Geometry-Aware Interactive World Simulator for 3D Robot Manipulation

Tianbao Zhang[1,2,5] Zeyu Liu[1] Shuyu Wu[1] Fanxing Li[1] Zheng Zhu[3]
Zhaoxin Fan[4] Danping Zou[1]
[1]Shanghai Jiao Tong University [2]Current Robotics [3]GigaAI [4]BUAA [5]NARWAL

***Abstract*—Interactive world models offer a promising route to scalable robot policy training and evaluation, but most existing robot simulators remain RGB-centric and are mainly studied in image-space or planar manipulation settings. This limits their usefulness for tasks that require 3D control, depth reasoning, and contact-aware robot-object interaction. We present *Geometry-Aware Interactive World Simulator*, an action-conditioned RGB-D world model for 3D manipulation. Our model learns compact RGB-D latents with a dedicated depth VAE and predicts future RGB-D observations conditioned on robot actions, enabling closed-loop rollouts that preserve both appearance and geometry. The predicted depths can be back-projected into point clouds, allowing the simulator to generate demonstrations and evaluate point-cloud policies directly. On Peg Insertion, Drawer Opening, and Block Stacking, simulator-generated RGB-D demonstrations train DP3 and FP3 policies that remain competitive with real-data training, and simulator scores show strong correlation with real-world scores. These results suggest that RGB-D interactive world models can serve as practical surrogates for data generation and policy evaluation in geometry-sensitive robot manipulation.**

I. Introduction

Interactive world models [2, 8, 13, 14, 19] can provide learned environments in which robot policies are trained, evaluated, and compared without repeatedly executing every trial on physical hardware. Recent action-conditioned video models [4, 9, 11] demonstrate this potential by predicting future observations in response to robot actions. However, most existing interactive robot simulators [14, 9, 1] are RGB-centric and are commonly evaluated in image-space or planar manipulation settings. Such settings do not capture the geometric requirements of manipulation skills involving vertical motion, insertion, lifting, stacking, or contact-rich object interaction.

For 3D manipulation, visually plausible RGB prediction is not sufficient [16, 18]. A useful world simulator must preserve depth ordering, occlusion, contact configuration, and robot-object spatial relationships over long horizons. These properties are especially important for point-cloud-based policies, whose observations depend directly on scene geometry. Without explicit geometric prediction, RGB-only world models may generate rollouts that look plausible but are not faithful enough for downstream 3D policy training or evaluation.

We introduce *Geometry-Aware Interactive World Simulator*, an action-conditioned RGB-D world model learned from real robot interaction data, as illustrated in Fig. 1. The simulator learns joint RGB-D latent states and predicts future RGB-D observations conditioned on robot actions. A lightweight depth VAE encodes geometric structure, while an action-conditioned dynamics model generates future latents autoregressively. The predicted depth maps are back-projected into point clouds, allowing the learned simulator to interface directly with 3D imitation policies. We validate the simulator on three real-world manipulation tasks: Peg Insertion, Drawer Opening, and Block Stacking. Our experiments evaluate three aspects: RGB-D prediction quality, simulator-generated demonstrations for policy training, and sim-to-real correlation for policy evaluation. Results show that the proposed simulator provides useful RGB-D rollouts for training DP3 [18] and FP3 [16] policies and that simulator performance strongly correlates with real-world performance.

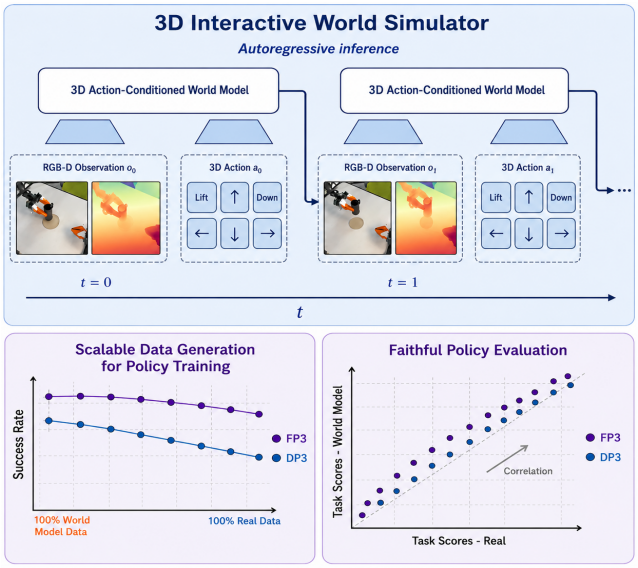

Fig. 1. **Overview of the 3D Interactive World Simulator.** The learned RGB-D world simulator enables autoregressive 3D interaction, scalable policy-training data generation, and faithful sim-to-real policy evaluation.

Our contributions are:

1) We formulate geometry-aware interactive world simulation as action-conditioned RGB-D generation for 3D robot manipulation.
2) We introduce a compact RGB-D latent modeling framework with depth VAE encoding and action-conditioned dynamics for temporally consistent rollouts.
3) We demonstrate that the learned simulator can generate useful policy training data and provide faithful policy ranking across three real-world 3D manipulation tasks.

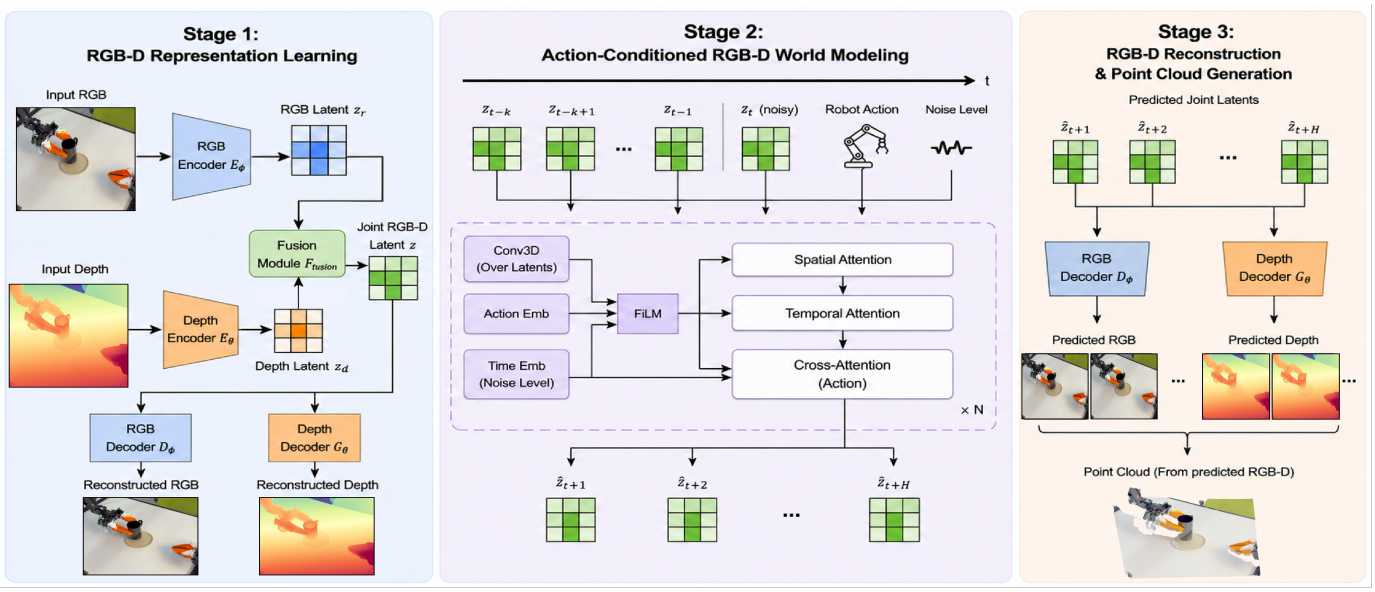

Fig. 2. **Geometry-Aware Interactive World Simulator.** Stage 1 learns compact RGB-D latents using RGB and depth encoders with modality-specific decoders. Stage 2 trains an action-conditioned latent dynamics model to predict future RGB-D latents from observation-action history. Stage 3 decodes predicted latents into RGB-D observations and reconstructs point clouds for 3D policy training and evaluation.

II. Related Work

Embodied Video Generation. Recent video generative models synthesize robot-world interactions by conditioning on different action representations. Text-conditioned methods [3, 7, 21] generate visual outcomes from high-level instructions but lack the precision needed for low-level manipulation. Latent-action methods such as IRASim [22] and Cosmos [1] encode instructions or 7-DoF poses into compact tokens, which may obscure fine-grained kinematics. Semantic approaches such as ORV [17] introduce 3D occupancy, but still lack explicit temporal interaction dynamics. Other methods use 2D visual prompts, including skeletons or masks [10, 5], yet such signals provide limited depth and geometric constraints. In contrast, our model predicts RGB-D observations conditioned on 3D robot actions, maintaining geometry-aware interaction over time.

3D/4D World Models. Recent 4D-native models such as Aether [23] and 4DNeX [6] generate spatiotemporal scenes, while TesserAct [20] and Robo4DGen [12] extend this idea toward 4D world generation. However, these methods often rely on static initial guidance, such as depth or Gaussians, and abstract action tokens, limiting fine-grained interactive control. Kinema4D [15] improves action grounding by transforming abstract actions into 4D pointmaps, providing explicit spatiotemporal cues. Our approach instead learns an action-conditioned RGB-D simulator that couples robot actions with reactive scene dynamics, enabling geometry-aware rollout, data generation, and policy evaluation.

III. Method

A. Problem Formulation

Given a robot interaction dataset

$$\mathcal{D} = \{(o_t, a_t)\}_{t=0}^{T}, \tag{1}$$

where $o_t = (r_t, d_t)$ contains an RGB image r_t and aligned depth map d_t, and a_t is the robot action, we learn an action-conditioned RGB-D world model:

$$\hat{o}_t = f_\theta(o_{t-N:t-1}, a_{t-N:t-1}). \tag{2}$$

The model predicts future RGB-D observations from recent observation-action history. Predicted depths are back-projected into point clouds for downstream 3D policies.

B. Geometry-Aware RGB-D World Model

Fig. 2 shows the overall framework. The model is trained with RGB-D representation learning followed by action-conditioned latent dynamics learning.

a) RGB-D Latent Representation.: Given (r_t, d_t), we encode RGB and depth into separate latents:

$$z_t^r = E_r(r_t), \qquad z_t^d = E_d(d_t), \tag{3}$$

where E_d is a lightweight depth VAE encoder. Unlike treating depth as an additional image channel, the depth VAE compresses metric geometry into a smooth latent space, reducing depth noise and stabilizing rollout. The RGB latent preserves appearance, while the depth latent captures compact scene geometry. We fuse them through a lightweight fusion module:

$$z_t = \Phi_{\text{fusion}}([z_t^r, z_t^d]), \tag{4}$$

and reconstruct both modalities with RGB and depth decoders.

Method	RGB			Depth			Point Cloud
	FVD↓	SSIM↑	PSNR↑	AbsRel↓	δ_1 ↑	δ_2 ↑	Chamfer L_1 ↓
Robo4DGen [12] (ICLR 2026)	21.82	75.13	20.02	22.45	66.04	82.39	0.2075
TesserAct [20] (ICCV 2025)	21.59	75.86	20.27	22.07	66.80	<u>82.60</u>	0.2030
Kinema4D [15] (ArXiv 2026)	**20.62**	<u>79.35</u>	**22.41**	<u>22.10</u>	<u>67.17</u>	**82.65**	**0.1886**
Ours	<u>21.18</u>	**79.40**	<u>22.35</u>	**21.96**	**67.50**	**82.65**	<u>0.1950</u>

TABLE I
COMPARISON OF RGB-D PREDICTION QUALITY. BEST RESULTS ARE HIGHLIGHTED IN BOLD AND SECOND-BEST RESULTS ARE UNDERLINED.

b) Action-Conditioned Latent Dynamics.: Given a history of RGB-D latents and actions, the dynamics model predicts the next latent:

$$\hat{z}_t = F_\psi(z_{t-N:t-1}, a_{t-N:t-1}). \tag{5}$$

Robot actions are embedded and injected through FiLM modulation and action cross-attention, allowing predicted RGB-D states to respond to 3D end-effector motions. We train the model with

$$\mathcal{L} = \mathcal{L}_z + \mathcal{L}_{rgb} + \mathcal{L}_{depth} + \mathcal{L}_{temp}, \tag{6}$$

where the first three terms supervise latent, RGB, and depth prediction. $\mathcal{L}_{temp}$ penalizes inconsistent changes between consecutive predictions, reducing long-horizon drift.

c) Closed-Loop Rollout.: At inference time, predicted latents are decoded into RGB-D observations, appended to the fixed context window, and used for the next prediction. The fixed window keeps per-step inference cost constant. Policies interact with the simulator by receiving RGB-D or point-cloud observations and outputting robot actions.

C. Policy Training and Evaluation in the Simulator

The learned simulator is used for both data generation and policy evaluation. For training, expert actions are executed inside the simulator to generate RGB-D demonstrations, which are converted into point clouds and used to train DP3 [18] and FP3 [16]. For evaluation, policies interact with the simulator in closed loop from controlled initial states, providing a low-cost proxy for comparing policies and checkpoints before real-world deployment.

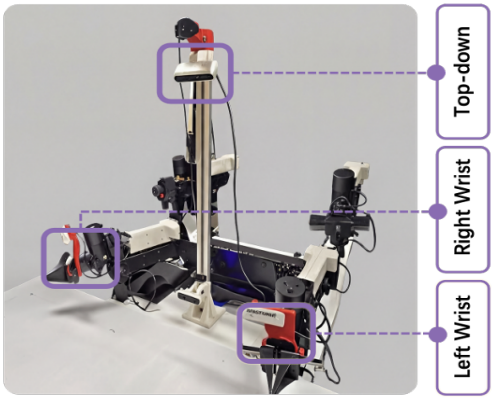

Fig. 3. **Real-world setup.** The bimanual AgileX platform equipped with three synchronized cameras.

IV. EXPERIMENTS

We evaluate whether the proposed simulator produces useful RGB-D rollouts, whether its generated demonstrations can replace part of real-world data, and whether simulator scores predict real-world policy performance.

A. Experimental Setup

We evaluate on **Peg Insertion**, **Drawer Opening**, and **Block Stacking**, which require alignment, contact-rich interaction, and object placement. Experiments are conducted on a bimanual AgileX platform equipped with three synchronized cameras, as shown in Fig. 3. We record aligned RGB-D observations, robot actions, and proprioceptive states when available, and collect about 600 play episodes per task, each with 100 steps.

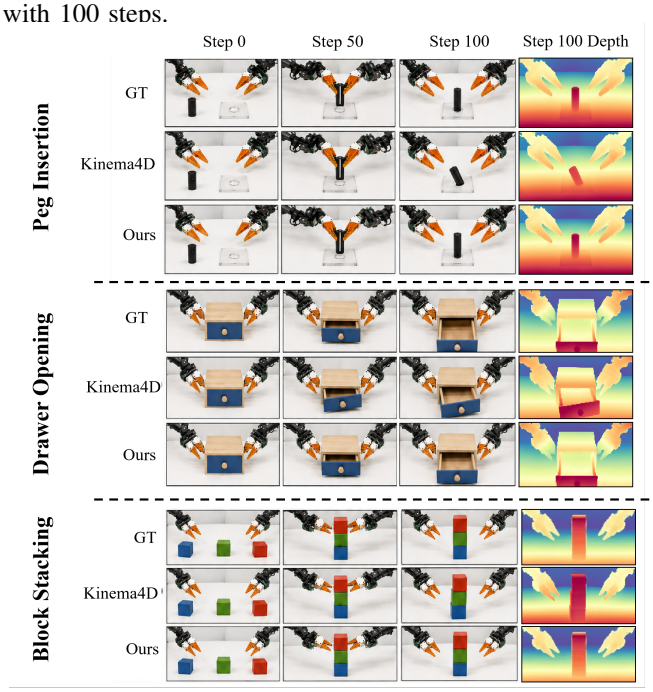

Fig. 4. **Qualitative RGB-D rollout comparison.** We compare GT, Kinema4D, and our method on Peg Insertion, Drawer Opening, and Block Stacking. Our method preserves more coherent object geometry, contact relationships, and final-step depth structure over long horizons.

B. RGB-D Prediction Quality

We compare RGB-D prediction quality with representative RGB-D and 4D generation baselines, including TesserAct [20], Robot4DGen [12] and Kinema4D [15]. These methods are used as RGB-D prediction references rather than closed-loop simulator baselines, since they are not designed

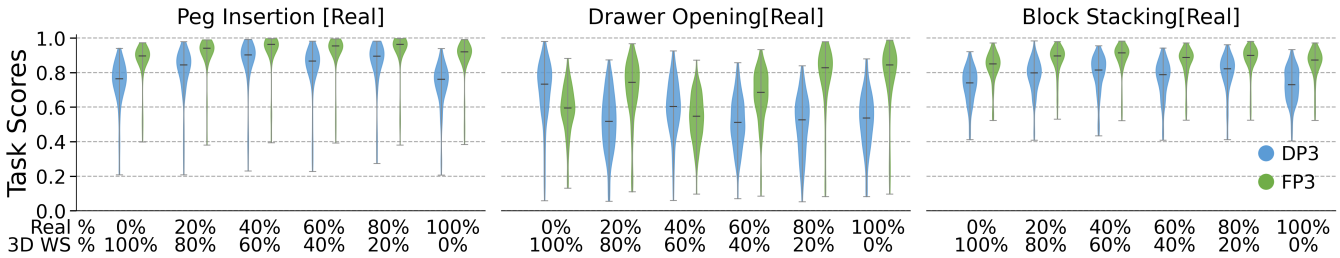

Fig. 5. **Policy training with mixtures of real-world and simulator-generated data.** We train DP3 [18] and FP3 [16] on fixed-size datasets with different ratios of real-world demonstrations and demonstrations generated by our RGB-D world simulator. "3D WS %" denotes the percentage of simulator-generated data, while "Real %" denotes the percentage of real-world data. Across Peg Insertion, Drawer Opening, and Block Stacking, policies trained with simulator-generated data achieve comparable task scores to those trained with real demonstrations, indicating that the learned simulator provides useful geometry-aware training data.

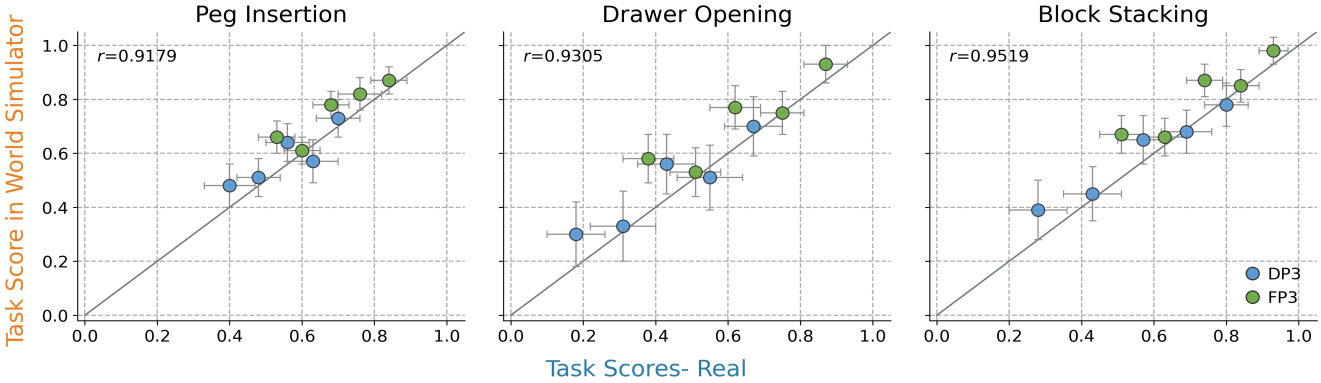

Fig. 6. **Correlation between simulator and real-world policy performance.** We evaluate DP3 [18] and FP3 [16] checkpoints in both our RGB-D world simulator and the real world under matched settings. Each point represents one trained policy, with error bars indicating evaluation variance. The strong positive correlations across all three tasks show that simulator scores reliably reflect real-world policy performance.

for action-conditioned policy interaction. All methods are trained with the same data and evaluated using matched initial observations and rollout horizons.

We report FVD, SSIM, and PSNR for RGB; AbsRel, δ_1, and δ_2 for depth; and Chamfer L_1 for reconstructed point clouds. Table I shows that our method achieves the best depth accuracy and competitive visual and geometric prediction quality. This indicates that the learned RGB-D rollouts are sufficiently accurate for downstream policy experiments. Qualitative results in Fig. 4 show that our simulator better preserves object geometry and contact relationships during long rollouts.

C. Policy Training with Simulator-Generated Data

We train DP3 [18] and FP3 [16] using fixed-size datasets with different simulator-real demonstration ratios. This evaluates whether simulator-generated data can replace real-world data while keeping the total number of demonstrations unchanged. All policies are trained with identical budgets and evaluated in the real world under held-out initial configurations. Fig. 5 shows that policies trained with simulator-generated RGB-D demonstrations remain competitive with those trained using real demonstrations. FP3 [16] consistently outperforms DP3 [18], and both methods maintain strong performance across mixture ratios. The results are strongest on geometry-sensitive tasks such as Peg Insertion and Block Stacking, suggesting that the simulator captures task-relevant spatial structure rather than only visual appearance.

D. Sim-to-Real Correlation for Policy Evaluation

We evaluate whether simulator performance can predict real-world performance. Intermediate and final DP3 [18]/FP3 [16] checkpoints are evaluated in the simulator and real world using matched initial configurations. Fig. 6 plots simulator scores against real-world scores. Across the three tasks, scores show strong positive correlation, indicating that policies ranked highly in the simulator also perform well in the real world. This supports using the learned RGB-D simulator as a reproducible policy-evaluation surrogate that reduces repeated physical trials.

V. Conclusion

We presented *Geometry-Aware Interactive World Simulator*, an action-conditioned RGB-D world model for 3D manipulation. By combining depth latent modeling with action-conditioned dynamics, the simulator produces closed-loop RGB-D rollouts that preserve appearance and geometry. Real-world experiments show that simulator-generated RGB-D data can train effective point-cloud policies and that simulator scores correlate with real-world performance, demonstrating the potential of geometry-aware world models for scalable

data generation and reproducible evaluation. One limitation is that evaluation focuses on tabletop tasks, while generalization to diverse objects, cluttered scenes, deformable materials, and longer-horizon interactions remains future work.

REFERENCES

[1] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

[2] Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.

[3] Haonan Chen, Bangjun Wang, Jingxiang Guo, Tianrui Zhang, Yiwen Hou, Xuchuan Huang, Chenrui Tie, and Lin Shao. World4omni: a zero-shot framework from image generation world model to robotic manipulation. *arXiv e-prints*, pages arXiv–2506, 2025.

[4] Taiye Chen, Xun Hu, Zihan Ding, and Chi Jin. Learning world models for interactive video generation. *Advances in Neural Information Processing Systems*, 38:154456–154483, 2026.

[5] Yixiang Chen, Peiyan Li, Jiabing Yang, Keji He, Xiangnan Wu, Yuan Xu, Kai Wang, Jing Liu, Nianfeng Liu, Yan Huang, et al. Bridgev2w: Bridging video generation models to embodied world models via embodiment masks. *arXiv preprint arXiv:2602.03793*, 2026.

[6] Zhaoxi Chen, Tianqi Liu, Long Zhuo, Jiawei Ren, Zeng Tao, He Zhu, Fangzhou Hong, Liang Pan, and Ziwei Liu. 4dnex: Feed-forward 4d generative modeling made easy. *arXiv preprint arXiv:2508.13154*, 2025.

[7] Xiao Fu, Xintao Wang, Xian Liu, Jianhong Bai, Runsen Xu, Pengfei Wan, Di Zhang, and Dahua Lin. Learning video generation for robotic manipulation with collaborative trajectory control. *arXiv preprint arXiv:2506.01943*, 2025.

[8] Junliang Guo, Yang Ye, Tianyu He, Haoyu Wu, Yushu Jiang, Tim Pearce, and Jiang Bian. Mineworld: a real-time and open-source interactive world model on minecraft. *arXiv preprint arXiv:2504.08388*, 2025.

[9] Yanjiang Guo, Lucy Xiaoyang Shi, Jianyu Chen, and Chelsea Finn. Ctrl-world: A controllable generative world model for robot manipulation. *arXiv preprint arXiv:2510.10125*, 2025.

[10] Yuxin Jiang, Shengcong Chen, Siyuan Huang, Liliang Chen, Pengfei Zhou, Yue Liao, Xindong He, Chiming Liu, Hongsheng Li, Maoqing Yao, et al. Enerverse-ac: Envisioning embodied environments with action condition. *arXiv preprint arXiv:2505.09723*, 2025.

[11] Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. *arXiv preprint arXiv:2503.00200*, 2025.

[12] Zeyi Liu, Shuang Li, Eric Cousineau, Siyuan Feng, Benjamin Burchfiel, and Shuran Song. Geometry-aware 4d video generation for robot manipulation. *arXiv preprint arXiv:2507.01099*, 2025.

[13] Xiaofeng Mao, Zhen Li, Chuanhao Li, Xiaojie Xu, Kaining Ying, and Kaipeng Zhang. Yume1. 5: A text-controlled interactive world generation model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7752–7761, 2026.

[14] Yixuan Wang, Rhythm Syed, Fangyu Wu, Mengchao Zhang, Aykut Onol, Jose Barreiros, Hooshang Nayyeri, Tony Dear, Huan Zhang, and Yunzhu Li. Interactive world simulator for robot policy training and evaluation. *arXiv preprint arXiv:2603.08546*, 2026.

[15] Mutian Xu, Tianbao Zhang, Tianqi Liu, Zhaoxi Chen, Xiaoguang Han, and Ziwei Liu. Kinema4d: Kinematic 4d world modeling for spatiotemporal embodied simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4495–4500, 2026.

[16] Rujia Yang, Geng Chen, Chuan Wen, and Yang Gao. Fp3: A 3d foundation policy for robotic manipulation. *arXiv preprint arXiv:2503.08950*, 2025.

[17] Xiuyu Yang, Bohan Li, Shaocong Xu, Nan Wang, Chongjie Ye, Zhaoxi Chen, Minghan Qin, Yikang Ding, Zheng Zhu, Xin Jin, et al. Orv: 4d occupancy-centric robot video generation. *arXiv preprint arXiv:2506.03079*, 2025.

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

[19] Yifan Zhang, Chunli Peng, Boyang Wang, Puyi Wang, Qingcheng Zhu, Fei Kang, Biao Jiang, Zedong Gao, Eric Li, Yang Liu, et al. Matrix-game: Interactive world foundation model. *arXiv preprint arXiv:2506.18701*, 2025.

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

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

[22] 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*, pages 9834–9844, 2025.

[23] Haoyi Zhu, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Chunhua Shen, Jiangmiao Pang, and Tong He. Aether: Geometric-aware unified world modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8535–8546, 2025.