0%	5%	9%	5%	1%	1%	0%	0%	0%	0%	3.2%
ProAct	15%	34%	**100%**	**100%**	**26%**	44%	48%	73%	**29%**	20%	**47.06%**



Figure 8: Representative successful real-world rollouts of PROACT on the six evaluation tasks. Within each task panel, the top row shows GALAXEA R1 Lite and the bottom row shows AgileX Cobot Magic; snapshots in each row are ordered from left to right. The eight snapshots are selected at semantic milestones rather than at uniform temporal intervals, covering repeated sorting, container and drawer manipulation, object transfer, and sustained wiping.

ularly large on several challenging tasks, including *Put Bottles Dustbin* (69% vs. 9%), *Pick Dual Bottles* (63% vs. 6%), *Pick Diverse Bottles* (57% vs. 3%), *Open Laptop* (76% vs. 35%), and *Dump Bin Bigbin* (78% vs. 42%). PROACT also retains strong performance on high-success tasks, reaching 100% on both bottle-shaking variants and 94% on *Grab Roller*. Although the relative advantage varies across individual tasks, the aggregate results show that the proposal–calibrate–refine formulation provides a stronger overall policy across the heterogeneous RoboTwin 2.0 task distribution.

C Real-World Evaluation Details

C.1 Evaluation Protocol and Metrics

We use a unified evaluation setup for all methods. Each policy is tested with the same task instructions, sensing configuration, camera placement, control interface, and randomized initialization protocol. Unless stated otherwise, every task contains $N = 25$ independent trials. Object and robot initial states are sampled from task-specific admissible ranges that are fixed before evaluation and shared across methods. No human correction is allowed once a trial starts.

Each task is represented by an ordered sequence of S_i checkpoints. Let $c_{i,j}^{(n)} \in \{0, 1\}$ denote whether trial n reaches checkpoint j of task i. To enforce sequential completion, we evaluate checkpoint j through the progressive indicator

$$\bar{c}_{i,j}^{(n)} = \prod_{k=1}^{j} c_{i,k}^{(n)}, \qquad \mathrm{SR}_{i,j} = \frac{1}{N} \sum_{n=1}^{N} \bar{c}_{i,j}^{(n)}. \tag{20}$$

Hence, a later checkpoint counts only if all preceding stages are completed.

We summarize real-world performance using two metrics. **Task Success Rate (TSR)** measures the fraction of trials that finish the entire task,

$$\mathrm{TSR}_i = \mathrm{SR}_{i,S_i}, \tag{21}$$

whereas **Cumulative Success Rate (CSR)** measures average checkpoint progress,

$$\mathrm{CSR}_i = \frac{1}{S_i} \sum_{j=1}^{S_i} \mathrm{SR}_{i,j}. \tag{22}$$

TSR therefore reflects complete task execution, while CSR additionally captures how far unsuccessful trials progress before failure. Aggregate suite- and platform-level results are computed by averaging the corresponding task-level metrics.

C.2 Task Descriptions

We deploy ProAct on GALAXEA R1 Lite and AgileX Cobot Magic and evaluate the same six sequential manipulation tasks on both platforms. The suite spans repeated object sorting, container interaction, articulated-object manipulation, retrieval, and contact-rich wiping, jointly testing visual grounding, precise grasping and placement, phase transitions, and robustness to accumulated execution error. Each task is evaluated through the ordered progressive checkpoints below, with identical semantic completion criteria across the two embodiments throughout evaluation.

- **Block Sorting.** *Instruction: "Place the red and blue blocks into the target bowl."* This task evaluates repeated object-centric pick-and-place over a multi-stage sequence, requiring the policy to preserve color identity, maintain the prescribed manipulation order, and recover a reliable grasp after completing the first placement. The four progressive stages are: 1) *Pick Red Block*—securely grasping and lifting the red block; 2) +*Place Red in Bowl*—placing the red block inside the target bowl; 3) +*Pick Blue Block*—locating and grasping the blue block after the scene has changed; and 4) +*Place Blue in Bowl*—placing the blue block inside the target bowl to complete the task.
- **Pot Lid Retrieval.** *Instruction: "Remove the lid and retrieve the object from the pot onto the plate."* This task tests interaction with a covered container and the transition from opening to retrieval. It requires the policy to remove the lid without destabilizing the pot, visually re-ground the previously occluded target, and transfer it to a constrained destination. The two progressive stages are: 1) *Remove Lid*—grasping and moving the lid until the pot is accessible; and 2) +*Retrieve Object to Plate*—extracting the target object and placing it stably within the designated area of the plate.
- **Drawer Manipulation.** *Instruction: "Open the drawer, place the toy inside, and close the drawer."* This task combines articulated-object control with object transport and evaluates whether a policy can maintain task coherence across distinct interaction modes. The four progressive stages are: 1) *Open Drawer*—grasping the handle and opening the drawer sufficiently; 2) +*Grasp Toy*—releasing the handle and securely grasping the toy; 3) +*Place Toy Inside*—placing the toy fully inside the open drawer; and 4) +*Close Drawer*—closing the drawer to its terminal state without displacing the inserted toy.
- **Drawer Toy Retrieval.** *Instruction: "Open the drawer, retrieve the toy, place it on the plate, and close the drawer."* This complementary drawer task tests target discovery under partial occlusion, collision-aware extraction from a confined space, and restoration of the articulated object after retrieval. The four progressive stages are: 1) *Open Drawer*—opening the drawer until the interior is accessible; 2) +*Retrieve Toy*—grasping and extracting the toy from the drawer; 3) +*Place Toy on Plate*—transferring the toy onto the plate; and 4) +*Close Drawer*—returning the drawer to its closed state.

Table 10: **Per-checkpoint real-world results on AgileX Cobot Magic.** For each of the six tasks we report the number of successful trials (out of 25) at every progressive checkpoint, so that a checkpoint is credited only when all preceding stages succeed. CSR is the mean success rate across a task's checkpoints, and its final checkpoint equals the task TSR. Best entries per column are in **bold**.

Method	Block Sorting					Pot Lid Retrieval			Drawer Manipulation				
	Pick Red Block	+Place Red in Bowl	+Pick Blue Block	+Place Blue in Bowl	CSR	Remove Lid	+Retrieve Object	CSR	Open Drawer	+Grasp Toy	+Place Toy Inside	+Close Drawer	CSR
OpenVLA-OFT Kim et al. (2025)†	17/25	15/25	12/25	11/25	55.0%	20/25	15/25	70.0%	16/25	15/25	11/25	10/25	52.0%
π_0 Black et al. (2024)†	19/25	17/25	14/25	13/25	63.0%	20/25	14/25	68.0%	18/25	16/25	13/25	12/25	59.0%
$\pi_{0.5}$ Intelligence et al. (2025)†	21/25	19/25	17/25	16/25	73.0%	21/25	18/25	78.0%	20/25	19/25	16/25	15/25	70.0%
PROACT	**22/25**	**21/25**	**20/25**	**19/25**	**82.0%**	**23/25**	**20/25**	**86.0%**	**22/25**	**21/25**	**19/25**	**18/25**	**80.0%**

Method	Drawer Toy Retrieval					Table Wiping			Color Sorting			
	Open Drawer	+Retrieve Toy	+Place Toy on Plate	+Close Drawer	CSR	Grasp Cloth	+Wipe Table	CSR	Red Block→Bin	+Blue Block→Bin	+Yellow Block→Bin	CSR
OpenVLA-OFT Kim et al. (2025)†	14/25	8/25	8/25	7/25	37.0%	19/25	12/25	62.0%	20/25	14/25	9/25	57.3%
π_0 Black et al. (2024)†	18/25	13/25	12/25	11/25	54.0%	20/25	14/25	68.0%	19/25	15/25	12/25	61.3%
$\pi_{0.5}$ Intelligence et al. (2025)†	20/25	16/25	15/25	14/25	65.0%	23/25	18/25	82.0%	22/25	19/25	16/25	76.0%
PROACT	**21/25**	**19/25**	**18/25**	**17/25**	**75.0%**	**24/25**	**20/25**	**88.0%**	**23/25**	**21/25**	**18/25**	**82.7%**

- **Table Wiping.** *Instruction: "Grasp the cloth and wipe the table."* This contact-rich task evaluates deformable-object grasping and sustained surface interaction. Beyond reaching the cloth, the policy must establish a stable grasp, bring the cloth into contact with the tabletop, and execute a coherent wiping trajectory over the designated region. The two progressive stages are: 1) *Grasp Cloth*—securely picking up the cloth; and 2) *+Wipe Table*—using the grasped cloth to complete the required wiping motion while maintaining effective surface contact throughout the designated trajectory.
- **Color Sorting.** *Instruction: "Sort the red, blue, and yellow blocks into the matching bins."* This task evaluates language-conditioned color grounding and repeated placement over a long action sequence. The policy must preserve the association between each object and its color-matched receptacle while operating in a scene that changes after every placement. The three progressive stages are: 1) *Red Block → Bin*—placing the red block into the matching bin; 2) *+Blue Block → Bin*—subsequently placing the blue block into its matching bin; and 3) *+Yellow Block → Bin*—placing the yellow block into its matching bin, thereby completing the full three-object sorting sequence.

C.3 Qualitative Real-World Rollouts

Fig. 8 provides a compact view of representative successful rollouts on GALAXEA R1 Lite and AgileX Cobot Magic. The six panels correspond to Block Sorting, Pot Lid Retrieval, Drawer Manipulation, Drawer Toy Retrieval, Table Wiping, and Color Sorting, with the top and bottom rows showing the two platforms and snapshots ordered from left to right. This selection covers repeated sorting, container interaction, articulated manipulation, object retrieval, and contact-rich wiping. In particular, the pot and drawer sequences make re-grounding after the scene changes visible, while Table Wiping shows sustained surface contact and the sorting tasks show repeated object localization and placement. Together, these sequences provide a qualitative view of how PROACT maintains coherent progress through distinct interaction phases.

C.4 Detailed Real-World Results and Ablations

We complement the aggregate comparison in Section 3.2 with per-checkpoint results on AgileX Cobot Magic (Tab. 10) and a checkpoint-level ablation on GALAXEA R1 Lite (Fig. 9). The progressive view isolates *where* along a task the improvements and the individual component contributions arise, which the task-level TSR/CSR summaries cannot expose.

Gains concentrate at transition checkpoints. The initial acquisition checkpoints are already reliable for every method: all policies open the drawer, remove the lid, or pick the first block on most trials, so the advantage of PROACT accrues almost entirely at the later stages that require re-grounding after the scene changes. On Drawer Toy Retrieval, PROACT loses only two trials at the Open → Retrieve transition ($21 \to 19$), whereas $\pi_{0.5}$ loses four ($20 \to 16$) and OpenVLA-OFT col-

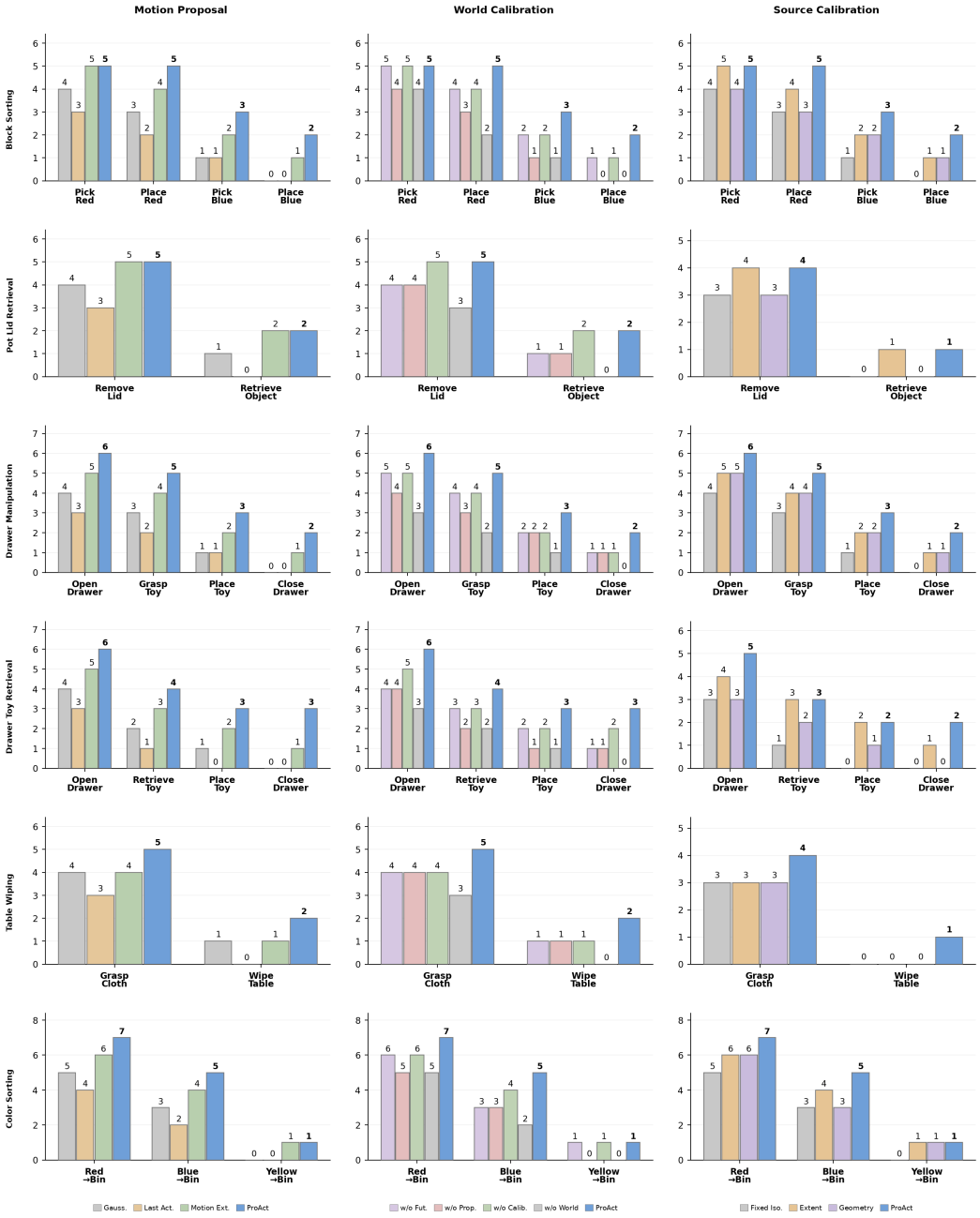


Figure 9: **Per-checkpoint real-world ablations across the six GALAXEA R1 Lite tasks.** Each panel plots per-checkpoint success for one ablation axis: motion proposal (left), world calibration (middle), and source calibration (right). Within every panel the full PROACT (blue) attains the highest bar, and its margin widens at the later transition and terminal checkpoints where local motion continuity becomes unreliable **across the full execution horizon**.

lapses ($14 \rightarrow 8$); the same pattern produces the $+10$-point CSR gaps on both drawer tasks, the largest in the suite. Because a checkpoint is credited only after all earlier ones succeed, these mid-sequence savings compound, so PROACT carries more trials into the terminal checkpoint on all six tasks — the effect that narrows the CSR–TSR gap reported in the main text.

Fewer late-stage failures. Read as survival curves, the PROACT rows decay most slowly: across the four-checkpoint tasks it forfeits at most three trials from the first to the last checkpoint (e.g. Block Sorting $22 \rightarrow 19$), against five for $\pi_{0.5}$ ($21 \rightarrow 16$) and six or more for the weaker baselines. The terminal contact-rich stages benefit most — +Close Drawer, +Wipe Table, and +Yellow→Bin

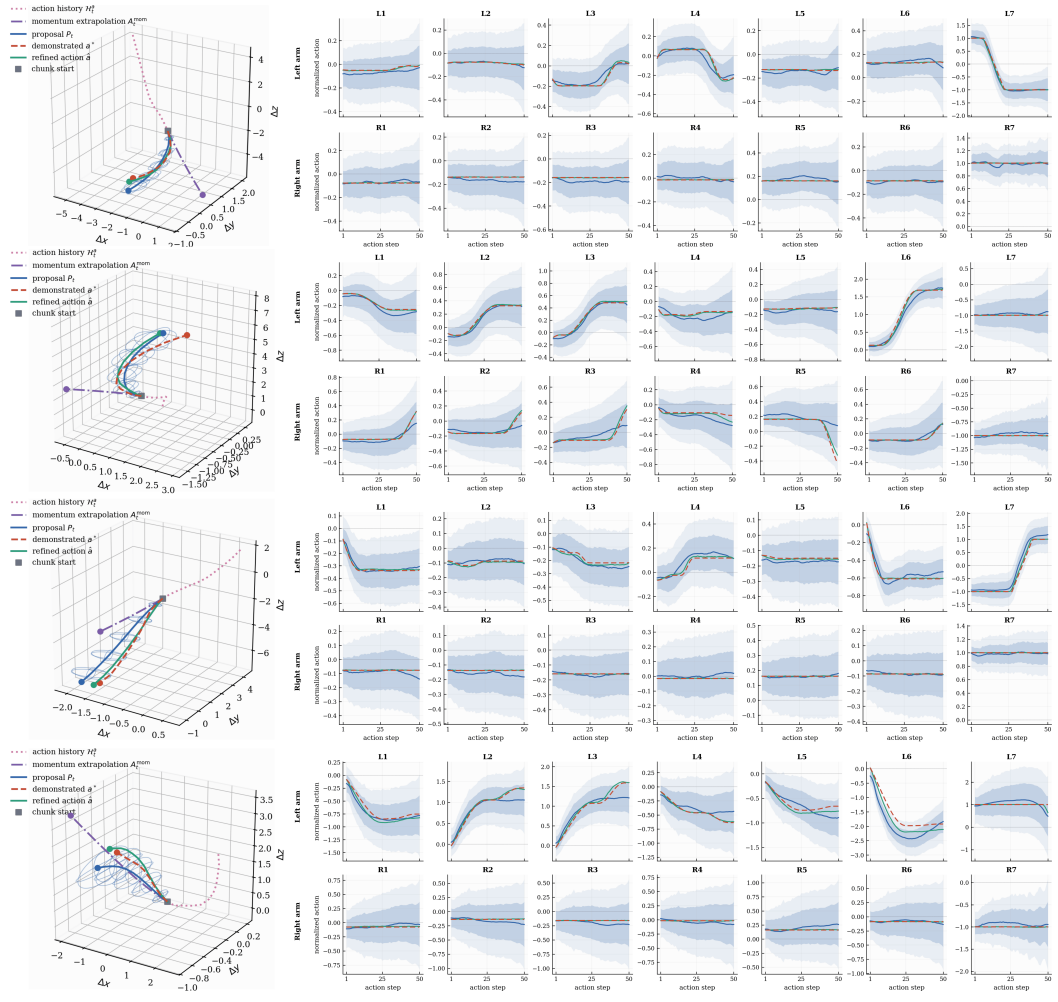


Figure 10: **Additional world-calibrated proposal refinements.** Further examples in the same format as Fig. 7, illustrating refinement behavior across more cases and execution phases.

each improve while their preceding stages are near-saturated — consistent with calibration acting precisely when the proposal must depart from recent motion.

Components matter at the transitions, not the grasps. Fig. 9 decomposes each ablation axis by checkpoint. Removing any single component leaves the first checkpoint of every task essentially unchanged — initial grasps and openings survive — but erodes the later checkpoints, so the aggregate TSR drop is almost entirely a terminal-checkpoint effect **over the full task horizon**. The World Expert is the exception: its removal already depresses the earliest checkpoints and then decays fastest, which is why it is the only ablation to lower performance across the whole sequence **from early to late stages** and yields the largest aggregate loss (-8.0 points in Tab. 5).

Extent carries more of the real-world gain. Within the source-calibration axis, both partial variants match ProAct on the initial grasps but trail it at the later checkpoints, and Extent Only ends higher than Geometry Only (76.0 vs. 74.7 real-world SR in Tab. 6). Adaptive magnitude therefore accounts for more of the real-robot improvement than low-rank geometry **throughout the later execution stages**, while only their combination recovers ProAct across the full checkpoint sequence — the checkpoint-level counterpart of the complementary roles isolated in that table.

D Extended Qualitative Analysis

Calibration adapts to each case. Fig. 10 collects several additional refinements in the format of Fig. 7, drawn from different execution phases **and diverse interaction conditions**. The same pattern

recurs across all of them: the calibrated support is anisotropic and phase-dependent, widening on the steps and dimensions where the proposal (blue) must leave the momentum prior and separate from the demonstration (dashed red), and tightening once the two agree.

Refinement moves only where it must. The refined trajectory (green) stays on the proposal where recent motion is already reliable and bends toward the demonstration only inside the widened support. This consistency across more examples and execution phases **under varying manipulation conditions** corroborates the quantitative trends of Section 3.3, indicating that the calibration responds to the local reliability of recent motion rather than applying a fixed correction.

E Derivations

Conventions and regularity conditions. Normalized and raw actions satisfy $\bar{a} = D_a^{-1}(a^{\mathrm{raw}} - \mu_a)$, where $D_a = \mathrm{diag}(d_1, \dots, d_{d_a}) \succ 0$; d_a is the number of effective action dimensions and d_i the scale of coordinate i. We assume $0 \le \beta, \lambda < 1$, $\tau_u > 0$, $0 < \sigma_{\min} \le \sigma_{\max} < \infty$, $d_a > 1$, $1 \le r < d_a$, and $0 \le S_{\max} < \infty$; H is even when temporal tubelets are used. Gaussian noises are mutually independent across sources and horizon steps unless stated otherwise. The stop-gradient sg is the identity forward and has zero derivative backward. Equation 1 suppresses the perception–language context; deterministic modules correspond to Dirac conditional kernels.

E.1 Three-Layer KV Fusion: Risk-Optimal Window and Weights

The locality and weights of $(\widetilde{K}^v_{t,j}, \widetilde{V}^v_{t,j})$ in Equation 3 follows from one risk. Let

$$Y_{t,l} = \mathrm{RMSNorm}\big([K^v_{t,l}; V^v_{t,l}]\big)\,, \tag{23}$$

and $c_j = \rho_p(j)$. On every admissible window we assume **(A1)** the conditional quadratic-mean model

$$Y_{t,c_j+r} = C_{t,j} + rG_{t,j} + \frac{r^2}{2} H_{t,j} + E_{t,c_j+r}, \tag{24}$$

where C, G, H are the local value and depth variations (a discrete working model, not a Taylor identity); **(A2)** at fixed depth and token layout, $\mathbb{E}[E_{t,l}] = 0$, $\mathbb{E}\|E_{t,l}\|_F^2 = \sigma_{t,l}^2 \in (0, \infty)$, $\mathbb{E}\langle E_{t,l}, E_{t,r}\rangle_F = 0$ for $l \neq r$, and $\langle G_{t,j}, H_{t,j}\rangle_F = 0$; and **(A3)** the estimand is the smooth context $C_{t,j}$, not the observed Y_{t,c_j}. An omitted deterministic higher-order term contributes bias rather than noise. Below write $\gamma_{t,j} = \|G_{t,j}\|_F$ and $\kappa_{t,j} = \|H_{t,j}\|_F$, suppressing their indices when unambiguous.

For $\mathcal{S}_j^{(m)} = \{c_j - m, \dots, c_j + m\}$ and $\widehat{C}^{(m)}(w) = \sum_r w_r Y_{t,c_j+r}$ with $\sum_r w_r = 1$, (A1)–(A2) give

$$\mathcal{R}(m, w) = \mathbb{E}\|\widehat{C}^{(m)}(w) - C_{t,j}\|_F^2 = \Big(\sum_r w_r r\Big)^2 \gamma^2 + \frac{1}{4}\Big(\sum_r w_r r^2\Big)^2 \kappa^2 + \sum_r w_r^2 \sigma^2_{t,c_j+r}, \tag{25}$$

from which we derive the window at uniform w and the weights at fixed $m = 1$.

Window size. For homoscedastic noise and $w_r = (2m+1)^{-1}$, $\sum_r w_r r = 0$ and $\sum_{r=-m}^{m} r^2 = m(m+1)(2m+1)/3$, hence

$$\mathcal{R}(m) = \frac{\sigma^2}{2m+1} + \frac{\kappa^2 m^2 (m+1)^2}{36}. \tag{26}$$

In particular, $\mathcal{R}(0) = \sigma^2$, $\mathcal{R}(1) = \sigma^2/3 + \kappa^2/9$, and $\mathcal{R}(2) = \sigma^2/5 + \kappa^2$. Moreover,

$$\Delta(m) := \mathcal{R}(m+1) - \mathcal{R}(m) = -\frac{2\sigma^2}{(2m+1)(2m+3)} + \frac{\kappa^2 (m+1)^3}{9}, \tag{27}$$

is strictly increasing in m. Thus $\mathcal{R}(1) < \mathcal{R}(0)$ iff $\kappa^2 < 6\sigma^2$, while $\mathcal{R}(1) < \mathcal{R}(2)$ iff $\Delta(1) > 0$ iff $\kappa^2 > 3\sigma^2/20$; the latter also implies $\Delta(m) > 0$ for all $m \ge 1$. Therefore, under (A1)–(A3),

$$\frac{3}{20}\sigma^2 < \kappa^2 < 6\sigma^2 \quad \Longrightarrow \quad m^\star = 1,\ |\mathcal{S}_j^\star| = 3. \tag{28}$$

and $m^\star = 1$ is unique throughout the open interval; an endpoint gives a tie. This is a conditional characterization, and one forward pass does not identify κ^2/σ^2.

Weights. At $m = 1$, linear exactness gives $w_{\pm 1} = u$, $w_0 = 1 - 2u$ and

$$u^\star = \frac{2\sigma^2_{t,c_j}}{\kappa^2 + \sigma^2_{t,c_j-1} + \sigma^2_{t,c_j+1} + 4\sigma^2_{t,c_j}}. \tag{29}$$

This optimum requires unknown κ and enforces symmetric neighbors. For the implemented weights we therefore use, only here, **(A4)** local stationarity: $G_{t,j} = H_{t,j} = 0$ on $\mathcal{S}_j = \{c_j - 1, c_j, c_j + 1\}$. Minimizing $\sum_l w_l^2 \sigma^2_{t,l}$ subject to $\sum_l w_l = 1$ then gives $w_l^\star \propto \sigma_{t,l}^{-2}$; the nonstationary bias is quantified next.

Estimating the variances. Under (A4), let $\bar{Y}_{t,j} = \frac{1}{3}\sum_{l\in\mathcal{S}_j} Y_{t,l}$ and $d_{t,j,l} = \|Y_{t,l} - \bar{Y}_{t,j}\|_F^2$. Then

$$\mathbb{E}[d_{t,j,l}] = \frac{1}{3}\sigma^2_{t,l} + \frac{1}{9}\sum_{r\in\mathcal{S}_j} \sigma^2_{t,r}, \qquad \text{i.e.} \qquad \mathbb{E}[\boldsymbol{d}_{t,j}] = A\,\boldsymbol{\sigma}^2_{t,j}, \quad A = \tfrac{1}{3}I_3 + \tfrac{1}{9}\mathbf{1}\mathbf{1}^\top. \tag{30}$$

The vectors use the order $(c_j - 1, c_j, c_j + 1)$. Thus raw dispersion contains a common offset, whereas $A^{-1} = 3I_3 - \frac{1}{2}\mathbf{1}\mathbf{1}^\top$ yields the unbiased method-of-moments estimator

$$\widehat{\sigma}^2_{t,j,l} = 3d_{t,j,l} - \frac{1}{2}\sum_{r\in\mathcal{S}_j} d_{t,j,r}, \tag{31}$$

under (A2),(A4). Under the quadratic mean (A1), its exact bias is

$$\mathbb{E}\big[\widehat{\boldsymbol{\sigma}}^2_{t,j} - \boldsymbol{\sigma}^2_{t,j}\big] = \left(2\gamma^2_{t,j}, \frac{1}{4}\kappa^2_{t,j} - \gamma^2_{t,j}, 2\gamma^2_{t,j}\right)^\top. \tag{32}$$

Hence the estimate is approximately unbiased only when these terms are small relative to the layer variances. The resulting parameter-free fusion is

$$w_{t,j,l} = \frac{\max(\widehat{\sigma}^2_{t,j,l}, \varepsilon_{t,j})^{-1}}{\sum_{r\in\mathcal{S}_j} \max(\widehat{\sigma}^2_{t,j,r}, \varepsilon_{t,j})^{-1}}, \qquad [\widetilde{K}^v_{t,j}; \widetilde{V}^v_{t,j}] = \sum_{l\in\mathcal{S}_j} w_{t,j,l} Y_{t,l}. \tag{33}$$

Here $\varepsilon_{t,j} > 0$ combines a fixed fraction of mean dispersion with a positive machine floor. Indeed, $\widehat{\sigma}^2_l \le 0$ iff $d_l \le \frac{1}{5}\sum_{r\neq l} d_r$, while $\sum_l \widehat{\sigma}^2_l = \frac{3}{2}\sum_l d_l \ge 0$; the floor therefore keeps all weights finite even when all caches coincide.

For the instantiation of $(\rho_p, \mathcal{S}_j)$ actually used with $L_v = 18$ and $L_p = 6$, see Appendix A.1.1.

E.2 Calibrated Source: Well-Posedness

Degeneracy and projection. Suppress (t, k) and write $d = d_a$. Equation 9 scores $\widetilde{C}$ against one residual $\widehat{r}$. For $u = \widehat{r}/\|\widehat{r}\|$ when $\widehat{r} \neq 0$ (any unit u otherwise), set $B = sue_1^\top$. Then

$$\widehat{r}^\top \widetilde{C}^{-1}\widehat{r} = \frac{\|\widehat{r}\|^2(d + s^2)}{d(1 + s^2)} \longrightarrow \frac{\|\widehat{r}\|^2}{d}, \tag{34}$$

$$\log\det\widetilde{C} = d\log\frac{d}{d + s^2} + \log(1 + s^2) = O(1) - 2(d - 1)\log s. \tag{35}$$

Thus the unconstrained pointwise objective is unbounded below; trace normalization fixes scale but not conditioning. An amortized empirical head would additionally need enough capacity to align with each residual, but the following projection removes the degeneracy without that capacity assumption:

$$\Pi(B) = B\min\left\{1, \frac{\sqrt{S_{\max}}}{\|B\|_F}\right\}, \qquad \Pi(0) := 0, \tag{36}$$

followed by $B \leftarrow \Pi(B)$, so $\|B\|_F^2 \le S_{\max}$. If s_i are the singular values of B, then $r < d$ leaves $d - r$ unit eigenvalues and

$$\kappa(\widetilde{C}_{t,k}) = 1 + \max_i s_i^2 \le 1 + \sum_i s_i^2 \le 1 + S_{\max}. \tag{37}$$

Recomputing $\alpha_{t,k} = d/(d + \|B\|_F^2)$ after projection preserves $\operatorname{tr} C_{t,k} = d\sigma_{t,k}^2$. Moreover, $C_{t,k}^a = D_a C_{t,k} D_a^\top$ is positive definite and $\kappa(C_{t,k}^a) \le \kappa(D_a)^2(1 + S_{\max})$; the sharper $1 + S_{\max}$ bound applies in normalized coordinates.

Lower bound. Let $S = \|B\|_F^2 = \sum_i s_i^2$. Since $\prod_i (1 + s_i^2) \ge 1 + S$,

$$\begin{aligned} \log\det \widetilde{C} &= d \log \frac{d}{d+S} + \sum_i \log(1 + s_i^2) \\ &\ge g(S) := d \log \frac{d}{d+S} + \log(1+S), \qquad g'(S) = -\frac{(d-1)S}{(d+S)(1+S)} \le 0. \end{aligned} \tag{38}$$

Using $S \le S_{\max}$ and the nonnegative Mahalanobis term gives

$$\mathcal{L}_B \ge \frac{1}{2}\left[d_a \log \frac{d_a}{d_a + S_{\max}} + \log(1 + S_{\max}) \right]. \tag{39}$$

For $d_a = 7$, $S_{\max} = 5$, the bound is -0.991. Equality requires zero residual, $S = S_{\max}$, and one nonzero singular direction; a nonzero residual has a strictly higher sample-wise minimum. Thus negative losses are possible, but divergence to $-\infty$ is excluded.

For the $O(d_a r^2)$ evaluation and sampling that this projection enables, see Appendix A.1.2.

F Limitations and Future Work

While ProAct improves generative action prediction by calibrating motion-informed proposals with prospective world representations, several limitations remain.

Counterfactual supervision. The current formulation is trained primarily from factual expert trajectories, where each action sequence is paired with the future evolution that actually occurred. As a result, the model does not explicitly observe the counterfactual world evolution that would be induced by alternative or erroneous proposals. The calibration module therefore learns how a proposal should be adjusted under expert future supervision, rather than directly comparing the predicted consequence of a proposal against the desired task-consistent future. A promising direction is to incorporate proposal perturbations, simulator rollouts, or action-conditioned world models to provide explicit counterfactual supervision and enable consequence-aware calibration.

Fixed-horizon prospective reasoning. Although ProAct is supervised with a temporal future sequence rather than a single endpoint, its prospective reasoning is currently performed over a fixed horizon aligned with the action chunk. However, different manipulation stages may require different temporal scales: fine-grained contact and alignment can often be resolved from short-term evolution, whereas the consequences of decisions in long-horizon tasks may only become evident after several action chunks. Extending the current formulation toward multi-scale or adaptive-horizon world reasoning could allow the policy to calibrate proposals using both immediate interaction cues and longer-term task progress **across diverse manipulation phases and tasks**.