WHAT SHOULD A WORLD-ACTION MODEL IMAGINE? 4D GEOMETRY AS AN INTERMEDIATE REPRESENTATION FOR CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

World Action Models (WAMs) imagine future observations to guide control, yet it remains unclear what a model should imagine for action. Most existing WAMs predict RGB futures, where physical structure and motion are entangled with texture, illumination and observer motion. We systematically study 4D geometry as a predictive intermediate representation for control. Rather than asking whether geometric supervision is useful, we ask when and how an imagined geometric future should participate in action generation. We represent the observable 4D state with world-frame pointmaps and scene flow, which jointly describe where visible surfaces are and how the same physical points move, together with wrist-view depth for complementary local geometry. Our controlled studies reveal several consistent findings. **First**, geometry is most useful when explicitly imagined and made directly accessible to action, outperforming feature alignment and auxiliary-only geometric supervision. **Second**, geometric and RGB futures are complementary rather than substitutable: removing either condition degrades control. **Third**, the value of 4D depends critically on its form and interface: world-frame pointmaps and scene flow outperform projective depth-and-optical-flow cues; and geometric supervision should be selected by view-specific reliability. These findings translate into substantial robustness gains. The resulting WAM improves a matched LingBot-VA baseline from 73.73% to 82.73%, with the largest gains under sensor noise, lighting, background, and viewpoint changes. An anonymous project page is available. Code will be publicly released.

1 INTRODUCTION

World–Action Models (WAMs) have emerged as a promising paradigm for robot control by coupling world prediction with action generation: rather than mapping observations directly to actions, they predict future states and use them to guide control (Du et al., 2023; 2024; Tian et al., 2025; Zhu et al., 2025a; Li et al., 2026a). This raises a fundamental question: *what should a WAM imagine for action?* Most existing WAMs rely primarily on future RGB or video predictions (Zhu et al., 2025a; Li et al., 2025; 2026a; Ye et al., 2026). RGB is expressive and naturally compatible with large-scale video pretraining (Wang et al., 2025a), but it is also an indirect description of the physical world. The same interaction can produce substantially different pixels under changes in illumination, texture, background, viewpoint, or sensor quality—conditions known to challenge visual robot policies and WAMs (Fei et al., 2026; Wang et al., 2026d). For control, however, the key quantities are often where surfaces are, how they move in 3D, and how the robot is geometrically related to the scene.

Simply adding geometry does not resolve this question. Existing geometric approaches span substantially different roles, including auxiliary 4D supervision, representation alignment, geometric priors, and geometry-aware action conditioning (Hou et al., 2025; He et al., 2026; Wang et al., 2026a; Zhang et al., 2026a; Wang et al., 2026e; Yang et al., 2026b; Zhao et al., 2026a; Guo et al., 2026a). As illustrated in Figure 1, geometry can regularize RGB features, align intermediate representations as in Geometry Forcing (Wu et al., 2025), or be explicitly generated as a future that action can directly access. Even with explicit geometric imagination, several choices remain: what geometric state to predict, which views to supervise, how to encode geometry into a pretrained video backbone, and how RGB and geometry should interact with action.

We focus on dense structure and motion. Depth describes visible structure but remains tied to camera projection, while optical flow captures apparent image motion and therefore entangles scene motion with observer motion. By contrast, pointmaps represent visible surfaces in 3D (Wang et al., 2024), and scene flow describes their physical displacement; recent 4D reconstruction methods similarly

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.

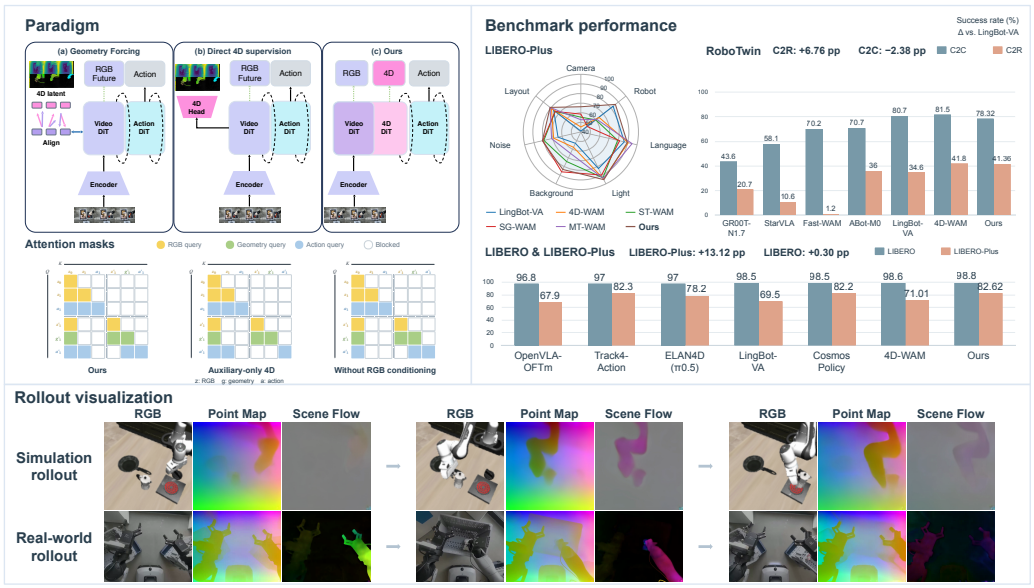


Figure 1: **Explicit geometric imagination for robust world–action modeling. Upper left**: We compare feature alignment, direct 4D supervision, and our explicit geometric imagination, together with action-conditioning controls. **Upper right**: Ours retains strong clean-task performance while improving robustness over the EGB-only baseline on LIBERO-Plus and randomized RoboTwin. **Bottom**: Predicted RGB, pointmaps, and scene flow across simulation and real-robot rollouts.

combine geometry, correspondence, and motion in a common 3D frame (Zhang et al., 2025b; Feng et al., 2025; Zhu et al., 2026). Together, world-frame pointmaps and scene flow describe where the observed world is and how it moves, while wrist-view depth provides complementary local geometry near manipulation. The directed attention masks in Fig. 1 further make explicit how predicted RGB and geometry are exposed to action. Thus, 4D world modeling for control is not merely a modality extension, but a joint *representation-and-interface* problem. We study this problem using a matched RGB WAM backbone, LingBot-VA (Li et al., 2026a). Rather than treating 4D as a single component, we systematically vary how geometry enters the model, what geometric quantities are predicted, which views they supervise, how they are encoded, and whether action directly accesses the imagined geometry. This controlled study yields four findings.

First, geometry is most effective when explicitly imagined and made available to action. The three paradigms in Figure 1 show that direct 4D reconstruction supervision already improves over RGB-only training, and feature-level Geometry Forcing improves further, but both remain weaker than generating an explicit geometric future for action conditioning. This separates the benefit of predictive supervision from that of directly using a generated future. **Second**, geometric imagination complements rather than replaces visual imagination. Our action model attends to completed RGB and geometric futures. Removing action access to the current imagined RGB while retaining generated 4D reduces success to 71.86%, below the RGB-only baseline, indicating that RGB and geometry provide complementary predictive information. **Third**, the choice of geometric representation matters. With the same wrist-depth complement and encoding interface, world-frame pointmaps and 3D scene flow outperform relative depth and optical flow. The former express position and displacement in a common 3D reference frame, where an ideally static physical point has zero scene flow despite camera motion. Consistent with this distinction, the matched robustness study shows improved camera-viewpoint performance, with the largest gains under sensor noise, lighting, and background perturbations that substantially alter image appearance while often preserving the underlying interaction. **Fourth**, more geometric supervision is not always better. Predicting full 4D for both the main and wrist cameras reduces performance relative to using main-view 4D with wrist depth. Diagnostics show that wrist-view pointmap gradients can conflict with main-view geometry and that wrist 4D supervision exhibits substantially higher gradient noise. A simpler wrist-depth target therefore provides more reliable complementary local structure.

These findings lead to the model summarized in Figure 1. Given RGB observations and language, it jointly predicts future RGB, world-frame pointmaps and scene flow, together with wrist-view

depth, and generates actions conditioned on the observed history and completed RGB–geometry future. The geometric reconstruction model (Zhu et al., 2026) is used only to construct offline supervision, so deployment still requires RGB sensing alone. The resulting model improves the matched LingBot-VA baseline from 73.73% to 82.73% on LIBERO-Plus Long (Fei et al., 2026), with gains of +17.38 points under sensor noise, +14.96 under lighting changes, +11.07 under background changes, and +10.02 under camera-viewpoint shifts. Across broader evaluations, it reaches 82.62% on LIBERO-Plus, 41.36% under RoboTwin randomization (Chen et al., 2025b), and 98.8% on standard LIBERO (Liu et al., 2023).

Our study therefore makes the following contributions: (1) We formulate 4D world imagination as a representation-and-interface problem for control and systematically compare how geometry can enter a WAM. (2) We identify empirical design principles showing that explicit geometry benefits from direct action access, complements RGB prediction, depends on the chosen spatial-motion representation, and should respect view-dependent supervision reliability. (3) We instantiate these findings with world-frame pointmaps and scene flow for global structure and motion, plus wrist-view depth for local geometry, while retaining RGB-only sensing at deployment. (4) We validate the resulting design through controlled simulation perturbations, representation and attention ablations, optimization and corruption diagnostics, and physical-robot experiments, with the strongest gains occurring when visual appearance changes while the underlying interaction remains similar.

2 Related Work

4D representations. Pointmaps support direct 3D reconstruction, while dynamic extensions connect scene geometry with temporal correspondence (Wang et al., 2024; Zhang et al., 2025b; Feng et al., 2025). Video priors further support reconstruction of evolving geometry (Jiang et al., 2025); MotionCrafter jointly represents world-frame pointmaps and dense 3D scene flow (Zhu et al., 2026). These signals describe both surface positions and physical displacement, separating scene motion from camera-induced image motion when reconstruction is accurate. We adapt this reconstruction capability offline to supervise task-conditioned *future* geometry, which the policy predicts from RGB observations without an online reconstruction teacher.

World Action Models. Video-based control ranges from decoding visual plans into actions to jointly generating video and action (Du et al., 2023; Tian et al., 2025; Zhu et al., 2025a; Li et al., 2025). Video-pretrained policies extend these approaches to diverse manipulation settings (Kim et al., 2026; Ye et al., 2026). Their shared prediction objective nevertheless permits different control pathways: LingBot-VA conditions action on predicted RGB (Li et al., 2026a), whereas Fast-WAM retains video co-training without test-time rollout (Yuan et al., 2026). This separates the benefit of predictive supervision from directly using a generated future. Building on LingBot-VA, we study how explicit geometry enriches action conditioning.

4D for World Action Modeling. Geometric methods differ in where structured information enters control. Depth and trajectory supervision can shape representations without supplying generated geometry to action (Li et al., 2026b; Zhang et al., 2026a; Yang et al., 2026b; Wang et al., 2026a). Other systems recover actions from generated RGB-D or RGB-D-normal futures (Zhen et al., 2025; Liu et al., 2026b; Wang et al., 2026b), or condition policies on partially denoised scene flow and predictive multimodal features (Wang et al., 2026e; Zhao et al., 2026a). X-WAM jointly predicts RGB-D and actions, but its depth branch reads RGB features unilaterally and can be omitted during action inference (Guo et al., 2026a). These precedents establish geometry as both supervision and a control representation. Ours studies a specific connection: RGB imagination conditions pointmaps, scene flow, and wrist depth; action then reads the completed geometric and RGB futures alongside observed history. Our within-backbone ablations distinguish this direct conditioning from auxiliary-only geometry (details in Section A.6).

3 Method

Given language and a history H_t of multi-view RGB observations and actions, our model predicts future RGB R, geometry G, and an action chunk A. Action generation conditions on the imagined RGB and geometry as well as history. After execution, new RGB observations and past actions form the next history; imagined geometry is used only for the current decision. Deployment requires RGB sensing, without online geometric reconstruction (Figure 2).

3.1 Geometric Representations and Supervision

Geometry and motion. Pointmaps $P_i \in \mathbb{R}^{H\times W\times 3}$ locate visible surface points; forward scene flow F_i gives their three-dimensional displacement. Thus $P_i(u) + F_i(u)$ is the next-frame position of the

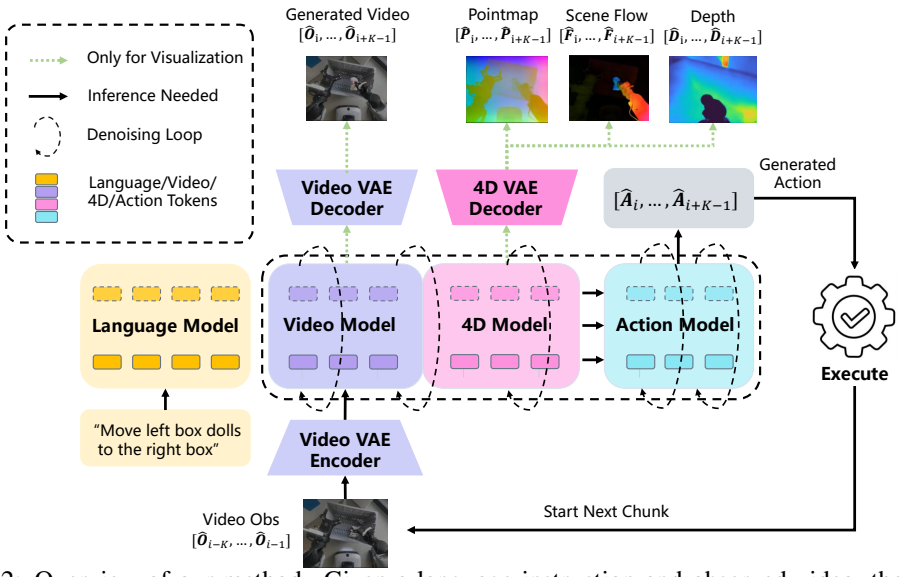


Figure 2: Overview of our method. Given a language instruction and observed video, the model predicts future video and 4D geometry (pointmaps, scene flow, and depth), then generates an action chunk conditioned on the imagined future. The action is executed and the next observation begins a new prediction cycle. The video and 4D decoders are used only to visualize the predicted streams.

same physical point, which may project to a different pixel or become occluded. Both signals use a common reference frame anchored at the clip's first camera pose. For exact targets, a static point has zero flow despite camera motion; reconstruction errors can violate this ideal.

Offline supervision. Using pooled LIBERO and RoboTwin simulator sequences, we adapt a pretrained 4D representation model (Zhu et al., 2026) in stages: train the geometry VAE, train the motion VAE with geometry frozen, then freeze both VAEs and train a deterministic video-to-4D reconstructor. Simulator depth, poses, and correspondences supervise these stages. The frozen reconstructor supplies pointmap and flow targets from robot training videos; it is not part of the deployed policy. Losses, validity masks, and coordinate conventions are in Sections A.4.1 and A.4.2.

Policy encoding. The final model uses main-view pointmaps and scene flow, plus wrist-view relative depth D from Video Depth Anything (Chen et al., 2025a): $G = \{P, F, D\}$. Coordinates and signed flow become three-channel pseudo-RGB; normalized depth is repeated across channels. RGB and geometric streams use the same frozen Wan VAE: $z^m = \mathcal{E}_{\text{Wan}}(\mathcal{N}_m(X^m)), \qquad m \in \{R, P, F, D\}$. Here $\mathcal{N}_m$ maps each signal to $[-1, 1]$ (Section A.4.3). Sequence-level normalization preserves relative structure but not a common metric scale across episodes. The policy predicts Wan latents; the separate 4D VAE belongs to offline reconstruction.

3.2 Geometry-Conditioned World–Action Modeling

Backbone. Starting from LingBot-VA (Li et al., 2026a), we add per-stream input/output projections and timestep conditioning, initialized from pretrained RGB. Separate interfaces distinguish geometric signals while their losses update the shared backbone. Language enters through cross-attention.
Training tokens. Figure 3 shows an initial RGB observation z_0 and one prediction chunk. Here z_1 and a_1 are the target RGB and action conditioning copies; z_1', g_1', and a_1' are noisy RGB, geometry, and action prediction copies. Subscripts index chunks, not noise levels. The geometry block g_1' groups pointmap, scene-flow, and wrist-depth tokens. Conditioning copies build the causal RGB/action context; they are distinct from the streams being denoised.

Directed visibility. Noisy RGB z_1' attends to z_0 and itself; geometry g_1' additionally attends to z_1', and action a_1' attends to z_0, z_1, g_1', and itself. Thus, geometry learns from noisy RGB prediction, while action uses target RGB and noisy geometry. Attention spans space and time within each visible block. Conditioning copies are causal, and prediction streams can access earlier RGB/action history across chunks; geometric history is not retained.

From training to inference. During training, the RGB conditioning copy z_1 uses ground-truth RGB with stochastic noise augmentation. At inference, the completed RGB prediction occupies this conditioning role, and completed geometric predictions replace the noisy geometry supplied during

Table 1: Zero-shot LIBERO-Plus success rate (%), grouped by policy family (Zhang et al., 2026b). External scores retain their published aggregation. OpenVLA-OFT$_m$ uses four-suite joint training; LingBot-VA is evaluated by EMERGE-Policy (Fang et al., 2026).

Method	Camera	Robot	Lang.	Light	Background	Noise	Layout	Overall
Vision Language Action Models								
OpenVLA-OFT$_m$ (Kim et al., 2025; Fei et al., 2026)	55.60	21.70	81.00	92.70	91.00	78.60	68.70	67.90
ABot-M0 (Yang et al., 2026d; Zhang et al., 2026b)	60.40	67.90	86.40	96.20	91.60	86.40	82.60	80.50
VLAct (Yang et al., 2026c)	73.90	68.40	81.50	**96.70**	96.70	86.00	83.30	82.60
VLAs with structured-future supervision								
VLA-JEPA (Zhang et al., 2026b)	64.20	67.70	88.10	91.80	93.40	65.80	**83.90**	77.90
ELAN4D ($\pi_{0.5}$) (He et al., 2026)	63.70	70.70	77.80	89.80	91.40	79.90	81.40	78.20
LaMP (Wang et al., 2026e)	64.50	69.60	88.20	95.30	**97.40**	76.90	73.80	79.30
Track4Action (Wang et al., 2026a)	**76.60**	60.70	87.10	96.30	96.10	88.40	78.70	82.30
World Action Models								
LingBot-VA (Li et al., 2026a)	40.90	83.00	86.40	82.30	53.10	64.40	76.20	69.50
4D-WAM (Yang et al., 2026b)	45.15	64.26	90.63	94.29	57.71	69.08	79.21	71.01
ST-WAM (Wang et al., 2026d)	55.40	60.10	79.30	93.00	74.20	79.50	74.30	72.80
SG-WAM (Zhao et al., 2026b)	58.60	48.90	81.40	89.80	86.10	80.70	74.20	73.00
MT-WAM (Yang et al., 2026e)	52.70	61.10	**94.70**	93.60	64.80	71.30	81.00	73.66
GE-Act (Zhang et al., 2026b)	60.70	77.00	77.40	95.80	86.00	90.90	80.20	80.30
Cosmos Policy (Kim et al., 2026; Zhang et al., 2026b)	75.80	63.30	81.70	96.50	88.90	**92.70**	82.20	82.20
Ours	66.23	**86.00**	89.59	95.18	83.36	81.14	80.98	**82.62**

Table 2: RoboTwin 2.0 success (%). External results follow the 4D-WAM comparison.

Method	Clean	Randomized
Vision Language Action Models		
StarVLA (StarVLA Community, 2026)	58.10	10.60
GR00T-N1.7 (NVIDIA, 2026)	43.60	20.70
ABot-M0 (Yang et al., 2026d)	70.70	36.00
World Action Models		
Fast-WAM (Yuan et al., 2026)	70.20	1.20
LingBot-VA (Li et al., 2026a)	80.70	34.60
4D-WAM (Yang et al., 2026b)	**81.50**	**41.80**
Ours	78.32	41.36

Table 3: Matched LIBERO-Plus Long-specialist success rate (%). Δ is Ours minus LingBot-VA in points.

Perturbation	LingBot-VA	**Ours**	Δ (pp)
Background textures	73.01	**84.08**	**+11.07**
Camera viewpoints	47.02	**57.04**	**+10.02**
Language instructions	91.05	**96.08**	**+5.03**
Light conditions	81.39	**96.35**	**+14.96**
Object layout	**88.14**	86.22	-1.92
Robot initial states	87.53	**92.62**	**+5.09**
Sensor noise	57.68	**75.06**	**+17.38**
Overall	73.73	**82.73**	**+9.00**

training. Action therefore uses both imagined RGB and geometry, without access to ground-truth future observations (Section A.4.4).

3.3 Training and Inference

Joint training. For $m \in \{R, P, F, D, A\}$, independently sampled noise levels per latent frame give $z_\sigma^m = (1-\sigma)z_0^m + \sigma\epsilon^m$, with $\epsilon^m \sim \mathcal{N}(0, I)$ and normalized actions z_0^A. A masked pass predicts velocity $\epsilon^m - z_0^m$:

$$\mathcal{L}_{\text{WAM}} = \sum_m \lambda_m \, \mathbb{E}\Big[w_m(\sigma) \, \|\widehat{v}^m - (\epsilon^m - z_0^m)\|^2_{\text{valid}} \Big] .$$

Errors are averaged over valid elements; w_m weights timesteps. RGB and action have $\lambda_m = 1$, and each geometric stream has $\lambda_m = 0.1$. We first optimize the added geometric interfaces, then the backbone and all modality interfaces; T5 and Wan VAE stay frozen.

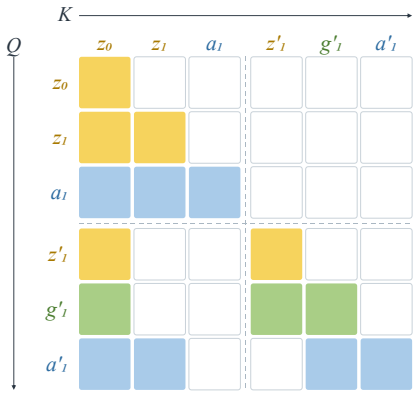
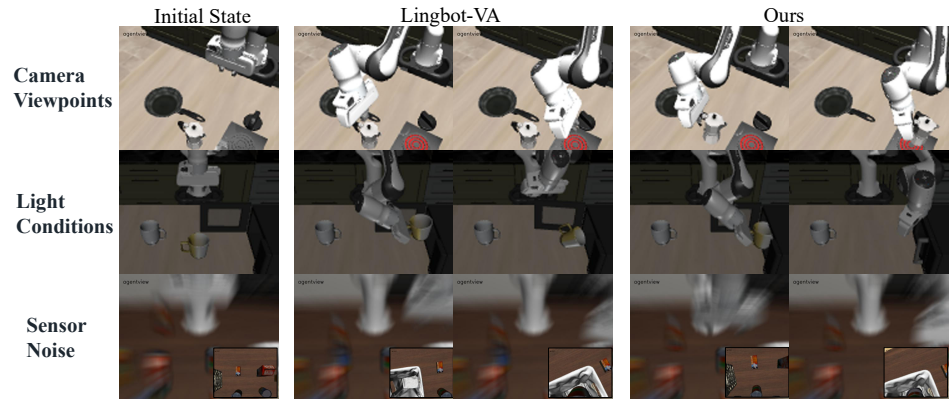


Figure 3: **Directed attention for world–action modeling.** Rows query columns; colored blocks indicate visible attention. z, g, and a denote RGB, geometry, and action, respectively; primes denote noisy prediction streams..

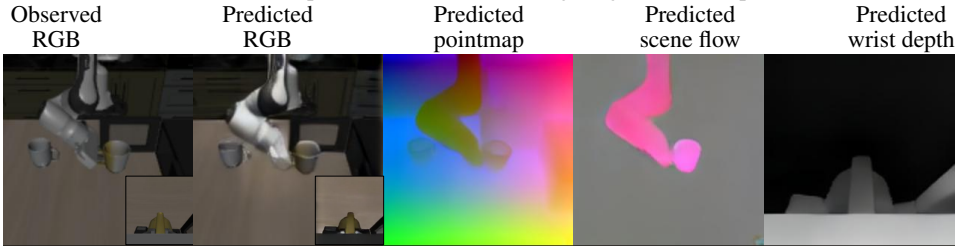
Closed-loop inference. World denoising runs from noise toward clean RGB and geometry on a shared schedule. At each step, the RGB pass supplies temporary keys and values to geometry. Action then follows a separate schedule conditioned on history and the completed world predictions. The robot executes an action chunk and updates its history (Sections A.4.4 and A.5.1). Neither teacher inference nor visual-stream decoding is needed to generate actions.

4 Experiments

We evaluate policy performance in simulation and on physical robots, and examine how predicted geometry contributes to control. Our study addresses five questions: *where robustness improves* (Q1), *whether direct action access to geometric futures is beneficial* (Q2), *how signal, view, and encoding choices affect performance* (Q3), *how the policy responds to corruption of its geometric condition* (Q4), *and whether similar gains are observed in real-robot manipulation* (Q5). We distinguish external benchmark comparisons from within-backbone studies.



(a) Matched closed-loop rollouts under camera, lighting, and sensor perturbations.



(b) Observed inputs and the generated RGB–4D future at one decision point.

Figure 4: **Closed-loop OOD rollouts and geometric imagination on LIBERO-Plus.** (a) Matched-start rollouts under camera shift, low light, and main-view zoom blur. Our policy completes all three tasks; LingBot-VA fails. Insets show the unperturbed wrist view. (b) Predicted RGB, pointmap, scene flow, and wrist depth during the low-light rollout.

4.1 Simulation: Robustness and Its Sources

Setup. We build on LingBot-VA (Li et al., 2026a) and train on clean demonstrations; the adapted 4D reconstructor also uses only clean training data. On LIBERO (Liu et al., 2023), a four-suite generalist is evaluated on standard LIBERO and on all 10,030 LIBERO-Plus episodes (Fei et al., 2026). The reported LIBERO-Plus generalist uses the 25K checkpoint. For controlled ablations, Long specialists share the base checkpoint, training steps, batch size, unfreezing schedule, and teacher within each comparison, and are evaluated on the same 2,519 episodes. LIBERO-Plus overall scores pool episodes. On RoboTwin (Chen et al., 2025b), we train for 15K steps on clean data and report task-macro success over 100 rollouts per task under clean and randomized conditions. External scores retain their published aggregation; the Full-generalist LingBot-VA result is a third-party evaluation. Protocols and variant definitions appear in Section A.2.

Q1: Where does 4D prediction improve robustness? On zero-shot LIBERO-Plus, our model reaches 82.62%, 13.12 points above the reported LingBot-VA result (Table 1). On RoboTwin, it obtains 78.32% clean and 41.36% randomized success, improving the reported LingBot-VA randomized score by 6.76 points (Table 2). Our model achieves 98.8% on standard LIBERO across Spatial, Object, Goal, and Long (Table 7), retaining strong clean-task performance while improving robustness. Together, the clean and perturbed evaluations show that geometric future prediction supports robust control without sacrificing the high success rate on standard tasks. The controlled Long comparison then examines the benefit within our training setup: success rises from 73.73% to 82.73%, a 9.00-point improvement.

Appearance changes: preserving an actionable future. The matched Long comparison shows the largest gains under sensor noise (+17.38 points), lighting (+14.96), and background changes (+11.07; Table 3). These perturbations alter image appearance while leaving the underlying interaction largely unchanged. RGB prediction must model task-relevant structure together with textures, highlights, and shadows; correlations with these appearance cues can become shortcuts on clean training scenes. Geometric prediction instead supervises surface positions and physical displacement, providing action with a structured description of the future beneath changing appearance. This additional condition helps explain the pronounced improvements under appearance perturbations.

The paired rollouts in Figure 4 illustrate this behavior. From matched initial states, our policy completes manipulation under a shifted camera, darker lighting, and severe main-view zoom blur, while LingBot-VA fails. In the low-light example, the predicted pointmap preserves the scene layout and scene flow concentrates on the robot and manipulated mug. These geometric futures expose the spatial structure and movement needed for the next action. **Separating observer and scene motion.** Camera-viewpoint success improves by 10.02 points. Whereas optical flow mixes object motion with camera motion and projection, world-frame scene flow directly describes physical displacement after camera-motion compensation. This removes camera-induced motion from the geometric prediction target and provides action with a consistent account of scene motion. The camera-shift rollout in Figure 4 demonstrates successful control under a changed view. Object-layout success decreases by 1.92 points; this perturbation changes the physical configuration itself.

4.2 Understanding and Designing Geometric Imagination

Q2: Does direct action access to geometric futures help? We first compare three ways of introducing the same native 4D targets (Table 4a). Direct supervision attaches a 4D reconstruction head to noisy RGB hidden states. Our Geometry Forcing adaptation (Wu et al., 2025) aligns layer-20 noisy RGB features with 4D latents. Explicit imagination instead models 4D as a generated stream that action can attend to. Their respective success rates are 75.63%, 77.42%, and 79.77%, versus 73.73% for RGB-only. Alignment and reconstruction supervision can shape RGB features, whereas explicit prediction exposes a separately represented geometric future to the action model.

Table 4: **Native-4D ablations on LIBERO-Plus Long.** (a) How 4D enters the model. (b) Action-conditioning controls. All variants retain RGB/action history. $\mathrm{RGB} \to A$ denotes action access to the current-chunk RGB condition, which is ground-truth RGB during training and generated RGB at inference.

(a) How 4D enters the model

Strategy	**Succ. (%)**
RGB-only LingBot-VA	73.73
Direct 4D reconstruction	75.63
Geometry Forcing	77.42
Explicit 4D imagination	**79.77**

(b) Action-conditioning controls

Configuration	$\mathbf{RGB} \to \mathbf{A}$	$\mathbf{4D} \to \mathbf{A}$	**Succ.**
Full native 4D	✓	✓	**79.77**
Auxiliary-only 4D	✓	–	77.54
w/o RGB imagination	–	✓	71.86

The attention-mask ablations distinguish learning from geometry from using it at inference (Table 4b). Auxiliary-only prediction, with 4D-to-action attention removed during training, reaches 77.54%; restoring the connection reaches 79.77%. Thus, geometric supervision helps even without direct access, and making the generated future available to action provides a further benefit. Removing action access to the current-chunk RGB condition reduces success to 71.86%. This condition is ground-truth RGB during training and completed RGB imagination at inference. RGB generation and its conditioning of geometry remain active; action reads historical conditions and generated 4D but not current RGB imagination. This retrained comparison supports complementary RGB and geometric conditions for action, rather than replacing RGB with geometry.

Q3: Which geometric representation and interface are useful? We compare pointmaps and 3D scene flow with relative depth predicted by Video Depth Anything (VDA) (Chen et al., 2025a) and 2D optical flow predicted by DPFlow (Morimitsu et al., 2025). These model names identify the offline teachers; the policy predicts the corresponding signal latents at deployment. We vary signal, view, and encoding in separate controlled panels.

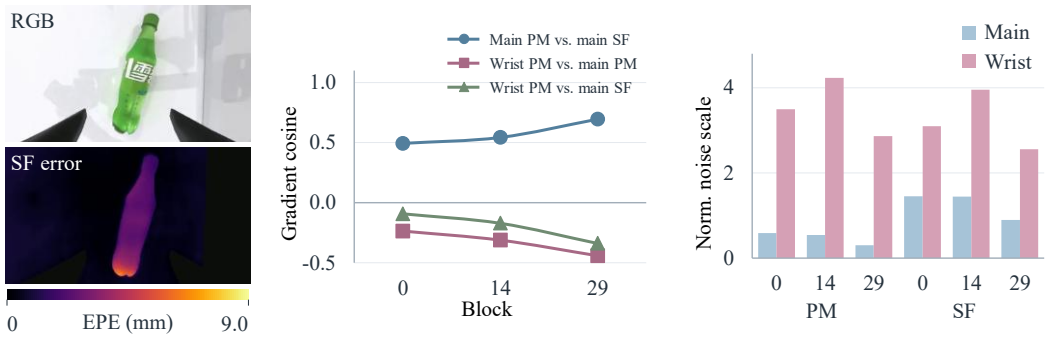
Three-dimensional structure versus projected cues. With depth in both views, adding optical flow raises success from 80.33% to 81.78% (Table 10), motivating depth plus flow as the stronger signal control. With the same wrist-depth complement and pseudo-RGB/Wan encoding, main-view pointmaps and scene flow obtain 82.73%, compared with 81.65% for depth and optical flow (Table 5a). Without wrist depth, the corresponding scores are 81.42% and 81.20%. Both representations already provide dense structure and motion, so density alone cannot explain their difference. The relevant distinction is whether positions and displacements are expressed jointly in a camera-motion-compensated 3D frame or through relative depth and image-plane motion. The larger observed margin with wrist depth also suggests that the value of a main-view representation depends on the complementary information available to action.

Table 5: **Representation ablations on LIBERO-Plus Long**. OF: optical flow; SF: scene flow.

(a) Main-view signal (pseudo-RGB)	**Wrist: None**	**Depth**
Depth + OF	81.20	81.65
Pointmap + SF	81.42	**82.73**

(b) Wrist signal (native main 4D)	**Succ.**
Native 4D	79.77
None	80.52
Depth	**81.68**

(c) Encoding (main pointmap + SF)	**Wrist: None**	**Depth**
Native 4D	80.52	81.68
Pseudo-RGB	81.42	**82.73**



(a) Scene-flow error. (b) Gradient alignment. (c) Gradient noise scale.

Figure 5: **Wrist-view 4D supervision.** (a) The bottle is stationary in world coordinates, yet wrist-view reconstruction estimates nonzero scene flow. RGB and endpoint error against aligned RoboTwin ground truth are shown above and below (0–9.0 mm). (b,c) At shared blocks 0/14/29, wrist pointmap (PM) gradients conflict with main-view geometry, and wrist PM and scene-flow (SF) gradients have higher noise scale. Diagnostic settings: Section A.3.1.

View-dependent reliability. With main-view native 4D fixed, wrist 4D lowers success from 80.52% to 79.77%, while wrist depth reaches 81.68% (Table 5b). Figure 5 explains this choice. In (a), the bottle rests on the table and is stationary in world coordinates, yet wrist-view reconstruction assigns it nonzero scene flow. This spurious motion directly illustrates wrist-view motion-estimation error. Wrist pointmap gradients also oppose main-view pointmaps (down to -0.44), and wrist pointmap/scene-flow gradient noise is 6.0–9.5×/2.1–2.8× higher than in the main view. Wrist depth supplies local structure without the conflicting, noisy 4D target.

Encoding for a pretrained RGB backbone. Encoding pseudo-RGB geometry with Wan improves success over native 4D latents by 0.90 points without wrist depth and 1.05 points with it (Table 5c). Sharing the pretrained video encoder supplies a common latent interface while separate projections retain modality identity. The optimization evidence in Figure 6 reveals the benefit of this design: pseudo-RGB geometry produces a 31-fold stronger gradient relative to action in shared block 14, and its block-0 gradient aligns with both RGB (0.47 cosine) and action (0.26). The native 4D gradient is weaker and nearly orthogonal to RGB in the same early block. Thus, the pseudo-RGB interface turns geometric future prediction into a strong, coherent training signal for the visual backbone used by action, rather than leaving geometry in a largely separate prediction pathway. Distribution and normalization analyses in Section A.3.2 further explain this pattern.

(a) Geometry gradient strength (b) Gradient cosine at shared block 0

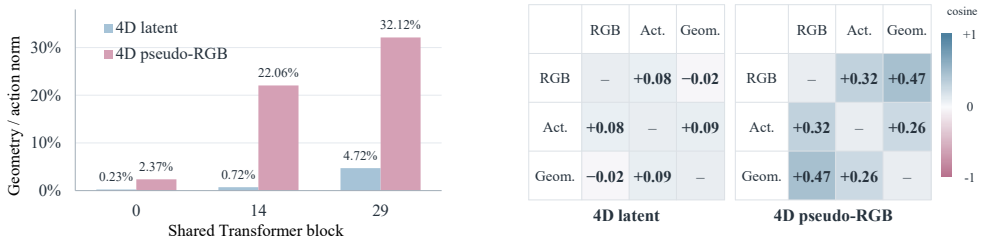


Figure 6: **Geometric gradients under native and pseudo-RGB encoding.** (a) Weighted geometry-to-action gradient-norm ratios across shared blocks. (b) Gradient cosines in block 0. Pseudo-RGB produces stronger geometric gradients and greater alignment with RGB and action; its geometry gradient sums the pointmap and scene-flow contributions.

Q4: How sensitive is the policy to geometric-condition corruption? We replace native predicted 4D with geometry reconstructed by the teacher from the policy's own predicted future RGB. Success changes from 79.77% to 80.41% (Figure 7); this alternative supplies no ground-truth future. We then mix the teacher latent with Gaussian noise as $\tilde{z} = (1-\lambda)z_{4D}^{\text{teacher}} + \lambda\epsilon$. Success remains at 80.41% for $\lambda = 0.25$ and 80.33% for $\lambda = 0.33$, then falls to 79.65% at $\lambda = 0.5$, 77.12% at $\lambda = 0.75$, and 75.35% with pure noise at $\lambda = 1$. The broad plateau followed

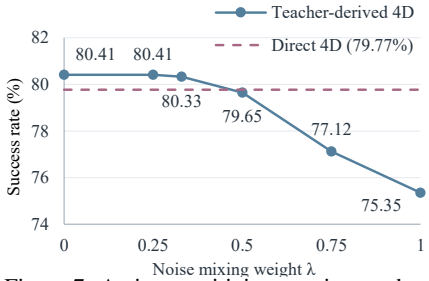


Figure 7: Action sensitivity to noisy teacher geometry. Dashed line: direct 4D prediction (79.77%). Exact values appear in Table 11.

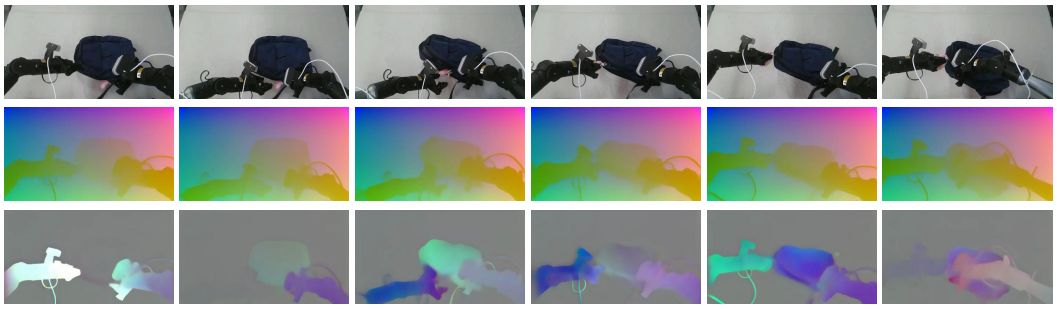


Figure 8: **Real-world backpack zipping.** Rows show observed RGB, offline-reconstructed pointmaps, and scene flow. Additional real-world sequences are provided in Figure 14.

by a decline shows that action can exploit an informative geometric future without exact latent reconstruction. Moderate corruption preserves much of its utility, whereas progressively destroying the geometric condition weakens control. Together with the attention ablations, this result connects the information supplied by the geometric pathway to the resulting action quality (details in Section A.2.4).

4.3 Real-World Evaluation

Q5: Do the gains also hold in real-robot manipulation? The real-robot suite comprises observer-motion manipulation on a humanoid robot, dynamic moving pick–place on AgileX ALOHA, and bimanual backpack zipping. These tasks probe observer motion, externally moving objects, and deformable contact with coordinated pulling, respectively. We compare $\pi_{0.5}$ (Physical Intelligence et al., 2025), RGB-only LingBot-VA (Li et al., 2026a), the matched Depth-flow WAM (Projective 4D), and our model. Policies receive RGB online, while geometric teachers supply offline training targets. Mobile and Dynamic retain their reported 700 demonstrations (630 training and 70 validation), one training seed, 5K updates, batch size 32, and 50 trials per method and split. Their training and inference hardware remain NVIDIA A800 and RTX 4090, respectively. Backpack is evaluated with 50 trials per method in each of the Nominal and configuration-OOD conditions (details in Section A.5.5). Ours achieves 70.00% Nominal and 50.00% OOD success (Table 6). The OOD score exceeds LingBot-VA by 10.67 points. On Dynamic, Ours improves both Nominal and OOD success over LingBot-VA, although both methods have a 24-point Nominal-to-OOD decrease. On Mobile and Backpack, Ours exhibits smaller observed decreases. Thus, the real-robot results support improved performance across the evaluated tasks, while the extent of robustness improvement is task-dependent.

Table 6: **Real-world success (%)** over 50 trials per method and condition. Suite averages tasks; Overall averages Nominal and OOD.

	Mobile		Dynamic		Backpack		Suite		
Method	**Nom.**	**OOD**	**Nom.**	**OOD**	**Nom.**	**OOD**	**Nom.**	**OOD**	**Overall**
$\pi_{0.5}$	**88**	66	38	18	54	26	60	36.67	48.33
LingBot-VA	84	60	52	28	58	30	64.67	39.33	52
Projective 4D	86	68	56	30	**62**	36	68	44.67	56.33
Ours	86	**72**	**62**	**38**	**62**	**40**	**70**	**50**	**60**

Figure 8 visualizes backpack zipping, where the bag deforms and rotates during coordinated bimanual pulling. The pointmaps and scene flow show offline-reconstructed spatial structure and motion, rather than online policy predictions. Ours achieves 62% Nominal and 40% configuration-OOD success (Table 14), outperforming LingBot-VA by 4/10 points and matching/exceeding Projective 4D by 0/4 points. The gain is concentrated under configuration shift, while the three tasks span observer motion, external motion, and deformable manipulation.

5 Conclusion

We study geometric imagination as both a *representation* and an *interface* for world–action modeling. Our results show that geometric supervision is most effective when the predicted future is directly accessible to action, while RGB remains complementary rather than replaceable. They further highlight that the choice of geometric signal matters: main-view pointmaps and scene flow provide structured global geometry and motion, whereas simpler wrist-view depth is more reliable for local manipulation. Encoding these signals through the pretrained visual interface further improves the resulting policy. Together, these design choices yield the strongest gains under appearance and viewpoint perturbations, while retaining RGB-only sensing at deployment. At the same time, the weaker improvement under physical layout changes suggests that geometric imagination improves robustness primarily when observations change more than the underlying interaction.

AI USE STATEMENT

Generative AI tools were used to support brainstorming and discussions of experimental design, to improve the clarity and language of the manuscript, and to assist with auxiliary research scripts. They were not used to implement the core model or training method. The authors reviewed all AI-assisted text and code and inspected and ran the auxiliary scripts to verify their behavior. The final method, experimental design, data analysis, and conclusions were determined by the authors. The authors take full responsibility for the manuscript, code, and reported results.

ETHICS STATEMENT

This work evaluates robotic manipulation in simulation and on physical robot platforms. Real-robot experiments were conducted under human supervision, with emergency-stop procedures and a separated robot workspace. The recorded data contain no sensitive information, and the research does not involve human-subject studies requiring ethics approval. The reported results characterize the evaluated tasks and conditions; deployment in other settings requires appropriate safety assessment.

REPRODUCIBILITY STATEMENT

The model and training procedure are described in Section 3, and the evaluation settings and results are reported in Section 4. The appendix documents geometry-target construction and implementation details (Section A.4), simulation protocols and variant definitions (Section A.2), and the real-robot tasks, execution settings, and scoring rules (Section A.5). The anonymous project page accompany this submission and code will be publicly released.

REFERENCES

Leonardo Barcellona, Andrii Zadaianchuk, Davide Allegro, Samuele Papa, Stefano Ghidoni, and Efstratios Gavves. Dream to Manipulate: Compositional World Models Empowering Robot Imitation Learning with Imagination. In *International Conference on Learning Representations*, pp. 56729–56763, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/8f0d446441a938d9de420a8ab8d7fd36-Abstract-Conference.html.

Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2Act: Human Video Generation in Novel Scenarios enables Generalizable Robot Manipulation. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 3936–3951. PMLR, 2025. URL https://proceedings.mlr.press/v305/bharadhwaj25a.html.

Qingwen Bu, Jia Zeng, Li Chen, Yanchao Yang, Guyue Zhou, Junchi Yan, Ping Luo, Heming Cui, Yi Ma, and Hongyang Li. Closed-Loop Visuomotor Control with Generative Expectation for Robotic Manipulation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 139002–139029. Curran Associates, Inc., 2024. doi: 10.52202/079017-4411. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/fad8962279154544ed69bb63eb14d677-Abstract-Conference.html.

Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, Hanbo Zhang, and Minzhao Zhu. GR-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024. URL https://arxiv.org/abs/2410.06158.

Sili Chen, Hengkai Guo, Shengnan Zhu, Feihu Zhang, Zilong Huang, Jiashi Feng, and Bingyi Kang. Video depth anything: Consistent depth estimation for super-long videos. *arXiv preprint arXiv:2501.12375*, 2025a. URL https://arxiv.org/abs/2501.12375.

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*, 2025b. URL https://arxiv.org/abs/2506.18088.

Tianxing Chen, Yao Mu, Zhixuan Liang, Zanxin Chen, Shijia Peng, Qiangyu Chen, Mingkun Xu, Ruizhen Hu, Hongyuan Zhang, Xuelong Li, and Ping Luo. G3Flow: Generative 3D Semantic Flow for Pose-aware and Generalizable Object Manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1735–1744, 2025c. URL https://openaccess.thecvf.com/content/CVPR2025/html/Chen_G3Flow_Generative_3D_Semantic_Flow_for_Pose-aware_and_Generalizable_Object_CVPR_2025_paper.html.

Yixiang Dai, Fan Jiang, Chiyu Wang, Mu Xu, and Yonggang Qi. FantasyWorld: Geometry-Consistent World Modeling via Unified Video and 3D Prediction. In *International Conference on Learning Representations*, pp. 103603–103622, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/a9500ba57161026d7f3f17f9f88d9a90-Abstract-Conference.html.

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. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1d5b9233ad716a43be5c0d3023cb82d0-Abstract-Conference.html.

Yilun Du, Sherry Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Brian Ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B. Tenenbaum, Leslie Kaelbling, Andy Zeng, and Jonathan Tompson. Video language planning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/848784373188ddf641079524e89e0ac9-Abstract-Conference.html.

Zhirui Fang, Qingchi Yu, Ziyang Chen, Longfei Li, Haoran Ma, Keru Zhou, Xinrun Xu, Samith Va, Yuxuan Hu, Peixuan Song, Qiang Du, Bin Qian, Yongkang Deng, Xin Li, Yezhen Wang, Zhe Li, Hao Luo, Shuyan Li, Ziwei Wang, Weijian Deng, and Xiu Li. EMERGE-Policy: A robot mind emerges beyond a single policy. *arXiv preprint arXiv:2608.29896*, 2026. URL https://arxiv.org/abs/2608.29896.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. LIBERO-Plus: A progressive robustness benchmark for visual-language-action models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Fei_LIBERO-Plus_A_Progressive_Robustness_Benchmark_for_Visual-Language-Action_Models_CVPR_2026_paper.html.

Haiwen Feng, Junyi Zhang, Qianqian Wang, Yufei Ye, Pengcheng Yu, Michael J. Black, Trevor Darrell, and Angjoo Kanazawa. St4RTrack: Simultaneous 4D reconstruction and tracking in the world. *arXiv preprint arXiv:2504.13152*, 2025. URL https://arxiv.org/abs/2504.13152.

Chongkai Gao, Haozhuo Zhang, Zhixuan Xu, Zhehao Cai, and Lin Shao. FLIP: Flow-Centric Generative Planning as General-Purpose Manipulation World Model. In *International Conference on Learning Representations*, pp. 21927–21948, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/37bf751554eb05465ed5dc60dced3f38-Abstract-Conference.html.

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*, 2026a. URL https://arxiv.org/abs/2604.26694.

Yanjiang Guo, Yucheng Hu, Jianke Zhang, Yen-Jen Wang, Xiaoyu Chen, Chaochao Lu, and Jianyu Chen. Prediction with action: Visual policy learning via joint denoising process. In *Advances in Neural Information Processing Systems*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/cbe25fa0e7c7084049276888a09acc8d-Abstract-Conference.html.

Yanjiang Guo, Lucy Shi, Jianyu Chen, and Chelsea Finn. Ctrl-World: A Controllable Generative World Model for Robot Manipulation. In *International Conference on Learning Representations*, pp. 6121–6138, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/0ae94013da7cd459402fd77874e09ee3-Abstract-Conference.html.

Zeyuan He, Bowen Yang, Zhirui Fang, Keru Zhou, Lei Jiang, Jingjing Qian, Fan Mo, Junchi Yan, Philip Torr, Xiu Li, Li Jiang, and Jialin Yu. ELAN4D: Embodiment-centric 4D supervision for vision-language-action models via plug-and-play adaptation. *arXiv preprint arXiv:2605.30484*, 2026. URL https://arxiv.org/abs/2605.30484.

Chengkai Hou, Yanjie Ze, Yankai Fu, Zeyu Gao, Songbo Hu, Yue Yu, Shanghang Zhang, and Huazhe Xu. 4D visual pre-training for robot learning. *arXiv preprint arXiv:2508.17230*, 2025. URL https://arxiv.org/abs/2508.17230.

Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video Prediction Policy: A Generalist Robot Policy with Predictive Visual Representations. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 24328–24346. PMLR, 2025. URL https://proceedings.mlr.press/v267/hu25g.html.

Wenlong Huang, Yu-Wei Chao, Arsalan Mousavian, Ming-Yu Liu, Dieter Fox, Kaichun Mo, and Li Fei-Fei. PointWorld: Scaling 3D World Models for In-The-Wild Robotic Manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/papers/Huang_PointWorld_Scaling_3D_World_Models_for_In-The-Wild_Robotic_Manipulation_CVPR_2026_paper.pdf.

Yuhang Huang, Jiazhao Zhang, Shilong Zou, Xinwang Liu, Ruizhen Hu, and Kai Xu. LaDi-WM: A Latent Diffusion-Based World Model for Predictive Manipulation. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 1726–1743. PMLR, 2025. URL https://proceedings.mlr.press/v305/huang25a.html.

Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, Yu Fang, Fengyuan Hu, Spencer Huang, Kaushil Kundalia, Yen-Chen Lin, Loïc Magne, Ajay Mandlekar, Avnish Narayan, You Liang Tan, Guanzhi Wang, Jing Wang, Qi Wang, Yinzhen Xu, Xiaohui Zeng, Kaiyuan Zheng, Ruijie Zheng, Ming-Yu Liu, Luke Zettlemoyer, Dieter Fox, Jan Kautz, Scott Reed, Yuke Zhu, and Linxi Fan. DreamGen: Unlocking Generalization in Robot Learning through Video World Models. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 5170–5194. PMLR, 2025. URL https://proceedings.mlr.press/v305/jang25a.html.

Zeren Jiang, Chuanxia Zheng, Iro Laina, Diane Larlus, and Andrea Vedaldi. Geo4D: Leveraging video generators for geometric 4D scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://arxiv.org/abs/2504.07961.

Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 3D Diffuser Actor: Policy Diffusion with 3D Scene Representations. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 1949–1974. PMLR, 2025. URL https://proceedings.mlr.press/v270/ke25a.html.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. In *Proceedings of Robotics: Science and Systems*, 2025. URL https://arxiv.org/abs/2502.19645.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026. URL https://arxiv.org/abs/2601.16163.

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. URL https://arxiv.org/abs/2601.21998.

Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.074. URL https://www.roboticsproceedings.org/rss21/p074.html.

Ying Li, Xiaobao Wei, Jiajun Cao, Hao Wang, Xiaowei Chi, Chengyu Bai, Qianpu Sun, Jiajun Li, Xiaojie Zhang, Peidong Jia, Jian Tang, Sirui Han, and Shanghang Zhang. WAM4D: Fast 4D world action model via spatial register tokens. *arXiv preprint arXiv:2606.14048*, 2026b. URL https://arxiv.org/abs/2606.14048.

Bokai Lin, Yifu Xu, Xinyu Zhan, Hongjie Fang, Jialin Tian, Fu-Cheng Zhang, Yong-Lu Li, Cewu Lu, and Lixin Yang. ChronoFlow-Policy: Unifying Past-Current-Future Interaction Flow in Visuomotor Policy Learning. *arXiv preprint arXiv:2606.31493*, 2026a. doi: 10.48550/arXiv.2606.31493. URL https://arxiv.org/abs/2606.31493.

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. In *International Conference on Machine Learning (ICML)*, 2026b. URL https://simonlinsx.github.io/RoboFlow4D_Page/.

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*, 2023.

Minghuan Liu, Zhengbang Zhu, Xiaoshen Han, Peng Hu, Haotong Lin, Xinyao Li, Jingxiao Chen, Jiafeng Xu, Yichu Yang, Yunfeng Lin, Xinghang Li, Yong Yu, Weinan Zhang, Tao Kong, and Bingyi Kang. Manipulation as in Simulation: Enabling Accurate Geometry Perception in Robots. In *International Conference on Learning Representations*, pp. 11238–11271, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/13183a224208671a6fc33ba1aa661ec4-Abstract-Conference.html.

Zeyi Liu, Shuang Li, Eric Cousineau, Siyuan Feng, Benjamin Burchfiel, and Shuran Song. Geometry-aware 4D video generation for robot manipulation. In *International Conference on Learning Representations*, 2026b. URL https://arxiv.org/abs/2507.01099.

Guanxing Lu, Shiyi Zhang, Ziwei Wang, Changliu Liu, Jiwen Lu, and Yansong Tang. ManiGaussian: Dynamic Gaussian Splatting for Multi-task Robotic Manipulation. In *European Conference on Computer Vision (ECCV)*, 2024. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/5194_ECCV_2024_paper.php.

Henrique Morimitsu, Xiaobin Zhu, Roberto M. Cesar Jr., Xiangyang Ji, and Xu-Cheng Yin. DPFlow: Adaptive optical flow estimation with a dual-pyramid framework. *arXiv preprint arXiv:2503.14880*, 2025. URL https://arxiv.org/abs/2503.14880.

NVIDIA. Isaac GR00T N1.7, 2026. URL https://github.com/NVIDIA/Isaac-GR00T.

Jing-Cheng Pang, Nan Tang, Kaiyuan Li, Yuting Tang, Xin-Qiang Cai, Zhen-Yu Zhang, Gang Niu, Masashi Sugiyama, and Yang Yu. Learning View-invariant World Models for Visual Robotic Manipulation. In *International Conference on Learning Representations*, pp. 54853–54876, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/8882d370cdafec9885b918a8cfac642e-Abstract-Conference.html.

Physical Intelligence. OpenPI: Open-source models and packages for robotics, 2025. URL https://github.com/Physical-Intelligence/openpi.

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. URL https://arxiv.org/abs/2504.16054.

Yichao Shen, Fangyun Wei, Zhiying Du, Yaobo Liang, Yan Lu, Jiaolong Yang, Nanning Zheng, and Baining Guo. VideoVLA: Video Generators Can Be Generalizable Robot Manipulators. In *Advances in Neural Information Processing Systems*, volume 38, pp. 95597–95621. Curran Associates, Inc., 2025. doi: 10.52202/085713-3197. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/89a3b655a8b68ae1c76b768152c9c19d-Abstract-Conference.html.

StarVLA Community. StarVLA: A lego-like codebase for vision-language-action model developing. *arXiv preprint arXiv:2604.05014*, 2026. URL https://arxiv.org/abs/2604.05014.

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. URL https://arxiv.org/abs/2602.10098.

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*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/e5b5c402bb7bd5e60bede6961d6fe39e-Abstract-Conference.html.

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

Chenyi Wang, Xinkai Wang, Bokai Lin, Jialin Tian, Fucheng Zhang, Cewu Lu, and Lixin Yang. Track4Action: Distilling world-centric 3D tracker into vision-language-action policies. *arXiv preprint arXiv:2608.03727*, 2026a. URL https://arxiv.org/abs/2608.03727.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual Geometry Grounded Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306, 2025b. URL https://openaccess.thecvf.com/content/CVPR2025/html/Wang_VGGT_Visual_Geometry_Grounded_Transformer_CVPR_2025_paper.html.

Jiaxu Wang, Yicheng Jiang, Tianlun He, Jingkai Sun, Qiang Zhang, Junhao He, Jiahang Cao, Zesen Gan, Mingyuan Sun, Qiming Shao, and Xiangyu Yue. MVISTA-4D: View-consistent 4D world model with test-time action inference for robotic manipulation. *arXiv preprint arXiv:2602.09878*, 2026b. URL https://arxiv.org/abs/2602.09878.

Junke Wang, Qihang Zhang, Shuai Yang, Yiming Luo, Yujun Shen, Zuxuan Wu, Yu-Gang Jiang, and Yinghao Xu. RepWAM: World action modeling with representation visual-action tokenizers. *arXiv preprint arXiv:2606.13674*, 2026c. URL https://arxiv.org/abs/2606.13674.

Mingxin Wang, Bin Hu, Bin Qian, Kaitao Jiang, Haoning Wu, Feng Yan, Bowen Jing, Ruiyang Hao, Enyi Wang, Kangning Niu, Yandan Yang, Mu Xu, Yan Wang, Houde Liu, and Tianlun Li. ST-WAM: Semantic-temporal world action model for robust manipulation under visual distribution shifts. *arXiv preprint arXiv:2607.28993*, 2026d. URL https://arxiv.org/abs/2607.28993.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3D Perception Model with Persistent State. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10510–10522, 2025c. URL https://openaccess.thecvf.com/content/CVPR2025/html/Wang_Continuous_3D_Perception_Model_with_Persistent_State_CVPR_2025_paper.html.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3D vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Wang_DUSt3R_Geometric_3D_Vision_Made_Easy_CVPR_2024_paper.html.

Xinkai Wang, Chenyi Wang, Yifu Xu, Mingzhe Ye, Fu-Cheng Zhang, Jialin Tian, Xinyu Zhan, Lifeng Zhu, Cewu Lu, and Lixin Yang. LaMP: Learning vision-language-action policy with 3D scene flow as latent motion prior. *arXiv preprint arXiv:2603.25399*, 2026e. URL https://arxiv.org/abs/2603.25399.

Youpeng Wen, Junfan Lin, Yi Zhu, Jianhua Han, Hang Xu, Shen Zhao, and Xiaodan Liang. VidMan: Exploiting Implicit Dynamics from Video Diffusion Model for Effective Robot Manipulation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 41051–41075. Curran Associates, Inc., 2024. doi: 10.52202/079017-1298. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/481c70828a4ff20d31a646cc6cc95f3d-Abstract-Conference.html.

Haoyu Wu, Diankun Wu, Tianyu He, Junliang Guo, Yang Ye, Yueqi Duan, and Jiang Bian. Geometry forcing: Marrying video diffusion and 3D representation for consistent world modeling. *arXiv preprint arXiv:2507.07982*, 2025. URL https://arxiv.org/abs/2507.07982.

Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing Large-Scale Video Generative Pre-training for Visual Robot Manipulation. In *International Conference on Learning Representations*, pp. 10641–10662, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/2c37c5bcef24b9541550261dcd63261b-Abstract-Conference.html.

Yuxi Xiao, Jianyuan Wang, Nan Xue, Nikita Karaev, Yuri Makarov, Bingyi Kang, Xing Zhu, Hujun Bao, Yujun Shen, and Xiaowei Zhou. SpatialTrackerV2: Advancing 3D Point Tracking with Explicit Camera Motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6726–6737, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Xiao_SpatialTrackerV2_Advancing_3D_Point_Tracking_with_Explicit_Camera_Motion_ICCV_2025_paper.html.

Jiazhi Yang, Kunyang Lin, Jinwei Li, Wencong Zhang, Tianwei Lin, Longyan Wu, Zhizhong Su, Hao Zhao, Ya-Qin Zhang, Li Chen, Ping Luo, Xiangyu Yue, and Hongyang Li. RISE: Self-improving robot policy with compositional world model. *arXiv preprint arXiv:2602.11075*, 2026a. URL https://arxiv.org/abs/2602.11075v2.

Lishan Yang, Wenxuan Song, Xi Wang, Pingyue Sheng, Zheng Fang, Ziyang Zhou, Junjie He, Haodong Yan, Jiayi Chen, Nan Sun, Qiao Sun, 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*, 2026b. URL https://arxiv.org/abs/2608.08023.

Senqiao Yang, Chengyao Wang, Yuxin Chen, Zixuan Wang, Longxiang Tang, Haokun Gui, Jinhui Ye, Changsheng Lu, Xiaoyang Wu, Mingkang Zhu, Pengguang Chen, Shu Liu, Zhuotao Tian, Hengshuang Zhao, Bei Yu, and Jiaya Jia. Beyond data scaling: Representation-centric continued pre-training for vision-language-action models. *arXiv preprint arXiv:2608.27550*, 2026c. URL https://arxiv.org/abs/2608.27550.

Yandan Yang, Shuang Zeng, Tong Lin, Xinyuan Chang, Dekang Qi, Junjin Xiao, Haoyun Liu, Ronghan Chen, Yuzhi Chen, Dongjie Huo, Feng Xiong, Xing Wei, Zhiheng Ma, and Mu Xu. ABot-M0: VLA foundation model for robotic manipulation with action manifold learning. *arXiv preprint arXiv:2602.11236*, 2026d. URL https://arxiv.org/abs/2602.11236.

Yiguang Yang, Jiankun Peng, Xiaoming Wang, Yiran Zhang, and Zhibo Fang. MT-WAM: Reorienting the one-pass predictive representation toward action generation. *arXiv preprint arXiv:2609.21474*, 2026e. URL https://arxiv.org/abs/2609.21474.

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. URL https://arxiv.org/abs/2602.15922.

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. *Robotics: Science and Systems*, 2024.

Hongyin Zhang, Pengxiang Ding, Shangke Lyu, Ying Peng, and Donglin Wang. GEVRM: Goal-Expressive Video Generation Model For Robust Visual Manipulation. In *International Conference on Learning Representations*, pp. 91789–91811, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/e538b85e58030aabb69945990b7f4139-Abstract-Conference.html.

Jianjun Zhang, Jian Zhu, Taiyi Su, Chong Ma, Zitai Huang, Yi Xu, and Hanli Wang. Learning 4D geometric priors for inference-efficient world action models. *arXiv preprint arXiv:2607.05468*, 2026a. URL https://arxiv.org/abs/2607.05468.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. MonST3R: A simple approach for estimating geometry in the presence of motion. In *International Conference on Learning Representations*, 2025b. URL https://arxiv.org/abs/2410.03825.

Zhanguang Zhang, Zhiyuan Li, Behnam Rahmati, Rui Heng Yang, Yintao Ma, Amir Rasouli, Sajjad Pakdamansavoji, Yangzheng Wu, Lingfeng Zhang, Tongtong Cao, Feng Wen, Xinyu Wang, Xingyue Quan, and Yingxue Zhang. Do world action models generalize better than VLAs? a robustness study. *arXiv preprint arXiv:2603.22078*, 2026b. URL https://arxiv.org/abs/2603.22078.

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*, 2026a. URL https://arxiv.org/abs/2607.06559.

Ruiteng Zhao, Zhengshen Zhang, Yue Su, Wenshuo Wang, Jiahui Li, Zhiyuan Yang, Francis E. H. Tay, Marcelo H. Ang, and Haiyue Zhu. SG-WAM: Self-guided world modeling in geometry-aware policy space. *arXiv preprint arXiv:2608.01397*, 2026b. URL https://arxiv.org/abs/2608.01397.

Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3D-VLA: A 3D Vision-Language-Action Generative World Model. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 61229–61245. PMLR, 2024. URL https://proceedings.mlr.press/v235/zhen24a.html.

Haoyu Zhen, Qiao Sun, Hongxin Zhang, Junyan Li, Siyuan Zhou, Yilun Du, and Chuang Gan. TesserAct: Learning 4D embodied world models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://arxiv.org/abs/2504.20995.

Gaoyue Zhou, Hengkai Pan, Yann Lecun, and Lerrel Pinto. DINO-WM: World Models on Pre-trained Visual Features enable Zero-shot Planning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79115–79135. PMLR, 2025. URL https://proceedings.mlr.press/v267/zhou25t.html.

Pengfei Zhou, Liliang Chen, Shengcong Chen, Di Chen, Wenzhi Zhao, Rongjun Jin, Guanghui Ren, and Jianlan Luo. Act2Goal: From World Model To General Goal-conditioned Policy. In *Proceedings of Robotics: Science and Systems*, 2026. doi: 10.15607/RSS.2026.XXII.015. URL https://www.roboticsproceedings.org/rss22/p015.html.

Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. In *Proceedings of Robotics: Science and Systems*, 2025a. doi: 10.15607/RSS.2025.XXI.015. URL https://www.roboticsproceedings.org/rss21/p015.html.

Fangqi Zhu, Hongtao Wu, Song Guo, Yuxiao Liu, Chilam Cheang, and Tao Kong. IRASim: A Fine-Grained World Model for Robot Manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9834–9844, 2025b. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhu_IRASim_A_Fine-Grained_World_Model_for_Robot_Manipulation_ICCV_2025_paper.html.

Haoyi Zhu, Honghui Yang, Yating Wang, Jiange Yang, Limin Wang, and Tong He. SPA: 3D Spatial-Awareness Enables Effective Embodied Representation. In *International Conference on Learning Representations*, pp. 26361–26391, 2025c. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/421fcf51a0e243f15f977553a6f482cb-Abstract-Conference.html.

Ruijie Zhu, Jiahao Lu, Wenbo Hu, Xiaoguang Han, Jianfei Cai, Ying Shan, and Chuanxia Zheng. MotionCrafter: Dense geometry and motion reconstruction with a 4D VAE. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Zhu_MotionCrafter_Dense_Geometry_and_Motion_Reconstruction_with_a_4D_VAE_CVPR_2026_paper.html.

Table 7: Standard LIBERO success rate (%). Avg. is the four-suite mean.

Method	Spatial	Object	Goal	Long	Avg.
Vision Language Action Models					
OpenVLA-OFT$_m$	97.7	98.0	96.1	95.3	96.8
$\pi_{0.5}$	98.8	98.2	98.0	92.4	96.9
ABot-M0	98.8	99.8	**99.0**	96.6	98.6
Structured-future VLAs					
Track4Action	95.0	99.6	97.6	95.8	97.0
ELAN4D ($\pi_{0.5}$)	98.2	98.8	96.8	94.2	97.0
VLA-JEPA	96.2	99.6	97.2	95.8	97.2
World Action Models					
Fast-WAM	98.2	**100.0**	97.0	95.2	97.6
MT-WAM	99.3	99.2	98.3	96.0	98.2
LaMP	**99.4**	99.8	97.4	96.7	98.3
LingBot-VA	98.5	99.6	97.2	**98.5**	98.5
SG-WAM	**99.4**	99.8	98.6	96.2	98.5
Cosmos Policy	98.1	**100.0**	98.2	97.6	98.5
4D-WAM	98.8	99.5	98.6	97.4	98.6
ST-WAM	99.0	**100.0**	**99.0**	96.8	98.7
Ours	**99.4**	99.6	98.6	97.6	**98.8**

A Appendix

A.1 Additional Quantitative Results

We provide the full-generalist perturbation breakdown and the complete Long-specialist study.

A.1.1 Standard LIBERO Results

Table 7 reports results on the standard LIBERO benchmark. Ours achieves a 98.8% average success rate across the four suites, remaining competitive with recent VLA and world–action models. This provides the clean-task reference for the robustness evaluations in the main paper.

A.1.2 Full-Generalist Perturbation Breakdown

Table 8 reports the suite-level counts underlying the Full-generalist result, and Table 9 reorganizes the same 10,030 episodes by perturbation family. The overall score is computed over all episodes and is therefore not an unweighted mean of either set of rows.

Table 8: LIBERO-Plus Full-generalist success by suite. One 25K policy is trained on the four clean suites and evaluated on their perturbed versions.

Suite	Successes / trials	Success rate (%)
Spatial	2,082 / 2,402	86.68
Object	2,342 / 2,518	**93.01**
Goal	1,894 / 2,591	73.10
Long	1,969 / 2,519	78.17
Overall	**8,287 / 10,030**	**82.62**

Table 9: LIBERO-Plus Full-generalist success by perturbation. Overall pools episodes across rows.

Perturbation	Successes / trials	Success rate (%)
Background textures	897 / 1,076	83.36
Camera viewpoints	1,059 / 1,599	66.23
Language instructions	1,377 / 1,537	89.59
Light conditions	1,087 / 1,142	**95.18**
Object layout	1,235 / 1,525	80.98
Robot initial states	1,333 / 1,550	86.00
Sensor noise	1,299 / 1,601	81.14
Overall	**8,287 / 10,030**	**82.62**

A.1.3 Complete Long-Specialist Variant Sweep

Table 10 collects the Long-specialist representation and action-conditioning variants.

Table 10: LIBERO-Plus Long-specialist success (%) on the same 2,519 episodes. VDA: relative depth; DPFlow: optical flow; PRGB: pseudo-RGB/Wan encoding.

Variant	Success	Δ vs. LingBot
Main 4D PRGB + wrist VDA	**82.73**	**+9.00**
Both-view VDA + DPFlow	81.78	+8.05
Main native 4D + wrist VDA	81.68	+7.95
Main VDA + DPFlow, wrist VDA	81.65	+7.92
Main 4D PRGB	81.42	+7.69
Main VDA + DPFlow	81.20	+7.47
Main native 4D	80.52	+6.79
Teacher 4D from predicted RGB (native)	80.41	+6.68
Both-view VDA PRGB	80.33	+6.60
Both-view native 4D	79.77	+6.04
4D auxiliary supervision only	77.54	+3.81
Geometry Forcing (layer-20 alignment)	77.42	+3.69
Direct 4D reconstruction supervision	75.63	+1.90
LingBot-VA	73.73	–
w/o RGB imagination conditioning (native)	71.86	-1.87

A.2 Experimental Protocols and Variant Definitions

A.2.1 Training and Evaluation

Standard LIBERO evaluates clean Spatial, Object, Goal, and Long tasks using the four-suite generalist. LIBERO-Plus Full-generalist evaluates a policy trained jointly on those four clean suites; the reported result uses the 25K checkpoint. The Long-specialist study trains on clean LIBERO-Long and uses the 2,519-case perturbed benchmark split. All Long-specialist variants are evaluated on the same 2,519 episodes. Both geometry-model adaptation and policy training exclude target OOD conditions.

Within each controlled comparison, variants share the LingBot-VA base checkpoint, training steps, batch size, gradient accumulation, unfreezing schedule, and teacher. The native/pseudo-RGB gradient diagnostics also use a matched probe batch. Representation-specific tokenization and timestep sampling are described below. On RoboTwin, the final policy trains for 15K steps; each evaluated task receives 100 rollouts per condition and tasks are equally weighted.

For our model, LIBERO-Plus overall success pools episodes rather than averaging the seven perturbation percentages. Standard LIBERO reports the four-suite mean. External scores retain their published aggregation and training protocols; these differ in pretraining data, model size, and optimization budget. VLAct uses additional multi-embodiment pretraining. The Full-generalist LingBot-VA row comes from EMERGE-Policy (Fang et al., 2026); it is distinct from our controlled Long baseline.

Standard LIBERO sources. The entries in Table 7 are taken from OpenVLA-OFT (Kim et al., 2025), $\pi_{0.5}$ (Physical Intelligence et al., 2025), ABot-M0 (Yang et al., 2026d), Track4Action (Wang et al., 2026a), ELAN4D (He et al., 2026), VLA-JEPA (Sun et al., 2026), Fast-WAM (Yuan et al., 2026), MT-WAM (Yang et al., 2026e), LaMP (Wang et al., 2026e), LingBot-VA (Li et al., 2026a), SG-WAM (Zhao et al., 2026b), Cosmos Policy (Kim et al., 2026), 4D-WAM (Yang et al., 2026b), and ST-WAM (Wang et al., 2026d). OpenVLA-OFT$_{\text{m}}$, ABot-M0, and our model use four-suite joint training.

A.2.2 Signals, Views, and Encoding Paths

Main view refers to the external scene camera (the `agentview` identifier in LIBERO configurations); wrist view is the end-effector camera. All geometric variants retain the same historical RGB observation interface. A main-only geometric variant omits wrist geometry, not wrist RGB.

The final signal set comprises main-view pointmaps, main-view 3D scene flow, and wrist-view relative depth. Video Depth Anything (VDA), using ViT-L with `metric=False`, supplies relative depth; DPFlow supplies pixel displacement between video frames. For depth, grayscale values are copied into three channels. Optical flow is encoded by horizontal displacement, vertical displacement, and magnitude using an episode-level robust scale. Both use the frozen Wan VAE. Thus the matched alternative to our model predicts main-view depth and optical flow plus wrist-view depth, using three geometric streams in either configuration.

The native 4D path instead consumes the reconstructor's geometry–motion latents through its dedicated token bridge. The pseudo-RGB path first decodes pointmaps and flow and re-encodes them as separate 48-channel Wan streams. The paths differ in latent space, stream layout, and tokenization. The final pseudo-RGB configuration samples training noise independently across streams. The native path and its wrist auxiliary stream share timestep samples in the supplied implementation.

A.2.3 Injection and Attention Ablations

The comparisons in Table 4 use native 4D targets and the both-view native reference (79.77%). Direct representation supervision predicts restored 4D latents with a head attached to noisy RGB hidden states. Geometry Forcing aligns noisy RGB hidden states from layer 20 to 4D latent features, adapting the intermediate-feature alignment approach of Wu et al. (2025). Explicit imagination predicts a separate noisy 4D stream and makes it available to action.

The direct-head attachment layer and parameter budget were not retained in the historical run metadata. Accordingly, this experiment compares recorded design variants rather than isolating the effect of the conditioning connection under a proven equal-parameter budget.

Auxiliary-only removes current geometry-to-action attention during training but retains geometric prediction supervision; geometry can be omitted at inference. The *w/o RGB imagination conditioning* variant removes only the current-chunk RGB conditioning connection to action: ground-truth RGB during training, and completed RGB imagination during inference. It retains RGB prediction, geometry prediction, and RGB-to-geometry attention. Action still reads observed RGB/action history and predicted 4D.

A.2.4 Teacher-Derived Geometry and Noise

In the teacher-derived condition, the policy first predicts future RGB and the 4D reconstructor processes that generated video. The resulting native 4D replaces the geometric condition used by action. With latent z^{teacher}, the perturbation condition supplies $\widetilde{z} = (1-\lambda)z^{\text{teacher}} + \lambda\epsilon$, where ϵ is Gaussian noise and $\lambda \in [0, 1]$.

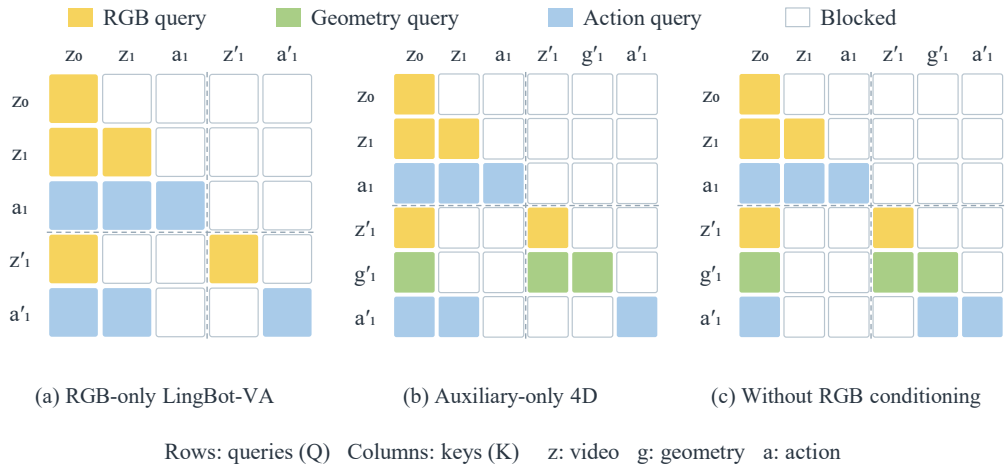


Figure 9: One-chunk attention-mask controls for the native-4D comparison. g' represents the geometric streams. Colors and the query/key convention match Figure 3. Auxiliary-only 4D blocks geometry-to-action attention. The RGB-conditioning ablation removes action access to the current-chunk RGB condition while retaining RGB generation, RGB-to-geometry attention, and predicted geometry.

Table 11: Action sensitivity to native-4D corruption. Teacher geometry is reconstructed from predicted future RGB and mixed with Gaussian noise using coefficient λ (Section A.2.4).

Geometry for action	λ	Success rate (%)
Predicted 4D	–	79.77
Teacher 4D	0.00	80.41
Teacher 4D + noise	0.25	80.41
Teacher 4D + noise	0.33	80.33
Teacher 4D + noise	0.50	79.65
Teacher 4D + noise	0.75	77.12
Teacher 4D + noise	1.00	75.35

Teacher-derived geometry reaches 80.41%, compared with 79.77% for direct 4D prediction. Success stays near this level at $\lambda = 0.25$ and 0.33, then decreases as the geometric condition is increasingly corrupted, reaching 75.35% at $\lambda = 1$. The curve measures sensitivity to this specific latent corruption. Teacher-derived geometry is reconstructed from generated RGB, not ground-truth future geometry, and its small improvement is not an oracle upper bound.

Here λ is the interpolation coefficient, not a calibrated error level. Gaussian scale, sampling axes, and run seeds were not retained in the run description; no repeated-run uncertainty is available for this probe.

A.3 Supervision and Optimization Diagnostics

A.3.1 Wrist-View 4D Supervision Quality

We compare a frozen MotionCrafter wrist-view reconstruction with RoboTwin simulator ground truth after one global 3D alignment for the sequence. In Figure 5a, the bottle rests on the table without moving in world coordinates, so its ground-truth scene flow is zero. The reconstruction nevertheless predicts nonzero scene flow on the bottle during wrist-view motion; its endpoint error therefore equals the magnitude of this spurious estimated motion. Matched RGB and error crops retain the gripper for context, and the error color scale spans 0–9.0 mm. We also measure per-loss gradients on the same shared Transformer parameters in a dual-view 4D run at step 1K (Figure 5). The cosine matrix uses a global batch of 32 across eight ranks; gradient noise scale is estimated from eight batch-8 gradients over 64 samples. In blocks 0/14/29, wrist pointmap versus main-view pointmap cosine is $-0.236/-0.312/-0.442$, while main-view pointmap versus scene-flow cosine

is $+0.494/+0.543/+0.695$. Wrist pointmap noise is 6.0/7.8/9.5$\times$ its main-view counterpart; wrist scene-flow noise is 2.1/2.7/2.8$\times$.

A.3.2 Pseudo-RGB Encoding and Geometric Gradients

We measured weighted, per-loss gradients with respect to the same trainable parameters in shared Transformer blocks 0, 14, and 29. The two step-10K checkpoints were evaluated on the same fixed set of 16 LIBERO-long training windows. RGB and action losses have weight 1; each geometric loss has weight 0.1. For pseudo-RGB, the geometry gradient is the vector sum of the pointmap and scene-flow gradients. The native path uses its single 4D-latent loss. The source training runs use matched training settings, and both checkpoints are probed on the same batch. Figure 11 resolves the pseudo-RGB pointmap and scene-flow gradients individually.

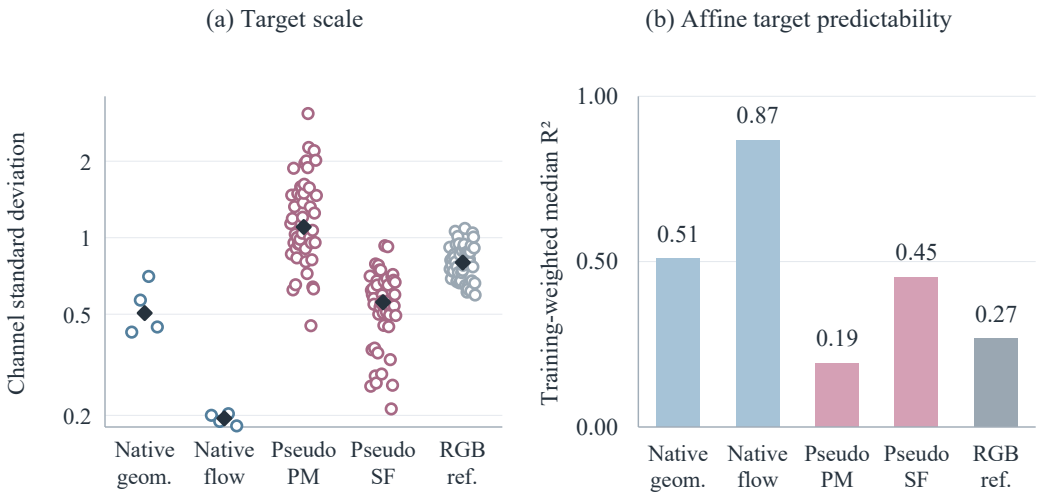


Figure 10: Geometric target statistics. (a) Per-channel standard deviations; diamonds mark medians. Native geometry and flow denote the geometry and motion channels of the 4D latent; PM and SF denote pseudo-RGB pointmap and scene-flow targets. (b) Training-weighted median R^2 of an affine predictor of the flow-matching target from its noisy input. RGB Wan latents provide a reference.

The geometry-to-action gradient-norm ratios for blocks 0/14/29 are $0.0023/0.0072/0.0472$ for native 4D and $0.0237/0.2206/0.3212$ for pseudo-RGB. The early-layer difference is directional as well as numerical: at block 0, pseudo-RGB geometry has cosine 0.468 with RGB and 0.261 with action, compared with -0.021 and 0.093 for native 4D. The pseudo-RGB geometry gradient also reaches 27% of the RGB gradient norm in this block, versus 2% for native 4D. These measurements show stronger geometric updates aligned with RGB and action in the early backbone. In pseudo-RGB blocks 14/29, geometry–RGB cosines fall to $0.148/0.008$ and geometry–action cosines to $0.018/-0.003$, indicating weaker cross-objective alignment in later blocks.

The target distributions in Figure 10 suggest why the native signal is harder to transmit. Across 20 LIBERO-long training episodes (10 tasks, two episodes each), native 4D scene-flow channels have standard deviations around 0.18–0.20, versus median values of 1.10 and 0.56 for Wan-encoded pseudo-RGB pointmaps and scene flow. Under the flow-matching path $x = (1-\sigma)z + \sigma\epsilon$ with target $\epsilon - z$, a small-amplitude z makes the target locally predictable from the noisy input. A best per-coordinate affine predictor computed from the observed variances has training-weighted median R^2 of 0.87 for native scene flow, compared with 0.19 for pseudo-RGB pointmaps and 0.45 for pseudo-RGB scene flow. This higher local predictability offers a potential shortcut in the native scene-flow objective; it does not demonstrate that the trained network actually uses that shortcut.

A short matched normalization control distinguishes gradient magnitude from useful alignment. With the same seed and training setup, per-channel standardization of native 4D latents raised the geometry-to-action gradient-norm ratio by $1.30/2.23/2.38$ times in blocks 0/14/29 at step 600. Yet its block-0 geometry gradients remained nearly orthogonal to RGB and action (cosines 0.014 and 0.023). Thus, standardization increases gradient magnitude without reproducing the early-layer

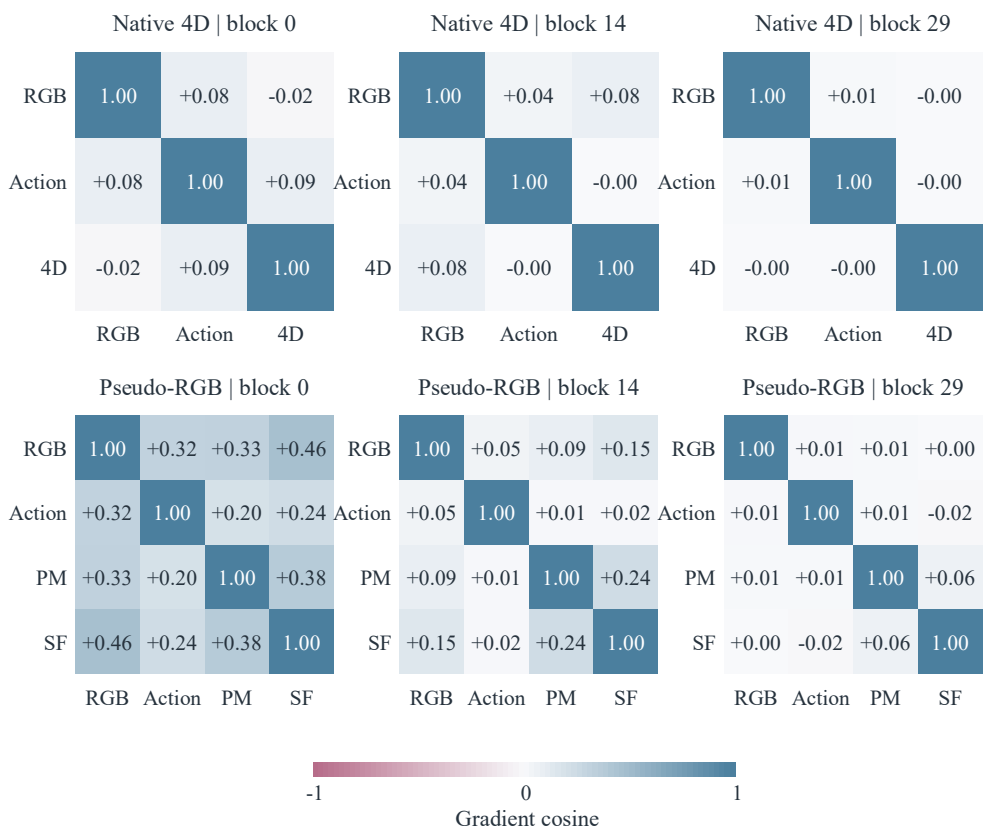


Figure 11: Per-objective gradient cosines by shared Transformer block (columns) and encoding path (rows). Pointmap and scene-flow gradients are shown separately; both align with RGB and action in block 0.

alignment observed with pseudo-RGB. The short probe does not establish a downstream success improvement. The separate encoding comparison in Table 5 reports downstream scores.

A.4 Method Details

A.4.1 4D Supervision Construction

The 4D VAEs and deterministic reconstruction network are trained in successive stages using pooled LIBERO and RoboTwin sequences. Pointmaps describe visible surface geometry, and forward scene flow describes the displacement of corresponding physical points. LIBERO pointmaps are obtained by back-projecting depth with camera intrinsics and transforming the resulting points using extrinsics; simulator correspondences supply motion supervision. The following details describe the RoboTwin data construction.

Geometry and validity. For each sequence and camera, we read RGB images, world-coordinate pointmaps, camera extrinsics, and the original point-validity flags. World points are transformed into each frame's camera coordinates. A pixel is geometrically valid only if its original validity flag is set, its coordinates are finite, and its camera-space depth is positive.

Rigid correspondences. Object segmentation identifies the rigid body associated with each valid point. Let $\bar{X}_t$ be a homogeneous point in world coordinates and T_t^i the object-to-world pose of its object i. Its corresponding position at the next frame is

$$\bar{X}_{t+1} = T_{t+1}^i (T_t^i)^{-1} \bar{X}_t. \tag{1}$$

The source and corresponding destination are transformed into a common reference frame, where their difference gives scene flow. This operation tracks the same physical point; it does not subtract

pointmaps at the same pixel index. In particular,

$$P_t^{\rightarrow}(u) = P_t(u) + F_t(u) \tag{2}$$

need not equal $P_{t+1}(u)$, because the corresponding point can project to another pixel or become occluded. Motion supervision requires a valid object identifier, source geometry, and object poses in both frames. The final frame has no successor and is excluded from flow supervision.

Samples are organized by interaction sequence and camera view. RGB is stored as video, with pointmaps, camera poses, scene flow, and validity masks in paired HDF5 files. The signals remain aligned frame by frame and pixel by pixel.

A.4.2 4D Representation Training

Clip preparation. Training uses 17-frame clips at a spatial resolution of 320×640. Points and motion are expressed in the first camera frame of the sampled clip. Let $\mathcal{V}$ collect its valid three-dimensional points in that frame. We compute their centroid c and mean centered radius s:

$$c = \frac{1}{|\mathcal{V}|} \sum_{P \in \mathcal{V}} P, \qquad s = \frac{1}{|\mathcal{V}|} \sum_{P \in \mathcal{V}} \|P - c\|_2. \tag{3}$$

Geometry and motion are normalized consistently as $P' = (P - c)/s$ and $F' = F/s$. Camera poses are adjusted to the same normalized coordinates; for a camera-to-reference pose, its translation becomes $(t - c)/s$ and its rotation is unchanged. Motion training samples consecutive frames so that each flow target retains its original temporal correspondence. This spatial normalization belongs to representation-model training and is separate from the policy's pseudo-RGB encoding below.

Geometry and motion VAEs. The 4D VAE is built on the pretrained LTX video-VAE architecture; adaptation begins from pretrained geometry and motion model weights (Zhu et al., 2026). Training first optimizes the geometry VAE with pointmap reconstruction, projected-depth, and surface-normal losses, collectively denoted $\mathcal{L}_{\text{geom}}$. The motion stage builds on the learned geometry representation: the geometry VAE is frozen, and the motion VAE is trained using scene-flow reconstruction and zero-flow regularization, collectively denoted $\mathcal{L}_{\text{motion}}$. Its decoder also receives geometry latents. Geometry and motion supervision use their respective validity masks.

Deterministic RGB-to-4D reconstruction. After VAE adaptation, both VAEs are frozen and the pretrained video-conditioned reconstruction network is trained to predict geometry and motion latents deterministically. Its objective combines latent matching with decoded supervision:

$$\mathcal{L}_{\text{rec}} = \mathcal{L}_{\text{latent}} + \beta_g \mathcal{L}_{\text{geom}} + \beta_f \mathcal{L}_{\text{motion}}. \tag{4}$$

Here $\mathcal{L}_{\text{latent}}$ matches predicted latents to encoded targets; decoded losses backpropagate through the frozen 4D VAE. The adapted reconstructor is subsequently frozen to generate policy-training targets offline. Thus, simulator geometry supervises the representation model, while the WAM learns future geometry from reconstructions of its training videos.

A.4.3 Policy Target Encoding

Pointmaps retain three coordinate channels, normalized separately using the third and ninety-seventh percentiles of valid sequence-level values. Scene flow uses symmetric ranges set by the ninety-ninth percentile of each absolute displacement component, preserving sign with zero motion at the midpoint. Wrist relative depth uses the first and ninety-ninth percentiles and is copied to three identical channels. Signals are clipped, resized, mapped to $[-1, 1]$, and encoded into 48-channel latents using the frozen Wan VAE's standardized posterior mean. These policy targets are distinct from the reconstructor's native 4D latents.

The adapted reconstructor and Video Depth Anything run offline over training videos, and their encoded outputs are cached. Neither teacher is called by our deployed policy. The teacher-from-predicted-RGB diagnostic in Section A.2.4 is a separate experimental intervention. Decoding policy predictions is needed only for visualization.

A.4.4 WAM TRAINING AND SAMPLING

Objective and optimization. For the linear probability path in Section 3.3, the velocity target follows directly from

$$\frac{\mathrm{d}}{\mathrm{d}\sigma}\big[(1-\sigma)z_0 + \sigma\epsilon\big] = \epsilon - z_0. \tag{5}$$

The final pseudo-RGB model samples noise timesteps independently per latent frame and stream. Squared errors are averaged over valid elements per frame and weighted by the scheduler's timestep weight $w_m(\sigma)$. RGB and action losses have unit modality weight; each geometric loss has weight 0.1. Differences in the native-4D path are specified in Section A.2.2.

During the first policy-training stage, only the geometric input projections, timestep-conditioning modules, and output projections are trainable. During the second, the full WAM Transformer and RGB, geometry, and action interfaces are jointly optimized. T5 and Wan VAE remain frozen. Evaluation-specific training budgets are reported in Sections A.2 and A.5.

Conditioning and cache conventions. The training sequence contains RGB/action conditioning copies and noisy prediction copies. In the one-chunk example of Figure 3, conditioning queries have key sets $z_0 : \{z_0\}$, $z_1 : \{z_0, z_1\}$, and $a_1 : \{z_0, z_1, a_1\}$. Prediction queries have key sets

$$\begin{aligned} z_1' &: \{z_0, z_1'\}, \\ g_1' &: \{z_0, z_1', g_1'\}, \\ a_1' &: \{z_0, z_1, g_1', a_1'\}. \end{aligned} \tag{6}$$

Each symbol denotes a block of tokens, and g_1' jointly includes all geometric streams. For later chunks, earlier RGB/action conditioning copies are also visible within the causal window. The current action conditioning copy a_1 is not visible to its prediction a_1'; it provides context for subsequent chunks. Conditioning queries never read noisy prediction copies. Language is supplied separately through cross-attention and is omitted from this self-attention diagram.

The current RGB conditioning copy z_1 is independently corrupted with probability 0.5 during training; its conditioning role is unchanged by this augmentation. Historical conditions comprise main- and wrist-view RGB and past actions, with no historical geometry. RGB prediction cannot read current geometry or actions, and geometry cannot read current actions.

At inference, RGB and geometry use a shared world denoising schedule. Each RGB pass provides temporary keys and values for the geometry pass at that step. After world denoising, completed RGB predictions fill the current RGB conditioning role, and completed geometry supplies the geometric condition. Action uses an independent denoising schedule with these conditions and history. Later cycles retain observed RGB and past actions as history rather than geometric imagination. On LIBERO, future RGB and action horizons are both 16; the architecture also supports different world and action horizons. Real-robot execution parameters appear in Section A.5.1.

A.5 REAL-ROBOT EVALUATION DETAILS

Mobile and Dynamic each use 700 demonstrations (630 training, 70 validation), excluding OOD configurations. All three tasks use 50 trials per method and condition.

Qualitative visualizations. **??** samples six synchronized frames per demonstration. RGB and offline geometric reconstructions share the same frame selection and crop.

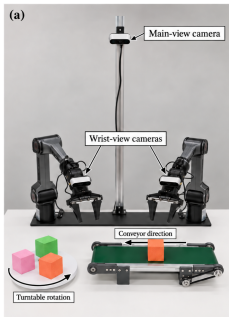
A.5.1 PLATFORMS AND COMMON PROTOCOL

Task 1 uses a humanoid robot. Task 2 uses an AgileX ALOHA bimanual robot system. For Tasks 1 and 2, Orbbec DaBai DCW binocular structured-light RGB-D cameras record 640×480 observations at 15 fps; all evaluated policies receive only RGB online. The task platforms are shown in Figure 12.

For Mobile and Dynamic, all four policies are trained once (one seed) for 5K updates with batch size 32 on NVIDIA A800 hardware. The final 5K checkpoint is evaluated, and real-robot inference



Humanoid robot



Dynamic ALOHA

Figure 12: Real-robot platforms: humanoid robot for observer-motion manipulation (left); AgileX ALOHA with a conveyor and turntable for moving pick–place (right).



Backpack zipping

Figure 13: Backpack-zipping demonstration: one gripper stabilizes the bag while the other pulls the zipper.

is deployed on one NVIDIA RTX 4090. We compare $\pi_{0.5}$ (Physical Intelligence et al., 2025), RGB-only LingBot-VA, the matched VDA-depth plus DPFlow control (Projective 4D), and our model. The WAM variants share the RGB observation interface and training budget; their geometric teachers generate cached training targets and are never queried during execution.

Horizon and execution protocol. LingBot-VA, Projective 4D, and our model use a LingBot-VA-style asynchronous deployment protocol (Li et al., 2026a). Camera observations are sampled at 12.5 Hz from the 15-fps capture stream. Each WAM predicts 16 future-world steps and a 32-step action chunk, executed at 30 Hz. A new inference request is launched after the first 16 actions, leaving 16 buffered actions (approximately 0.53 s) to execute while the next chunk is generated. With the optimized deployment setting, our model takes approximately 0.36 s per chunk on the RTX 4090, within that remaining action interval. The $\pi_{0.5}$ baseline follows the official OpenPI real-ALOHA runtime (Physical Intelligence, 2025), using a synchronous 25-step `ActionChunkBroker`; its execution schedule differs from the WAMs.

Evaluation scheduling and scoring. For Mobile and Dynamic, the same frozen final checkpoint is evaluated in Nominal and OOD conditions. Each split contains 10 predefined configurations with five repeats. All methods use the same reset identities, and method order is randomized within each configuration block. A predeclared hardware fault unrelated to policy behavior is rerun with the same reset; policy-caused collisions, object loss, and timeouts remain failures.

For each split c, the suite score is defined as

$$s_{\text{suite},c} = \frac{s_{\text{mobile},c} + s_{\text{dynamic},c} + s_{\text{backpack},c}}{3}. \tag{7}$$

Overall averages the Nominal and OOD suite scores.

A.5.2 Result Aggregation

Task success is $100k/n$, where k is the number of successful autonomous trials and n is the number of eligible trials. With 50 trials, task-level rates change in 2-point increments. Under a binomial approximation, the 95% Wilson intervals for our model are $[58.3, 82.5]$ for Mobile OOD (36/50) and $[25.9, 51.8]$ for Dynamic OOD (19/50). For Backpack, our model obtains 31/50 Nominal and 20/50 OOD successes, with Wilson intervals $[48.2, 74.1]$ and $[27.6, 53.8]$, respectively. These intervals use a binomial approximation and describe rollout uncertainty, not training-seed variability. Repeats within the same reset configuration may be correlated; method-level differences therefore require paired outcomes and repeated training runs for stronger statistical claims.

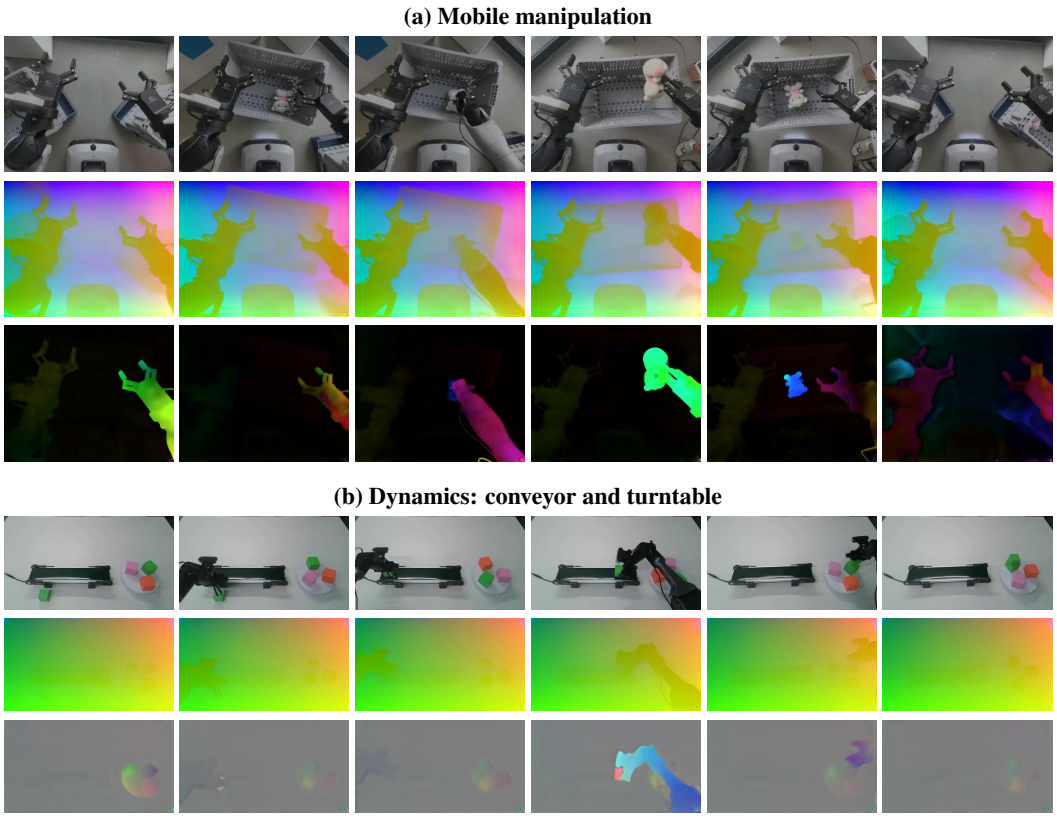


Figure 14: **Additional real-world manipulation sequences.** (a) Mobile manipulation with a changing onboard viewpoint. (b) Moving pick–place with a conveyor and rotating turntable. Rows show observed RGB, offline-reconstructed pointmaps, and scene flow from demonstration sequences.

A.5.3 Task 1: Observer-Motion Mobile Manipulation

Table 12: Task 1 mobile manipulation: successes/trials (rate) and nominal-to-viewpoint-OOD drop in points.

Method	Nominal	View OOD	Drop
$\pi_{0.5}$	**44/50 (88%)**	33/50 (66%)	22
LingBot-VA	42/50 (84%)	30/50 (60%)	24
Projective 4D	43/50 (86%)	34/50 (68%)	18
Ours	43/50 (86%)	**36/50 (72%)**	**14**

Task and evaluated capability. The humanoid robot yaws its mobile base, grasps an object with its right arm, places it in a destination bin, and returns to its initial orientation. Base translation, torso motion, and the left arm are disabled. Because the head and wrist cameras move while the external scene is largely static, the task is the physical counterpart of the simulated camera-viewpoint evaluation.

Evaluation protocol. Object and bin locations are held fixed across the viewpoint comparison so that observer motion is not conflated with layout change. A rollout succeeds if the robot places the object fully inside the destination bin, retains it there for 2 s, and returns within 10° of its initial yaw before the 60-s timeout, without dropping the object, collision, or safety intervention. The Nominal and viewpoint-OOD conditions compare performance under familiar and shifted observer configurations while keeping the object arrangement fixed.

A.5.4 TASK 2: DYNAMIC MOVING PICK–PLACE

Table 13: Task 2 moving pick–place: successes/trials (full-task rate).

Method	Nominal	Motion OOD
$\pi_{0.5}$	19/50 (38%)	9/50 (18%)
LingBot-VA	26/50 (52%)	14/50 (28%)
Projective 4D	28/50 (56%)	15/50 (30%)
Ours	**31/50 (62%)**	**19/50 (38%)**

Task and evaluated capability. On ALOHA, the pickup object moves on a conveyor and the placement target moves on a turntable. Successful execution therefore requires prediction of external scene motion, interception of the moving object, and release into a moving target.

Evaluation protocol. A rollout succeeds if the robot grasps the moving object, releases it inside the moving placement target, and the object remains within the target region for 2 s before the 45-s timeout, without collision, drop after placement, or safety intervention. Both conditions involve externally moving pickup and placement targets; Nominal and OOD success are reported separately in Table 13.

A.5.5 TASK 3: BIMANUAL BACKPACK ZIPPING

Task and evaluated capability. Two grippers close a backpack zipper: one stabilizes the bag while the other engages and pulls the zipper. Evaluation measures zipper closure; the preceding packing operations in the demonstration are outside the evaluated task.

Evaluation protocol. Each method is evaluated for 50 autonomous trials under Nominal conditions and 50 under configuration OOD. The target is zipper closure, separately from the preceding packing operations. Figure 13 shows the bag and two cooperating grippers from the head-camera view. The same complete-closure criterion is used for both conditions, so partial progress on the longer packing sequence is not counted as successful zipping.

Success criterion. Success requires complete closure to the zipper endpoint during autonomous execution, without human assistance, bag loss, or a safety intervention. Partial closure counts as failure.

Table 14: Backpack-zipping successes/trials (success rate). Mean averages Nominal and configuration OOD equally.

Method	Nominal	Configuration OOD	Mean (%)
$\pi_{0.5}$	27/50 (54%)	13/50 (26%)	40
LingBot-VA	29/50 (58%)	15/50 (30%)	44
Projective 4D	**31/50 (62%)**	18/50 (36%)	49
Ours	**31/50 (62%)**	**20/50 (40%)**	**51**

Related deformable-object tasks. RISE studies backpack packing (Yang et al., 2026a), and LingBot-VA studies clothes and pants folding (Li et al., 2026a); our task focuses on zipper closure.

A.6 EXTENDED RELATED WORK

We expand the three themes in Section 2, focusing on how geometric representations, future prediction, and action generation interact. Two distinctions guide the discussion: recovering an observed scene differs from predicting its future, and learning from a predictive target differs from supplying a generated prediction to the action model.

A.6.1 4D REPRESENTATIONS: GEOMETRY, CORRESPONDENCE, AND MOTION

From spatial reconstruction to dynamic correspondence. Pointmaps place image-indexed surface points in a shared 3D frame, providing a direct prediction target for geometric reconstruc-

tion (Wang et al., 2024). Dynamic reconstruction extends this formulation along two complementary directions: estimating geometry at successive times and establishing which physical points correspond across those times (Zhang et al., 2025b; Feng et al., 2025). The distinction matters because a sequence of depth maps or pointmaps need not identify the motion of the same surface point. MonST3R demonstrates geometry estimation in moving scenes without an explicit motion representation, whereas St4RTrack combines reconstruction and tracking in world coordinates. Video diffusion priors offer another route to temporal consistency by jointly estimating complementary geometric maps (Jiang et al., 2025). Together, these approaches make geometry and motion available from ordinary video, while differing in their coordinate frames, correspondence assumptions, and inference procedures.

Geometric targets for control. MotionCrafter jointly reconstructs pointmaps and dense scene flow in a shared coordinate system (Zhu et al., 2026). This representation makes the location and displacement of visible surfaces explicit. For exact world-frame targets, a static surface has zero scene flow even under camera motion; the same property does not hold for image-plane optical flow. It provides a useful representational bias, although learned reconstruction, occlusion, and changes of reference frame still limit invariance. Depth and optical flow remain strong alternatives: they supply spatial ordering and temporal correspondence with different estimation demands. Video Depth Anything provides the temporally consistent depth targets used for our wrist view (Chen et al., 2025a). Our view-specific choice reflects the reliability of available supervision, rather than a claim that every camera benefits equally from dense 4D prediction.

The connection to robot learning extends beyond reconstruction. Predicting future point clouds can improve representations subsequently used by policies (Hou et al., 2025). In our setting, an adapted reconstructor supplies offline targets for a WAM that must predict future geometry from its observation history and instruction. The reconstructor's native 4D VAE and the policy's video VAE have separate roles: the former supports geometric recovery, while the latter encodes normalized geometric targets in the WAM's existing latent interface. This design concerns transfer into a pretrained control model. Reconstruction fidelity alone does not determine whether a representation will be useful for action.

A.6.2 World Action Models: Prediction Objectives and Control Pathways

Coupling prediction and action. Video planning treats future visual observations as an intermediate plan and recovers executable actions afterward (Du et al., 2023; 2024). More integrated approaches share predictive representations or generative computation between vision and action. Predictive inverse dynamics conditions actions on anticipated visual states, while joint denoising and unified video–action models learn both modalities within a common architecture (Tian et al., 2025; Guo et al., 2024; Zhu et al., 2025a; Li et al., 2025). UWM's modality-specific diffusion times make policy, forward-dynamics, inverse-dynamics, and video-generation conditionings available within one formulation. Large-scale video pretraining further supplies dynamic priors for robot adaptation (Cheang et al., 2024; Kim et al., 2026; Ye et al., 2026). These methods share an interest in predictive learning, but their conditioning structures need not make action depend on a completed future rollout.

When the predicted future is consumed. LingBot-VA uses a causal video–action architecture in which visual prediction supports action generation (Li et al., 2026a). Fast-WAM instead preserves video co-training while removing future-video generation from the deployed policy (Yuan et al., 2026). Their distinction motivates separating two possible benefits: predictive training can improve the features available from current observations, and generated future states can provide additional conditions during action inference. Representation-centric tokenization also shows that the interface learned for prediction matters beyond image reconstruction quality (Wang et al., 2026c). Our question concerns the content and use of the future within this interface. We retain RGB imagination and add geometric motion streams whose predictions are directly available to action, then compare that pathway with auxiliary-only geometric training in the same backbone.

A.6.3 Representation Learning for VLA and WAM

Beyond the architectures discussed above, representation learning connects language-conditioned policies, predictive control, and geometric perception. We organize these connections by the information represented and its role in control, distinguishing policy representations from the reconstruction and simulation models that can support them.

Spatial representations for action. 3D Diffusion Policy encodes sparse point clouds for action diffusion, while 3D Diffuser Actor combines 3D scene features, language, and proprioception to denoise robot pose trajectories (Ze et al., 2024; Ke et al., 2025). SPA learns spatially aware visual features through differentiable rendering of multi-view observations (Zhu et al., 2025c). G3Flow instead maintains object-centric 3D semantic features using generated digital twins and pose tracking (Chen et al., 2025c); its semantic flow is distinct from dense physical scene flow. Manipulation as in Simulation improves the geometric input itself by denoising RGB-D measurements with simulation-trained Camera Depth Models (Liu et al., 2026a). Extending spatial representations to generative reasoning, 3D-VLA connects a 3D language model with action tokens and goal-image and point-cloud generation (Zhen et al., 2024). These approaches establish several routes for incorporating spatial structure, without requiring the same future representation or action interface.

Representing future motion. ManiGaussian uses future scene reconstruction in a dynamic Gaussian representation to supervise features for manipulation (Lu et al., 2024). RoboFlow4D predicts multi-frame 3D flows from visual observations and instructions, then conditions action on these explicit predictions (Lin et al., 2026b). PointWorld models action-conditioned scene evolution through 3D point flows, representing both scene changes and robot actions in a shared geometric space for model-predictive control (Huang et al., 2026). ChronoFlow-Policy represents past, current, and future interactions through sparse object and gripper keypoints and jointly learns flows and actions (Lin et al., 2026a). FLIP uses image-plane flows as a general action representation for flow-conditioned video planning (Gao et al., 2025). These methods differ in motion density, coordinate system, and whether predictions are conditioned on instructions or candidate actions. Explicit flow-guided control is therefore an established direction; our comparison concerns the particular geometric streams and their integration into a video–action backbone.

Predictive visual and latent features. GR-1 couples future-image and action prediction after video pretraining, and VideoVLA adapts video generation to joint visual and action forecasting (Wu et al., 2024; Shen et al., 2025). VidMan transfers video-diffusion dynamics into an inverse-dynamics model through shared parameters and attention adapters (Wen et al., 2024). Video Prediction Policy directly conditions action learning on predictive features inside a video diffusion model (Hu et al., 2025). Predictive targets need not be pixels: LaDi-WM forecasts features aligned with visual foundation models for diffusion-policy conditioning, while DINO-WM predicts DINOv2 patch features for action-sequence optimization (Huang et al., 2025; Zhou et al., 2025). These distinctions separate learning dynamics-aware features, consuming predicted latent states, and using a predictive model to evaluate candidate actions.

Visual plans and feedback representations. Gen2Act conditions robot execution on generated human videos (Bharadhwaj et al., 2025). Act2Goal uses imagined intermediate states at multiple temporal scales to combine local control with long-horizon goal progress (Zhou et al., 2026). CLOVER measures deviations from generated visual plans in a learned embedding space and uses feedback to refine actions and trigger replanning (Bu et al., 2024). GEVRM combines generated visual goals with contrastively learned internal embeddings for handling perturbations (Zhang et al., 2025a). ReViWo addresses viewpoint variation by separating view-invariant and view-dependent representations before world-model and policy learning (Pang et al., 2025). Together, these works show that robustness depends on both the representation of an anticipated state and how observations correct the resulting behavior.

Representations for imagined interaction and data. Ctrl-World provides action-controlled, multi-view video rollouts with memory for policy evaluation and improvement, while IRASim models fine-grained action-conditioned video dynamics for manipulation (Guo et al., 2026b; Zhu et al., 2025b). DreamGen uses embodiment-adapted video models to generate synthetic robot trajectories and obtains action labels through latent-action or inverse-dynamics models (Jang et al., 2025).

DreMa combines Gaussian scene representations with physics simulation to augment demonstrations through imagined interactions (Barcellona et al., 2025). Here, predictive representations can support training-data generation or evaluation, so their use does not by itself imply that the deployed policy reads a generated geometric future.

Geometric perception and world-model foundations. VGGT jointly predicts camera parameters, pointmaps, depth, and tracks; CUT3R maintains a recurrent state for continuous 3D reconstruction; and SpatialTrackerV2 couples 3D point tracking with depth and camera-motion estimation (Wang et al., 2025b;c; Xiao et al., 2025). FantasyWorld adds a geometric branch to a video foundation model to jointly represent video and 3D structure (Dai et al., 2026). These are relevant sources of geometric representations, rather than interchangeable manipulation policies. Our use of offline geometric supervision connects this perception capability to task-conditioned prediction, while the following discussion focuses on which predicted signals actually reach the action model.

A.6.4 4D for World Action Modeling: Where Geometry Enters Control

Geometry as training supervision. Geometric feature alignment can improve video generation by supervising internal features with a geometric teacher (Wu et al., 2025). Related WAM approaches supervise future depth, relational geometry, or trajectory structure through branches and objectives used during training (Li et al., 2026b; Zhang et al., 2026a; Yang et al., 2026b). These mechanisms transfer spatial and temporal information into features used by the policy. VLA approaches pursue a related objective through embodiment keypoint tracks or a frozen 3D tracker (He et al., 2026; Wang et al., 2026a). The supervision is not interchangeable: robot keypoints describe embodiment motion, whereas dense scene signals can also describe manipulated objects and surrounding surfaces. Track4Action retains learned track queries for action conditioning while discarding its tracker; it therefore should not be interpreted as an absence of motion-related features at inference. The relevant distinction is whether action reads a generated geometric future with an explicit reconstruction target.

Geometry as a predicted control representation. Generating structured futures for control has clear precedents. RGB, depth, and normal prediction supports geometric reconstruction and inverse dynamics; multi-view geometric generation also enables pose-based action recovery and optimization of a trajectory latent (Zhen et al., 2025; Liu et al., 2026b; Wang et al., 2026b). These methods establish that imagined geometry can mediate executable behavior. A policy can also consume predictive internal features without completing a geometric rollout. LaMP conditions a VLA action expert on hidden states from a partially denoised scene-flow expert (Wang et al., 2026e). RynnWorld-4D co-models RGB, depth, and optical flow, and its policy reads intermediate multimodal features extracted in one backbone pass (Zhao et al., 2026a). Thus, the presence of geometric generation during training does not by itself specify the representation supplied to action at deployment.

Closest architectural relationship and our scope. X-WAM combines video–action modeling with a depth branch and encodes depth through a shared video VAE (Guo et al., 2026a). Its unilateral connection lets depth read main-branch features, while the main branch does not read depth tokens; depth prediction can be disabled during action inference. This is a close precedent for adapting video priors to geometry. Our distinguishing design combines main-view pointmaps and scene flow with wrist depth, and explicitly exposes their completed predicted latents to action alongside RGB imagination and observed history. The geometric future is transient: it conditions the current action chunk and is not maintained as a persistent scene map. Neither teacher inference nor decoding geometric latents is required online.

Our contribution therefore concerns the combination of representation and information flow, rather than the introduction of geometry to robot control in general. The native-4D ablations compare alignment, auxiliary prediction, and direct action conditioning within one backbone; the final representation study further examines view selection and compatibility with the pretrained video encoder. These comparisons support the utility of the proposed pathway under the evaluated conditions. They do not establish that complete geometric denoising is always preferable to intermediate predictive features, or that pointmaps and scene flow universally outperform depth and optical flow. Published systems also differ in pretraining, sensing, and evaluation protocols, so their reported success rates alone cannot isolate these architectural choices.

A.7 LIMITATIONS

Several limitations remain. The geometric targets inherit reconstruction errors from the offline teacher, and sequence-level normalization removes a consistent metric scale across episodes. Our gradient analyses are diagnostic rather than causal explanations of the observed policy gains, and small differences between closely related variants require repeated training runs to quantify uncertainty. Moreover, pointmaps and scene flow describe geometry and motion but do not explicitly model contact, force, or object affordances, which may become important for more contact-rich manipulation. Future work could therefore combine stronger geometric teachers with contact-aware representations and larger-scale geometric pretraining, and study whether richer physical futures further improve generalization beyond visual perturbations.