MULTIWORLD: INTERACTIVE REAL-TIME MULTI-AGENT MULTI-VIEW VIDEO WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Interactive multi-agent video world modeling requires joint action control, consistent observations across viewpoints, and timely responses to actions. We present **MultiWorld**, a framework that generates synchronized views of a shared environment conditioned on multiple agents' actions. Agent Identity Embedding associates actions with agents, while a Dynamic World State Model provides shared context for view generation. Causal Teacher Distillation trains a few-step generator, with data-guided correction stabilizing distillation. View-parallel inference and pipelined decoding support real-time deployment. The framework supports varying numbers of agents and views. Experiments in multiplayer games and multi-robot manipulation show improvements in FVD, action following, and 3D consistency over several representative baselines. Our few-step model supports 24 FPS generation for both views, serving as a real game engine.

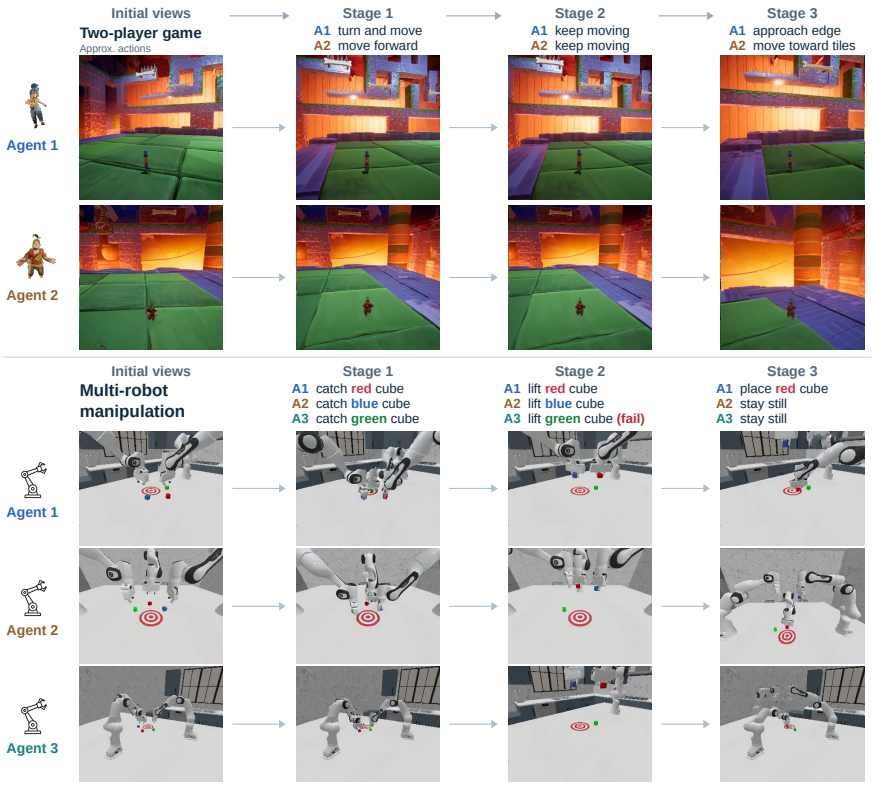

Figure 1: **MultiWorld generates multi-agent multi-view videos.** Given initial views and per-frame actions, our model produces action-controllable, multi-view consistent videos in both multi-player video game and multi-robot manipulation scenarios. Game action labels are approximate visual summaries of the depicted motion.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Video world models (Team et al., 2026; Kong et al., 2024; Mao et al., 2025; Huang et al., 2025a; Wang et al., 2024; Yang et al., 2026; Wu et al., 2025b) predict environmental dynamics from text or actions. Many approaches focus on a single agent, whereas collaborative robotics and multiplayer games require several agents to act within a shared environment. Each agent also observes that environment from a different viewpoint. Multi-agent simulation therefore requires joint action control and consistent observations across views.

This setting poses three linked challenges. (1) **Multi-Agent Controllability**: associate actions with the correct agents and model their joint effects. (2) **Multi-View Consistency**: generate compatible observations of the same environment from different viewpoints. (3) **Real-Time Interactivity**: respond to high frequency input actions, while remaining stable as generated frames accumulate. Moreover, the framework must also accommodate varying agent and view counts.

MultiWorld addresses these challenges through an interactive autoregressive framework (Fig. 1). Agent Identity Embedding associates actions with agents, and a Dynamic World State Model maintains shared context for multi-view generation (Section 3.1). We first pretrain a bidirectional model and fine-tune it to a multi-step autoregressive model via teacher forcing. In the distillation stage, unlike previous methods Huang et al. (2025b); Zhu et al. (2026) using a bidirectional model as a teacher, we propose Causal Teacher Distillation to distill a few-step autoregressive model from a multi-step autoregressive model. Data-Guided Correction uses paired data targets under generated histories to stabilize the transition from multi-step to few-step sampling (Section 3.2). View-parallel inference and pipelined decoding provide the system-side support for online interaction (Section 3.3).

We evaluate MultiWorld on two multi-agent, multi-view datasets. A real-player dataset from *It Takes Two* captures two-player interactions with distinct viewpoints and interdependent actions. A multi-robot manipulation dataset collected with RoboFactory (Qin et al., 2025) contains several thousand episodes with varying agent and camera counts. The evaluation measures visual quality, action following, and geometric consistency in both settings.

In summary, our key contributions include:

- We propose **MultiWorld**, a framework for scalable multi-agent, multi-view world modeling with joint action conditioning and synchronized dynamic environment modeling.
- We introduce Agent Identity Embedding and a Dynamic World State Model to associate actions with agents and maintain shared geometric context for consistent multi-view generation.
- We develop Causal Teacher Distillation with Data-Guided Correction to stabilize few-step autoregressive distillation, and combine view-parallel inference and pipelined decoding for real-time interactive deployment.

2 RELATED WORK

Interactive Video World Models. Interactive video world models extend video generation with camera and action control (He et al., 2024; Decart et al., 2024; Guo et al., 2025). Recent technical reports broaden this interface: EchoWM (Zhang et al., 2026) couples continuous navigation with audiovisual synthesis, while LingBot-World 2.0 (Gao et al., 2026) supports long-horizon action control and text-driven events. Geometry-aware memory in Matrix-Game 3.5 (Qian et al., 2026) and DreamX-World 1.0 (DreamX Team et al., 2026) helps preserve scenes during extended exploration. Causal generation and few-step distillation further reduce inference cost (Huang et al., 2025b; Zhu et al., 2026; Zhao et al., 2026a; Liu et al., 2025; Yuan et al., 2026). Context-Matched Distillation (CMD) (Bandyopadhyay et al., 2026) aligns teacher supervision with causal generation by using history-conditioned scores. Building on these advances in control, consistency, and efficiency, MultiWorld focuses on joint multi-agent actions and synchronized multi-view observations for real-time interaction.

Multi-Agent Modeling. Multi-agent planning (Feng et al., 2025) aims to coordinate multiple agents in a shared environment to achieve individual or shared goals. Multi-agent planning requires an envi-

ronment for execution. RoboFactory (Qin et al., 2025) investigated reasonable and safe multi-agent cooperation with compositional constraints in robotic manipulation tasks. SeqWM-MARL (Zhao et al., 2025) employs a sequential paradigm for multi-agent reinforcement learning in robotics tasks. Similarly, TeamCraft (Long et al., 2024) and CausalMACE (Chai et al., 2025) developed systems for multi-agent cooperation within Minecraft environments.

However, these works rely on physical simulators or game engines that require complex manual environment design. In contrast, video world models offer a generalizable alternative for simulating multi-agent environments, yet multi-agent video world models remain underexplored. Developing a multi-agent simulator presents several challenges, including controllability among multiple agents and ensuring multi-view consistency. MultiVerse (team, 2025) simulates a two-player racing game. COMBO (Zhang et al., 2024) attempts to integrate several single-agent video world models into a single multi-agent model, but this approach fundamentally neglects inter-agent interactions.

Solaris (Savva et al., 2026) builds a two-player video world model in Minecraft. Recent multi-agent and multi-view world models (Liu et al., 2026a; Sun et al., 2026; Hu et al., 2026; Zhao et al., 2026b) also study shared-world generation; our focus is causal distillation with data-guided self-correction for real-time interaction in a scalable multi-agent, multi-view framework.

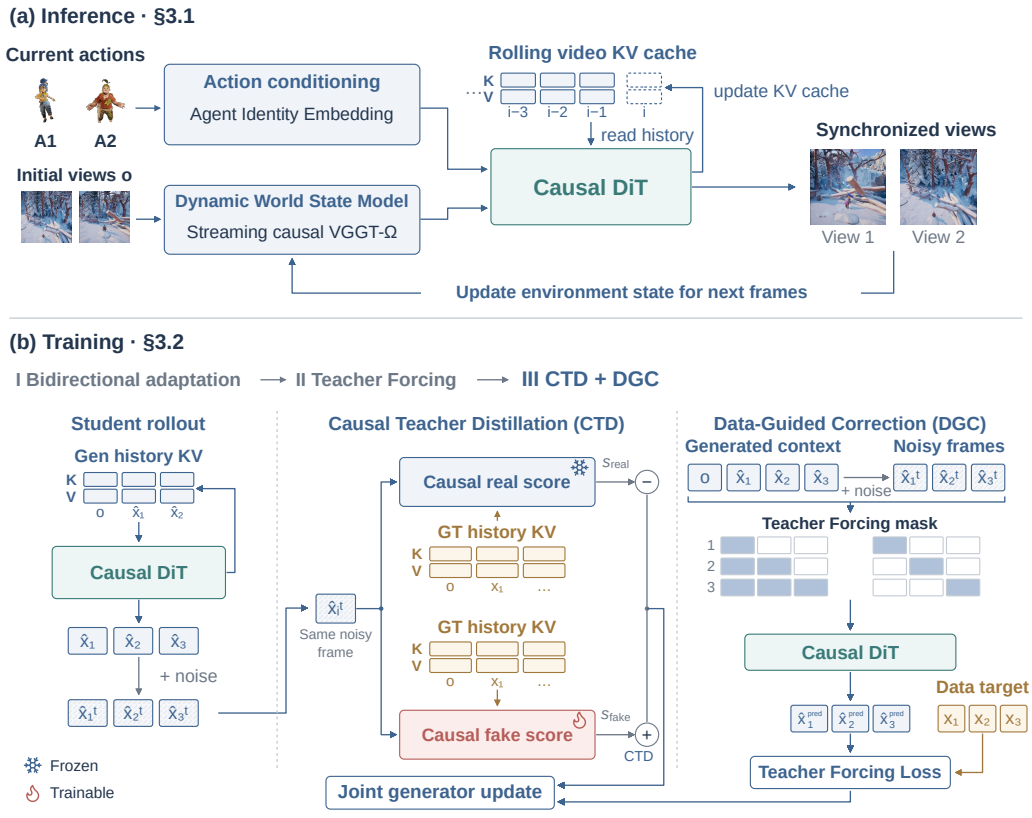

Figure 2: **MultiWorld inference and training.** (a) Interactive generation (Section 3.1): action conditioning and features from the Dynamic World State Model condition a Causal DiT with a rolling KV cache; generated views feed back into the Dynamic World State Model. (b) Training (Section 3.2): after bidirectional training and Teacher Forcing, Causal Teacher Distillation (CTD) combines a frozen real-score model and a trainable fake-score model, both conditioned on GT history. Data-Guided Correction (DGC) uses generated context and noisy generated frames in a Teacher Forcing pass, supervised by GT frames.

3 METHOD

MultiWorld generates synchronized views of a shared environment in response to multiple agents' actions. We first describe interactive inference and action conditioning and the Dynamic World State Model, then the three-stage training procedure, and interactive deployment optimizations.

3.1 MULTI-AGENT AUTOREGRESSIVE GENERATION

Interactive autoregressive inference. We consider K agents and V views, with initial observations $o = \{o^v\}_{v=1}^V$ and a text description c of the static environment. At latent frame index i, $a_i = (a_i^1, \ldots, a_i^K)$ collects the agents' actions, and $x_i = \{x_i^v\}_{v=1}^V$ contains one latent frame per view. The distribution over N latents factorizes as

$$p_\theta(x_{1:N} \mid o, c, a_{1:N}) = \prod_{i=1}^{N} p_\theta(x_i \mid x_{<i}, o, c, a_{1:i}), \tag{1}$$

where $x_{<i}$ denotes the preceding latent frames and $a_{1:i}$ contains only actions received through frame i. Let $h_{<i}$ denote the context before frame i, initialized from o. For each new frame, the model receives the current actions, generates the next latent frame, and commits its context:

$$(\hat{x}_i, b_i) = G_\theta(\epsilon_i, a_{1:i}, h_{<i}, c), \qquad h_{<i+1} = \mathrm{Update}(h_{<i}, b_i), \tag{2}$$

where ϵ_i is standard Gaussian noise and G_θ returns the current latent frame together with a KV block b_i from generated latent $\hat{x}_i$. The update commits this block for subsequent conditioning. FIFO rolling KV cache is adopted for long-term generation.

Multi-agent action conditioning. An action encoder independently embeds each a_i^k over time. Accurate action control requires distinguishing which agent performs each action. We therefore introduce Agent Identity Embedding (AIE), which applies rotary position embeddings (RoPE) (Su et al., 2024) along the agent-index axis to associate actions with their corresponding agents. Without AIE, the model may confuse action assignments, such as when two agents perform mirrored actions.

Dynamic World State Model. The Dynamic World State Model maintains lightweight memory tokens as an evolving representation of the shared environment. We fine-tune bidirectional VGGT-Ω (Wang et al., 2026) into a causal model that integrates synchronized observations from all viewpoints. Features read from these memory tokens condition the causal DiT, without passing an accumulating point cloud to the generator. After generating each latent frame, the synchronized views update the memory for subsequent frames. This update uses only past and current observations, never future frames.

3.2 TRAINING PROCEDURE

MultiWorld training includes three stages: Bidirectional Training; Teacher Forcing; and Causal Teacher Distillation (CTD) with Data-Guided Correction. The first two stages produce a multi-step causal model; the final stage distills a few-step causal model. History conditions $h_{<i}$ denote cached KV features of preceding frames. We omit the common action and text conditions for brevity. **Stage I: Bidirectional Training.** We first add action conditioning to a pretrained text-to-video model and train on video-action pairs. Random action dropout enables classifier-free guidance.

Stage II: Teacher Forcing. We adapt the bidirectional model for autoregressive generation using teacher forcing. Causal attention prevents access to future frames. Let $h_{<i}^{\mathrm{GT}}$ denote the history KV cache from clean data frames $x_{<i}$. At noise level $t \in (0, 1)$, we form $x_i^t = (1-t)x_i + t\epsilon_i$ and minimize the teacher forcing loss:

$$L_{\mathrm{TF}} = \mathbb{E}\left[\sum_{i=1}^{N} \left\| v_\theta(x_i^t, t \mid h_{<i}^{\mathrm{GT}}) - (\epsilon_i - x_i) \right\|_2^2\right]. \tag{3}$$

Stage III: Causal Teacher Distillation. A bidirectional teacher scores a clip jointly, so its supervision for a frame can depend on future frames unavailable during autoregressive generation. Causal Teacher Distillation (CTD) instead uses history-conditioned frame scores, matching this temporal

information boundary. The generator produces $\hat{x}_i$ conditioned on its own rollout $h_{<i}^{\text{gen}}$. We perturb each generated frame as $\hat{x}_i^t = (1-t)\hat{x}_i + t\epsilon_i$. Two causal score models compute the score of each noisy frame conditioned on preceding ground-truth history frames $h_{<i}^{\text{GT}}$:

$$s_i^{\text{real}} = s_{\text{T}}(\hat{x}_i^t, t \mid h_{<i}^{\text{GT}}), \qquad s_i^{\text{fake}} = s_\psi(\hat{x}_i^t, t \mid h_{<i}^{\text{GT}}). \tag{4}$$

The real-score model s_{T} is frozen, whereas the fake-score model s_ψ is trainable. Following distribution matching distillation (Yin et al., 2023), we update the generator using

$$\frac{\partial L_{\text{CTD}}}{\partial \theta} = \mathbb{E}\left[\sum_{i=1}^{N} w(t)(1-t)(s_i^{\text{fake}} - s_i^{\text{real}})^\top \left.\frac{\partial G_\theta}{\partial \theta}\right|_i\right], \tag{5}$$

where $w(t)$ weights the update across noise levels. We train the same fake-score model by flow matching on detached generated samples with

$$L_{\text{fake}} = \mathbb{E}\left[\sum_{i=1}^{N}\left\|\frac{-\hat{x}_i^t - t\, s_\psi(\hat{x}_i^t, t \mid h_{<i}^{\text{GT}})}{1-t} - (\epsilon_i - \hat{x}_i)\right\|_2^2\right]. \tag{6}$$

Data-Guided Correction (DGC). At the start of CTD, the generator is initialized from a multi-step model but sampled with a few-step algorithm. This mismatch causes large sampling errors that accumulate through autoregressive generation, severely degrading later frames. On these degraded states, real-score estimation can fail to provide useful corrective gradients, leading to training collapse. We introduce Data-Guided Correction (DGC) to supply an explicit correction signal during this transition.

For each generated frame, we retain its history $h_{<i}^{\text{gen}}$, perturb the frame to obtain $\hat{x}_i^t$, and train the generator to denoise it toward the paired data latent x_i. The data target therefore supervises recovery under the model's own imperfect history. To apply this supervision to all frame positions, including the degraded later frames, we use a packed Teacher Forcing pass with causal attention. Let ℓ_{corr} denote the per-frame data-supervision loss:

$$L_{\text{corr}} = \mathbb{E}\left[\frac{w_{\text{TF}}(t)}{N}\sum_{i=1}^{N}\ell_{\text{corr}}(\theta; \hat{x}_i^t, t, h_{<i}^{\text{gen}}, x_i)\right]. \tag{7}$$

Here, $w_{\text{TF}}(t)$ weights data-guided correction across noise levels. The initial observation remains a condition, while all N subsequent frames receive correction. This paired-data supervision complements the distribution-matching update:

$$\frac{\partial L_{\text{G}}}{\partial \theta} = \frac{\partial L_{\text{CTD}}}{\partial \theta} + \lambda_{\text{TF}}\frac{\partial L_{\text{corr}}}{\partial \theta}, \tag{8}$$

where $\lambda_{\text{TF}} \geq 0$. Both terms contribute to the same generator optimizer step.

3.3 Interactive Deployment Optimization

We combine view-parallel generation, pipelined execution, and an adaptive inference NFE schedule to support real-time interaction at approximately 24 FPS for two synchronized views. Our architecture enables parallel inference by decomposing multi-view generation into separate per-view generation processes conditioned on joint actions and shared features from the Dynamic World State Model. Appendix Fig. 6 illustrates the deployment pipeline. **View-parallel generation.** MultiWorld decomposes multi-view generation into separate single-view generation processes conditioned on shared context. This decomposition allows additional views to be assigned to additional GPUs. With sufficient hardware, per-frame generation latency can theoretically remain nearly constant as the number of views increases.

In our two-view deployment, 2 GPUs independently perform DiT denoising, while two additional GPUs decode the corresponding latents with separate VAE decoders. The causal Dynamic World State Model is colocated with the VAE decoding workers and updates the environment representation for subsequent generation. This configuration produces two 480×480 views at the same time.

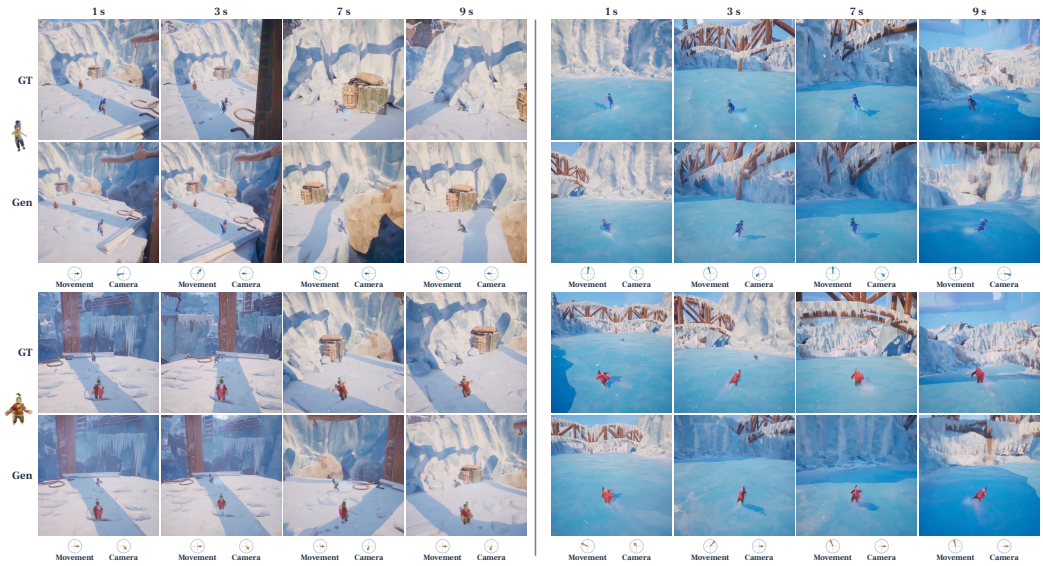

Figure 3: **Multiplayer game generation.** Within each panel, the upper and lower groups correspond to two players. Movement and camera signals are visualized below. The results show good action following and cross-view consistency even when the view discrepancy is large.

Pipelined execution. We initialize the video KV cache from the initial observations and update it through a clean forward pass after each generated latent frame. VAE decoding and delivery for latent frame i overlap with DiT denoising of latent frame $i + 1$, while incoming actions are queued for generation. This overlap reduces the time spent waiting for decoding and delivery between generation steps.

Adaptive Inference NFE. Inspired by Diagonal Forcing (Liu et al., 2026b), we allocate four denoising steps to the first generated latent frame and two to each subsequent frame. The larger initial budget helps establish a reliable starting context, while later frames reuse historical KV features and are generated with fewer denoising steps.

4 Experiments

We evaluate visual quality, action following, and geometric consistency in a two-player video game and multi-robot manipulation tasks involving two to four agents.

4.1 Experimental Setup

Datasets and Metrics. We construct approximately 200 hours real-player gameplay from *It Takes Two* after processing. The videos are resized to 960×480 at 24 FPS. The RoboFactory dataset contains manipulation tasks involving 2–4 agents. We measure visual quality using FVD (Unterthiner et al., 2018), PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018). We report reprojection error (RPE) (Wu et al., 2025a) for geometric consistency and use an inverse dynamics model (IDM) to assess action following ability. Details are provided in Appendix A and Appendix B.

Baselines. We compare three approaches. **Standard** models each view independently using an image-action-to-video backbone. **Concat-View** combines multiple views into a single video input; the evaluated implementation uses a fixed view count. **COMBO** (Zhang et al., 2024) first trains single-agent models and then composes them into a multi-agent model.

Main Results. MultiWorld achieves the best results on most metrics among the comparable baselines in both domains. In the game setting, it leads on FVD, LPIPS, SSIM, action following, and RPE, while PSNR remains comparable to COMBO. In robotics, it leads on FVD, PSNR, action following, and RPE, and ties on LPIPS and SSIM. Concat-View* uses a different training-view

Table 1: Comparison of multi-view conditioning strategies using bidirectional models for all methods. Action denotes discrete-action accuracy (%) for games and $100(1 - \mathrm{MSE})$ for robotics. Bold and underlined values mark the best and second-best distinct values, including ties. Concat-View* was trained on only two camera views per episode and is excluded from the ranking.

Method	FVD↓	LPIPS↓	SSIM↑	PSNR↑	Action↑	RPE↓
Multi-Player Video Game						
Standard	245	0.36	0.50	17.48	88.4	0.75
Concat-View	215	0.36	0.49	17.54	89.1	0.74
COMBO	<u>207</u>	<u>0.34</u>	<u>0.51</u>	**17.82**	<u>89.3</u>	<u>0.72</u>
Ours	**159**	**0.33**	**0.52**	<u>17.58</u>	**91.2**	**0.59**
Multi-Robot Manipulation						
Standard	100	**0.07**	**0.90**	26.39	88.2	1.60
Concat-View*	106	0.06	0.90	27.44	92.0	0.82
COMBO	<u>99</u>	<u>0.08</u>	**0.90**	<u>26.49</u>	<u>88.5</u>	<u>1.54</u>
Ours	**96**	**0.07**	**0.90**	**26.60**	**88.7**	**1.52**

configuration and is excluded from this ranking. These results demonstrate the effectiveness of the framework in two separately trained domains; the following ablations examine its components.

4.2 Ablation Study

Agent Identity Embedding. Table 2 compares no identity encoding with two AIE settings. The base controls the rotation periods of the identity embedding. AIE with base 20 achieves the best results across the three reported metrics.

Dynamic World State Model. Table 3 compares observation encoders. Causal VGGT-Ω has the lowest FVD, LPIPS, and RPE and the highest SSIM, while VGGT has the highest PSNR. None omits conditioning from the Dynamic World State Model. These comparisons evaluate backbone configurations, not the isolated effect of online memory. Additional component comparisons appear in Appendix D.

Table 2: **AIE Identity Embedding.**

Config	FVD ↓	PSNR ↑	Action ↑
w/o AIE	236	17.52	88.6
base=10k	234	17.53	89.2
base=20	**228**	**17.56**	**89.7**

Table 3: **Dynamic World State Model backbones.**

Encoder	FVD ↓	LPIPS ↓	SSIM ↑	PSNR ↑	RPE ↓
None	228	0.36	0.51	17.56	0.75
Wan VAE	256	0.36	0.50	17.38	0.71
DINOv2	232	0.36	0.50	17.48	0.72
VGGT	179	0.35	0.51	**17.72**	0.67
Causal VGGT-Ω	**159**	**0.33**	**0.52**	17.58	**0.59**

Distillation Design. Table 4 shows that CTD outperforms the bidirectional-teacher baseline, and that adding DGC further significantly improves generation quality. Two factors may explain these results. First, the per-frame action condition requires each generated frame to reflect action controls on the agent's states. A causal teacher respects this temporal information boundary and may therefore provide supervision better aligned with autoregressive generation. Second, our Teacher Forcing model achieves an FVD of 154 and a PSNR of 17.64 dB, slightly outperforming our bidirectional model. Meanwhile, the causal teacher scores generated frames given ground-truth history. Together, the teacher's generation quality and clean historical context may support more reliable score estimation.

Deployment Efficiency. Table 5 reports results on 4 NVIDIA H200 GPUs at batch size one, generating two 480×480 views. FPS counts two-view frame pairs per second. Interactive measurements include the browser-to-server-to-browser path after warm-up; first-output latency runs from the first action request to receipt of the first generated frame, excluding model loading. The bidirectional baseline is measured offline. At 4 sampling steps, system optimization increases throughput from 10.42 to 13.14 FPS without changing first-output latency (0.325 s). Reducing the inference NFE to 2 and 1 steps yields 24.80 and 41.74 FPS, with latencies of 0.24 and 0.15 s, respectively. Deployment optimization significantly increases throughput, enabling interactivity.

Table 4: **Stage III distillation.** Bidi Teacher uses bidirectional score estimation; CTD uses a causal teacher. DGC adds paired-data supervision under generated histories.

Model	FVD ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Bidi Teacher	414	14.27	0.44	0.49
CTD w/o DGC	397	16.29	0.49	0.40
CTD w/ DGC	**290**	**17.80**	**0.53**	**0.34**

Table 5: **Deployment efficiency.** Interactive timings are end-to-end after warm-up; the bidirectional baseline is offline. With system optimization, Multiworld supports real-time interactions.

Configuration	Steps	FPS ↑	Latency (s) ↓
Bidi, offline	30	6.90	34.20
CTD, serial	4	10.42	0.33
CTD, optimized	4	13.14	0.33
CTD, optimized	2	**24.80**	**0.24**

4.3 Qualitative Results

Action Following and Cross-View Consistency. Figure 3 compares generated frames with ground truth. The examples show movement through a snowy environment and across an icy surface, with movement and camera controls displayed alongside the frames. Appendix F includes the baseline comparison (Fig. 9) and further interaction examples.

Multi-Robot Failure Trajectory Simulation. Figure 4 illustrates a failed cooperative manipulation sequence involving an inter-robot collision. Modeling both successful and unsuccessful trajectories is relevant to robotic simulation, where collecting failures on physical hardware can risk damage. Our dataset includes successful and perturbed failure trajectories, constructed as described in Appendix A. An example of successful coordination is shown in Fig. 14.

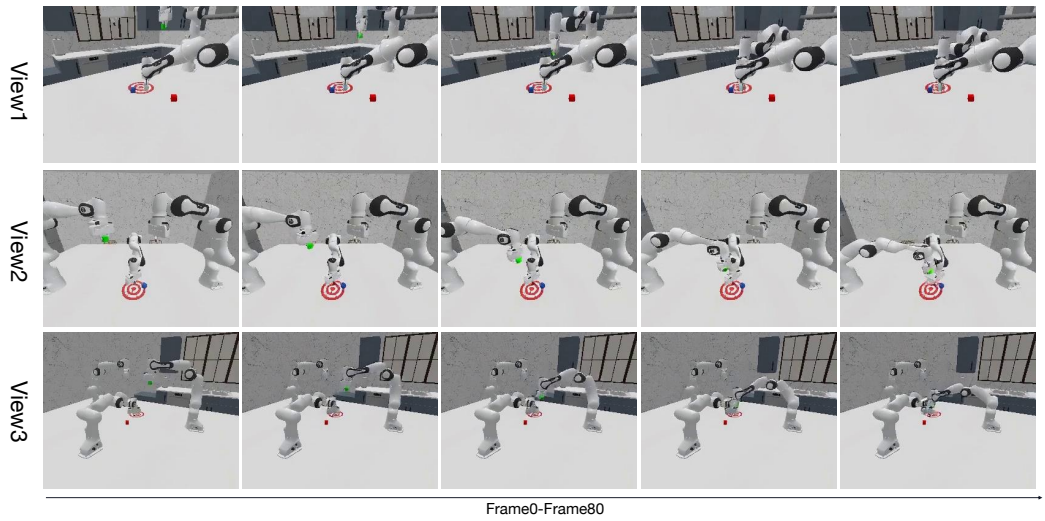

Figure 4: **Failure trajectory example in multi-robot manipulation.**

Long-Horizon Generation. Figure 5 shows multi-step autoregressive rollouts extending beyond the 10s training length to 20s clips remains consistent and stable.

Implementation Details. We initialize MultiWorld from Wan2.2-5B (Wan et al., 2025). Game training uses 10-second sequences at 24 FPS (240 RGB frames), with a resolution of 480×480

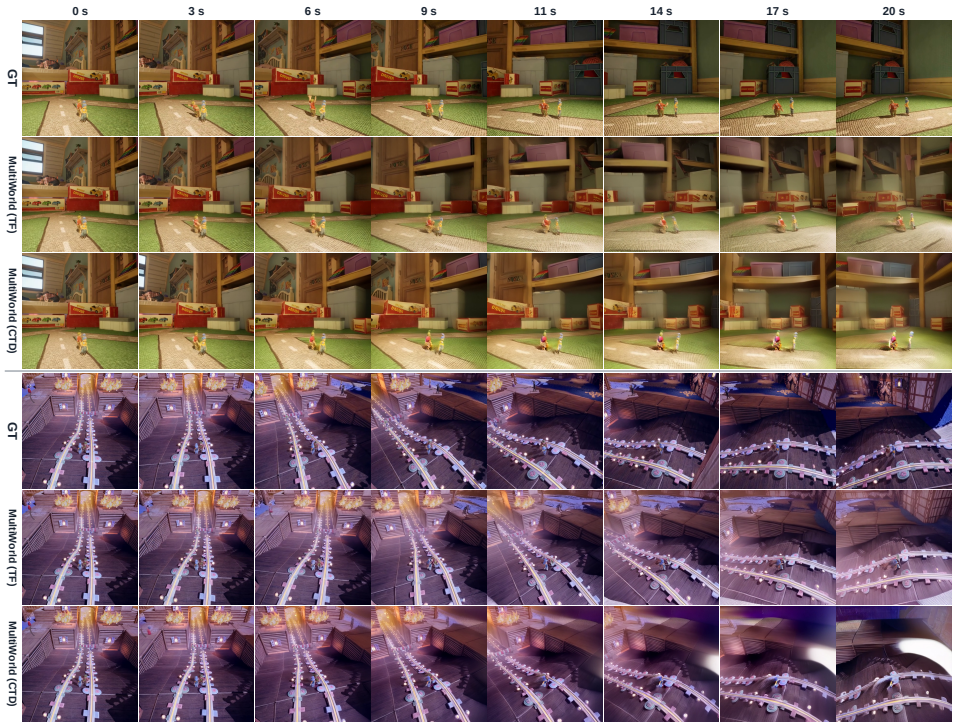

Figure 5: **Long-Horizon Generation.** The two examples compare ground truth (GT) with autoregressive MultiWorld rollouts. Models are trained on 10s clips and evaluated on approximately 20s sequences. Within each example, rows are ordered as GT, MultiWorld (TF), and MultiWorld (CTD).

per view. We use AdamW on 32 NVIDIA H200 GPUs, with 40,000 steps of bidirectional training followed by 40,000 steps of Teacher Forcing. Appendix C provides the training configuration. We train separate models for the game and robotics domains. All methods in Table 1, including MultiWorld, are evaluated as bidirectional models. This comparison evaluates multi-view conditioning rather than few-step autoregressive deployment.

5 Conclusion

MultiWorld generates synchronized views of a shared environment from multiple agents' actions. Agent Identity Embedding associates actions with agents, and a Dynamic World State Model provides shared context for view generation. Causal Teacher Distillation with data-guided self-correction supports few-step autoregressive generation, while view-parallel inference and pipelined decoding support interactive deployment. Evaluation in multiplayer games and multi-robot manipulation shows improvements in FVD, action following, and geometric consistency over the comparable baselines, with trade-offs in other visual metrics.

Limitations. Our evaluation covers a two-player game and manipulation tasks involving two to four robots. Larger agent populations and more diverse camera configurations remain outside this evaluation. Small or distant agents can appear visually ambiguous (Fig. 13), and computational constraints have limited exploration of larger-scale training.

Future Work. We will investigate long-horizon state drift and memory requirements, as well as interaction latency as the number of agents and views increases.

AI USE STATEMENT

Generative AI tools were used for translation and language editing to improve readability, sentence ordering, and logical flow, as well as to assist in preparing schematic diagrams. They were not used to construct datasets, generate research ideas, or design the proposed method. All AI-assisted text and figures were reviewed by the authors, who take full responsibility for their accuracy and integrity.

ETHICS STATEMENT

The gameplay data were collected by the authors for non-commercial research. The underlying game content remains subject to the rights of its respective owners; we do not claim ownership of these assets or intend to release them for commercial use. All robotic experiments were conducted in simulation, without physical robot deployment.

REPRODUCIBILITY STATEMENT

The main text and appendix describe the data collection and preprocessing procedures, model architecture, training configurations, and evaluation protocols to support reproduction of our experiments. Additional implementation details are provided in the appendix.

REFERENCES

Bowen Baker, Ilge Akkaya, Peter Zhokov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video pretraining (vpt): Learning to act by watching unlabeled online videos. *Advances in Neural Information Processing Systems*, 35:24639–24654, 2022. 2

Hmrishav Bandyopadhyay, Xuanchi Ren, Zijian Huang, Jay Zhangjie Wu, Tianshi Cao, Ruilong Li, Bryan Chu, Sanja Fidler, Yi-Zhe Song, and Zian Wang. Context-matched distillation: Teacher causality for autoregressive video distillation. *arXiv preprint arXiv:2608.13391*, 2026. URL https://arxiv.org/abs/2608.13391. 2

Qi Chai, Zhang Zheng, Junlong Ren, Deheng Ye, Zichuan Lin, and Hao Wang. Causalmace: Causality empowered multi-agents in minecraft cooperative tasks. *arXiv preprint arXiv:2508.18797*, 2025. 3

Decart, Quevedo Julian, McIntyre Quinn, Campbell Spruce, Chen Xinlei, and Wachen Robert. Oasis: A universe in a transformer. 2024. URL https://oasis-model.github.io/. 2

DreamX Team, Yancheng Bai, Rui Chen, Xiangxiang Chu, Rujing Dang, Hao Dou, Bingjie Gao, Qiwen Gu, Siyu Hong, Jiachen Lei, Geng Li, Jifan Li, Ruimin Lin, Qingfeng Shi, Bingze Song, Lei Sun, Jing Tang, Ruitian Tian, Jun Wang, Jiahong Wu, Pengfei Zhang, Shen Zhang, and Jiashu Zhu. DreamX-World 1.0: A general-purpose interactive world model, 2026. URL https://arxiv.org/abs/2606.16993. 2

Haoyi Duan, Hong-Xing Yu, Sirui Chen, Li Fei-Fei, and Jiajun Wu. Worldscore: A unified evaluation benchmark for world generation. *arXiv preprint arXiv:2504.00983*, 2025. 2

Leonhard Euler. *Institutionum calculi integralis*, volume 4. impensis Academiae imperialis scientiarum, 1845. 3

Zhaohan Feng, Ruiqi Xue, Lei Yuan, Yang Yu, Ning Ding, Meiqin Liu, Bingzhao Gao, Jian Sun, Xinhu Zheng, and Gang Wang. Multi-agent embodied ai: Advances and future directions. *arXiv preprint arXiv:2505.05108*, 2025. 2

Zelin Gao, Qiuyu Wang, Jiapeng Zhu, Jingye Chen, Zichen Liu, Qingyan Bai, Jiahao Wang, Yufeng Yuan, Hanlin Wang, Yichong Lu, Ka Leong Cheng, Haojie Zhang, Jian Gao, Tianrui Feng, Yuzheng Liu, Yao Yao, Yinghao Xu, Xing Zhu, Yujun Shen, and Hao Ouyang. Infinite worlds with versatile interactions, 2026. URL https://arxiv.org/abs/2607.07534. 2

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. 2

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024. 2

Anthony Hu, Václav Volhejn, Adrien Ramanana Rahary, Chris Mulder, Aditya Makkar, Alyx Liao, Amélie Royer, Manu Orsini, Adam Jelley, Eloi Alonso, Florian Laurent, Fredrik Norén, James Swingos, Jan Hünermann, Kent Rollins, Lucas Hosseini, Matthieu Le Cauchois, Maxim Peter, Pim de Witte, Tim Brown, Vincent Micheli, Moritz Böhle, Gabriel de Marmiesse, Viktoriia Sharmanska, Lucia Specia, Michael Black, and Patrick Pérez. Multiplayer interactive world models with representation autoencoders, 2026. URL https://arxiv.org/abs/2607.05352. 3

Siqiao Huang, Jialong Wu, Qixing Zhou, Shangchen Miao, and Mingsheng Long. Vid2world: Crafting video diffusion models to interactive world models. *arXiv preprint arXiv:2505.14357*, 2025a. 2

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv: 2506.08009*, 2025b. 2

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024. 2

Fangfu Liu, Kai He, Tianchang Shen, Tianshi Cao, Sanja Fidler, Yueqi Duan, Jun Gao, Igor Gilitschenski, Zian Wang, and Xuanchi Ren. Gamma-world: Generative multi-agent world modeling beyond two players, 2026a. URL https://arxiv.org/abs/2605.28816. 3

Jinxiu Liu, Xuanming Liu, Kangfu Mei, Yandong Wen, Ming-Hsuan Yang, and Weiyang Liu. Streaming autoregressive video generation via diagonal distillation, 2026b. URL https://arxiv.org/abs/2603.09488. 6

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time. *arXiv preprint arXiv:2509.25161*, 2025. 2

Qian Long, Zhi Li, Ran Gong, Ying Nian Wu, Demetri Terzopoulos, and Xiaofeng Gao. Teamcraft: A benchmark for multi-modal multi-agent systems in minecraft. *arXiv preprint arXiv:2412.05255*, 2024. 3

Xiaofeng Mao, Shaoheng Lin, Zhen Li, Chuanhao Li, Wenshuo Peng, Tong He, Jiangmiao Pang, Mingmin Chi, Yu Qiao, and Kaipeng Zhang. Yume: An interactive world generation model. *arXiv preprint arXiv:2507.17744*, 2025. 2

Runjia Qian, Zile Wang, Jihai Zhang, Kai Zou, Wei Yu, Jiaxing Li, Zexiang Liu, Yaokun Li, Fei Kang, Kaichen Huang, Mengyin An, Haobo Zhang, Biao Jiang, Jiahua Wang, Haofeng Sun, Yang Liu, and Yangguang Li. Matrix-Game 3.5: Enhancing real-time streaming interactive world models with patch memory, 2026. URL https://arxiv.org/abs/2608.29910. 2

Yiran Qin, Li Kang, Xiufeng Song, Zhenfei Yin, Xiaohong Liu, Xihui Liu, Ruimao Zhang, and Lei Bai. Robofactory: Exploring embodied agent collaboration with compositional constraints. *arXiv preprint arXiv:2503.16408*, 2025. 2, 3

Georgy Savva, Oscar Michel, Daohan Lu, Suppakit Waiwitlikhit, Timothy Meehan, Dhairya Mishra, Srivats Poddar, Jack Lu, and Saining Xie. Solaris: Building a multiplayer video world model in minecraft. *arXiv preprint arXiv:2602.22208*, 2026. 3

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 4

Huiqiang Sun, Zhan Peng, Size Wu, Kun Wang, Kang Liao, Dianyi Wang, Xingyu Zeng, Sheng Jin, Yangguang Li, Zhiguo Cao, Ziwei Liu, and Wei Li. Prisma-world: Camera-controllable multi-agent video world model, 2026. URL https://arxiv.org/abs/2606.09507. 3

Enigma team. Introducing multiverse: The first ai multiplayer world model, 2025. URL https://enigma.inc/blog. 3

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, et al. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026. 2

Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Advances in neural information processing systems*, 34:16558–16569, 2021. 2

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 6

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 8

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. VGGT-Ω. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. 4, 3

Xiaofeng Wang, Zheng Zhu, Guan Huang, Boyuan Wang, Xinze Chen, and Jiwen Lu. Worlddreamer: Towards general world models for video generation via predicting masked tokens. *arXiv preprint arXiv:2401.09985*, 2024. 2

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6

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*, 2025a. 6, 2

Tong Wu, Shuai Yang, Ryan Po, Yinghao Xu, Ziwei Liu, Dahua Lin, and Gordon Wetzstein. Video world models with long-term spatial memory. *arXiv preprint arXiv:2506.05284*, 2025b. 2

Ying Yang, Zhengyao Lv, Tianlin Pan, Haofan Wang, Binxin Yang, Hubery Yin, Chen Li, Ziwei Liu, and Chenyang Si. Stableworld: Towards stable and consistent long interactive video generation. *arXiv preprint arXiv:2601.15281*, 2026. 2

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. *arXiv preprint arXiv: 2311.18828*, 2023. 5

Shenghai Yuan, Yuanyang Yin, Zongjian Li, Xinwei Huang, Xiao Yang, and Li Yuan. Helios: Real real-time long video generation model. *arXiv preprint arXiv: 2603.04379*, 2026. 2

Hongxin Zhang, Zeyuan Wang, Qiushi Lyu, Zheyuan Zhang, Sunli Chen, Tianmin Shu, Yilun Du, and Chuang Gan. Combo: Compositional world models for embodied multi-agent cooperation. *arXiv preprint arXiv:2404.10775*, 2024. 3, 6

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018. 6

Songchun Zhang, Yaowei Li, Junhao Zhuang, Weiyang Jin, Haoyu Wang, Xin Lu, Yilang Sun, Shiyi Zhang, Haoran Li, Xiaoxiao Ma, Yuming Li, Yijun Liu, Yaofeng Su, Yanwen Ma, Haoyu Wu, Zihan Su, Yue Ma, Lvmin Zhang, Haoyang Huang, Zeyue Xue, Anyi Rao, and Nan Duan. EchoWM: Open and enterable omnimodal world models, 2026. URL https://arxiv.org/abs/2608.23189. 2

Min Zhao, Hongzhou Zhu, Kaiwen Zheng, Zihan Zhou, Bokai Yan, Xinyuan Li, Xiao Yang, Chongxuan Li, and Jun Zhu. Causal forcing++: Scalable few-step autoregressive diffusion distillation for real-time interactive video generation. *arXiv preprint arXiv:2605.15141*, 2026a. 2

Renjie Zhao, Yuxiang Wu, Mingyu Zhang, Jiaxin Li, Sisi Li, He Li, Yimin Sheng, Tianxi Tan, Zhenkai Zhang, Jiao Liang, Jianyi Zhu, and Yong-Lu Li. Population-scalable multi-agent world modeling, 2026b. URL https://arxiv.org/abs/2608.08600. 3

Zijie Zhao, Honglei Guo, Shengqian Chen, Kaixuan Xu, Bo Jiang, Yuanheng Zhu, and Dongbin Zhao. Empowering multi-robot cooperation via sequential world models. *arXiv preprint arXiv:2509.13095*, 2025. 3

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. *arXiv preprint arXiv: 2602.02214*, 2026. 2

APPENDIX FOR *MultiWorld*: INTERACTIVE REAL-TIME MULTI-AGENT MULTI-VIEW VIDEO WORLD MODELS

A DATASET CONSTRUCTION

Video Game Dataset. We collect two-player *It Takes Two* gameplay at 60 FPS and 2K resolution, together with synchronized control events from both players. After filtering and preprocessing, the training dataset contains approximately 200 hours of synchronized two-view video. Duration is counted once per synchronized sequence, rather than separately for each view. The final dataset contains 10-second clips at 24 FPS with two 480×480 views.

Collection Protocols. We specify four collection protocols with different behavioral constraints (Table 6). These constraints are imposed during recording, not during training. We mix data from all four categories for training without curriculum scheduling.

Table 6: **Gameplay collection protocols.** Durations are approximate collection totals before final filtering. Each synchronized two-view sequence is counted once.

Category	Recording constraints	Hours
Simple	Camera movement only.	$\sim$10
Medium	Camera and locomotion controls have restricted rates of change and action combinations. Players follow clear movement directions and avoid repeated direction switching.	$\sim$100
Complex	Task-oriented gameplay focused on progressing through and completing the game.	$\sim$20
Extremely complex	Unrestricted gameplay without constraints on player actions.	$\sim$100

Robotics Dataset. We collect four RoboFactory manipulation tasks involving two to four robots: striking, stacking with two robots, stacking with three robots, and passing with four robots. Each task contains 1,000 successful and 2,000 perturbed failure episodes. Episodes are recorded from multiple viewpoints at a reported resolution of 256×320.

Data Release. We plan to release the **full filtered gameplay dataset** of *It Takes Two*. We also plan to release the complete multi-robot cooperation dataset.

A.1 DATASET PREPROCESSING FOR IT TAKES TWO

Video Filtering and Segmentation. We associate each player's actions with the corresponding half of the left–right split-screen video. We remove intervals in which both players remain stationary, as well as cutscenes, death screens, and large-motion segments. Retained videos are resized to 960×480 at 24 FPS and divided into consecutive, non-overlapping 10-second clips; shorter remainders are discarded.

Action Alignment. We align both players' control events with the video timeline and resample them to 24 Hz. Within each $1/24$-second interval, a discrete button is marked as pressed if any press occurs. For continuous action channels represented as accumulated control inputs, we sum the recorded values along the temporal dimension within the interval, rather than averaging them. The resulting action vector is paired with the corresponding video frame.

Action Processing. Dead-zone normalization sets control magnitudes below 0.1 to zero and rescales the remaining magnitudes to $[0, 1]$. For a nonzero game-stick input $s = (s_x, s_y)$, we encode its magnitude $r = \|s\|_2$ and direction $\phi = \mathrm{atan2}(s_y, s_x)$ as

$$\gamma(s) = [r, \sin(\phi), \cos(\phi), \ldots, \sin(H\phi), \cos(H\phi)], \tag{9}$$

where H is the number of angular harmonics. These periodic features avoid discontinuities at angular wrap-around. Zero inputs use a canonical inactive representation because their direction is undefined.

Unifying Heterogeneous Action Spaces. The two players use different controllers (keyboard/mouse vs. gamepad), resulting in heterogeneous action spaces. To avoid separate action

encoders, we concatenate the two action types into a single vector. For each player, we mask the other's actions to zero, enabling a single encoder for both agents.

A.2 Dataset Preprocessing for the Robotics Dataset

We use the tasks provided in RoboFactory (Qin et al., 2025).

Failure Case Construction. We construct failure episodes from correct task plans by perturbing their execution. For example, in the strike-cube task, we preserve the sequence of arm movements and gripper operations but introduce controlled randomness at each step. This retains the task structure while introducing execution errors that can lead to failure.

B Metrics

We define the RPE and Action metrics reported in Table 1.

B.1 Reprojection Error

Following Duan et al. (2025); Wu et al. (2025a), we use DROID-SLAM (Teed & Deng, 2021) to reconstruct camera poses and pixel-wise depths from frame-to-frame features, refined through differentiable dense bundle adjustment. Reprojection error measures the mean distance between observed pixel positions and projected reconstructed points:

$$\mathrm{RPE} = \frac{1}{|\mathcal{V}|} \sum_{(i,j)\in\mathcal{V}} \left\| \mathbf{p}_{ij}^{*} - \Pi(\mathbf{P}_{ij}) \right\|_2 .$$

Here, $\mathcal{V}$ is the set of valid correspondences, $\mathbf{p}_{ij}^{*}$ is an observed pixel position, $\mathbf{P}_{ij}$ is the reconstructed 3D point, and Π is the camera projection. Lower values indicate smaller reprojection residuals under the reconstruction procedure. We use both temporal correspondences and synchronized cross-view correspondences to jointly reconstruct scene points and camera poses. Reprojecting these points into the corresponding views evaluates consistency across both time and viewpoint.

B.2 Action Following

Inverse Dynamics Model. Following VPT (Baker et al., 2022), we train a bidirectional IDM with a ResNet-50 backbone and a temporal processing layer to predict discrete and continuous actions. Training uses the same dataset for 20 epochs with a learning rate of 10^{-4}. We apply the IDM to generated videos and compare its predictions with the conditioning actions. Table 1 reports discrete-action accuracy for the game and $100\big(1 - \mathrm{MSE}(a_{\mathrm{pred}}, a_{\mathrm{gt}})\big)$ for robotics. The robot score is a transformed regression error, not classification accuracy; higher values indicate closer action predictions under this metric.

C Training and Implementation Details

C.1 Video Model Training

Table 7 summarizes the game-training configuration. Teacher Forcing follows bidirectional training; CTD and Data-Guided Correction then train the few-step generator. Bidirectional training uses a learning rate of 10^{-4}, with a 1,000-step warm-up from 10^{-5}. Its per-GPU batch size is four across 32 workers, giving a global batch size of 128. Teacher Forcing runs for 40,000 steps at 10^{-4} with a per-GPU batch size of two across 32 workers (global batch size 64). CTD uses 16 workers with one sample per worker and a generator-to-fake-score update ratio of $1 : 5$. All stages use one gradient-accumulation step.

C.2 Flow-Matching Sampling

The training objectives and noise interpolation are defined in Section 3.2. Using the same noise-level convention, the flow-matching sampler integrates the learned velocity from noise toward the clean

Table 7: Game-training configuration.

Setting	Value
Initialization	Wan2.2-5B
Sequence	10 seconds, 24 FPS, 240 RGB frames
Resolution	480×480 per view
Optimizer	AdamW
Hardware	32 NVIDIA H200 GPUs (bidirectional training and TF)
Bidirectional training	40,000 steps
Teacher Forcing	40,000 steps

endpoint:

$$\mathrm{d}x^t = v_\theta(x^t, t)\,\mathrm{d}t.$$

We use Euler updates (Euler, 1845) at decreasing noise levels. Conditions are omitted here for brevity; each forward pass uses the conditions specified in Section 3.1.

C.3 Dynamic World State Model Training

Causal Geometry Distillation. We initialize the student from pretrained VGGT-Ω (Wang et al., 2026). Block-causal attention in the aggregator and camera head allows interaction within each stereo pair but excludes future observations. The student processes pairs sequentially using its own recurrent state.

A bidirectional teacher supplies geometry targets from overlapping windows aligned by Sim(3). Supervision combines depth, camera, world-point, stereo pose, reprojection, and confidence losses. Training uses an eight-observation KV cache and truncated backpropagation every three observations. After a 900-second memory-only warm-up, we jointly train the memory, aggregator, camera head, and dense head, keeping the DINO encoder frozen. Learning rates are 2×10^{-5} for memory and 2×10^{-7} for pretrained components.

State Representation and Conditioning. Sixteen learned queries read a fixed-size geometry memory and inject the resulting history features into the two views' VGGT-Ω register embeddings. The output camera/register features are mapped through a frame-wise FiLM module to shift and scale features in the video DiT. The memory is read before processing the current stereo pair and updated afterward. Generated views subsequently update the geometry state for the next generation step.

C.4 KV Cache Management

The deployment configuration uses a persistent sink entry and a rolling history cache. The initial observation is retained as the sink while older non-sink entries are evicted as generation proceeds.

D Additional Ablation Studies

We report additional comparisons of action conditioning and the Dynamic World State Model.

Main Architectural Components. Table 8 compares the Standard baseline, the addition of Agent Identity Embedding (AIE), and the full configuration with conditioning from the Dynamic World State Model. Adding AIE increases action accuracy from 88.4% to 89.7%. The full model obtains 91.2% action accuracy and reduces FVD to 159.

Table 8: **Architectural components.** Standard denotes the image-action-to-video baseline. + AIE adds agent identity encoding; Both also includes conditioning from the Dynamic World State Model.

Config	FVD ↓	LPIPS ↓	SSIM ↑	PSNR ↑	Action ↑	RPE ↓
Standard	245	0.36	0.50	17.48	88.4	0.75
+ AIE	228	0.36	0.51	17.56	89.7	0.76
Both	**159**	**0.33**	**0.52**	**17.58**	**91.2**	**0.59**

Table 9: **Adaptive Action Weighting.** Comparison with and without AAW in the action-conditioning branch.

Config	FVD ↓	PSNR ↑	Action ↑
w/o AAW	245	17.48	88.4
w/ AAW	**236**	**17.52**	**88.6**

Adaptive Action Weighting. Table 9 compares the original action-conditioning variants. With AAW, action accuracy increases from 88.4% to 88.6%, while FVD decreases from 245 to 236.

E Interactive Application

The interactive application separates the web interface from model inference. Each client collects player inputs over a $1/24$-second interval and sends the resulting action packet to the server. The backend maintains the action queue and requests new observations from the model. Returned frames update the corresponding player views.

The interface exposes model-loading status, queued actions, GPU utilization, initial-scene selection, and controls to start or stop generation. The inference service maintains generation context across requests; per-view DiT execution and decoding and pipelined delivery are described in Section 3.3. Figure 6 illustrates the four-GPU streaming implementation. Two DiT workers independently generate the left and right views, while two dedicated VAE workers decode the corresponding latents. Decoding and frontend delivery overlap with generation of the next chunk.

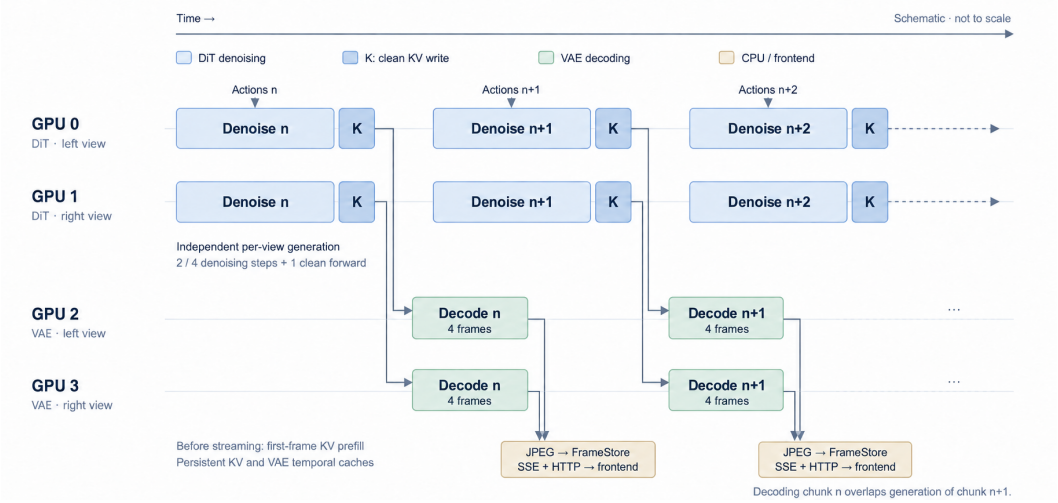

Figure 6: **Four-GPU streaming pipeline.** GPUs 0 and 1 independently run the DiT for the left and right views. Their latents are routed to GPUs 2 and 3, respectively, for VAE decoding. Each denoising block is followed by a clean forward pass that writes the KV cache (K). Latents pass through a CPU queue to GPUs 2–3 for per-view VAE decoding, followed by JPEG encoding and frontend delivery. Each depicted decoding chunk produces four frames per view. Initial-frame KV prefill precedes streaming, and KV and VAE temporal caches persist across chunks. The timeline is schematic and not to scale; it illustrates overlap, not measured stage durations.

F Supplementary Visualizations

Comparison Across Training Stages. Figure 7 compares ground-truth frames with outputs from the bidirectional, Teacher Forcing (TF), and Causal Teacher Distillation (CTD) models at selected times in two gameplay examples. Figure 8 provides two additional examples.

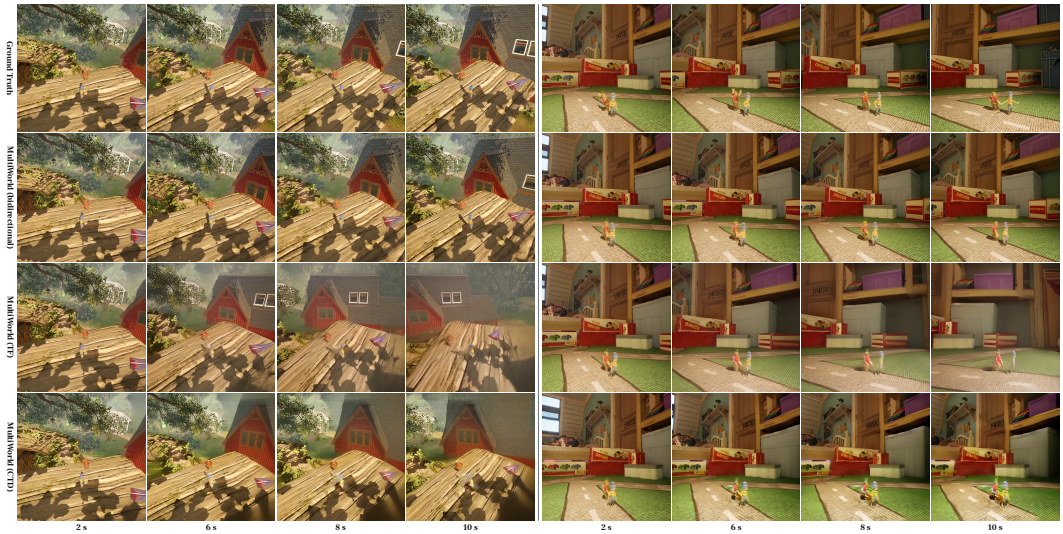

Figure 7: **Qualitative comparison across training stages.** The left and right panels show two gameplay examples at 2, 6, 8, and 10 seconds. From top to bottom, rows show ground truth, the bidirectional model, the model trained with Teacher Forcing (TF), and the model trained with Causal Teacher Distillation (CTD).

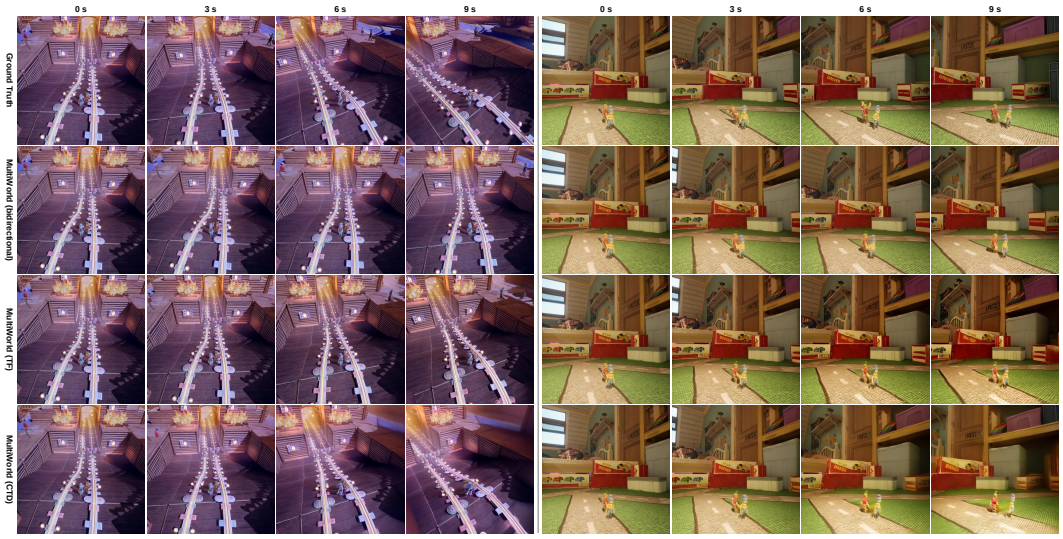

Figure 8: **Additional comparison across training stages.** The left and right panels show cases 948 and 2637, respectively, at 0, 3, 6, and 9 seconds. Rows show ground truth, the bidirectional model (40k), the Teacher Forcing model (72k, 50 NFE), and the Causal Teacher Distillation model (experiment 19, generator step 150). Each column compares the same timestamp across models.

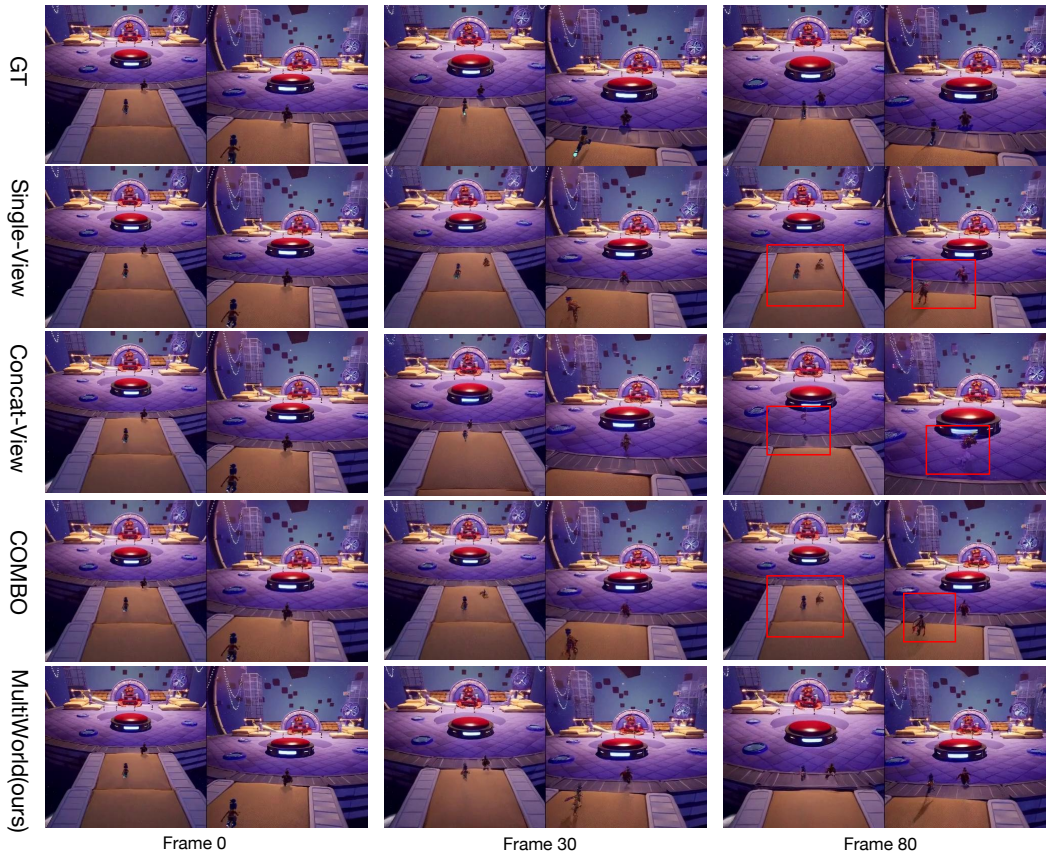

Figure 9: **Qualitative comparison in a multiplayer game.** Rows show the reference video (GT), Standard (labeled Single-View), Concat-View, COMBO (Zhang et al., 2024), and MultiWorld. Columns show frames 0, 30, and 80. Red boxes mark examples of incorrect actions, missing agents, or inconsistent scene states across views.

These examples complement the main-text qualitative comparisons with static-action inputs, physical effects across views, multi-agent interactions, and a failure case.

Action Controllability. Figure 10 shows a static-action sequence in which MultiWorld avoids unintended agent motion.

Physical Consistency. Figure 11 shows shadows across opposite views and snow footprints that remain visible from both camera perspectives.

Multi-Agent Interactions. Figure 12 illustrates coordinated actions on a shared object across synchronized views.

Failure Case Analysis. The small-agent artifacts in Fig. 13 illustrate a limitation of visual fidelity for distant objects.

F.1 Earlier Multi-Robot Rollout

Figure 15 shows an earlier 162-frame rollout in which three robots stack colored cubes. This sequence is shorter than the current 240-frame training length and therefore does not demonstrate extrapolation beyond the updated training horizon.

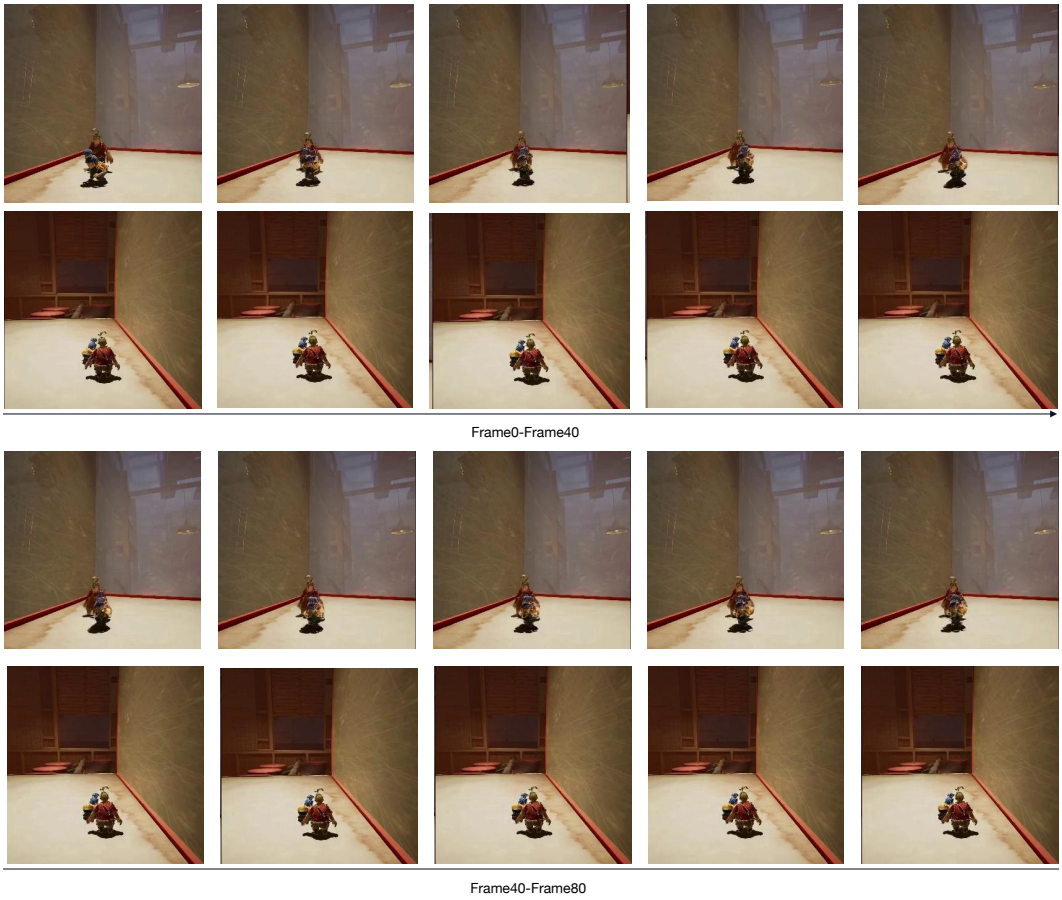

Figure 10: **Static-action example.** The sequence shows generated views under static action inputs.

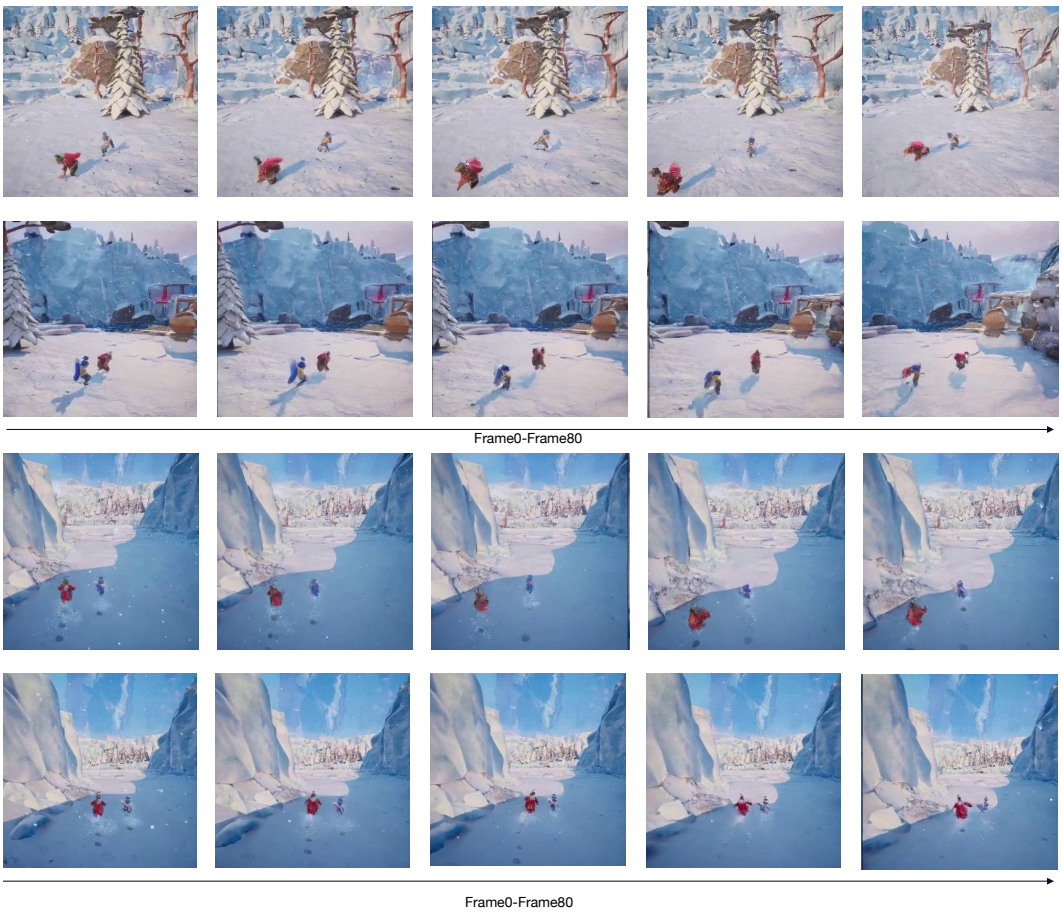

Figure 11: **Cross-view physical effects.** The upper example shows shadows from opposite viewpoints; the lower example shows footprints left by two agents in snow.

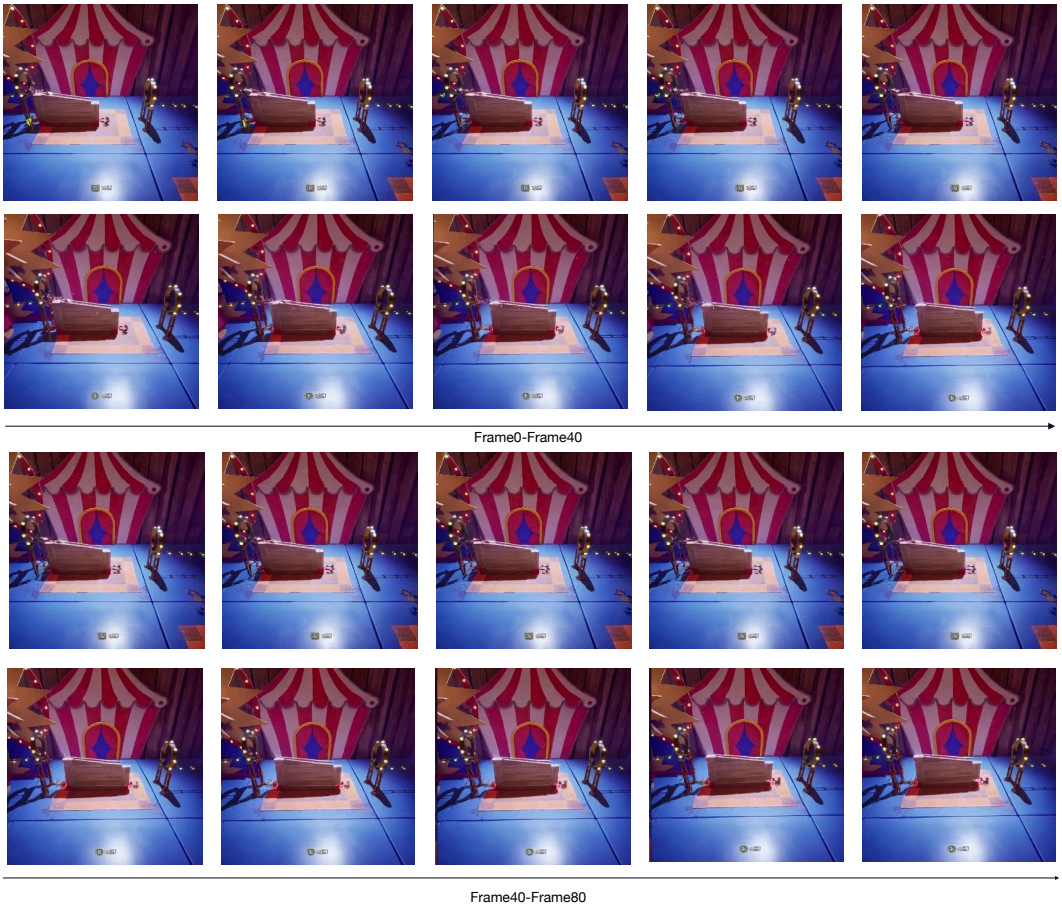

Figure 12: **Two-agent interaction example.** One agent pushes and the other pulls a board from left to right; the views show the same cooperative sequence.

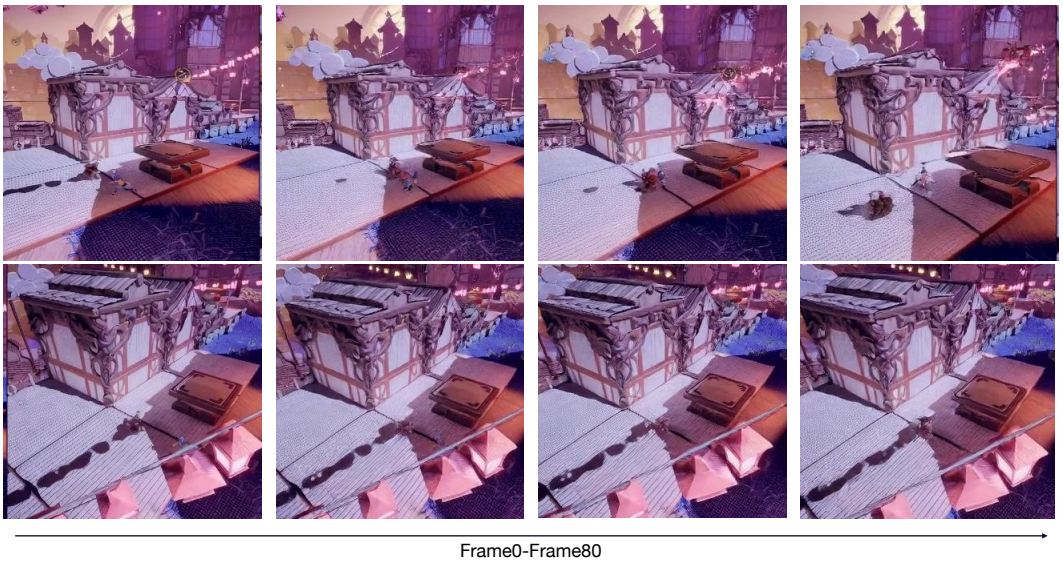

Figure 13: **Small-agent failure case.** Agents occupying small image regions have ambiguous shapes.

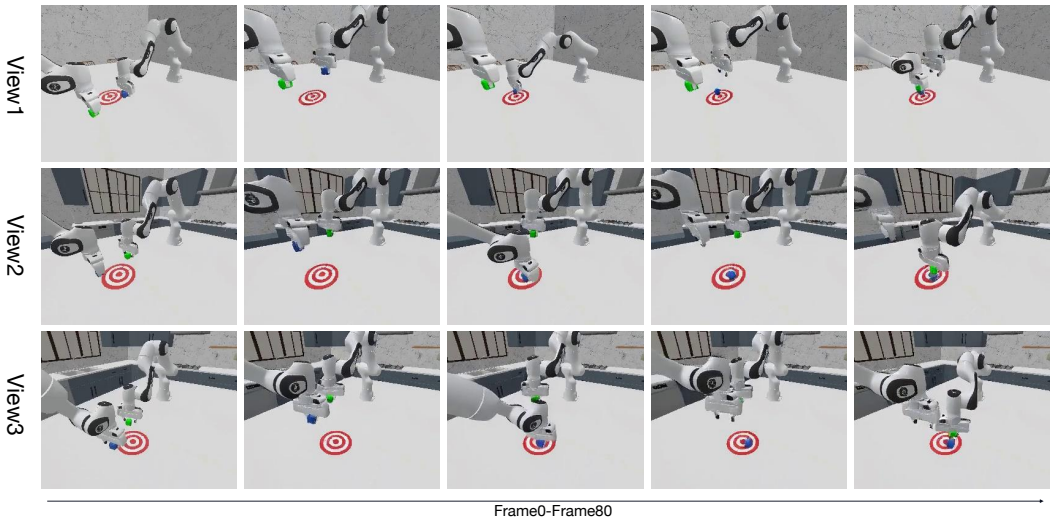

Figure 14: **Successful coordination example.** The sequence shows robots completing a cooperative manipulation task.

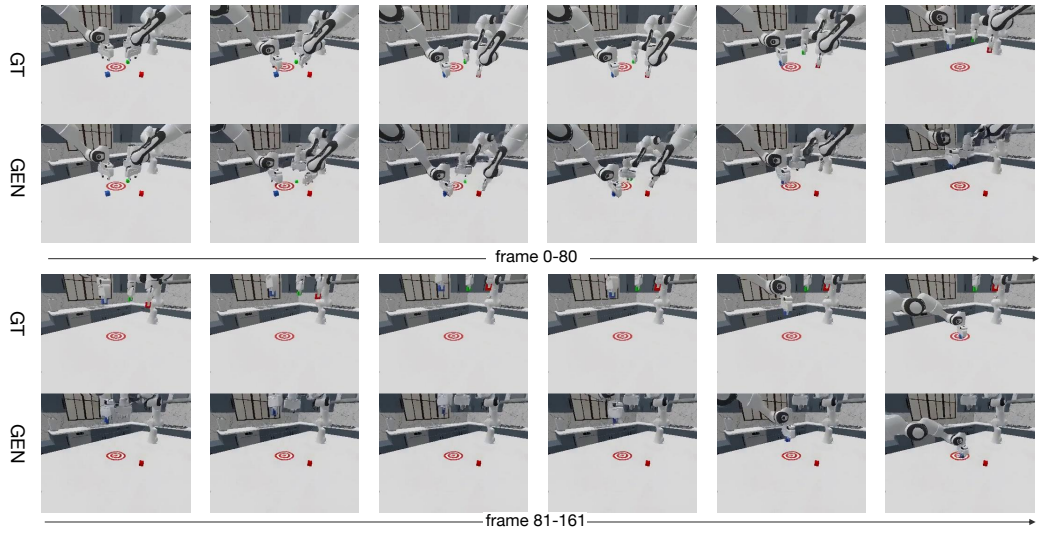

Figure 15: **Earlier multi-robot rollout.** Rows compare reference (GT) and generated (GEN) sequences. The upper and lower groups show frames 0–80 and 81–161, respectively.