VERIFIABLE 4D WORLD-ACTION MODEL FOR ROBUST ROBOT CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

World-action models couple imagined visual futures with robot actions, but RGB-only imagination requires recovering geometry and motion from appearance that can change across scenes. We propose S4NDBOX, a 4D world-action model that jointly predicts RGB, depth, view-specific geometric motion fields, and robot actions, explicitly connecting visual evolution to 3D robot motion. The explicit 4D future supports action generation and candidate search based on geometric agreement between predicted futures and paired actions before execution. Across both simulated and real-world distribution shifts, S4NDBOX improves robustness over RGB-only control. On RoboTwin, after training only on clean scenes, S4NDBOX improves success in the hard (domain-randomized) setting by 13.3 percentage points over the RGB-only baseline. We further evaluate zero-shot simulation-to-real transfer on Stack Cups, training on simulated demonstrations and deploying on a physical robot without real-world task demonstrations or fine-tuning. Under additional shifts in object pose, lighting, and clutter, with camera-pose shifts additionally evaluated in the single-view setting, our 4D model improves the mean task score over RGB-only by 41.7 and 40.0 points in three-view and single-view settings, respectively. Website: https://demo1-indol-gamma.vercel.app/.

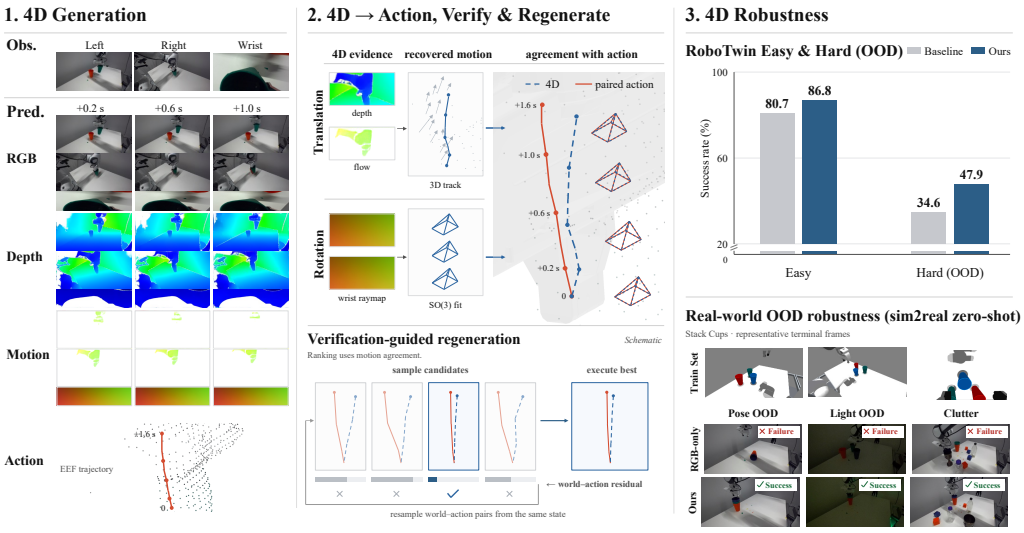

Figure 1: **S4NDBOX structures visual imagination as a control-aligned 4D future. Imagine**: given the observation, S4NDBOX predicts a 4D visual future whose geometry and motion inform action generation. **Verify**: the predicted future and paired action describe the same robot evolution, enabling world-action consistency checking and candidate selection before execution. **Act**: the robot executes the selected actions. S4NDBOX improves generalization to RoboTwin's Hard (domain-randomized) setting and zero-shot simulation-to-real deployment with additional pose, lighting, clutter, and camera-pose changes.

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.

1 INTRODUCTION

Deploying manipulation policies in new scenes requires generalizing across differences in visual appearance and scene configuration, including when transferring zero-shot from simulation to the real world. World-action models (WAMs) use predictions of how scenes evolve and robots move to support action generation (Du et al., 2023; Li et al., 2026a; Ye et al., 2026). To guide actions, these predictions need to convey how spatial relationships between the robot and objects evolve. However, when these futures are represented mainly in RGB, as in most existing WAMs, the action model must implicitly infer geometry and motion relevant to control from predictions whose appearance can change across environments. For example, aligning a gripper with a cup requires inferring their relative position and orientation in three dimensions from the predicted images.

Explicitly predicting geometry and motion offers a promising approach, as explored by recent WAMs that incorporate geometry (Zhen et al., 2025; Guo et al., 2026; Zhao et al., 2026; Yang et al., 2026). By explicitly representing spatial structure and its evolution, depth and motion may support generalization across novel scenes by capturing task-relevant geometric and dynamic structure that remains consistent despite changes in visual appearance. This is particularly relevant to transfer from simulation to reality, where explicit geometry may reduce the burden of recovering these relationships from appearance alone. When object poses and scene layouts also change, depth and motion make the resulting spatial relationships and their evolution explicit, providing information for adapting actions to the new configuration. Joint supervision of appearance, depth, and motion encourages predictions to describe a geometrically coherent future, yet coherence among these predictions does not ensure that the paired action agrees with the robot motion they describe.

We propose S4NDBOX, a 4D WAM that augments RGB imagination with explicit geometry and motion to support action generation and enable verification through geometric agreement. As illustrated in Fig. 1, S4NDBOX jointly predicts future RGB, metric depth, and view-specific motion fields, and generates actions conditioned on the latent representation of this predicted future. These modalities serve complementary roles: depth anchors visual prediction in metric 3D structure, and optical flow from fixed cameras captures projected robot and object motion. For a camera mounted on the robot wrist, predicted world-frame ray directions expose wrist orientation, which is associated with end-effector rotation. Together, these predictions make the geometry and motion of the imagined future explicit to the action model, rather than requiring them to be inferred from RGB alone.

Beyond informing action generation, explicit geometry and motion allow candidate actions to be checked against the futures they are intended to realize. The predicted future and its paired action describe the same robot evolution from complementary representations and should therefore agree geometrically. We quantify this world-action agreement by comparing the projected robot motion and wrist orientation decoded from the predicted future with the corresponding quantities implied by the paired action. We investigate this agreement as a proxy for candidate quality, using it to rank future-action hypotheses sampled from the same state. This gives the predicted future an additional role in control: guiding search among candidate actions before execution, including when deploying in unfamiliar scenes.

We evaluate S4NDBOX under two forms of distribution shift: generalization to RoboTwin domain randomization and zero-shot simulation-to-real transfer under additional real-world shifts. On RoboTwin, policies are trained only on clean demonstrations and evaluated in the Easy (clean) and Hard (domain-randomized) settings, where scene appearance and layout are randomized. S4NDBOX improves Hard-setting success by 13.3% points over the RGB-only baseline. We further evaluate Stack Cups on a Franka Research 3, training on simulated demonstrations and deploying directly in the real world under additional shifts in object pose, lighting, and clutter, with camera-pose shifts additionally evaluated in the single-view setting. Compared with RGB-only control, our 4D policy improves the mean graded task score by 41.7 points with three views and 40.0 points with a single view. Beyond task success, we analyze the geometric coherence of predicted robot evolution. Experiments in randomized simulated scenes suggest that ranking sampled actions by agreement can prioritize successful candidates under limited execution budgets.

Our main contributions are as follows:

- We propose S4NDBOX, a 4D WAM that jointly predicts RGB, metric depth, motion fields, and robot actions. Joint prediction across appearance, geometry, and motion provides complementary constraints on the same imagined future.

- We develop an autoregressive 4D-to-action formulation in which the predicted 4D future precedes and informs action generation. Fixed-view flow and wrist-orientation rays expose motion quantities related to robot translation and rotation, enabling both action generation and pre-execution world-action consistency checks.
- We demonstrate improved generalization in zero-shot simulation-to-real transfer and under simulated scene distribution shifts, including generalization to RoboTwin's Hard (domain-randomized) setting.

2 Related Work

Policies and world–action models. Diffusion Policy and DP3 learn action distributions from visual and 3D observations (Chi et al., 2023; Ze et al., 2024). Generalist policies broaden task and environment coverage through large-scale robot learning and vision–language pretraining (Brohan et al., 2022; 2023), with diverse datasets such as DROID supporting improved generalization (Khazatsky et al., 2024). UniPi and DreamZero leverage video prediction during training or inference to support action generation (Du et al., 2023; Ye et al., 2026). TesserAct augments future prediction with RGB, depth, and surface normals to construct temporally coherent 4D scenes, while Structured4D predicts future 3D scene evolution in a structured latent space and converts predicted futures into actions through inverse dynamics (Zhen et al., 2025; Li et al., 2026b). Several models explicitly evaluate real-world generalization to unseen environments, objects, or spatial configurations (Physical Intelligence et al., 2025; Ye et al., 2026). Nevertheless, robustness remains uneven: Latent Interface Training identifies sensitivity to unseen visual conditions (Lin et al., 2026a). In simulation, RoboTwin reports Easy (clean) and Hard (domain-randomized) settings; after clean-only task training, the Hard setting exposes substantial generalization gaps for several policies (Chen et al., 2025; RoboTwin Team, 2026a). We use LingBot-VA for controlled backbone comparisons and DreamZero as an external WAM comparator.

Geometry-bearing world models. TesserAct predicts RGB, depth, and normals; PointWorld represents action effects as 3D point flow; GEM-4D introduces geometric correspondence supervision; and RynnWorld-4D jointly predicts RGB, depth, and flow with a policy head (Zhen et al., 2025; Huang et al., 2026; Zhou et al., 2026; Zhao et al., 2026). Track2Act uses predicted point tracks for generalizable manipulation, while 4D-WAM and X-WAM incorporate spatiotemporal or 3D structure into world–action modeling (Bharadhwaj et al., 2024; Yang et al., 2026; Guo et al., 2026). Geometry-aware models also investigate distribution shifts: 4D-WAM evaluates generalization to RoboTwin's Hard setting, and X-WAM reports real-robot tests with unseen placements, tablecloths, and distractors (Yang et al., 2026; Guo et al., 2026). Our contribution therefore centers on a controlled analysis of causal multi-view generation and view-dependent motion targets—flow for fixed views and rays for moving wrist views—rather than the addition of depth and flow alone. Our model supports OOD generalization under simulated scene shifts and real-world conditions.

3 Method

3.1 Problem Formulation and Overview

Given the observation and robot history H_k, we denote the visual-space 4D future by $\mathcal{Y}_k = \{R_k^v, D_k^v, M_k^v\}_{v=1}^V$, containing RGB, depth, and view-specific motion representations for each camera v (Section 3.2). We denote its shared-VAE latent representation by Z_k. S4NDBOX predicts Z_k before the corresponding action chunk A_k (Section 3.3), with

$$p_\theta(Z_k, A_k \mid H_k) = p_\theta(Z_k \mid H_k)\, p_\theta(A_k \mid H_k, Z_k). \tag{1}$$

At inference, current RGB-D observations are given with zero motion conditioning. Decoding Z_k yields $\mathcal{Y}_k$, whose predicted geometry and motion can be compared with the motion implied by A_k to rank candidate world–action pairs before execution (Section 3.4).

3.2 Geometric 4D Future Representation

We next specify the pre-VAE depth and motion representations in $\mathcal{Y}_k$.

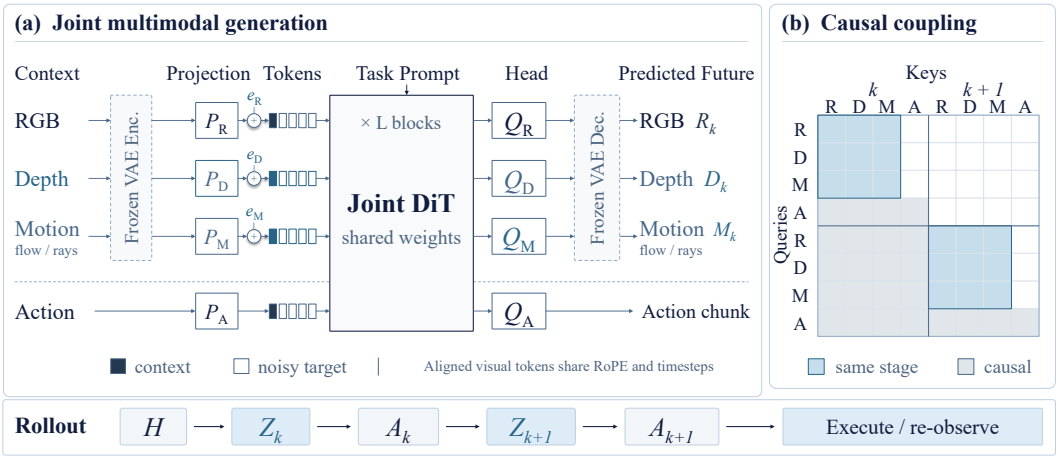

Figure 2: **Causal autoregressive 4D world–action modeling. (a)** RGB, depth, and motion (flow/rays) are encoded by a shared frozen VAE into the latent future Z_k, mapped into a shared Transformer by modality-specific input layers P and embeddings e, and projected back by Q for VAE decoding into the predicted visual future (R_k, D_k, M_k). The action stream bypasses the VAE and produces A_k through P_A and Q_A. **(b)** Visual modalities interact within the same stage, while action tokens attend to the completed Z_k and visual tokens cannot access current or future actions. Execution and re-observation update the history for the next rollout step.

Metric depth. Let $d_k^v(u)$ denote the metric depth at pixel u for camera v. We normalize it as

$$q(d) = 1 - \left(1 - \frac{d}{\lambda_D s_D}\right)^{\lambda_D+1},$$

where λ_D and s_D are fixed codec parameters (Gabeur et al., 2026). The normalized depth map is represented as the three-channel depth field D_k^v and passed to the VAE.

Fixed-view motion. For a fixed camera v, we compute endpoint flow between visual timestamps t and t' within the prediction horizon. For a source pixel u on rigid entity j with world point $X_t(u)$,

$$X_{t'}(u) = T_{wj,t'} T_{wj,t}^{-1} X_t(u), \quad F_{t\to t'}^v(u) = \pi_{t'}^v(X_{t'}(u)) - \pi_t^v(X_t(u)).$$

Here $T_{wj,t} \in SE(3)$ is the entity-to-world transform and π_t^v denotes projection into camera v. The endpoint flow is converted to the three-channel motion field M_k^v using polar-RGB encoding, with magnitude normalized by $\sigma_F\sqrt{H^2+W^2}$, following Chefer et al. (2025).

Wrist-view orientation. For a wrist camera v, we instead compute the world-frame ray and its three-channel motion representation as

$$r_t^v(u) = \frac{R_{wc,t}^v K_v^{-1} \bar{u}}{\left\| K_v^{-1} \bar{u} \right\|_2}, \qquad M_k^v(u) = \frac{r_t^v(u) + 1}{2}.$$

Here $\bar{u} = [u_x, u_y, 1]^\top$, K_v is the camera intrinsic matrix, and $R_{wc,t}^v$ is the camera-to-world rotation, exposing wrist orientation tied to end-effector rotation.

3.3 Causal Autoregressive 4D World-Action Modeling

With the visual-space 4D fields defined above, we next describe how their shared-VAE latents are jointly generated and causally coupled to action prediction.

Causal 4D-action coupling. Future latent tokens for RGB, depth, and motion are predicted jointly from the observed history, allowing the three modalities to interact while remaining independent of future actions. The action tokens are then decoded conditioned on both the history and the latent 4D future Z_k. Complementary 4D modalities first mutually constrain the same imagined future, which then serves as the basis for action inference.

Unified multimodal modeling. As illustrated in Fig. 2, each camera view of the RGB, depth, and motion fields is encoded by the shared frozen VAE (Team Wan et al., 2025), yielding the visual latents in Z_k. For each modality, multi-view latents are arranged using the same T-shaped spatial layout, aligning corresponding camera views across modalities. Each modality uses linear input and output layers (P_m, Q_m), with $m \in \mathcal{M} = \{R, D, M\}$: P_m maps patchified VAE latents into the shared Transformer space, while Q_m maps Transformer outputs back to latent patches. We retain the pretrained RGB layers (P_R, Q_R) and initialize (P_D, Q_D) and (P_M, Q_M) from their RGB counterparts. After input projection, zero-initialized modality embeddings e_R, e_D, and e_G are added to the RGB, depth, and motion streams, respectively. Aligned visual tokens share the same 3D RoPE (Su et al., 2024) coordinates. Actions use P_A for tokenization and Q_A for decoding, with temporal positions between consecutive visual timestamps. A single Transformer jointly processes all visual and action tokens.

Training objective. For each visual modality $m \in \mathcal{M}$, let z_0^m denote the clean VAE latent target in Z_k. We sample $z_s^m = (1-s)z_0^m + s\epsilon^m$ and train the model to predict the flow-matching target $\epsilon^m - z_0^m$ (Lipman et al., 2023; Liu et al., 2023). Visual modalities aligned at the same latent frame share the same noise level s while using independent Gaussian noise. Actions are trained analogously with a separate flow-matching loss ℓ_A. With timestep-weighted visual losses ℓ_m, we optimize

$$\mathcal{L} = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} \ell_m + \ell_A. \tag{2}$$

The observed visual prefix is kept clean and excluded from the prediction loss.

3.4 4D Verification and Regeneration

A decoded visual-space 4D future $\mathcal{Y}_k$ and its paired action describe the same robot evolution in complementary representations. Although the action is generated from the predicted future, this causal dependence alone does not guarantee geometric consistency between the two. Their agreement therefore provides a natural pre-execution consistency signal for evaluating a world–action hypothesis. Given a fixed history, we sample multiple hypotheses, measure their geometric agreement, and execute the most consistent candidate.

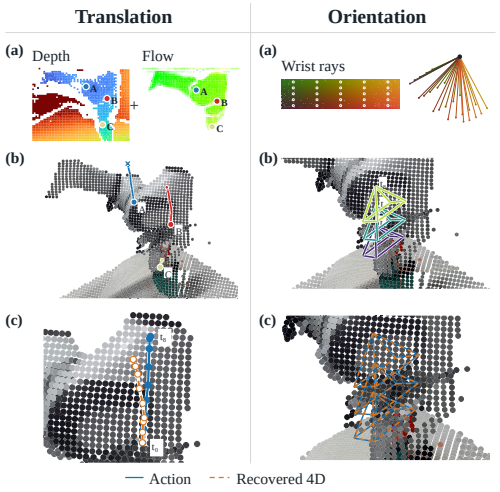

Figure 3: **World–action agreement.** Left: translation; right: orientation. (a) Predicted depth, flow, and wrist rays. (b) Recovered 3D point motion and camera orientation. (c) Comparison with the paired action.

Recovering comparable motion. Figure 3 illustrates how predicted 4D fields are connected to action-space geometry. For translation, predicted flow projects a source pixel to its future image location, and the corresponding target-frame depth lifts this point into 3D. Repeating this across time yields a recovered 3D trajectory, which we compare with the displacement implied by the paired action. For rotation, the predicted world-frame wrist rays encode the orientation of the wrist camera; aligning them with canonical camera rays recovers an $\mathrm{SO}(3)$ orientation, which can be compared with the orientation implied by the end-effector action.

World–action agreement. Using the correspondences above, we measure whether the decoded 4D fields in $\mathcal{Y}_k$ and their paired action describe consistent robot motion. For fixed-view motion, the predicted flow $\hat{F}$ defines a tracked image trajectory $\hat{u}_{a,t}^{(i)}(u)$ for robot arm a at future timestep t under candidate i, where u denotes the tracked source pixel. We independently obtain the corresponding action-implied trajectory $\tilde{u}_{a,t}^{(i)}(u)$ by transporting the observed 3D robot point with candidate action i and projecting it back into the fixed camera. Their disagreement is measured as

$$S_{\text{motion}}^{(i)} = \max_{a,t} \frac{1}{|\mathcal{P}_a|} \sum_{u \in \mathcal{P}_a} \rho\Big(\big\| \hat{u}_{a,t}^{(i)}(u) - \tilde{u}_{a,t}^{(i)}(u) \big\|_2 \Big), \tag{3}$$

Algorithm 1 4D verification-guided candidate selection

Require: Observation history H, world–action model p_θ, candidate budget $N \geq 1$
Ensure: One selected action chunk and updated observation history
for $i = 1, \ldots, N$ **do**
  $Z^{(i)} \sim p_\theta(Z \mid H)$ ▷ Predict a 4D future
  $A^{(i)} \sim p_\theta(A \mid H, Z^{(i)})$ ▷ Predict its paired action
  $\hat{u}^{(i)} \leftarrow$ trajectory induced by predicted fixed-view flow
  $\tilde{u}^{(i)} \leftarrow$ corresponding trajectory implied by $A^{(i)}$
  $\hat{r}^{(i)} \leftarrow$ predicted world-frame wrist rays
  $\tilde{r}^{(i)} \leftarrow$ corresponding wrist rays implied by $A^{(i)}$
  $S^{(i)}_{\text{motion}} \leftarrow \textsc{MotionAgreement}(\hat{u}^{(i)}, \tilde{u}^{(i)})$
  $S^{(i)}_{\text{rot}} \leftarrow \textsc{RotationAgreement}(\hat{r}^{(i)}, \tilde{r}^{(i)})$
end for
Combine the within-bank motion and rotation ranks for each candidate
$i^\star \leftarrow$ valid candidate with the best combined rank
Execute $A^{(i^\star)}$
Update H with the realized observations and robot state

where $\mathcal{P}_a$ denotes the tracked points for arm a and ρ is a normalized endpoint-error function. For wrist rotation, let $\hat{r}^{(i)}_{c,t}(u)$ denote the predicted world-frame ray, and let $\tilde{r}^{(i)}_{c,t}(u)$ denote the corresponding ray implied by the candidate end-effector rotation. We define

$$S^{(i)}_{\text{rot}} = \max_{c,t} Q_{0.9,u}\left[\angle\left(\hat{r}^{(i)}_{c,t}(u), \tilde{r}^{(i)}_{c,t}(u)\right)\right]. \tag{4}$$

Here $Q_{0.9,u}$ denotes the 90th percentile over pixels u, and the maximum is taken over wrist cameras c and evaluated future timestep t. We take the maximum over arms or wrist views and future endpoints so that a severe inconsistency at any future endpoint is not hidden by averaging over the rest of the trajectory. Lower scores indicate stronger agreement between the predicted future and paired action. Since the motion and rotation residuals have different scales, we combine their within-bank ranks.

Verification-guided regeneration. Algorithm 1 summarizes the full inference loop. From the same observation history, we sample multiple joint hypotheses, each containing a predicted 4D future and its corresponding action. We score each hypothesis by its world–action agreement and select the most consistent valid candidate for execution. After execution, new observations update the history and the procedure continues in closed loop.

4 Experiments

We evaluate S4NDBOX under two distribution shifts: generalization to RoboTwin's Hard (domain-randomized) setting (Section 4.1) and zero-shot simulation-to-real transfer to a Franka Research 3 under pose, lighting, clutter, and camera-pose shifts (Section 4.2). We further study modality contributions (Section 4.3), physical consistency (Section 4.4), and world–action verification (Section 4.5).

4.1 Generalization under Scene Distribution Shifts

Evaluation protocol. We train our policy using only 50 clean demonstrations per task (2,500 demonstrations in total over 50 tasks), then evaluate the policy in both clean and domain-randomized scenes, following RoboTwin's hard protocol (Chen et al., 2025). Our main evaluation uses 50 trials per task and setting across all 50 tasks. The clean setting uses fixed white backgrounds, lighting, and table height, without additional distractors. The domain-randomized setting varies textures, lighting, and table height and introduces clutter, creating shifts in scene appearance and layout. This protocol tests whether a policy trained only in clean environments remains effective under changes in scene appearance and layout. We build our method on the LingBot-VA backbone (Li et al., 2026a). Additional implementation and evaluation details are provided in Appendix A.

Table 1: **RoboTwin generalization under scene shifts.** Average success rate (%). Easy and Hard evaluate clean and domain-randomized scenes after clean-only task training. Our evaluation uses 50 trials per task and setting across all 50 tasks. Motion in our model combines head-camera flow and wrist rays. Evaluation details are in Appendix B. *RoboTwin 2.0 trajectories reported in the pretraining corpus. GigaBrain-0.7 additionally pretrains on 37.3K embodied hours (LingBot-VA: 16K). ♦Same LingBot-VA backbone.

Method	4D	Easy	**Hard**	Mean
Single-task supervised fine-tuning				
π_0 (Black et al., 2025)		46.4	16.3	31.4
Robotics Diffusion Transformer (Liu et al., 2025)		34.5	13.7	24.1
3D Diffusion Policy (Ze et al., 2024)		55.2	5.0	30.1
Action Chunking Transformer (Zhao et al., 2023)		29.7	1.7	15.7
Diffusion Policy (Chi et al., 2023)		28.0	0.6	14.3
Vision–language–action models				
GigaBrain-0.7* (GigaBrain Team et al., 2026)		66.8	**67.9**	**67.3**
$\pi_{0.5}$ (Physical Intelligence et al., 2025)		70.7	46.0	58.4
X-VLA (Zheng et al., 2025)		68.0	20.9	44.5
World–action models				
AHA-WAM (Cai et al., 2026)		64.3	3.2	33.8
FastWAM (Yuan et al., 2026)		77.8	1.9	39.9
X-WAM* (Guo et al., 2026)	✓	70.0	25.8	47.9
LingBot-VA♦ (Li et al., 2026a; Yang et al., 2026)		80.7	34.6	57.7
4D-WAM♦ (Yang et al., 2026)	✓	81.5 +0.8	41.8 +7.2	61.6 +3.9
Ours: S4NDBOX (RGB + depth + motion)♦	✓	**86.8** +6.1	<u>47.9</u> +13.3	**67.3** +9.6

Quantitative Results. As shown in Tab. 1, single-task supervised fine-tuning baselines perform substantially worse and deteriorate further under scene randomization. Our method achieves 86.8% success in Easy (clean) and 47.9% in Hard (domain-randomized). Compared with the LingBot-VA RGB-only baseline, this corresponds to gains of 6.1% and 13.3%, respectively, with the larger improvement appearing under scene randomization. FastWAM, which retains video co-training but omits future prediction at inference (Yuan et al., 2026), drops from 77.8% in Easy to 1.9% in Hard, indicating that video co-training alone does not ensure robustness to domain-randomized scenes.

Among 4D world–action models, our model exceeds 4D-WAM and X-WAM in the Hard setting by 6.1% and 22.1%, respectively. Architecturally, 4D-WAM uses trajectory-feature alignment during training (Yang et al., 2026), while X-WAM introduces an auxiliary depth branch whose depth tokens are not exposed to action decoding (Guo et al., 2026). In contrast, our action decoder directly accesses predicted depth, head-camera flow, and wrist-ray directions, providing an explicit geometric interface between the imagined future and robot control.

Our model also outperforms X-VLA and $\pi_{0.5}$. GigaBrain-0.7 remains stronger in the Hard setting at 67.9%; however, it uses substantially broader pretraining—37.3K embodied hours and 272M vision-language samples, compared with 16K manipulation hours for the inherited LingBot-VA backbone (GigaBrain Team et al., 2026; Li et al., 2026a). Overall, these results show that our method improves generalization under scene distribution shifts while retaining strong performance in the clean setting. We further ablate the contributions of individual 4D modalities in Section 4.3.

4.2 Zero-Shot Simulation-to-Real Transfer

Evaluation protocol. We evaluate zero-shot simulation-to-real transfer on Stack Cups using a Franka Research 3. Both RGB-only and geometry-augmented policies are trained on the same 1,000 scripted demonstrations collected in CoppeliaSim and deployed on the calibrated physical setup without real-world task demonstrations or fine-tuning. We compare post-convergence checkpoints. We evaluate two observation layouts: three views (two side cameras and one wrist camera) and a single left-side view. In addition to the simulation-to-real domain gap, we introduce four real-world shifts in object pose, lighting, clutter, and camera pose. Camera-pose OOD is evaluated only in the

Table 2: **Zero-shot sim-to-real transfer on Stack Cups.**

Method	Pose	Light	Clutter	Cam pose	Mean
Three views: two side + one wrist camera					
RGB-only	0.0	12.5	0.0	–	4.2
(Ours) RGB + depth + motion	**37.5**	**25.0**	**75.0**	–	**45.8**
Single view: left-side camera only					
RGB-only	37.5	0.0	37.5	34.4	28.8
(Ours) RGB + depth + flow	**87.5**	**75.0**	**50.0**	**65.6**	**68.8**

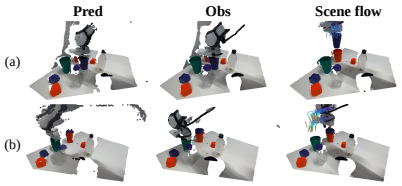

Figure 4: **4D prediction under clutter.** Predicted/observed clouds and illustrative point tracks for (a) cup geometry and (b) reaching. Examples from a single-view clutter rollout; track colors indicate nominal time.

single-view setting. Both policies follow the same protocol, with four episodes per view layout for cup configuration, lighting (two strong, one dim, and one medium), and added clutter, and eight single-view episodes for camera-pose shifts: orbit, vertical translation, combined orbit/elevation, and roll, with two episodes each. We use the same progress-based graded task score for all OOD shifts and pool all evaluated episodes within each view layout to compute the mean. The scoring definition, camera perturbations, and per-condition outcomes are provided in Appendix C.

Sim-to-real generalization. As shown in Tab. 2, our policy substantially improves zero-shot simulation-to-real performance over RGB-only control by 41.7 and 40.0 score points with three views and the left view alone, respectively. Under **lighting shifts**, RGB-only scores 0.0 in the single-view setting, whereas S4NDBOX achieves 75.0 across lighting conditions and remains effective even in the very dim-light episode, suggesting that explicit 4D cues support control when RGB observations become less informative. Under **clutter**, our three-view policy likewise scores 75.0 versus 0.0, demonstrating greater robustness to visual distractors. Under **object pose shifts**, S4NDBOX improves scores in both view layouts, suggesting better generalization to changed cup configurations. Under **camera pose shifts**, evaluated with the single left-side view, S4NDBOX scores 65.6 versus 34.4 for RGB-only, a 31.3-point improvement. Fig. 4 provides qualitative examples under real-world clutter. The predicted point clouds preserve recognizable scene structure, while the scene-flow tracks visualize the predicted motion of the manipulated cup and robot, qualitatively showing that the 4D representation learned in simulation continues to produce structured geometry and motion after transfer to the physical robot.

4.3 Modality Ablation

Role of 4D modalities. All variants share the LingBot-VA backbone and are trained for 5k updates on identical clean demonstrations in Robotwin, with five paired trials per task and setting. Starting from RGB and action prediction (RGB-A), adding depth (RGBD-A), adding all-view flow (RDF-A), and replacing wrist flow with rays (RDFR-A) yield successive gains of 11.25%, 3.33%, and 3.75% in randomized success (Tab. 3). RDFR-A performs best, suggesting that geometry and view-specific motion cues jointly improve robustness to scene shifts.

Role of the 4D–action interface. Blocking direct action attention to predicted depth and motion tokens at test time reduces overall success from 67.3% to 58.6%, an 8.7% drop (Tab. 4), suggesting that the trained policy uses predicted geometry and motion through this pathway for action generation. We further remove all observed depth inputs from the same checkpoint, without retraining, while retaining depth prediction and action access to predicted 4D tokens. Success remains 55.8% overall (80.4% Easy, 31.2% Hard), suggesting that the policy can rely on predicted 4D cues.

4.4 Geometric Consistency Analysis

Using the same single-view Stack Cups setup as the precision probe (Appendix D.2), we evaluate the geometric and motion consistency of imagined futures.

Motion consistency. On 50 paired Stack Cups initial scenes, selected WorldArena (Shang et al., 2026) and Physics-IQ (Motamed et al., 2026) diagnostics show that RDF improves spatiotemporal

Table 3: **Modality ablation.**

Method	Easy	Hard	Mean
RGB-A	74.00	22.08	48.04
RGBD-A	**78.40**	33.33	55.87
RDF-A	72.00	36.67	54.33
RDFR-A	72.80	**40.42**	**56.61**

Table 4: **Interface ablation.**

	w/o 4D attn.	w/o depth obs.	Ours
Easy	76.4	80.4	**86.8**
Hard	40.8	31.2	**47.9**
Mean	58.6	55.8	**67.3**

Table 5: **Physical plausibility.**

Metric ↑	RGB-A	RDF-A
Motion Smoothness	0.7287	**0.7451**
Spatiotemporal IoU	0.6459	**0.7184**
Weighted Spatial IoU	0.8494	**0.8785**

IoU from 0.6459 to 0.7184 and weighted spatial IoU from 0.8494 to 0.8785 relative to RGB-only, while also achieving higher motion smoothness (Tab. 5).

Geometric coherence. We overlay action-implied robot geometry on the generated future, using known kinematics to map predicted actions to robot configurations. As shown in Fig. 5, RGB-only predictions can appear visually plausible even when individual links bend from the action-implied configuration. In contrast, jointly predicted RGB, depth, and flow make geometry and motion explicit, encouraging coherent robot evolution without hard rigid-body constraints.

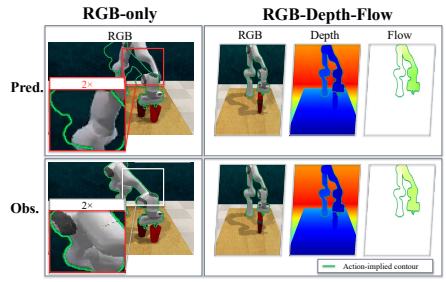

Figure 5: **Geometric coherence.** Red marks RGB-only mismatch.

4.5 World–Action Verification

To test whether 4D world–action agreement can guide candidate selection, we evaluate flow+raymap ranking on 50 randomized Stack Blocks Two scenes in RoboTwin. For each scene, the same 16 candidates are ordered either by the verifier or by the exact expectation under random ordering. Candidates are executed from identical scene resets with standard closed-loop inference after the first action chunk.

As shown in Fig. 6, verification reaches 86.0% success within three attempts, compared with 78.1% for random order, an observed gain of 7.9 percentage points. The advantage is most visible at small attempt budgets and narrows as more candidates are tried. The gain suggests that explicit 4D world–action agreement can provide a useful ranking signal under limited execution budgets.

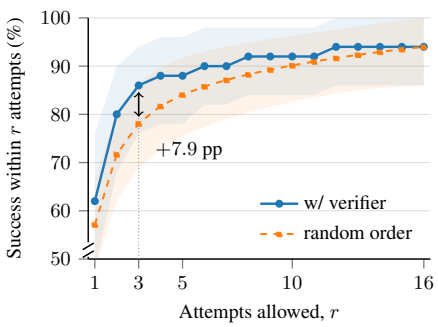

Figure 6: **Verifier-ranked retries.** Verifier ranking versus random ordering. Shading denotes pointwise 95% bootstrap intervals.

5 Discussion

Explicit 4D prediction provides a geometric interface between imagined futures and robot actions. Explicit geometry and motion expose spatial structure to action generation through the predicted future's latent representation. The RoboTwin results support improved robustness to scene shifts. Real-world Stack Cups evaluations after zero-shot transfer show higher graded task scores for our 4D policy than RGB-only under pose, lighting, and clutter shifts. These results demonstrate the practical value of geometry-augmented control when appearance and scene configuration change.

Experiments in randomized simulated scenes suggest that world-action agreement can prioritize successful candidates under limited execution attempts. Limited tasks and trials, partial visibility, and calibration errors bound these findings. Agreement measures self-consistency, while physical safety and successful execution require further validation. More broadly, predicted 4D futures can both inform action generation and guide candidate search before execution.

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on research methodology and experimental design, and to assist with the interpretation and presentation of experimental results. We did not use generative AI to generate synthetic datasets, formulate or prove mathematical claims, or implement the reported learning methods; other required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for literature search and retrieval, manuscript editing and restructuring, code inspection and editing of supporting scripts, and preparation and refinement of scientific schematics and figures. All AI-assisted content was reviewed by the authors. References and factual claims were checked against the cited sources, code changes were inspected and tested by the authors where applicable, and reported experimental results were verified against the underlying measurements. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

REFERENCES

Homanga Bharadhwaj, Roozbeh Mottaghi, Abhinav Gupta, and Shubham Tulsiani. Track2Act: Predicting point tracks from internet videos enables generalizable robot manipulation. *arXiv preprint arXiv:2405.01527*, 2024. URL https://arxiv.org/abs/2405.01527.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.010.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, et al. RT-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022. URL https://arxiv.org/abs/2212.06817.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023. URL https://arxiv.org/abs/2307.15818.

Jisong Cai, Long Ling, Shiwei Chu, Zhongshan Liu, Jiayue Kang, Zhixuan Liang, Wenjie Xu, Yinan Mao, Weinan Zhang, Xiaokang Yang, Ru Ying, Ran Zheng, and Yao Mu. AHA-WAM: Asynchronous horizon-adaptive world-action modeling with observation-guided context routing. *arXiv preprint arXiv:2606.09811*, 2026. URL https://arxiv.org/abs/2606.09811.

Hila Chefer, Uriel Singer, Amit Zohar, Yuval Kirstain, Adam Polyak, Yaniv Taigman, Lior Wolf, and Shelly Sheynin. VideoJAM: Joint appearance-motion representations for enhanced motion generation in video models. In *International Conference on Machine Learning*, 2025.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, 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.

Yixiang Chen, Peiyan Li, Yuan Xu, et al. FlowWAM: Optical flow as a unified action representation for world action models. *arXiv preprint arXiv:2607.13017*, 2026.

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. In *Robotics: Science and Systems*, 2023.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Joshua B. Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *Advances in Neural Information Processing Systems*, 2023.

Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T. Barron, Kyle Genova, Nithish Kannen,

Sherry Ben, Yandong Li, Mandy Guo, Suhas Yogin, Yiming Gu, Huizhong Chen, Oliver Wang, Saining Xie, Howard Zhou, Kaiming He, Thomas Funkhouser, Jean-Baptiste Alayrac, and Radu Soricut. Image generators are generalist vision learners. *arXiv preprint arXiv:2604.20329*, 2026.

GigaBrain Team, Angen Ye, Axiang Sun, Can Jin, Chenxi Cheng, Chong Shi, Dengke Shang, Dingqian Zhang, Guan Huang, Guangqiang Wang, Guangqing Ding, Guo Li, Hangcong Li, Hengyu Zhong, Hongtao Lu, Jianbo Qin, Jiming Mao, Jing Zhu, Jindi Lv, Jingzhi Cui, Junjie Xie, Junyi Bao, Kai Liu, Lei Yuan, Limin Long, Lv Feng, Mingming Yu, Peng Li, Pengfei Yi, Qi Li, Qianli Zhang, Qingfang Li, Qitang Hu, Rui Zhang, Shaoyan Sun, Shibo Sun, Shiying Duan, Tenghui Chen, Tianze Liu, Weijie Ke, Wenyao Xue, Xiaofeng Wang, Xiaoyu Tian, Xinyu Liu, Xinze Chen, Yang Wang, Yankai Wang, Yejun Zeng, Yifan Li, Yifei Nie, Yilong Li, Yilong Liu, Yongchao Feng, Yumeng Wang, Yun Ye, Zhichao Liu, Ziheng He, Zonghai Yang, and Zheng Zhu. GigaBrain-0.7: Scaling embodied foundation models to emergent capabilities with a three-system architecture. *arXiv preprint arXiv:2608.15875*, 2026. URL https://arxiv.org/abs/2608.15875.

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. URL https://arxiv.org/abs/2604.26694.

Wenlong Huang, Yu-Wei Chao, Arsalan Mousavian, Ming-Yu Liu, Dieter Fox, Kaichun Mo, and Fei-Fei Li. Pointworld: Scaling 3d world models for in-the-wild robotic manipulation. *arXiv preprint arXiv:2601.03782*, 2026.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, et al. DROID: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024. URL https://arxiv.org/abs/2403.12945.

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

Zhiyi Li, Peilin Wu, Xiaoshen Han, Ruojin Cai, and Yilun Du. Structured 4d latent predictive model for robot planning, 2026b. URL https://arxiv.org/abs/2607.01166.

Jianman Lin, Shailesh Shailesh, Zhongyi Luo, and Jiafei Duan. Breaking the vision-action shortcut: Latent interface training for generalizable robotics foundation models. *arXiv preprint arXiv:2609.12641*, 2026a. URL https://arxiv.org/abs/2609.12641.

Sixu Lin, Junliang Chen, Huaiyuan Xu, Zhuohao Li, Guangming Wang, Yixiong Jing, Sheng Xu, Runyi Zhao, Brian Sheil, Lap-Pui Chau, and Guiliang Liu. RoboFlow4D: A lightweight flow world model toward real-time flow-guided robotic manipulation. *arXiv preprint arXiv:2605.17522*, 2026b.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. RDT-1B: A diffusion foundation model for bimanual manipulation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=yAzN4tz7oI.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. Do generative video models understand physical principles? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 948–958, March 2026. doi: 10.1109/WACV61042.2026.00099.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

RoboTwin Team. RoboTwin 2.0 leaderboard: Benchmark setting. Official benchmark protocol, 2026a. URL https://robotwin-platform.github.io/leaderboard. Accessed September 9, 2026.

RoboTwin Team. RoboTwin: Task configuration reference, 2026b. URL https://robotwin-platform.github.io/doc/usage/configurations.html. Upstream demo_clean and demo_randomized configurations checked September 12, 2026.

RoboTwin Team. RoboTwin: Domain randomization, 2026c. URL https://robotwin-platform.github.io/doc/usage/domain-randomization.html. Accessed September 12, 2026.

Yu Shang, Zhuohang Li, Yiding Ma, et al. WorldArena: A unified benchmark for evaluating perception and functional utility of embodied world models. *arXiv preprint arXiv:2602.08971*, 2026.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Lishan Yang, Wenxuan Song, Xi Wang, Pingyue Sheng, Zheng Fang, Ziyang Zhou, Junjie He, Haodong Yan, Jiayi Chen, Nan Sun, Qiao Sun, Pengwei Wang, Lingqiao Liu, Yan Wang, Yuxiang Gao, Feras Dayoub, and Haoang Li. 4D-WAM: Infusing spatiotemporal awareness into world action models through trajectory fields. *arXiv preprint arXiv:2608.08023*, 2026.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, 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. URL https://arxiv.org/abs/2603.16666.

Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Robotics: Science and Systems*, 2024.

Haoyu Zhao, Xingyue Zhao, Siteng Huang, Xin Li, Deli Zhao, and Zhongyu Li. RynnWorld-4D: 4d embodied world models for robotic manipulation. *arXiv preprint arXiv:2607.06559*, 2026.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware, 2023. URL https://arxiv.org/abs/2304.13705.

Haoyu Zhen, Qiao Sun, Hongxin Zhang, Junyan Li, Siyuan Zhou, Yilun Du, and Chuang Gan. Tesseract: Learning 4d embodied world models. *arXiv preprint arXiv:2504.20995*, 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. *arXiv preprint arXiv:2510.10274*, 2025.

Kaichen Zhou, Yuzhen Chen, Fangneng Zhan, Hang Hua, Grace Chen, Xinhai Chang, Ao Qu, Yilun Du, Zhuang Liu, Paul Pu Liang, and Mengyu Wang. GEM-4D: Geometry-enhanced video world models for robot manipulation. *arXiv preprint arXiv:2605.22882*, 2026.

SUPPLEMENTARY MATERIAL

Implementation details are provided in Appendix A, followed by RoboTwin evaluation (Appendix B), real-world Stack Cups (Appendix C), geometric coherence and precision-sensitive control (Appendix D), and world–action verification (Appendix E).

A IMPLEMENTATION DETAILS

All variants use the LingBot-VA backbone. Table 6 lists observation layouts and modality settings.

Table 6: **Generation settings.** Image dimensions are height × width. Each view is resized and VAE-encoded before latent packing.

Item	RoboTwin	Stack Cups sim-to-real
Views	Head + two wrists	Two fixed sides + one wrist; or left only
Image grids	Head 256×320; wrists 128×160	Each view 160×320
Latent layout	Wrists above head	Wrist above sides; or left only
Depth codec	Power/Hilbert, $\lambda = -3.15$, scale 0.35	Bounded Hilbert, $\lambda = -3$, $c = 3.333\ldots$; $[0, 4]$ m
Motion stream	Head flow + wrist rays; all-view flow in RDF	Three views: side flow + wrist rays; left only: flow
Flow bandwidth	Polar $\sigma = 0.40$	Polar $\sigma = 0.15$
Training action sampling	50 Hz; 16 samples/non-anchor latent	10 Hz
Training visual sampling	50 Hz / stride 4	$10/3$ Hz

Test-time observations. RGB-only policies receive current RGB; geometry-augmented policies also receive current depth. Flow conditioning encodes zero displacement. Future RGB, depth, and motion are predicted, not ground-truth inputs.

B ROBOTWIN EVALUATION

B.1 EVALUATION PROTOCOL

Policies use the Aloha-AgileX dual-arm embodiment and 50 clean demonstrations per task across 50 tasks (2,500 total). Easy and Hard denote clean and randomized evaluation, respectively (Table 7) (RoboTwin Team, 2026b;c).

Table 7: **Easy and Hard scene settings.** Table height is the randomization amplitude.

Setting	Easy	Hard
Background/textures	Not randomized	Randomized
Table clutter	Off	On
Lighting	Fixed	Randomized
Table-height amplitude	0	0.03 m
Head-camera displacement	0	0
Clean-background probability	1	0.02
Extreme-light probability	0	0.02

B.2 EVALUATION SETTINGS

The main RoboTwin results are evaluated on all 50 tasks with 50 trials per task and setting. Infrastructure-failure seeds are replaced; ordinary task failures are retained, and duplicate runs use the first valid result. Scores average task success rates within each setting and weight Easy and Hard equally.

B.3 MODALITY ABLATION

All four variants in Tab. 3 share the backbone, training data, 5,000 updates, and evaluation seeds: five trials on each of 50 tasks in both settings (250 Easy and 250 Hard). Modality definitions follow Section 4.3; conditioning follows Appendix A.

C REAL-WORLD STACK CUPS EVALUATION

C.1 SIMULATION TRAINING AND DEPLOYMENT

Policies are initialized independently from `lingbot-va-base` and trained on the same 1,000 scripted demonstrations from CoppeliaSim 4.1.0, PyRep, and a project-modified RLBench, using the Panda arm and default grasp attachment. No real-world task demonstrations or fine-tuning are used. We compare post-convergence checkpoints.

The scene and cameras are calibrated to the laboratory setup; fixed base and gripper-tip-to-TCP transforms map Panda poses to FR3 targets. Calibration and deployment development include real-robot rollouts, but no real-data task training. Each view is resized and VAE-encoded separately before latent packing; the single-view setting uses only the left view (Table 6). Deployment is asynchronous.

Actions. Actions are $[p_t - p_0, q_0^{-1} q_t, g]$ relative to the episode's first frame, with `xyzw` quaternions and gripper openness g (1 open, 0 closed). All variants share simulation-derived q01/q99 normalization and clipping to $[-1.5, 1.5]$.

C.2 EVALUATION AND SCORING

We evaluate two policies under pose, lighting, clutter, and camera-pose OOD shifts. Pose, lighting, and clutter are evaluated in both three-view and single-view layouts, with four episodes per policy, view layout, and condition (48 episodes total). Three-view policies use two fixed side cameras and one wrist camera; single-view policies use only the left camera. Pose changes the initial cup configuration; lighting comprises two strong-light, one dim-light, and one medium-light episode; clutter adds distracting objects. The task requires the target stack to remain stable after release.

Camera-pose OOD is evaluated only in the single-view setting, with eight episodes per policy: orbit $= -5°$, vertical translation $dz = +0.03$ m, orbit $= +2°$ combined with elevation $= -2°$, and roll $rz = +2°$, with two episodes each (16 additional episodes; 64 real-world evaluation episodes in total). Here, orbit rotates the camera around the table, elevation raises or lowers the camera, dz translates the camera vertically, and rz rotates the camera about its optical axis.

Scoring protocol. All real-world OOD conditions use the same progress-based task score: 0.25 for successfully picking up the first cup, 0.5 for successfully stacking the first cup, 0.75 for successfully picking up the second cup, and 1.0 for completing the full stack. Episodes that do not reach the first milestone receive 0. Condition scores average the corresponding episodes. The mean reported in Tab. 2 pools total graded reward over all evaluation episodes for that view layout (r/n): 12 episodes for the three-view setting and 20 episodes for the single-view setting, including the eight camera-pose episodes. Development rollouts are not evaluation episodes.

D GEOMETRIC COHERENCE AND PRECISION-SENSITIVE CONTROL

Geometric visualization. In Fig. 5, predicted actions are mapped through the controller and kinematics and projected onto the generated RGB future. The overlay compares action-implied geometry, not measured future motion.

D.1 RECOVERED MOTION FROM 4D PREDICTIONS

Figure 7 expands Fig. 3. Flow is sampled on source pixels and depth at their target locations. The background is the observed anchor point cloud; wrist rays recover orientation, while displayed camera origins come from actions. Colors identify tracked points, not time; crosses mark lost tracks.

Table 8: **Real-world pose, lighting, and clutter episode scores.** Four episodes per condition under the shared progress-based scoring protocol.

Policy / view	OOD	Episode scores	Score
RGB, 3-view	Pose	0, 0, 0, 0	0.0
	Lighting	0.5, 0, 0, 0	12.5
	Clutter	0, 0, 0, 0	0.0
Ours, 3-view	Pose	0.5, 0.5, 0, 0.5	37.5
	Lighting	0.5, 0, 0.5, 0	25.0
	Clutter	1, 0.5, 0.5, 1	75.0
RGB, left	Pose	0.5, 0, 0, 1	37.5
	Lighting	0, 0, 0, 0	0.0
	Clutter	0.5, 0, 0.5, 0.5	37.5
Ours, left	Pose	1, 1, 0.5, 1	87.5
	Lighting	0, 1, 1, 1	75.0
	Clutter	0.5, 1, 0.5, 0	50.0

Table 9: **Single-view camera-pose OOD.** Two episodes per perturbation under the shared progress-based scoring protocol.

Camera perturbation	RGB episode scores	Ours episode scores	RGB	Ours
Orbit -5°	0.5, 0.5	0.5, 0.5	50.0	50.0
$dz = +0.03$ m	0, 0.5	0.5, 0.5	25.0	50.0
Orbit $+2^\circ$, elev. -2°	0.5, 0.25	0.25, 1.0	37.5	62.5
Roll $rz = +2^\circ$	0, 0.5	1.0, 1.0	25.0	100.0
Pooled r/n	$2.75/8$	$5.25/8$	34.4	65.6

For this rollout, the final reference-point discrepancy is 24.9 mm, orientation differences span 0.21–1.21°, and 23/170 tracks survive to the horizon. These are single-rollout diagnostics, not aggregate accuracy or the verifier scores used for candidate ranking.

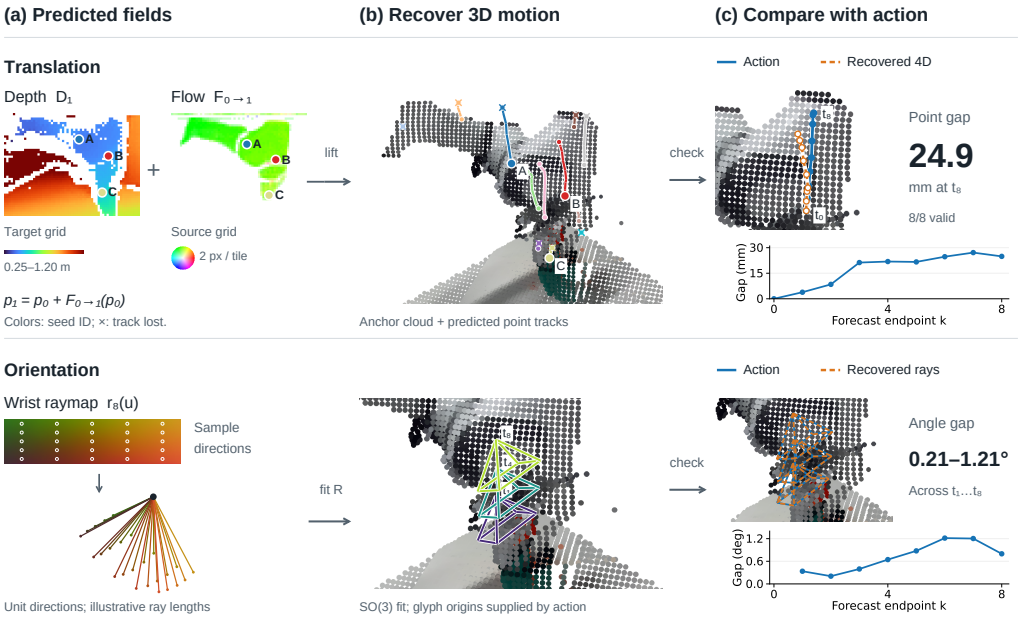

Figure 7: **Recovered motion and action alignment.** Depth–flow tracks, anchor point-cloud context, wrist orientations, and per-endpoint discrepancies for the rollout shown in Figure 3.

Table 10: **Lighting breakdown.** Graded task scores.

Policy / view	Strong ($n = 2$)	Dim ($n = 1$)	Medium ($n = 1$)
RGB, 3-view	0.0	50.0	0.0
Ours, 3-view	25.0	50.0	0.0
RGB, left	0.0	0.0	0.0
Ours, left	50.0	100.0	100.0

Method	Geometric interface	Seen (%)	Unseen (%)	PSNR ↑	SSIM ↑	LPIPS ↓
Diffusion Policy (Chi et al., 2023)	RGB	46.9	0.0	–	–	–
3D Diffusion Policy (Ze et al., 2024)	3D point cloud	85.4	4.1	–	–	–
RoboFlow4D + Diffusion Policy (Lin et al., 2026b)	RGB + future 3D motion	0.0	0.0	–	–	–
RoboFlow4D + 3D Diffusion Policy (Lin et al., 2026b)	3D point cloud + future 3D motion	5.0	0.0	–	–	–
Structured4D (Li et al., 2026b)	3D scene generation + IDM	0.0	0.0	7.242	0.176	0.824
FlowWAM (Chen et al., 2026)	RGB + robot flow	0.0	0.0	20.689	0.721	0.118
Ours	**RGB + depth + head flow**	**100.0**	**90.0**	**26.255**	**0.763**	**0.048**

Native geometry/motion diagnostics (method-specific). RoboFlow4D: 3D path error 0.0598; Structured4D: depth RMSE 0.377 m and $\delta_1 = 28.8\%$; FlowWAM: 2D motion error 6.281 px on the robot region and 0.721 px on valid labeled regions; Ours: depth AbsRel 0.0369, $\delta_1 = 99.10\%$, and motion-region flow EPE 1.532 px.

Table 11: **Single-view precision probe on Stack Cups.** Success rates on seen and unseen configurations using a single front view. Baseline trial counts differ, so the comparison is descriptive rather than a matched statistical benchmark. Higher success rates are better.

D.2 Precision-Sensitive Control

The geometric correspondences in Figure 3 expose motion in coordinates that can be compared with robot actions. We next separately test whether explicit geometry and motion support precision-sensitive control in a single fixed-view setting.

We evaluate precision-sensitive manipulation in a *single-view* Stack Cups setting, reporting success on both seen and unseen configurations. The seen split measures attainable control precision under familiar configurations, while the unseen split tests whether this precision transfers to novel ones. All methods use the same front-camera observation. We compare Diffusion Policy (DP) (Chi et al., 2023), which predicts actions from RGB; DP3 (Ze et al., 2024), which operates on 3D point-cloud observations; RoboFlow4D (Lin et al., 2026b), which predicts future 3D motion to guide DP or DP3; and FlowWAM (Chen et al., 2026), which jointly models robot flow and actions. Our model instead exposes an explicit 4D future to action prediction.

As shown in Tab. 11, our 4D WAM reaches 100.0% success on seen configurations and 90.0% on unseen configurations. DP3 reaches 85.4%/4.1%, while RoboFlow4D+DP3 reaches 5.0%/0.0%. The gains from DP to DP3 and from RoboFlow4D+DP to RoboFlow4D+DP3 further indicate that explicit 3D structure substantially strengthens the visual–action interface. Because the adapted baselines use different trial counts, we treat this experiment as a descriptive precision probe rather than a matched statistical ranking.

D.3 Prediction–Execution Motion Diagnostics

Protocol. We evaluate prediction–execution consistency on 50 paired simulated Stack Cups initial scenes using RGB-only and RDF converged models. We report selected WorldArena (Shang et al., 2026) and Physics-IQ (Motamed et al., 2026) diagnostics that directly measure temporal smoothness and agreement between predicted and executed motion.

Metrics. Motion Smoothness evaluates temporal consistency of the generated video. Spatiotemporal IoU and Weighted Spatial IoU compare predicted motion regions with those observed during each model's execution. Higher is better for all three metrics; results are reported in Tab. 5. These are targeted motion-consistency diagnostics rather than official WorldArena or Physics-IQ benchmark scores.

E World–Action Verification

This appendix provides protocol details for the fixed-pool verifier-ranked retry study in Section 4.5. For each scene, multiple world–action candidates are sampled from the same observation history. The verifier changes only their execution order: candidate pools and recorded outcomes are held fixed across rankings. Candidate selection applies to the first action chunk; later chunks use ordinary closed-loop inference.

Scoring. The flow+raymap verifier used in the main-text experiment combines the within-bank ranks of the motion-agreement score in Eq. (3) and the rotation-agreement score in Eq. (4). Lower disagreement corresponds to an earlier rank. Task-success outcomes are not used to compute the verifier ranking.

E.1 Fixed-pool ranked-retry evaluation

Protocol and baseline. The 15k model generates 16 candidates for each of 50 randomized Stack Blocks Two scenes. Each candidate is executed from the same scene reset (800 rollouts total), and the verifier and random-order baseline share these recorded outcomes. Success within r attempts means that at least one of the first r candidates succeeds. Every attempt resets the scene rather than recovering from a failed state.

For a pool containing c successful candidates, the exact success probability under a uniformly random ordering is

$$p_r(c) = 1 - \frac{\binom{16-c}{r}}{\binom{16}{r}}, \qquad r = 1, \ldots, 16, \tag{5}$$

with zero numerator when $r > 16 - c$. We average this quantity across scenes. Thus the baseline is the exact expectation over random permutations of the same fixed candidate pools, rather than a separately executed random-selection baseline.

Results. Using flow+raymap ranking, success within three attempts is 86.0%, compared with an exact random-order expectation of 78.11%, an observed gain of 7.89 percentage points. A paired scene-bootstrap with 20,000 resamples gives a 95% interval of $[-0.49, 16.05]$ percentage points for this gain. The advantage is largest at small attempt budgets and narrows as more candidates are tried, consistent with the main-text result in Fig. 6.

This fixed-pool study isolates candidate ordering: it does not measure candidate-generation scaling, adaptive recovery after a failed execution, or latency. Because the scenes and candidate pools were examined during development, we treat the result as an exploratory same-scene evaluation.