MM-WORLD: A UNIFIED VIDEO WORLD MODEL FOR EGOCENTRIC NAVIGATION AND MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Action-conditioned video world models enable users to explore and interact with virtual worlds through their own actions. However, existing models typically focus on either navigation or local manipulation, leaving their combination poorly supported, even though most real-world tasks require moving and manipulating in tandem. To address this, we present MM-World, a unified video world model for egocentric navigation and manipulation controlled by head and hand motion. Our key insight is to express these heterogeneous action-conditioning signals in a shared image space. We reproject scene geometry reconstructed from the initial image along the head trajectory and overlay hands mesh rendered in their prescribed poses. The resulting visual condition reuses the pretrained model's native visual-conditioning pathway without action-specific input modules, while providing a shared interface for joint training on navigation and manipulation videos. We further adopt an autoregressive generation scheme to support long-horizon interactions. Moreover, to evaluate these capabilities comprehensively, we curate a unified dataset and establish a four-track benchmark with 846 test cases spanning navigation, manipulation, mobile manipulation, and long-horizon generation. Extensive experiments show that MM-World outperforms previous methods by a large margin across visual quality and action-following.

1 INTRODUCTION

Action-conditioned video world models aim to simulate how the visual world evolves in response to a user's actions, predicting what the user will see next given an observation and a sequence of actions (Bruce et al., 2024; Parker-Holder & Fruchter, 2025; Robbyant Team et al., 2026). In the egocentric setting, two forms of action are fundamental, head motion determining how the user moves through and looks at the scene while hand motion determines how the user interacts with objects. These two are rarely separable in practice, since everyday embodied activities routinely require moving to a place and acting on it, for example, walking up to a table and picking up a book (Figure 1). A world model that only conditions on one of the two actions therefore cannot faithfully simulate such mobile manipulation behavior, even though it pervades real egocentric activity. A world model for egocentric interaction should thus respond jointly to head and hand motion. Such a model could enable immersive interaction beyond the physical world and provide scalable egocentric interaction data for downstream learning.

Existing action-conditioned video world models, however, satisfy only one side of this need. Navigation world models let users steer the viewpoint and synthesize large, explorable scenes (Bruce et al., 2024; Robbyant Team et al., 2026) and some further let the scene itself change, but only through explicitly specified changes, such as text prompts that trigger scripted scene events (DreamX Team et al., 2026), or trajectories that prescribe object motion (Gu et al., 2026), not as a consequence of the user's physical interaction with the scene. Manipulation world models take the complementary view, modeling how hands or grippers act on nearby objects (Zhu et al., 2025; Liao et al., 2025), typically under a fixed or slowly moving camera, and even recent models conditioning on camera or head motion (Xie et al., 2026; Wang et al., 2026c) still target limited viewpoint changes rather than navigation-scale movement (Sec. 5.3). Thus, neither line of work was built for mobile manipulation and manipulation models in particular struggle to follow the prescribed camera trajectory under navigation-scale motion, which motivates our work.

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.

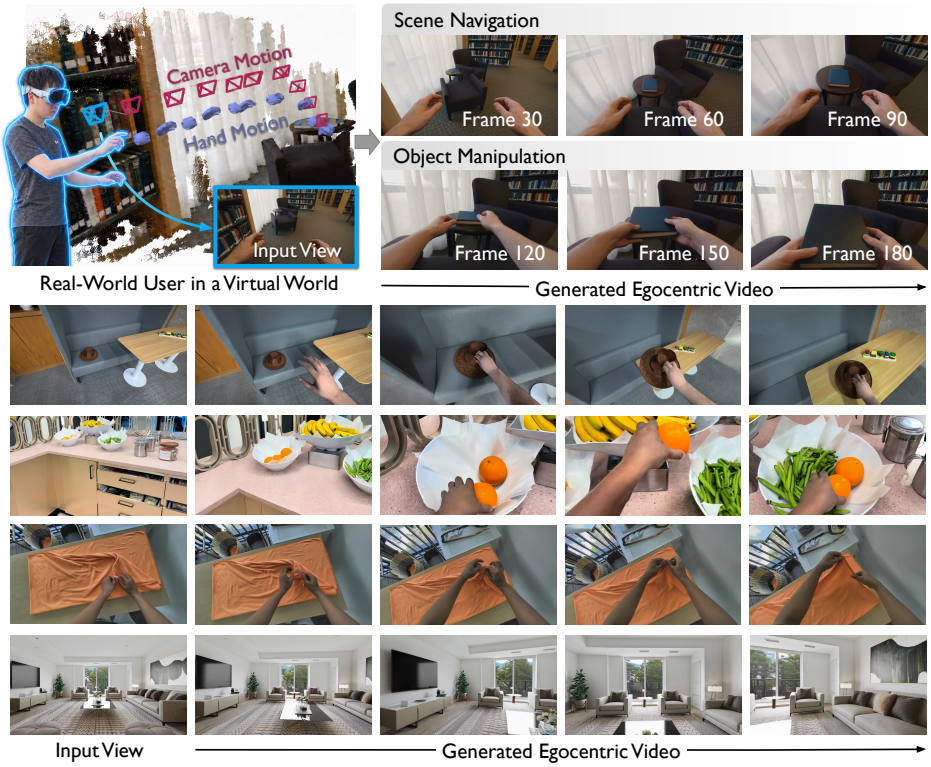

Figure 1: MM-World turns a real user's head and hand motion into egocentric video of a scene given by a single input view. Head and hand motion recorded from the user drives the video generation.

Combining the two action conditions requires more than adding a hand-control branch to a navigation model. The two actions differ in both representation and supervision, as head motion is rigid while hand motion is articulated, and their data cover different regimes, navigation videos rarely showing hands and manipulation clips rarely showing camera motion. Our key observation is that, despite these differences, both actions condition the model through geometric effects in image space, head motion reprojecting the observed scene and hand motion projecting hand geometry. Overlaying rendered hands on the reprojected scene thus yields a single, image-based visual condition, letting us reuse the pretrained visual-conditioning pathway without action-specific input modules. The renders with geometric relation further guide the model toward newly revealed content and hand-object responses. More importantly, it lets every clip share the same training interface, each supplying the action renders available to it while all clips train the same conditioning pathway.

We build MM-World on this observation (Figure 2). rendering scene geometry reconstructed from the initial image together with MANO hand meshes (Romero et al., 2017), that follow the prescribed head and hand motions. Since these renders cover the scene only partially, we complement them with camera-ray conditioning (Yin et al., 2026) for viewpoint cues in uncovered regions. Together with the initial image and a static scene description, these conditions drive a video diffusion model to generate egocentric interactions, jointly trained on navigation and manipulation videos and extended to autoregressive generation for longer interaction sequences.

We curate a unified dataset from four sources (Ling et al., 2024; Hoque et al., 2026; Punamiya et al., 2026) with a shared annotation pipeline for camera trajectories, hand configurations, and static scene descriptions. We additionally establish a benchmark for evaluation spanning four tracks, namely navigation, manipulation at three levels of difficulty, mobile manipulation, and long-horizon generation. Each track measures frame quality together with how faithfully the video follows the camera and the hands. Although no training clip was curated for mobile manipulation and head motion accounts for a small portion, MM-World follows both the head and hands around tracks. It outperforms all the other world models in frame quality, camera accuracy, and, where applicable, hand accuracy. Our contributions are as follows:

- We present MM-World, a unified video world model for egocentric navigation and manipulation, supporting camera and hand motion control within a single framework.

- We develop a unified visual conditioning scheme that represents navigation and manipulation controls in a common space for action-conditioned video generation.
- We curate a comprehensive dataset and establish a novel benchmark spanning navigation and manipulation, demonstrating consistent improvements in visual quality and action accuracy over existing baselines.

2 Related Work

Navigation and camera control. Navigation world models generate views controlled by latent actions (Bruce et al., 2024) or keyboard, mouse, and camera poses (He et al., 2025; Sun et al., 2025; Robbyant Team et al., 2026; DreamX Team et al., 2026; Zhu et al., 2026; Huang et al., 2026; Qian et al., 2026). Real-time rollouts often distill bidirectional generators into causal ones (Huang et al., 2025; Yang et al., 2025). Camera-controlled generation uses injected camera parameters (He et al., 2024; Yin et al., 2026), motion-reference videos (Luo et al., 2025), or target-view geometry renders (Yu et al., 2024; Ren et al., 2025; Chen et al., 2026c), sometimes with 3D layouts (Wang et al., 2025). Renders provide pixel-aligned guidance only for observed geometry. Other changes are specified by text (Parker-Holder & Fruchter, 2025; Gao et al., 2026b; DreamX Team et al., 2026) or object trajectories (Gu et al., 2026). MM-World combines first-view reprojection with camera rays for every token, while hands drive world changes.

Manipulation and embodiment control. Manipulation world models predict hand- or gripper-induced changes, often from fixed cameras. They use robot trajectories (Zhu et al., 2025; Liao et al., 2025), hand–object interactions (Akkerman et al., 2025; Gao et al., 2026a), or rendered skeletons (Wu & Gao, 2026; Zhao et al., 2026), robot geometry (Kim et al., 2026a), and depth, segmentation, or keypoints (Gao et al., 2026a). Egocentric models also use head motion, conditioning on whole-body motion (Tu et al., 2025), head and hand poses (Xie et al., 2026), projected hand meshes and camera rays (Wang et al., 2026c), or hand-surface Plücker maps (Chen et al., 2026b). EgoSim (Hao et al., 2026) renders a first-frame point cloud with hand keypoints and updates the cloud after interactions; DWM (Kim et al., 2026b) renders a known 3D scene and hand meshes. MM-World instead starts from one image and synthesizes unseen content. It also trains on navigation videos: omitting hands from the shared visual condition lets one model handle navigation, manipulation, and both together. Earlier unified simulators connect datasets through language and low-level actions (Yang et al., 2024).

3 Method

This section details the architecture and training strategy of our MM-World, a unified video world model designed for egocentric navigation and manipulation (Fig. 2). We begin by formalizing the prediction problem in Sec. 3.1. Next, we introduce the central component of MM-World, the unified visual condition in Sec. 3.2, which renders head and hand actions into the image space of the target frames. We then describe our condition injection method in Sec. 3.3, and illustrate the training strategy in Sec. 3.4. Finally, Sec. 3.5 extends the model to autoregressive long-video generation.

3.1 Problem Formulation

In this work, we consider an egocentric observer navigating a scene and performing bimanual interactions. The observer's actions determine the head viewpoint and hand configurations, while a world model predicts their visual consequences: scene appearance from new viewpoints and object responses to manipulation.

Formally, an episode is initialized with a reference image I_0 and a text description s of the scene. At each time step $t = 1, \ldots, T$, the action is defined as $a_t = (\mathbf{C}_t, \mathcal{H}_t)$, where $\mathbf{C}_t \in \mathrm{SE}(3)$ denotes the camera-to-world pose of the head camera with associated intrinsics $\mathbf{K}_t$. All poses are expressed relative to the initial camera frame, such that $\mathbf{C}_0$ is the identity. The term $\mathcal{H}_t$ represents a (possibly empty) set of hand configurations, each specifying the MANO articulation and shape (Romero et al., 2017) of a hand, along with its camera-relative position. Given this formulation, we model: $p_\theta(V_{0:T} \mid I_0, a_{1:T}, s)$, where $V_{0:T}$ is the generated video sequence starting from the initial observation $V_0 = I_0$. The action sequence $a_{1:T}$ encapsulates two regimes of this problem: pure navigation

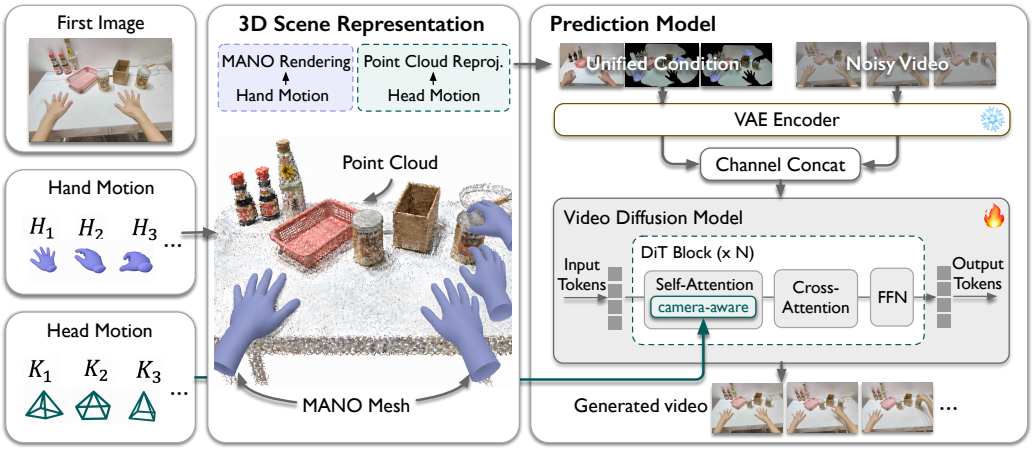

Figure 2: **Pipeline of MM-World.** Given the first image together with future hand and head motions (left), MM-World reconstructs the scene and represents the hands as MANO meshes (middle). These representations are rendered into a unified visual condition, which, together with camera-ray cues, conditions a video diffusion model to generate the egocentric interaction video (right).

and egocentric manipulation. In navigation, the action consists solely of viewpoint changes (i.e., the hand state $\mathcal{H}_t = \emptyset$ for all t). In egocentric manipulation, the action contains both head and hand motions. The text s describes only the static scene and its objects, excluding actions and events (Sec. 4.1). Thus, dynamic changes must be inferred from $a_{1:T}$ rather than supplied through the text.

3.2 Unified Visual Condition

Head motion and hand articulation exhibit distinct parameterizations: head movement follows a rigid camera transformation, whereas hands are defined via an articulated mesh. Splitting these actions into dedicated modules would fragment the control interface, requiring the generator to learn independent mappings to pixel space. However, both modalities ultimately exert a direct geometric influence on the egocentric perspective. We render these effects directly to obtain a unified, pixel-aligned condition for navigation and manipulation.

Back-projecting the reference image. To render head motion, we back-project the reference image I_0 into a colored point cloud $\mathcal{P}_0$ in the coordinate frame of the first camera: $\mathcal{P}_0 = \{(D_0(\mathbf{u})\mathbf{K}_0^{-1}\tilde{\mathbf{u}},\ I_0(\mathbf{u})) \mid \mathbf{u} \in \Omega_0\}$, where D_0 is the depth map of I_0, $\mathbf{K}_0$ is the camera intrinsics, $\tilde{\mathbf{u}}$ is the homogeneous pixel coordinates, and Ω_0 is the set of valid pixels. To prevent artifacts such as hands freezing into the static scene, Ω_0 excludes depth discontinuities and pixels covered by the estimated hand regions in I_0. From the remaining pixels, we keep the fraction ρ with the highest predicted confidence, discarding the least reliable depth estimates. We set ρ to 75% for navigation and 50% for manipulation and use the same value in training and at test time; Sec. 5.4 examines lower, zero, and randomly values.

While Ω_0 determines which pixels are lifted, the depth D_0 comes from different sources in training and at test time. During training, D_0 and the camera trajectory come from the same Pi3X reconstruction of the full clip (Sec. 4.1), so the lifted geometry and the trajectory share one scale. At test time, D_0 is predicted from I_0 alone with Pi3X (Wang et al., 2026b). The approximately metric scale of this single-view prediction is not calibrated to the given trajectory, so the lifted geometry matches the trajectory only approximately; Sec. 5.4 measures the effect of this gap. In both cases, $\mathcal{P}_0$ contains only the surfaces visible in I_0: the full-clip reconstruction changes how their depth is estimated but adds no surfaces from later frames.

Rendering scene and hand motion. Let $\Pi_{\mathbf{K}}(\cdot)$ denote a rendering operator that takes 3D geometry defined in camera coordinates with intrinsics $\mathbf{K}$, returning an RGB image along with its binary mask. For each frame $t = 1, \ldots, T$, we render the scene and the hands as follows:

$$\big(R_t^{\mathrm{s}}, M_t^{\mathrm{s}}\big) = \Pi_{\mathbf{K}_t}\big(\mathbf{C}_t^{-1}\mathcal{P}_0\big), \qquad \big(R_t^{\mathrm{h}}, M_t^{\mathrm{h}}\big) = \Pi_{\mathbf{K}^{\mathrm{h}}}\big(\mathrm{MANO}(\mathcal{H}_t)\big), \tag{1}$$

where $\mathbf{C}_t^{-1}\mathcal{P}_0$ transforms the scene point cloud to the t-th camera frame, exclusively simulating head motion. Conversely, the hand meshes $\mathrm{MANO}(\mathcal{H}_t)$ reside in the local camera space and thus

naturally move with the head. Rendering the hands as uniformly colored, shaded meshes provides explicit geometric conditioning for pose and shape while leaving texture synthesis to the generator.

We render hands with the monocular estimator's fixed intrinsics $\mathbf{K}^{\mathrm{h}}$ (Sec. 4.1) rather than the scene intrinsics $\mathbf{K}_t$. This preserves the estimator's 2D alignment, which is more reliable than its metric depth. The render therefore specifies hand shape and position in image space, with apparent size providing only a coarse distance cue.

Compositing the unified condition. We composite the rendered hands over the scene and compute the union of their masks:

$$R_t = (1 - M_t^{\mathrm{h}}) \odot R_t^{\mathrm{s}} + M_t^{\mathrm{h}} \odot R_t^{\mathrm{h}}, \qquad M_t = M_t^{\mathrm{s}} \vee M_t^{\mathrm{h}}, \qquad t = 1, \ldots, T, \tag{2}$$

where $\odot$ denotes element-wise multiplication and $\vee$ is the pixel-wise logical OR. We set $R_0 = I_0$ and $M_0 = \mathbf{1}$ to retain the complete reference image. Because the hand and scene renders lack metric alignment, we overlay the hands without depth testing and leave realistic hand–scene occlusions to the generator. The resulting $(R_{0:T}, M_{0:T})$ is our unified visual condition. Pure navigation uses the same format with $M_t^{\mathrm{h}} = \mathbf{0}$ and $R_t = R_t^{\mathrm{s}}$.

Resolving unobserved and dynamic regions. As a static projection of a single view, $\mathcal{P}_0$ cannot depict newly revealed regions or object responses. The support mask distinguishes rendered pixels from holes, while the generator uses the render as a structural reference to synthesize missing content and responses to hand interaction.

3.3 Conditioning a Pretrained Video Generator

We build our generator upon the 14B-parameter Wan2.1 Image-to-Video (I2V) diffusion transformer (Wan et al., 2025), which operates in the latent space of a frozen video VAE (encoder $\mathcal{E}$). We reuse its existing input pathway, which concatenates the noisy latent, encoded visual condition, and availability mask. At diffusion timestep τ, the transformer input is: $X_\tau = \mathrm{Concat}_{\mathrm{ch}}\big[z_\tau,\ \mathcal{F}(M_{0:T}),\ \mathcal{E}(R_{0:T})\big]$, where z_τ is the noisy video latent, and $\mathcal{F}$ resamples the binary masks to match the spatio-temporal resolution of the latent (detailed in Appendix A.1). Unlike standard first-frame conditioning, the renders provide structural cues across all frames, with the mask indicating validity spatially rather than per frame. The input format remains unchanged, requiring no new input layer and allowing the original patch embedding to remain frozen.

Camera rays. The scene render provides no geometric cues in newly revealed regions, so we condition each latent token on its viewing ray. For each token i, we compute its ray origin $\mathbf{o}_i$ and unit direction $\mathbf{d}_i$ in the first camera's frame. We inject its Plücker coordinates $\mathbf{r}_i = (\mathbf{d}_i, \mathbf{o}_i \times \mathbf{d}_i)$ into the self-attention mechanism via SCoPE (Yin et al., 2026), which adds learned ray projections to the queries and keys of every transformer block (Appendix A.1). We zero-initialize the ray projections and derive the rays from the same poses used for scene rendering. SCoPE gates the ray encoding using the log-norm of the ray moment $\mathbf{o}_i \times \mathbf{d}_i$. Because trajectory scale is only approximately metric, we randomly offset this gate during training to reduce sensitivity to scale (Appendix A.1).

Reference image and text. Following the base model, frozen encoders provide I_0 and s as cross-attention conditions. The image supplies scene appearance, while the text supplies scene and object semantics that help synthesize content where the render lacks support.

3.4 Joint Training and Inference

Joint training. We train a unified model on a mixture of navigation and manipulation clips. Because pure navigation clips simply omit the hand renders, both tasks seamlessly share an identical input format and all trainable parameters. Every clip is converted into a standard set of conditions c: the unified visual condition, the camera rays, the reference image, and the text description. We optimize the flow-matching objective (Lipman et al., 2023) of the pretrained backbone. Let $z = \mathcal{E}(V_{0:T})$ be the target latent, $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ the sampled Gaussian noise, and $\tau \in [0, 1]$ the diffusion timestep. The training objective is defined as:

$$z_\tau = (1 - \tau)\, z + \tau\, \epsilon, \qquad \mathcal{L} = \mathbb{E}_{z,c,\tau,\epsilon}\Big[w(\tau)\ \mathrm{MSE}\big(v_\theta(z_\tau, \tau; c),\ \epsilon - z\big)\Big], \tag{3}$$

where v_θ is the predicted velocity and $w(\tau)$ is the scheduler's timestep weight. To preserve the backbone's powerful generative prior, we freeze its original weights, the VAE, and the image and text

encoders. Training is strictly limited to the SCoPE modules and newly injected LoRA adapters (Hu et al., 2022) within the attention and feed-forward blocks.

Inference. Given I_0, $a_{1:T}$, and s, we predict the depth of I_0 from the image alone and back-project it into a 3D scene using the same ρ as in training. We then render the unified visual condition and compute the camera rays. These conditions are built once and stay fixed across denoising steps. Starting from Gaussian noise at $\tau = 1$, we integrate the predicted velocity to $\tau = 0$ with Euler steps and decode the result with the VAE decoder. All $T+1$ frames are denoised jointly with bidirectional attention, so MM-World generates a fixed-length clip rather than rolling it out frame by frame.

3.5 Autoregressive Extension

For long, interactive rollouts where actions arrive dynamically, we introduce an autoregressive variant that generates video in sequential chunks. This extension employs sliding-window block-causal attention and a KV cache of past chunks. To maintain strict causality, all conditions are temporally aligned: each chunk receives only its corresponding slices of the unified visual condition and camera rays, and depends exclusively on actions up to its final frame. Following LongLive (Yang et al., 2025; Chen et al., 2026a), we train this variant in two stages: autoregressive pretraining via teacher forcing, followed by distribution matching distillation (DMD) (Yin et al., 2024b;a) on the model's own streaming rollouts to produce an efficient few-step generator. Details are provided in App. B.

4 Data and Benchmark

4.1 Data Construction

We build our data from four egocentric video sources. For navigation, we use DL3DV-10K (Ling et al., 2024), whose videos feature substantial camera motion through static scenes. For manipulation, we use three sources that differ in capture conditions. EgoDex (Hoque et al., 2026) and EgoVerse (Punamiya et al., 2026) are public egocentric manipulation datasets, and KAI-Ego is our in-house dataset, which offers high-quality indoor recordings of everyday hand–object interactions across diverse settings. Appendix C.1 reports the scale and composition of each source.

We annotate every clip with camera trajectories, hand meshes, and a scene description, using only its RGB frames and the same off-the-shelf models for every source. We feed all frames of a clip to Pi3X (Wang et al., 2026b) to obtain per-frame camera poses and intrinsics, together with the first-frame points used to render the scene condition. At test time, these points are instead predicted from the first frame alone, so that the model gains no unfair advantage from future frames (Sec. 3.2). We then detect the hands in every frame and recover their MANO parameters with WiLoR (Potamias et al., 2025), rendering the resulting meshes in the estimator's fixed camera $\mathbf{K}^{\mathrm{h}}$. Finally, Qwen3.7-Plus writes the text s, which describes only the static scene and its objects, without actions or state changes, so that it does not reveal the interaction (Sec. 3.1). Appendix C details each step.

4.2 Benchmark Design

The benchmark comprises three short-horizon tracks with 81-frame sequences, and a long-horizon track for autoregressive generation. We manually select clips and sequences for all four tracks. **(1) Navigation.** This track evaluates camera control and scene consistency without hand interaction, using clips in the evaluation split of DL3DV-10K. **(2) Manipulation.** This track evaluates object interaction, organized into three tiers by capture conditions and environmental diversity: *in-lab* EgoDex clips covering tabletop tasks; *indoor* KAI-Ego clips across diverse indoor settings; and *in-the-wild* EgoVerse clips. **(3) Mobile manipulation.** This track evaluates scene geometry and interaction dynamics under simultaneous camera motion and hand-object interaction. We select KAI-Ego clips from service, office, and retail settings, covering occlusion, contact, object displacement, and state changes. **(4) Long-horizon generation.** This track evaluates whether appearance, geometry, and control alignment remain stable beyond the model's local context. For metrics to evaluate the predictions, we group them into five categories based on the aspect evaluated: frame fidelity, video quality, camera accuracy, motion consistency, and hand accuracy. Further details on all four tracks and metric suite are provided in Appendix D.

Table 1: Navigation track results. Best results are in **bold** and second-best results are underlined.

	Frame quality			Camera accuracy				Motion / video	
Method	PSNR↑	SSIM↑	LPIPS↓	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
HY-World1.5	11.1412	0.2397	0.6148	87.7593	0.6036	1.3670	0.0888	14.6512	398.4690
LingBot-World	11.3661	0.2725	0.6080	40.0449	0.3481	0.8632	0.0651	11.9340	294.2601
SCoPE	11.5271	0.2549	0.5838	36.7438	0.3800	1.0626	0.0557	13.9007	271.1343
DreamX-World	11.7813	0.2680	0.5800	31.0412	0.3374	1.0594	0.0627	12.1287	252.2664
SANA-WM	11.3250	0.2384	0.5830	19.5000	0.2334	0.8336	0.0478	10.6041	366.1390
SolarWM	12.7223	0.3081	0.5123	15.2753	0.2204	0.8657	0.0433	10.7819	229.2066
LingBot-World v2	10.8589	0.2383	0.5938	13.2135	0.1633	0.7628	0.0428	11.8168	299.3180
Matrix-Game-3.5	13.2348	0.3632	0.5254	7.9319	0.1483	0.3935	0.0358	11.4787	242.2346
Ours	**14.5036**	**0.4240**	**0.4232**	**5.9907**	**0.0953**	**0.3157**	**0.0239**	**5.5838**	**207.8224**

5 EXPERIMENTS

We evaluate MM-World on the benchmark of Sec. 4.2 to answer three questions. First, can a single model follow each control on its own, namely head motion in navigation and hand motion in manipulation (Sec. 5.2)? Second, can it combine the two controls without training on clips curated for mobile manipulation (Sec. 5.3)? Third, how much of this ability comes from the scene render, and how much coverage and depth accuracy does the render require (Sec. 5.4)?

5.1 EXPERIMENTAL SETUP

MM-World. All results in Secs. 5.2 and 5.3 come from a single MM-World model, trained jointly on the four data sources of Sec. 4.1 for 30K steps on 64 A100 GPUs with a global batch size of 64. The mobile manipulation track tests whether the model composes both controls at test time because no training clip was curated for this task. Appendix A.1 details the remaining training hyperparameters and inference settings. The autoregressive variant (Sec. 3.5) is evaluated only on the long-horizon track. We describe its training and inference in App. B and report results in App. E.2.

Baselines. For each track, we select baselines whose publicly released inference interfaces support the required controls. We use their bidirectional checkpoints on the three short-horizon tracks and their causal autoregressive versions on the long-horizon track. Appendix D.2 details the inference configuration of each model and how the evaluation settings are aligned across models.

5.2 NAVIGATION AND MANIPULATION

Navigation. Table 1 compares MM-World with camera-controlled world models on the navigation track. MM-World follows the prescribed camera more closely than every baseline: relative to the strongest one, Matrix-Game-3.5, it lowers the rotation error from 7.93° to 5.99° and the trajectory error by 36%. Its frames are also closer to the ground truth, with a PSNR of 14.50 dB against at most 13.23 dB for the baselines, and its videos have the lowest FVD. The gain in frame quality is expected from how the condition is built: surfaces seen in the first frame reach every target view as pixel-aligned renders, whereas the baselines must regenerate them from the first frame as the camera moves. Figure 3 shows an example.

Manipulation. On the three manipulation tiers (Table 2), MM-World is best on every metric. Its advantage in hand accuracy holds in every tier: it reduces the root-aligned MPJPE of the strongest baseline, Wan2.1-Fun-14B-Control, by 33–43% (10.75 vs. 18.74 on EgoDex, 6.58 vs. 9.87 on KAI-Ego, and 12.22 vs. 21.26 on EgoVerse), and its PSNR is 5–6 dB higher. Its advantage in camera accuracy grows from in-lab to near-in-the-wild clips, whose viewpoints vary more: relative to the most accurate baseline in each tier, the rotation error falls by 43% on EgoDex, 61% on KAI-Ego, and 69% on EgoVerse. In Figure 3, most baselines alter the scene or the manipulated objects, or drift from the target viewpoint, whereas MM-World stays close to the ground truth.

5.3 MOBILE MANIPULATION

Mobile manipulation combines navigation-scale head motion with hand–object interaction, the setting that motivates a unified model. Table 3 compares MM-World with EgoSim and Hand2World,

Table 2: Manipulation results on EgoDex (ED), KAI-Ego (KE), and EgoVerse (EV). Best and second-best results for each dataset and metric are in **bold** and underlined, respectively. Table 8 reports all metrics. RynnWorld denotes RynnWorld-Teleop; Gen. Reality denotes Generated Reality; Wan2.2-5B and Wan2.1-14B denote the corresponding Control baselines.

	PSNR↑			LPIPS↓			MPJPE↓			PA-MPJPE↓			RotErr↓			FVD↓		
Method	ED	KE	EV	ED	KE	EV	ED	KE	EV	ED	KE	EV	ED	KE	EV	ED	KE	EV
EgoSim	20.34	16.40	14.15	0.213	0.325	0.401	31.12	40.02	39.33	9.60	11.02	11.02	86.67	91.39	75.63	192.8	390.3	378.2
RynnWorld	24.08	20.89	15.41	0.209	0.229	0.430	28.95	13.92	32.17	9.21	5.03	9.68	108.70	92.09	109.99	248.6	185.3	299.6
Hand2World	13.00	15.65	11.41	0.640	0.448	0.649	28.07	17.33	50.83	9.22	5.72	10.12	126.10	120.90	107.69	914.3	779.1	1333.6
Wan2.2-5B	22.99	20.50	15.31	0.203	0.252	0.436	22.95	13.11	24.94	7.33	4.35	7.72	102.64	99.37	120.39	149.0	202.5	347.2
Gen. Reality	23.98	21.00	15.88	0.189	0.248	0.420	20.73	12.96	22.81	7.13	4.22	7.41	99.41	94.23	112.11	174.1	224.7	454.2
Wan2.1-14B	24.21	21.08	15.32	0.194	0.218	0.415	18.74	9.87	21.26	6.13	3.49	6.59	98.01	87.66	113.48	162.1	192.5	378.6
Ours	**29.85**	**26.18**	**21.70**	**0.088**	**0.111**	**0.179**	**10.75**	**6.58**	**12.22**	**4.55**	**2.58**	**4.90**	**49.25**	**34.16**	**23.37**	**67.5**	**82.5**	**136.8**

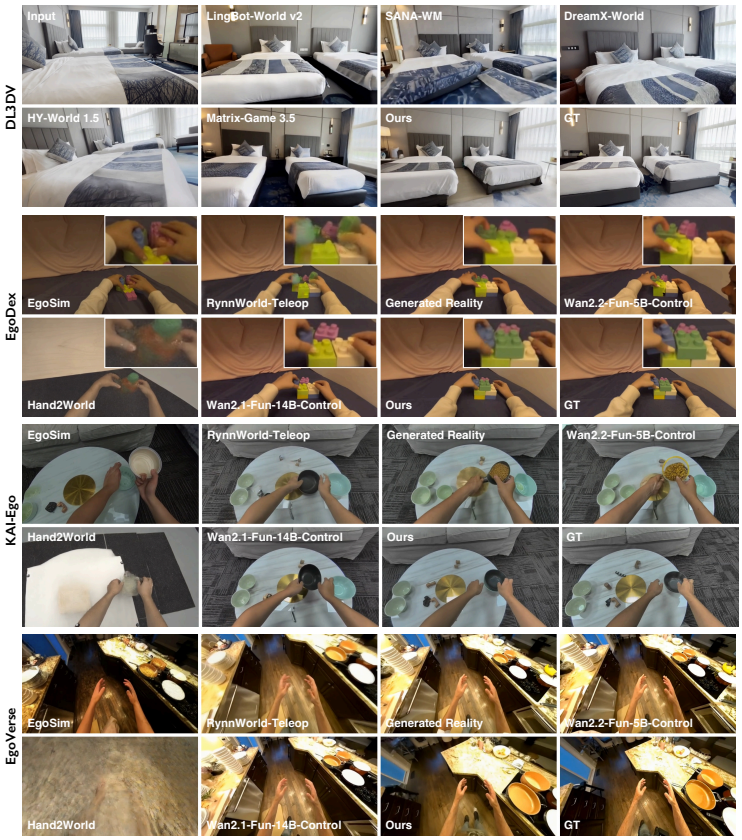

Figure 3: Qualitative comparison on four datasets.

the baselines whose architectures support mobile manipulation. Both drift far from the prescribed head trajectory, with rotation errors of $60.08°$ and $86.44°$ against $18.22°$ for MM-World, and their hand errors are three to seven times larger (root-aligned MPJPE of 30.62 and 68.79 against 10.28). MM-World also yields the best visual quality. No training clip was curated for this setting: navigation clips contain no hands, and head motion in the manipulation clips is mostly small (Appendix C). We attribute the result to the unified visual condition, which expresses both actions in the same image space, so that the model can compose them at test time although it has rarely seen them together.

5.4 Ablation Studies

We ablate the scene render, the part of the unified visual condition that carries head motion (Sec. 3.2), along three axes: whether it is present, how many first-frame points it retains, and how accurately

Table 3: Mobile manipulation track results. Best and second-best results are in **bold** and underlined. Table 9 reports all metrics.

	Frame quality		Hand accuracy		Camera accuracy		Video
Method	PSNR↑	LPIPS↓	MPJPE↓	PA-MPJPE↓	RotErr↓	ATE↓	FVD↓
EgoSim	12.90	0.480	68.79	13.40	60.08	0.078	1255.8
Hand2World	10.96	0.683	30.62	10.05	86.44	0.150	2497.6
Ours	**18.64**	**0.253**	**10.28**	**5.05**	**18.22**	**0.055**	**545.3**

Table 4: Render coverage. Rows give the fraction ρ of first-frame points retained in training and columns the fraction at test time. Appendix E.3 reports all metrics.

	DL3DV						KAI-Ego					
	RotErr↓			PSNR↑			PSNR↑			MPJPE↓		
Training	High	Low	Zero	High	Low	Zero	High	Low	Zero	High	Low	Zero
High (default)	6.37	8.22	99.16	15.24	14.90	11.19	25.13	24.90	21.54	8.14	8.11	8.21
Low	5.59	7.46	93.83	15.37	15.10	11.25	25.12	25.13	21.70	8.31	8.33	8.32
Random	6.63	6.50	54.10	15.25	14.96	11.57	24.97	24.83	21.96	8.35	8.24	8.20
Zero	29.34	25.87	33.16	9.34	9.99	11.96	14.58	16.76	22.39	25.15	15.52	7.97

their depth is estimated. For the first two, we train four models that differ only in the fraction ρ of first-frame points retained during training, *High* (the setting of the main model), *Low*, *Zero* (no scene render), and *Random* (a level drawn per clip), and test each at the High, Low, and Zero levels (Table 4); we write High/Zero for the High model tested at the Zero level, and so on. To limit the cost, these models are trained on DL3DV and KAI-Ego only and for a shorter schedule than the main model. Appendix E.3 details their training and evaluation.

Scene render. Without the scene render, head motion reaches the model only through the camera rays. Comparing High/High with Zero/Zero, each model tested as it was trained, shows that the render is the main carrier of head motion: on DL3DV, it lowers the rotation error and raises PSNR. On KAI-Ego, it raises PSNR but leaves hand accuracy unchanged (MPJPE 8.14 with the render and 7.97 without), because both models receive the hand render. Removing the scene render thus changes the camera and the scene but not the hands, consistent with the model's ability to combine the two actions on mobile manipulation (Sec. 5.3).

Render coverage. The render need not be dense: halving its points causes only a 0.34 dB PSNR drop and a 1.85° increase in rotation error on DL3DV, and Low performs comparably to High whenever a render is present. In contrast, removing the render from a model trained to always use one causes severe failure, indicating a train–test mismatch rather than a failure of ray conditioning. Training with one-third render-free clips reduces this error to 54.10° with at most a 0.16 dB PSNR cost when renders are available. While Random is preferable when test-time renders may be missing, MM-World always reconstructs one from the first frame; under this setting, High matches the alternatives on DL3DV and gives the lowest KAI-Ego MPJPE, motivating our default choice.

Scene geometry. At test time, MM-World predicts depth from the first frame alone, unlike the full-clip reconstruction used during training. Oracle full-clip depth improves navigation PSNR from 14.50 to 15.46 dB, but barely changes camera accuracy. This is because rotation is largely depth-independent and camera rays already encode the trajectory, while depth errors mainly affect translated renderings and thus appearance. The gap combines the lower accuracy of single-image depth with its mismatch to the training geometry, and closing it would mainly improve appearance.

6 Conclusion

We presented MM-World, a unified video world model that turns head and hand motion into egocentric video from a single image. It renders both actions into one image-space condition, reprojecting geometry lifted from that image along the head trajectory and overlaying hand meshes, so that one model learns from navigation and manipulation videos alike. We also curated a unified dataset and a four-track benchmark. MM-World outperforms the evaluated baselines in visual quality and action following on navigation and manipulation, and also on mobile manipulation, where it composes the two actions although no training clip was curated for this setting.

AI USE STATEMENT

Generative AI tools assisted with language refinement and manuscript organization. The authors reviewed all AI-assisted material and take full responsibility for the final manuscript.

REFERENCES

Rick Akkerman, Haiwen Feng, Michael J. Black, Dimitrios Tzionas, and Victoria Fernández Abrevaya. InterDyn: Controllable interactive dynamics with video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12467–12479, 2025.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Maria Elisabeth Bechtle, Feryal Behbahani, Stephanie C.Y. Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando De Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 4603–4623, 2024. URL https://proceedings.mlr.press/v235/bruce24a.html.

Yukang Chen, Luozhou Wang, Wei Huang, Shuai Yang, Bohan Zhang, Yicheng Xiao, Ruihang Chu, Weian Mao, Qixin Hu, Shaoteng Liu, Yuyang Zhao, Huizi Mao, Ying-Cong Chen, Enze Xie, Xiaojuan Qi, and Song Han. LongLive-2.0: An NVFP4 parallel infrastructure for long video generation. *arXiv preprint arXiv:2605.18739*, 2026a. doi: 10.48550/arXiv.2605.18739. URL https://arxiv.org/abs/2605.18739.

Yushuo Chen, Xiaoyu Shi, Xiaoshi Wu, Xintao Wang, Pengfei Wan, and Yebin Liu. HandsOnWorld: Unconstrained egocentric video generation with camera-disentangled hand control. *arXiv preprint arXiv:2607.02075*, 2026b. URL https://arxiv.org/abs/2607.02075.

Yutian Chen, Shi Guo, Renbiao Jin, Tianshuo Yang, Xin Cai, Yawen Luo, Mingxin Yang, Mulin Yu, Linning Xu, and Tianfan Xue. AnyRecon: Arbitrary-view 3D reconstruction with video diffusion model. *arXiv preprint arXiv:2604.19747*, 2026c.

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. *arXiv preprint arXiv:2606.16993*, 2026. URL https://arxiv.org/abs/2606.16993.

Mingju Gao, Kaisen Yang, Huan ang Gao, Bohan Li, Ao Ding, Wenyi Li, Yangcheng Yu, Jinkun Liu, Shaocong Xu, Yike Niu, Haohan Chi, Hao Chen, Hao Tang, Yu Zhang, Li Yi, and Hao Zhao. PAM: A pose-appearance-motion engine for sim-to-real HOI video generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026a.

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. *arXiv preprint arXiv:2607.07534*, 2026b. URL https://arxiv.org/abs/2607.07534.

Bohai Gu, Taiyi Wu, Yueyang Yuan, Jian Liu, Xiaocheng Lu, Dazhao Du, Jie Zhang, Jinxiang Lai, Shuai Yang, Xiaotong Zhao, Alan Zhao, and Song Guo. WorldCraft: From camera navigation to object manipulation in interactive video world models. *arXiv preprint arXiv:2605.25077*, 2026. URL https://arxiv.org/abs/2605.25077.

Jinkun Hao, Mingda Jia, Ruiyan Wang, Hongrui Zhu, Jiafei Cao, Xihui Liu, Ran Yi, Lizhuang Ma, Jiangmiao Pang, and Xudong Xu. EgoSim: Egocentric world simulator for embodied interaction generation. *arXiv preprint arXiv:2604.01001*, 2026. URL https://arxiv.org/abs/2604.01001.

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.

Xianglong He, Chunli Peng, Zexiang Liu, Boyang Wang, Yifan Zhang, Qi Cui, Fei Kang, Biao Jiang, Mengyin An, Yangyang Ren, Baixin Xu, Hao-Xiang Guo, Kaixiong Gong, Size Wu, Wei Li, Xuchen Song, Yang Liu, Yangguang Li, and Yahui Zhou. Matrix-Game 2.0: An open-source real-time and streaming interactive world model. *arXiv preprint arXiv:2508.13009*, 2025. URL https://arxiv.org/abs/2508.13009.

Ryan Hoque, Peide Huang, David J. Yoon, Mouli Sivapurapu, and Jian Zhang. EgoDex: Learning dexterous manipulation from large-scale egocentric video. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2505.11709.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Junchao Huang, Guian Fang, Shengju Qian, Xianghao Kong, Zhuoran Zhao, Wei Huang, Yihua Du, Zixin Zhang, Justin Cui, Yuchao Gu, et al. SolarWM: Open data and scalable training for long-horizon video world models. *arXiv preprint arXiv:2609.02886*, 2026.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2506.08009.

Byungjun Kim, Taeksoo Kim, Hyunsoo Cha, and Hanbyul Joo. Robot-factored world models via robot rendering. *arXiv preprint arXiv:2607.22535*, 2026a. URL https://arxiv.org/abs/2607.22535.

Byungjun Kim, Taeksoo Kim, Junyoung Lee, and Hanbyul Joo. Dexterous world models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026b. URL https://arxiv.org/abs/2512.17907.

Haopeng Li, Yitong Li, Junsong Chen, Tian Ye, Haozhe Liu, Jincheng Yu, Duomin Wang, Ruihua Zhang, Zeke Xie, Enze Xie, and Song Han. Sol-Attn: Accelerating video generation inference via on-the-fly attention sparsification. *arXiv preprint arXiv:2607.24027*, 2026a. doi: 10.48550/arXiv.2607.24027. URL https://arxiv.org/abs/2607.24027.

Yitong Li, Junsong Chen, Haopeng Li, Haozhe Liu, Jincheng Yu, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Sol Video Inference Engine: Agent-native full-stack acceleration framework for efficient video generation. *arXiv preprint arXiv:2606.23743*, 2026b. doi: 10.48550/arXiv.2606.23743. URL https://arxiv.org/abs/2606.23743.

Yue Liao, Pengfei Zhou, Siyuan Huang, Donglin Yang, Shengcong Chen, Yuxin Jiang, Yue Hu, Jingbin Cai, Si Liu, Jianlan Luo, Liliang Chen, Shuicheng Yan, Maoqing Yao, and Guanghui Ren. Genie Envisioner: A unified world foundation platform for robotic manipulation. *arXiv preprint arXiv:2508.05635*, 2025. URL https://arxiv.org/abs/2508.05635.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, Xuanmao Li, Xingpeng Sun, Rohan Ashok, Aniruddha Mukherjee, Hao Kang, Xiangrui Kong, Gang Hua, Tianyi Zhang, Bedrich Benes, and Aniket Bera. DL3DV-10K: A large-scale scene dataset for deep learning-based 3D vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22160–22169, 2024.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Yawen Luo, Xiaoyu Shi, Jianhong Bai, Menghan Xia, Tianfan Xue, Xintao Wang, Pengfei Wan, Di Zhang, and Kun Gai. CamCloneMaster: Enabling reference-based camera control for video generation. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–10, 2025.

Yawen Luo, Xiaoyu Shi, Junhao Zhuang, Yutian Chen, Quande Liu, Xintao Wang, Pengfei Wan, and Tianfan Xue. Shotstream: Streaming multi-shot video generation for interactive storytelling. In *European Conference on Computer Vision*, pp. 193–209. Springer, 2026.

Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. Do generative video models understand physical principles? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 948–958, 2026. URL https://openaccess.thecvf.com/content/WACV2026/html/Motamed_Do_Generative_Video_Models_Understand_Physical_Principles_WACV_2026_paper.html.

Jack Parker-Holder and Shlomi Fruchter. Genie 3: A new frontier for world models. Google DeepMind Blog, 2025. URL https://deepmind.google/discover/blog/genie-3-a-new-frontier-for-world-models/. Published August 5, 2025.

Rolandos Alexandros Potamias, Jinglei Zhang, Jiankang Deng, and Stefanos Zafeiriou. WiLoR: End-to-end 3D hand localization and reconstruction in-the-wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://arxiv.org/abs/2409.12259.

Ryan Punamiya, Simar Kareer, Zeyi Liu, Josh Citron, et al. EgoVerse: An egocentric human dataset for robot learning from around the world. *arXiv preprint arXiv:2604.07607*, 2026. URL https://arxiv.org/abs/2604.07607.

Runjia Qian, Zile Wang, Jihai Zhang, Kai Zou, Wei Yu, Jiaxing Li, Zexiang Liu, Yaokun Li, Fei Kang, Kaichen Huang, et al. Matrix-Game 3.5: Enhancing real-time streaming interactive world models with patch memory. *arXiv preprint arXiv:2608.29910*, 2026.

Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. GEN3C: 3D-informed world-consistent video generation with precise camera control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://arxiv.org/abs/2503.03751.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, Yihang Chen, Jie Liu, Yansong Cheng, Yao Yao, Jiayi Zhu, Yihao Meng, Kecheng Zheng, Qingyan Bai, Jingye Chen, Zehong Shen, Yue Yu, Xing Zhu, Yujun Shen, and Hao Ouyang. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026. URL https://arxiv.org/abs/2601.20540.

Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics*, 36(6):245:1–245:17, 2017. doi: 10.1145/3130800.3130883. URL https://mano.is.tue.mpg.de/.

Wenqiang Sun, Haiyu Zhang, Haoyuan Wang, Junta Wu, Zehan Wang, Zhenwei Wang, Yunhong Wang, Jun Zhang, Tengfei Wang, and Chunchao Guo. WorldPlay: Towards long-term geometric consistency for real-time interactive world modeling. *arXiv preprint arXiv:2512.14614*, 2025. URL https://arxiv.org/abs/2512.14614.

Yuanpeng Tu, Hao Luo, Xi Chen, Xiang Bai, Fan Wang, and Hengshuang Zhao. PlayerOne: Egocentric world simulator. *arXiv preprint arXiv:2506.09995*, 2025. URL https://arxiv.org/abs/2506.09995.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric and challenges. *arXiv preprint arXiv:1812.01717*, 2019. doi: 10.48550/arXiv.1812.01717.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan

Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. doi: 10.48550/arXiv.2503.20314. URL https://arxiv.org/abs/2503.20314.

Qinghe Wang, Yawen Luo, Xiaoyu Shi, Xu Jia, Huchuan Lu, Tianfan Xue, Xintao Wang, Pengfei Wan, Di Zhang, and Kun Gai. CineMaster: A 3D-aware and controllable framework for cinematic text-to-video generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–10, 2025.

Weijie Wang, Haoyu Zhao, Yifan Yang, Feng Chen, Zeyu Zhang, Yefei He, Zicheng Duan, Donny Y. Chen, Yuqing Yang, and Bohan Zhuang. Latent spatial memory for video world models. *arXiv preprint arXiv:2606.09828*, 2026a. doi: 10.48550/arXiv.2606.09828. URL https://arxiv.org/abs/2606.09828.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=DTQIjngDta.

Yuxi Wang, Wenqi Ouyang, Tianyi Wei, Yi Dong, Zhiqi Shen, and Xingang Pan. Hand2World: Autoregressive egocentric interaction generation via free-space hand gestures. *arXiv preprint arXiv:2602.09600*, 2026c. URL https://arxiv.org/abs/2602.09600.

World Labs Team. Atlas: A world model for spatial intelligence. World Labs Blog, 2026. URL https://www.worldlabs.ai/blog/atlas. Published September 1, 2026.

Zhuoyuan Wu and Jun Gao. OSCAR: Omni-embodiment action-conditioned world model for robotics. *arXiv preprint arXiv:2606.04463*, 2026. URL https://arxiv.org/abs/2606.04463.

Linxi Xie, Lisong C. Sun, Ashley Neall, Tong Wu, Shengqu Cai, and Gordon Wetzstein. Generated Reality: Human-centric world simulation using interactive video generation with hand and camera control. *arXiv preprint arXiv:2602.18422*, 2026. URL https://arxiv.org/abs/2602.18422.

Sherry Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=sFyTZEqmUY.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. LongLive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025. doi: 10.48550/arXiv.2509.22622. URL https://arxiv.org/abs/2509.22622.

Minghao Yin, Jiahao Lu, Wenbo Hu, Wang Zhao, Ying Shan, and Kai Han. SCoPE: Sightline-coordinate positional encoding for video diffusion transformers. *arXiv preprint arXiv:2606.27345*, 2026. doi: 10.48550/arXiv.2606.27345. URL https://arxiv.org/abs/2606.27345.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Frédo Durand, and William T. Freeman. Improved distribution matching distillation for fast image synthesis. In *Advances in Neural Information Processing Systems*, 2024a. URL https://arxiv.org/abs/2405.14867.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024b. URL https://arxiv.org/abs/2311.18828.

Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. ViewCrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024. URL https://arxiv.org/abs/2409.02048.

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

Haoyu Zhao, Xingyue Zhao, Hangyu Li, Biao Gong, Kehan Li, Siteng Huang, Xin Li, Deli Zhao, and Zhongyu Li. RynnWorld-Teleop: An action-conditioned world model for digital teleoperation. *arXiv preprint arXiv:2607.06558*, 2026. URL https://arxiv.org/abs/2607.06558.

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*, 2025. URL https://arxiv.org/abs/2406.14540.

Haoyi Zhu, Haozhe Liu, Yuyang Zhao, Tian Ye, Junsong Chen, Jincheng Yu, Tong He, Song Han, and Enze Xie. SANA-WM: Efficient minute-scale world modeling with hybrid linear diffusion transformer. *arXiv preprint arXiv:2605.15178*, 2026.

A Implementation Details

A.1 Generator Implementation Details

Scene rendering. Depth discontinuities are detected on the predicted depth map with a relative threshold of 0.03. Hand pixels are identified by the first-frame hand render, resized to the reconstruction resolution without dilation; if this mask is unavailable or would remove every remaining candidate, hand exclusion is skipped. Of the N eligible pixels, the $\max(1, \lfloor \rho N \rfloor)$ with the highest confidence are kept. During rendering, points less than 0.1 in front of the camera are discarded, and the rest are sorted from far to near and splatted with bilinear weights onto their 2×2 pixel neighborhoods using over-compositing. The support mask M_t^{s} marks pixels with accumulated opacity above 0.01; unsupported pixels are black.

Hand rendering. The hand camera $\mathbf{K}^{\mathrm{h}}$ follows the estimator's crop-to-image conversion: its principal point is the image center, and its focal length is the estimator's canonical focal length of 5000 pixels at a 256-pixel crop, rescaled to the longer image side. Each hand is rendered as a lit mesh with a single base color on a black background. The hand mask M_t^{h} is the union of visible hand pixels, so occlusion between the two hands is resolved by the rasterizer, whereas occlusion by scene objects is not modeled.

Latent conditioning. The mask transform $\mathcal{F}$ binarizes the support masks, resizes them to the latent resolution with nearest-neighbor sampling, repeats the initial mask four times to match the temporal layout of the video VAE, and folds groups of four temporal positions into channels. For $T + 1 = 81$ frames, this gives 21 latent frames with four mask channels. The noisy latent, folded mask, and condition latent contribute 16, 4, and 16 channels, respectively, and the backbone's patch embedding maps the resulting 36-channel input to spatiotemporal tokens.

Camera encoding. Rays are computed at every fourth frame, $t = 0, 4, \ldots, T$, which aligns them with the latent frames. For token i with Plücker coordinates $(\mathbf{d}_i, \mathbf{m}_i)$, where $\mathbf{m}_i = \mathbf{o}_i \times \mathbf{d}_i$, SCoPE decomposes the moment into its direction $\hat{\mathbf{m}}_i = \mathbf{m}_i / \|\mathbf{m}_i\|$ and log-norm $\lambda_i = \log \|\mathbf{m}_i\|$, and forms the query and key features $\mathbf{f}_i^q = (\mathbf{d}_i, \hat{\mathbf{m}}_i, \lambda_i)$ and $\mathbf{f}_i^k = (\hat{\mathbf{m}}_i, \mathbf{d}_i, \lambda_i)$. At block ℓ, the query and key of token i become

$$\begin{aligned} \tilde{\mathbf{q}}_i^{(\ell)} &= \mathbf{q}_i^{(\ell)} + \alpha_q^{(\ell)}\, g^{(\ell)}(\lambda_i) \odot \mathrm{RMSNorm}\big(E_q^{(\ell)}(\mathbf{f}_i^q)\big), \\ \tilde{\mathbf{k}}_i^{(\ell)} &= \mathbf{k}_i^{(\ell)} + \alpha_k^{(\ell)}\, g^{(\ell)}(\lambda_i) \odot \mathrm{RMSNorm}\big(E_k^{(\ell)}(\mathbf{f}_i^k)\big), \end{aligned} \tag{4}$$

where $\mathbf{q}_i^{(\ell)}$ and $\mathbf{k}_i^{(\ell)}$ have already undergone query/key normalization and rotary positional encoding. $E_q^{(\ell)}$ and $E_k^{(\ell)}$ are bias-free two-layer MLPs with 64 hidden units whose output layers are initialized to zero, $g^{(\ell)}$ is a sigmoid gate shared by queries and keys, and the scalars $\alpha_q^{(\ell)}$ and $\alpha_k^{(\ell)}$ are initialized to one. The 40 blocks of the backbone each receive an independent module, adding 289M trainable parameters. For scale augmentation, with probability 0.3 per clip we add an offset drawn uniformly from $[-1.2, 1.6]$ to the gate input λ_i; the features $\mathbf{f}_i^q$ and $\mathbf{f}_i^k$ are left unchanged.

Optimization. LoRA adapters have rank 32 and $\alpha = 32$ and are applied to the query, key, value, and output projections and to both feed-forward layers. We use AdamW with a learning rate of 10^{-4} for LoRA and 2×10^{-5} for SCoPE, weight decay 0.01, and a constant schedule. Training clips are drawn from the four sources with the mixture weights given in Appendix C.1.

Objective and sampling. The training scheduler discretizes time into 1000 levels, $\tau_k = 5u_k/(1 + 4u_k)$ with $u_k = 1 - k/1000$ for $k = 0, \ldots, 999$, and each sample draws its level index uniformly. The weight $w(\tau)$ in Eq. (3) is proportional to $\exp\big(-2(\tau - \frac{1}{2})^2\big)$, shifted to a minimum of zero and normalized to unit mean over the grid, and the MSE is averaged over latent entries. At inference, we use 50 Euler steps on the same shifted schedule without classifier-free guidance.

B Autoregressive Extension Details

This appendix details the autoregressive variant of Sec. 3.5.Following recent streaming generation paradigms (Yang et al., 2025; Chen et al., 2026a; Luo et al., 2026), we train it in two stages: au-

toregressive pretraining with teacher forcing, which yields a causal multi-step model, followed by distribution matching distillation (DMD) (Yin et al., 2024b;a) on the model's own streaming rollouts, which yields a few-step model. The rollout protocol of the long-horizon track is given in Appendix D.

Chunk-wise causal generation. We divide the target latent $z = \mathcal{E}(V_{0:T})$ along time into chunks $z^1, \dots, z^N$ of f latent frames each. Every condition of Secs. 3.2 and 3.3 is aligned with the frames it describes, so we split it in the same way: chunk n receives the slice of the unified visual condition (R, M) and the camera rays that cover its frames, together with the reference image I_0 and the description s. We denote these conditions by c^n. Chunk n thus depends only on the actions up to its last frame, and the model can consume the action stream as it arrives. We replace the bidirectional self-attention with block-causal attention: tokens attend to all tokens in their own chunk and to earlier chunks within a sliding window, and the keys and values of generated chunks are kept in a KV cache during inference. The resulting conditional distribution factorizes as

$$p_\theta\big(z^{1:N} \mid c^{1:N}\big) = \prod_{n=1}^{N} p_\theta\big(z^n \mid z^{<n},\, c^n\big)\,. \tag{5}$$

Stage 1: Autoregressive pretraining with teacher forcing. We pretrain the causal model on a large corpus of synthetic videos with teacher forcing, following LongLive-2.0 (Chen et al., 2026a): each chunk is denoised given the ground-truth clean latents of the preceding chunks. All chunks are trained in one forward pass by concatenating the clean sequence $z^{1:N}$ with its noisy copy and applying a block-sparse mask under which the noisy chunk $z^n_{\tau_n}$ attends to its own tokens and to the clean chunks $z^{<n}$. With the flow-matching objective of Eq. (3), the loss is

$$z^n_{\tau_n} = (1-\tau_n)\, z^n + \tau_n\, \epsilon^n, \qquad \mathcal{L}_{\mathrm{TF}} = \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N} w(\tau_n)\ \mathrm{MSE}\Big(v_\theta\big(z^n_{\tau_n}, \tau_n;\, z^{<n}, c^n\big),\, \epsilon^n - z^n\Big)\right], \tag{6}$$

where $\epsilon^n \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and τ_n is the diffusion time of chunk n. At inference, the resulting model generates in a streaming fashion: it denoises chunk n from Gaussian noise with multiple Euler steps while attending to the KV cache of the previously generated chunks, then runs the clean chunk once more at $\tau = 0$ to append its keys and values to the cache.

Stage 2: Streaming DMD distillation. Teacher forcing trains every chunk on ground-truth history, whereas at inference each chunk is conditioned on the model's own outputs, so errors accumulate over long rollouts (Huang et al., 2025). We therefore distill the Stage 1 model into a few-step generator G_θ on its own rollouts, following the streaming long tuning of LongLive (Yang et al., 2025). DMD matches the distribution of generated videos to that of a frozen teacher: with $\hat{z} = G_\theta(\epsilon; c)$ and its noised version $\hat{z}_\tau$, the generator is updated with

$$\nabla_\theta \mathcal{L}_{\mathrm{DMD}} = -\,\mathbb{E}_{\tau,\epsilon}\Big[\big(s_{\mathrm{real}}(\hat{z}_\tau, \tau; c) - s_{\mathrm{fake}}(\hat{z}_\tau, \tau; c)\big)\, \frac{\partial G_\theta(\epsilon; c)}{\partial \theta}\Big], \tag{7}$$

where s_{real} is the score of the frozen teacher and s_{fake} is the score of a critic trained online with the flow-matching loss on the generator's outputs. Distillation proceeds in two steps. We first distill on the initial 5-second window: the generator rolls out this window chunk by chunk from scratch, and Eq. (7) is applied to it. We then extend training to later windows (5–10 s, 10–15 s, and so on): each window is generated by continuing the rollout from the KV cache of the frames generated so far, and the DMD loss is applied only to the new window. Earlier frames serve as detached context, so gradients flow only through the current window and the memory cost does not grow with the rollout length. Throughout this stage, self-attention uses a sliding window of 21 latent frames. Training thus follows the same streaming procedure as inference.

Inference acceleration. We deploy the distilled model with the Sol Video Inference Engine (Li et al., 2026b), a training-free acceleration framework for video diffusion models, and use its kernel-level optimizations. Kernel fusion rewrites each transformer block so that memory-bound operations are fused into the adjacent matrix multiplications, reducing kernel launches and memory traffic, along with Sol-Attn (Li et al., 2026a) for efficient sparse attention computation.

C Data and Annotation Details

C.1 Data Sources

Table 5 lists the four sources with their numbers of source videos and clips and their sampling weights during training. Every clip has 81 frames at 848×480, giving 2.52M clips in total. A source video is a scene for DL3DV-10K and a recorded episode for KAI-Ego and EgoVerse. DL3DV-10K contributes navigation clips with substantial camera motion through static scenes and no hand interaction. EgoDex consists of tabletop tasks recorded with Apple Vision Pro. KAI-Ego covers everyday hand–object interactions in diverse indoor settings, including service, office, and retail environments. The EgoVerse clips come from the subsets contributed by its industry partners Scale AI and Mecka AI, with 6,039 and 38,855 episodes, respectively; they are recorded in real-world environments and show more varied scenes, viewpoints, and demonstrators. Although EgoDex and EgoVerse provide their own camera and hand pose annotations, we do not use them: all annotations are estimated from the RGB frames (Appendices C.2–C.4).

During training, we sample KAI-Ego, EgoVerse, EgoDex, and DL3DV with probabilities 0.3, 0.3, 0.3, and 0.1; this raises DL3DV from about 2% of the clips to 10% of the training samples, so its clips are repeated during training. The benchmark clips (Sec. 4.2) come from the same sources and are processed with the same pipeline, but they are held out and never used for training.

Table 5: Sources of the processed data. Every clip has 81 frames at 848×480. The weight is the probability with which a training sample is drawn from each source.

Source	Regime	Content	Source videos	Clips	Weight
DL3DV-10K	Navigation	Static scenes	10,510	55,551	0.1
EgoDex	Manipulation	Tabletop tasks	—	802,896	0.3
KAI-Ego (ours)	Manipulation	Everyday indoor interaction	138,873	1,076,188	0.3
EgoVerse	Manipulation	Real-world environments	44,894	582,036	0.3
Total				2,516,671	1.0

C.2 Camera Trajectories and Training Geometry

We reconstruct every ground-truth clip from all of its frames with Pi3X (Wang et al., 2026b), without camera priors. The reconstruction provides per-frame camera poses, which we express relative to the first camera; per-frame intrinsics, recovered from the predicted rays with the principal point fixed at the image center; and the first-frame points used to render training conditions (Sec. 3.2). Translations retain the approximate metric scale predicted by Pi3X. The poses and intrinsics serve as the camera actions for both training and evaluation; at evaluation, the scene condition is instead lifted from the first frame alone.

C.3 Hand Annotations

We detect hands in every frame with the detector of WiLoR (Potamias et al., 2025) at a confidence threshold of 0.3 and reconstruct MANO parameters from each detection with the same model. For each side, we keep the highest-scoring detection. When a hand is missed for at most three consecutive frames but detected on both sides of the gap, we interpolate its bounding box, linearly in the center and log-linearly in size, and reconstruct the hand from the interpolated box; gaps are not filled when the box would move more than 40 pixels per frame or change in area by more than a factor of two, and hands are not extrapolated beyond the first or last detection. On EgoDex, this post-processing keeps 118.5M of 122.6M raw detections and adds 0.62M interpolated boxes; on KAI-Ego, it keeps 172.7M of 175.2M raw detections and adds 0.21M interpolated boxes. The reconstructed hands are placed in the estimator's fixed camera $\mathbf{K}^{\mathrm{h}}$ (Appendix A.1) rather than in the Pi3X camera.

C.4 Scene–Object Descriptions

We annotate each clip with Qwen3.7-Plus, queried at temperature zero with nine frames sampled uniformly across the clip. The model returns a scene sentence, an object list, a condition description, and a separate action description; only the condition description is used as s. The prompt asks for the stable environment and a few representative objects (three to six for manipulation, four to eight for navigation), described by identity and intrinsic appearance, and allows objects that become visible only in later frames. It forbids mentioning people, body parts, or the camera, as well as actions, interaction targets, temporal order, motion, results, viewpoint changes, and object states caused within the clip. For manipulation clips, reading the description alone should not reveal which object is manipulated or what changes (Sec. 3.1).

D Benchmark and Evaluation Details

D.1 Benchmark Composition and Data Splits

Our benchmark contains three short-horizon tracks and one long-horizon track. The short-horizon benchmark comprises 809 evaluation clips: 156 navigation clips, 623 manipulation clips, and 30 mobile-manipulation clips. Every short-horizon clip contains 81 frames and is represented at 848×480 resolution. The long-horizon benchmark contains 37 complete sequences whose durations substantially exceed the model's local generation window. Table 6 summarizes the composition and native temporal properties of each subset. All examples are manually selected rather than randomly sampled.

Navigation. We construct the navigation split from the official evaluation partition of DL3DV-10K. We select 31 scenes and extract 156 clips: 30 scenes contribute five clips each and one scene contributes six. Each clip contains 81 frames at 15 fps. The selected scenes cover residential, retail, hospitality, institutional, and recreational indoor environments, and contain substantial translational or rotational camera motion without visible hand interaction. DL3DV scenes used for evaluation are excluded from model training at the scene level, preventing different temporal segments of the same scene from appearing in both splits.

Manipulation. The manipulation split contains 623 clips from three complementary sources, organized by increasing variation in capture conditions and environments. The *in-lab* tier contains 222 EgoDex clips, uniformly covering 111 tabletop tasks with two clips per task. The selected tasks include pick-and-place, insertion, folding, tool use, articulated-object manipulation, and bimanual coordination.

The *indoor* tier contains 200 KAI-Ego clips sampled from 200 distinct recording groups. Each clip depicts an everyday object interaction in a natural indoor environment. Restricting the subset to one clip per recording group reduces redundancy and prevents a small number of long recordings from dominating the evaluation.

The *in-the-wild* tier contains 201 EgoVerse clips, drawn from 201 distinct episodes, including 170 Mecka episodes and 31 Scale episodes. These clips also contain 81 frames at 15 fps. Compared with the other tiers, this subset exhibits larger variation in illumination, background appearance, camera pose, object arrangement, and demonstrator behavior.

Mobile-manipulation. We manually curate a mobile-manipulation split comprising 30 evaluation units from KAI-Ego, each containing 81 frames. Units are selected to ensure temporal overlap between camera motion and hand–object interaction. The complete source assets and their parent sequences are held out exclusively for evaluation and excluded from training.

Long-horizon split. We manually curate the long-horizon split by selecting 37 complete sequences: 14 navigation sequences from DL3DV and 23 manipulation sequences from KAI-Ego. Each underlying sequence is counted once, rather than treating different condition-density configurations as separate examples. The selected DL3DV sequences contain 301–511 frames at 30 fps (10.03–17.03 seconds), while the selected KAI-Ego sequences contain 241–481 frames at 15 fps (16.07–32.07 seconds). We use point-retention rates of 75% for DL3DV and 50% for KAI-Ego.

Table 6: Detailed composition of the benchmark.

Horizon	Track	Source	Examples	Frames	Duration
Short	Navigation	DL3DV	156	81	5.40 s
Short	Manipulation: in-lab	EgoDex	222	81	2.70 s
Short	Manipulation: indoor	KAI-Ego	200	81	5.40 s
Short	Manipulation: in-the-wild	EgoVerse	201	81	5.40 s
Short	Mobile manipulation	KAI-Ego	30	81	5.06–5.40 s
	Short-horizon total		**809**		
Long	Navigation	DL3DV	14	301–511	10.03–17.03 s
Long	Manipulation	KAI-Ego	23	241–481	16.07–32.07 s
	Long-horizon total		**37**		

Each example provides the initial RGB frame, temporally aligned sparse scene and interaction conditions, their validity masks, and per-frame camera parameters. The remaining RGB frames are generated autoregressively. We report short- and long-horizon results separately; the 37 complete sequences are excluded from the 809-clip short-horizon total.

Manual curation criteria. We apply four criteria consistently across the benchmark: (i) *Informative initial observation.* We favor clips whose first frame captures a broad portion of the scene and relevant objects appearing in subsequent frames, providing visual context for prediction and reducing the need to infer unseen scene content. (ii) *Temporal and interaction coherence.* Each clip must be temporally continuous and capture a coherent motion or interaction segment, with interpretable hand trajectories and, where applicable, consistent contact transitions and object responses. We exclude abrupt cuts, discontinuities, and ambiguous interactions whose progression cannot be reliably determined. (iii) *Multimodal alignment.* RGB frames and the corresponding camera, hand, and mask conditions must remain temporally synchronized and spatially consistent throughout the selected interval. (iv) *Visual usability.* The scene and interaction must remain sufficiently discernible to assess motion and object behavior. We exclude severe corruption, blur, or persistent loss of visibility that prevents reliable interpretation, while retaining meaningful occlusions and other challenging events. At the subset level, we seek coverage across environments, objects, viewpoints, hand laterality, and interaction types, including both simple motions and challenging cases involving occlusion, contact, object displacement, or state changes. For mobile manipulation, we additionally require global viewpoint change to occur concurrently with local hand–object interaction. Together, these criteria prioritize interpretable, well-aligned sequences while preserving the diversity and difficulty of the source data.

D.2 Baseline Configurations

All baselines use their recommended resolutions and frame counts. The ground-truth first frame is cropped and/or resized accordingly for conditioning, and the same spatial transformations are applied to the corresponding GT frames for fair evaluation. For models supporting text conditioning, we use the benchmark captions, except for a subset of MAMO samples whose prompts are left empty. These captions describe the overall scene and content rather than fine-grained actions. Hand and camera inputs are adapted to each baseline's required representations and input formats.

Native-interface comparison. The baselines expose heterogeneous conditioning interfaces. We therefore evaluate each model through its released inference protocol rather than reducing all controls to a single common representation. Each method is initialized from the same ground-truth first frame, after applying its required crop, resize, and preprocessing operations, and receives the most informative hand and camera signals accepted by its public interface. For camera-conditioned navigation models, we convert the calibrated benchmark trajectory to the coordinate convention and pose serialization required by each implementation. For manipulation models, the common MANO/WiLoR estimates are converted to their native representations.

When an interface supports sample-specific text, we provide the shared clip-level scene description. These captions describe the scene and visible content, but deliberately omit fine-grained action descriptions so that the action must be inferred from the control signal. The designated subset of MAMO samples retains an empty prompt. Models whose released protocols use a fixed prompt or empty text condition are evaluated without modifying that behavior. For every baseline, the same spatial transform is applied to the conditioning image, target video, rendered controls, and any camera intrinsics used to construct those controls.

Camera motion is supplied only through interfaces that explicitly support it. Hand2World receives the calibrated trajectory in first-frame-relative form, $T_t^{\mathrm{rel}} = (T_0^{\mathrm{c2w}})^{-1} T_t^{\mathrm{c2w}}$, together with crop-adjusted intrinsics, which the model converts to dense Plücker-ray conditioning. EgoSim instead incorporates camera motion into its rendered point-cloud scene prior and coverage mask. For RynnWorld-Teleop and Generated Reality, camera calibration is used only when projecting or expressing the hand controls; no additional camera matrices are passed to the generator. Wan2.1-Fun-14B-Control receives the aligned frame-0 image as both its start image and CLIP-image condition, an 81-frame black-background 2D hand-control video rendered from MANO/WiLoR estimates, and the shared clip-level scene description, but no camera matrices. Thus, the comparison measures each system under its intended native conditioning contract rather than under an artificially standardized input interface.

Autoregressive long-video protocols. For navigation, we evaluate HY-WorldPlay, DreamX-World-5B, LingBot-World Fast, Matrix-Game-3.5 Distilled, LingBot-World v2 Causal Fast, and SolarWM-5B Stage2 using their native autoregressive interfaces. HY-WorldPlay runs at 832×480 and 24 FPS; DreamX-World-5B and Matrix-Game-3.5 at 1280×704 and 16 FPS; both LingBot variants take an 832×480 input and produce 832×464 videos at 16 FPS; and SolarWM-5B runs at 864×480 and 16 FPS. Each model receives the ground-truth first frame, the calibrated camera trajectory in its required representation, and one clip-level caption. Sequence lengths are rounded to the nearest valid native chunk length while preserving the trajectory endpoints. Complete native outputs are retained without temporal or spatial normalization, including Matrix-Game's 84-frame generation blocks and SolarWM's released tail-padding behavior. For manipulation, Hand2World is run as one continuous causal session at 832×480 and 15 FPS, using four denoising steps and a 21-latent-slot sliding KV cache with camera re-anchoring. RynnWorld-Teleop is rolled out at 832×480 and 16 FPS using 81-frame windows with stride 80, 50 denoising steps, guidance 1.0, and additive control. Its first window uses the ground-truth first frame, while later windows use the preceding window's final generated frame. One caption is reused throughout each sequence, and padded or duplicated boundary frames are removed before evaluation.

D.3 Metric Computation and Aggregation

We group the metrics into five categories based on the aspect evaluated: frame fidelity, video quality, camera accuracy, motion consistency, and hand accuracy. We assess frame fidelity using PSNR, SSIM, and the perceptual similarity metric LPIPS (Zhang et al., 2018), and distribution-level video quality using Fréchet Video Distance (FVD) (Unterthiner et al., 2019). Camera accuracy is measured by rotation error (RotErr), absolute trajectory error (ATE), and relative pose error (RPE), while motion consistency is measured by optical flow endpoint error (Flow-EPE). For manipulation and mobile-manipulation clips only, we also evaluate hand accuracy using root-aligned mean per-joint position error (MPJPE) and Procrustes-aligned joint and vertex position errors (PA-MPJPE and PA-MPVPE). We report metrics separately, with one or two per aspect shown in the main tables.

Evaluation protocol. Generated and reference videos are paired by decoded frame index, with identical temporal sampling and evaluation ending when either video ends. Unless otherwise specified, every decoded frame is used. Before evaluation, each generated sequence is standardized to begin with exactly one copy of the original conditioning image, prepending it or replacing a reconstructed initial frame as needed. This frame is included in all paired evaluations, including the first transition used for optical-flow comparison.

Image fidelity: PSNR, SSIM, and LPIPS. Image metrics are computed on decoded RGB frames, with generated frames resized to the reference resolution when necessary. PSNR measures pixel-wise fidelity through the mean squared error, while SSIM compares local image statistics to assess

structural similarity; both are computed using PIQ. LPIPS measures perceptual differences using the official AlexNet-based model Zhang et al. (2018). All three metrics are evaluated over the full frame and averaged over frames within each video. Higher PSNR and SSIM indicate better fidelity, whereas lower LPIPS indicates better fidelity.

Camera accuracy: RotErr, ATE, and RPE. Camera trajectories are estimated independently from both videos using the same Pi3X model and checkpoint. We retain poses available at common frame indices and align the generated trajectory to the reference using a single sequence-level Umeyama similarity transform, accounting for global translation, rotation, and isotropic scale.

After alignment, RotErr is the mean angular difference between corresponding camera orientations, and ATE is the root mean squared distance between corresponding camera centers. RPE measures the mean rotational and translational discrepancies between local motions of consecutive poses in the retained sequence. RotErr and rotational RPE are reported in degrees; ATE and translational RPE are expressed in Pi3X reference-trajectory units, rather than meters. Lower values indicate better agreement.

These metrics compare independently estimated trajectories. The similarity alignment is applied only during evaluation: MM-World does not fit its scene-geometry scale to the test sequence during generation. Clips with failed pose estimation, fewer than three common poses, or undefined similarity alignment are excluded from camera evaluation. We report the number and fraction of valid trajectories.

Motion consistency: Flow-EPE. Flow-EPE compares optical flow estimated by RAFT-Large between consecutive evaluated frames in the reference and generated videos. Both videos are placed on the reference-resolution grid before flow estimation. The Euclidean distance between corresponding flow vectors is averaged over all image pixels and then over temporal transitions, including the transition from the conditioning frame to the first predicted frame. Flow-EPE is reported in pixels, with lower values indicating better motion agreement.

Video-distribution quality: FVD. FVD compares reference and generated video distributions using the Fréchet distance between Gaussian distributions fitted to their I3D features. We use the StyleGAN-V I3D implementation and extract one feature vector per video from 16 uniformly sampled frames, including the first and last frames of the evaluated window. FVD is computed jointly over each reported split, with lower values indicating closer distributional agreement. A video with fewer than 16 frames or a feature-extraction failure invalidates FVD for the affected split.

Hand accuracy: root-aligned MPJPE, PA-MPJPE, and PA-MPVPE. Hand geometry is estimated independently from both videos using the same WiLoR checkpoint and MANO model Potamias et al. (2025); Romero et al. (2017). Detections are matched within each frame and handedness class using Hungarian assignment based on normalized bounding-box center distances, with distant matches rejected.

Root-aligned MPJPE is the mean Euclidean distance between the 21 corresponding joints after independently subtracting each hand's wrist position. PA-MPJPE additionally aligns each predicted joint set to its reference using a fitted similarity transform. PA-MPVPE measures the mean distance between the 778 corresponding mesh vertices after a separate similarity alignment fitted to the vertices. Thus, root alignment removes translation only, whereas Procrustes alignment also removes rotation and isotropic scale. All hand errors are reported in millimeters and averaged over valid matched hand instances within each video. Lower values indicate better agreement. Because reference geometry is also estimated by WiLoR, these metrics assess reconstruction consistency under a fixed estimator, rather than global hand position or error against motion-capture annotations.

Aggregation and validity. Except for FVD, scores are first aggregated within each video and then averaged equally across videos with valid scores. Unmatched hands contribute no geometric error, and videos without valid hand matches are excluded from hand-metric averages. Hand metrics are evaluated only for manipulation and mobile-manipulation clips.

The overall manipulation score is computed across valid videos in the full 623-clip set, comprising tiers of 222, 200, and 201 videos. This weights tiers by their valid-video counts rather than equally.

For FVD, the overall score is recomputed from the pooled reference and generated feature sets; it is never obtained by averaging per-video or tier-level scores.

E Additional Results

E.1 Full Results of the Main Comparisons

Tables 7, 8, and 9 report all metrics at full precision for the comparisons in Tables 1, 2, and 3, including SSIM, the relative pose errors, ATE on the manipulation track, PA-MPVPE, and the optical-flow error Flow-EPE.

Table 7: All metrics for the navigation comparison of Table 1, at full precision. All models are bidirectional. Given the first frame and camera poses for the subsequent 80 frames, each model produces an 81-frame video. ↑/↓ indicate higher/lower is better. Rows are ordered by decreasing RotErr (worst to best). Best results are in **bold**, and second-best results are underlined, across all rows.

	Frame quality			Camera accuracy				Motion / video	
Method	PSNR↑	SSIM↑	LPIPS↓	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
HY-World1.5	11.1412	0.2397	0.6148	87.7593	0.6036	1.3670	0.0888	14.6512	398.4690
LingBot-World	11.3661	0.2725	0.6080	40.0449	0.3481	0.8632	0.0651	11.9340	294.2601
SCoPE	11.5271	0.2549	0.5838	36.7438	0.3800	1.0626	0.0557	13.9007	271.1343
DreamX-World	11.7813	0.2680	0.5800	31.0412	0.3374	1.0594	0.0627	12.1287	252.2664
SANA-WM	11.3250	0.2384	0.5830	19.5000	0.2334	0.8336	0.0478	10.6041	366.1390
SolarWM	12.7223	0.3081	0.5123	15.2753	0.2204	0.8657	0.0433	10.7819	229.2066
LingBot-World v2	10.8589	0.2383	0.5938	13.2135	0.1633	0.7628	0.0428	11.8168	299.3180
Matrix-Game-3.5	13.2348	0.3632	0.5254	7.9319	0.1483	0.3935	0.0358	11.4787	242.2346
Ours	**14.5036**	**0.4240**	**0.4232**	**5.9907**	**0.0953**	**0.3157**	**0.0239**	**5.5838**	**207.8224**

Table 8: All metrics for the manipulation comparison of Table 2, at full precision. Within each tier, rows are ordered by decreasing root-aligned MPJPE (worst to best). Both panels use the same row order. ↑/↓ indicate higher/lower is better. Best and second-best results within each tier are **bold** and underlined, respectively.

(a) Frame quality and hand accuracy

	Frame quality			Hand accuracy		
Method	PSNR↑	SSIM↑	LPIPS↓	Root-aligned MPJPE↓	PA-MPJPE↓	PA-MPVPE↓
Tier 1: EgoDex (in-lab; easiest)						
EgoSim	20.3381	0.8082	0.2134	31.1192	9.6029	9.0352
RynnWorld-Teleop	24.0767	0.8159	0.2088	28.9539	9.2083	8.6550
Hand2World	12.9969	0.6266	0.6399	28.0714	9.2226	8.6921
Wan2.2-Fun-5B-Control	22.9884	0.8055	0.2031	22.9535	7.3266	6.9145
Generated Reality	23.9767	0.8191	0.1888	20.7317	7.1252	6.7169
Wan2.1-Fun-14B-Control	24.2108	0.8122	0.1937	18.7429	6.1289	5.7649
Ours	**29.8549**	**0.9157**	**0.0877**	**10.7484**	**4.5462**	**4.2757**
Tier 2: KAI-Ego (indoor; intermediate)						
EgoSim	16.4009	0.6159	0.3254	40.0178	11.0246	10.4676
Hand2World	15.6487	0.5930	0.4475	17.3270	5.7230	5.4683
RynnWorld-Teleop	20.8915	0.6652	0.2291	13.9211	5.0309	4.7568
Wan2.2-Fun-5B-Control	20.4974	0.6814	0.2520	13.1094	4.3521	4.1396
Generated Reality	21.0008	0.6899	0.2479	12.9565	4.2208	4.0085
Wan2.1-Fun-14B-Control	21.0829	0.6876	0.2184	9.8669	3.4855	3.3094
Ours	**26.1849**	**0.8624**	**0.1106**	**6.5762**	**2.5830**	**2.4619**
Tier 3: EgoVerse (near in-the-wild; hardest)						
Hand2World	11.4099	0.4416	0.6493	50.8309	10.1164	9.5378
EgoSim	14.1472	0.52313	0.4009	39.3332	11.0207	10.3966
RynnWorld-Teleop	15.4124	0.5173	0.4300	32.1740	9.6813	9.1387
Wan2.2-Fun-5B-Control	15.3092	0.5235	0.4362	24.9442	7.7163	7.3334
Generated Reality	15.8833	0.5406	0.4200	22.8119	7.4052	7.0150
Wan2.1-Fun-14B-Control	15.3215	0.5144	0.4149	21.2562	6.5920	6.2060
Ours	**21.7009**	**0.7692**	**0.1785**	**12.2173**	**4.9021**	**4.6223**

(b) Camera accuracy and motion / video quality

	Camera accuracy				Motion / video	
Method	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
Tier 1: EgoDex (in-lab; easiest)						
EgoSim	86.6691	0.0289	0.2041	0.0062	1.0267	192.8476
RynnWorld-Teleop	108.6965	0.0342	0.2589	0.0063	1.1627	248.6226
Hand2World	126.0974	0.0393	1.1647	0.0074	1.5574	914.3043
Wan2.2-Fun-5B-Control	102.6416	0.0367	0.2687	0.0068	1.9191	148.9877
Generated Reality	99.4102	0.0361	0.2396	0.0066	1.4853	174.1311
Wan2.1-Fun-14B-Control	98.0084	0.0328	0.2555	0.0062	1.1332	162.1087
Ours	**49.2454**	**0.0240**	**0.1410**	**0.0045**	**0.6336**	**67.4600**
Tier 2: KAI-Ego (indoor; intermediate)						
EgoSim	91.3946	0.0339	0.1567	0.0063	0.9086	390.2976
Hand2World	120.8987	0.0478	0.6066	0.0079	0.7916	779.1409
RynnWorld-Teleop	92.0903	0.0466	0.2011	0.0071	1.0274	185.2659
Wan2.2-Fun-5B-Control	99.3725	0.0459	0.1958	0.0074	1.4662	202.5144
Generated Reality	94.2336	0.0459	0.1963	0.0074	1.3252	224.7240
Wan2.1-Fun-14B-Control	87.6597	0.0448	0.1930	0.0070	0.8997	192.5219
Ours	**34.1617**	**0.0267**	**0.0975**	**0.0046**	**0.4466**	**82.4980**
Tier 3: EgoVerse (near in-the-wild; hardest)						
Hand2World	107.6900	0.1128	1.4255	0.0220	2.9861	1333.5780
EgoSim	75.6303	0.0803	0.5073	0.0164	3.0811	378.2288
RynnWorld-Teleop	109.9943	0.1149	0.7538	0.0213	4.3385	299.5526
Wan2.2-Fun-5B-Control	120.3866	0.1217	0.6714	0.0227	5.9810	347.1635
Generated Reality	112.1093	0.1199	0.6361	0.0228	5.5024	454.1505
Wan2.1-Fun-14B-Control	113.4812	0.1157	0.6662	0.0211	3.8837	378.5693
Ours	**23.3736**	**0.0496**	**0.2073**	**0.0096**	**1.2389**	**136.8482**

Table 9: All metrics for the mobile manipulation comparison of Table 3, at full precision. Only models whose architectures support mobile manipulation are evaluated. Rows are ordered by decreasing root-aligned MPJPE (worst to best). Both panels use the same row order. ↑/↓ indicate higher/lower is better. Best and second-best results across all rows are **bold** and underlined, respectively.

(a) Frame quality and hand accuracy

	Frame quality			Hand accuracy		
Method	PSNR↑	SSIM↑	LPIPS↓	Root-aligned MPJPE↓	PA-MPJPE↓	PA-MPVPE↓
EgoSim	12.9024	0.4322	0.4799	68.7865	13.4038	12.6701
Hand2World	10.9638	0.4155	0.6828	30.6236	10.0549	9.4586
Ours	**18.6401**	**0.6513**	**0.2528**	**10.2829**	**5.0536**	**4.7513**

(b) Camera accuracy and motion / video quality

	Camera accuracy				Motion / video	
Method	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
EgoSim	60.0773	0.0780	0.4913	0.0151	3.3116	1255.8163
Hand2World	86.4367	0.1504	1.4913	0.0269	3.1201	2497.5500
Ours	**18.2240**	**0.0549**	**0.2230**	**0.0100**	**1.3795**	**545.2918**

E.2 LONG-HORIZON GENERATION

Tables 10 and 11 report the long-horizon track of Sec. 4.2. We compare the autoregressive variant of Sec. 3.5 (Appendix B) only with models that support autoregressive long-video generation. On manipulation, it is best on every metric. On navigation, it has the best PSNR, SSIM, LPIPS, and Flow-EPE, but not the best camera accuracy or FVD: SolarWM-5B Stage2 has the lowest rotation error, LingBot-World v2 Causal Fast the lowest ATE and translational RPE, and Matrix-Game-3.5 Distilled the lowest rotational RPE and FVD.

Table 10: Long-horizon navigation results on the 14 DL3DV sequences of the long-horizon track. Only models supporting autoregressive long-video generation are evaluated. Rows are ordered by decreasing RotErr (worst to best). Both panels use the same row order. ↑/↓ indicate higher/lower is better. Best and second-best results across all rows are **bold** and underlined, respectively.

(a) Frame quality

Method	PSNR↑	SSIM↑	LPIPS↓
HY-WorldPlay	11.0065	0.2577	0.6432
DreamX-World-5B	11.4226	0.2473	0.6305
LingBot-World Fast	9.3337	0.2164	0.7059
Ours	**14.9847**	**0.4364**	**0.4576**
Matrix-Game-3.5 Distilled	12.5384	0.2867	0.5463
LingBot-World v2 Causal Fast	10.4530	0.2226	0.6340
SolarWM-5B Stage2	12.2635	0.2963	0.5443

(b) Camera accuracy and motion / video quality

	Camera accuracy				Motion / video	
Method	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
HY-WorldPlay	122.1279	0.5300	0.3191	0.0297	3.4401	1089.5114
DreamX-World-5B	48.5330	0.2410	0.2334	0.0155	8.4852	1119.0666
LingBot-World Fast	40.4422	0.3098	0.4108	0.0118	10.6765	1421.8805
Ours	36.0932	0.3271	0.2799	0.0209	**0.6529**	1102.0646
Matrix-Game-3.5 Distilled	29.0213	0.1814	**0.1525**	0.0132	1.5019	**969.1707**
LingBot-World v2 Causal Fast	27.3362	**0.1700**	0.2382	**0.0087**	6.8620	1183.7035
SolarWM-5B Stage2	**25.7826**	0.1728	0.6596	0.0116	7.7117	1161.1823

Table 11: Long-horizon manipulation results on the 23 KAI-Ego sequences of the long-horizon track. Only models supporting autoregressive long-video generation are evaluated. Rows are ordered by decreasing root-aligned MPJPE (worst to best). Both panels use the same row order. ↑/↓ indicate higher/lower is better. Best and second-best results across all rows are **bold** and underlined, respectively.

(a) Frame quality and hand accuracy

	Frame quality			Hand accuracy		
Method	PSNR↑	SSIM↑	LPIPS↓	Root-aligned MPJPE↓	PA-MPJPE↓	PA-MPVPE↓
RynnWorld-Teleop	15.7414	0.4988	0.4763	37.9195	11.0088	10.3718
Hand2World	11.8215	0.4239	0.6303	16.3386	5.8830	5.6194
Ours	**19.2823**	**0.6648**	**0.2747**	**10.4696**	**4.5112**	**4.2730**

(b) Camera accuracy and motion / video quality

	Camera accuracy				Motion / video	
Method	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
RynnWorld-Teleop	102.5193	0.1065	0.4343	0.0108	3.2277	1447.2502
Hand2World	122.1660	0.1030	0.4845	0.0105	2.4098	1692.8832
Ours	**47.9475**	**0.0760**	**0.2236**	**0.0091**	**1.3761**	**1018.1876**

E.3 Render-Coverage Ablation

Models. The four models of Sec. 5.4 differ only in the fraction ρ of first-frame points retained during training. High keeps 75% for navigation and 50% for manipulation, as in the main model;

Low keeps 40% and 25%; Zero renders no scene points after the first frame, although manipulation clips keep the hand render; and Random draws one of these three levels with probability $1/3$ for each training clip, so its average retention (38.3% on DL3DV and 25% on KAI-Ego) is close to that of Low. All four start from the pretrained backbone with newly initialized LoRA and SCoPE modules and otherwise follow the configuration of the main model (Appendix A.1), including the scene descriptions. They share the initialization, the seed, the data order, and the number of training steps. To limit the cost, they are trained only on a subset of 110,000 KAI-Ego clips and on the 55,274 DL3DV training clips, sampled in proportion to their sizes, with a global batch of 64 on 64 GPUs; one pass over these data takes 2,583 steps, whereas the main model is trained on four sources for 30K steps (Sec. 5.1).

Evaluation. Each model is tested at the High, Low, and Zero levels. Because the retained points are those with the highest predicted confidence, lower levels drop the least confident points. The renders of this study use the full-clip depth of training rather than the single-image depth of test time (Sec. 3.2). The High and Low models never see a clip without a scene render during training, and the Zero model never sees one with it, so the corresponding cells of Table 4 are outside the training distribution.

Results. Table 12 reports all metrics after 2,500 steps, the checkpoint of Table 4, and Table 13 repeats Table 4 after 2,000 steps. The conclusions of Sec. 5.4 hold at both checkpoints, although the differences quoted there refer to the 2,500-step checkpoint. The order of High, Low, and Random at the levels with a render changes between the checkpoints, which is why the tables do not rank these models.

Table 12: All metrics of the render-coverage ablation after 2,500 steps, the checkpoint of Table 4. *Train* and *Test* give the retained fraction ρ used in training and at test time. Panel (a) reports DL3DV, and panels (b) and (c) report KAI-Ego. Values are given to two decimals.

(a) DL3DV

		Frame quality			Camera accuracy				Motion / video	
Train	Test	PSNR↑	SSIM↑	LPIPS↓	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
High (default)	High	15.24	0.47	0.39	6.37	0.10	0.36	0.03	5.50	191.35
	Low	14.90	0.45	0.40	8.22	0.12	0.40	0.03	5.85	206.99
	Zero	11.19	0.24	0.61	99.16	0.73	1.70	0.12	17.40	544.27
Low	High	15.37	0.48	0.39	5.59	0.10	0.35	0.03	5.36	191.71
	Low	15.10	0.46	0.40	7.46	0.11	0.37	0.03	5.55	197.01
	Zero	11.25	0.25	0.62	93.83	0.74	1.69	0.12	16.88	501.43
Random	High	15.25	0.47	0.39	6.63	0.11	0.38	0.03	5.63	194.15
	Low	14.96	0.46	0.40	6.50	0.12	0.41	0.03	5.93	201.09
	Zero	11.57	0.27	0.61	54.10	0.53	1.47	0.08	17.16	321.51
Zero	High	9.34	0.29	0.64	29.34	0.42	0.86	0.08	8.34	717.41
	Low	9.99	0.29	0.64	25.87	0.38	0.89	0.07	8.40	603.23
	Zero	11.96	0.28	0.57	33.16	0.36	1.05	0.06	12.85	275.08

(b) KAI-Ego: frame quality and hand accuracy

		Frame quality			Hand accuracy		
Train	Test	PSNR↑	SSIM↑	LPIPS↓	MPJPE↓	PA-MPJPE↓	PA-MPVPE↓
High (default)	High	25.13	0.83	0.12	8.14	3.18	3.02
	Low	24.90	0.81	0.12	8.11	3.21	3.05
	Zero	21.54	0.66	0.21	8.21	3.22	3.06
Low	High	25.12	0.84	0.12	8.31	3.21	3.04
	Low	25.13	0.83	0.12	8.33	3.19	3.02
	Zero	21.70	0.67	0.21	8.32	3.22	3.06
Random	High	24.97	0.83	0.13	8.35	3.29	3.12
	Low	24.83	0.82	0.13	8.24	3.25	3.08
	Zero	21.96	0.69	0.20	8.20	3.17	3.01
Zero	High	14.58	0.58	0.52	25.15	6.43	6.14
	Low	16.76	0.64	0.41	15.52	5.01	4.76
	Zero	22.39	0.70	0.18	7.97	3.07	2.92

(c) KAI-Ego: camera accuracy and motion / video quality

		Camera accuracy				Motion / video	
Train	Test	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
High (default)	High	36.37	0.02	0.10	0.00	0.49	125.94
	Low	40.20	0.02	0.11	0.00	0.50	126.74
	Zero	76.70	0.04	0.18	0.01	0.90	195.30
Low	High	38.74	0.02	0.10	0.00	0.49	130.41
	Low	37.91	0.02	0.10	0.00	0.49	130.93
	Zero	75.82	0.04	0.18	0.01	0.88	202.07
Random	High	39.78	0.02	0.10	0.00	0.49	127.30
	Low	39.93	0.03	0.11	0.00	0.50	129.82
	Zero	64.70	0.03	0.17	0.01	0.83	180.06
Zero	High	102.93	0.05	0.28	0.01	0.78	1097.96
	Low	85.06	0.04	0.21	0.01	0.67	711.92
	Zero	58.73	0.03	0.15	0.01	0.74	153.61

Table 13: Render coverage after 2,000 steps, in the layout of Table 4. Rows give the level used in training and columns the level used at test time.

	DL3DV						KAI-Ego					
	RotErr↓			PSNR↑			PSNR↑			MPJPE↓		
Training	High	Low	Zero	High	Low	Zero	High	Low	Zero	High	Low	Zero
High (default)	6.06	7.19	99.22	15.34	14.98	11.21	25.36	24.99	21.53	8.07	8.18	8.30
Low	6.37	7.45	92.14	15.22	15.00	11.28	25.02	25.18	21.68	8.29	8.38	8.17
Random	6.36	8.87	60.24	15.15	14.85	11.49	25.10	24.84	21.59	8.40	8.43	8.38
Zero	30.84	29.01	34.77	9.49	10.10	11.93	14.87	16.77	22.21	24.48	15.96	8.27

E.4 Scene-Geometry Ablation

Table 14 reports all metrics of the geometry comparison in Sec. 5.4. Both rows use the main model on the navigation track and render only the points of the first frame; they differ only in how the depth of these points is estimated. *Single image* predicts it from I_0 alone, the test-time setting of Sec. 3.2. *Full clip* takes it from the Pi3X reconstruction of the full reference clip, which also provides the camera trajectory, as in training (Sec. 4.1). Because this reconstruction uses the future ground-truth frames, the full-clip depth is an oracle rather than a valid test-time input. The oracle improves all nine metrics. The comparison cannot separate two effects: the lower accuracy of single-image depth and its mismatch with the full-clip geometry on which the model is trained.

Table 14: All metrics of the scene-geometry ablation on the DL3DV navigation track, at full precision. Both rows use the main model and differ only in the depth of the first-frame points. *Single image* is the test-time setting and the Ours row of Table 7; *Full clip* uses future ground-truth frames and is an oracle.

	Frame quality			Camera accuracy				Motion / video	
Depth	PSNR↑	SSIM↑	LPIPS↓	RotErr↓	ATE↓	RPE_r ↓	RPE_t ↓	Flow-EPE↓	FVD↓
Single image	14.5036	0.4240	0.4232	5.9907	0.0953	0.3157	0.0239	5.5838	207.8224
Full clip (oracle)	15.4572	0.4879	0.3780	5.8691	0.0885	0.3057	0.0228	5.0000	192.5591

E.5 Additional Qualitative Results

Figures 4, 5, 6, and 7 show all baselines at four time steps, on further clips for DL3DV, EgoDex, and KAI-Ego and on the EgoVerse clip of Figure 3.

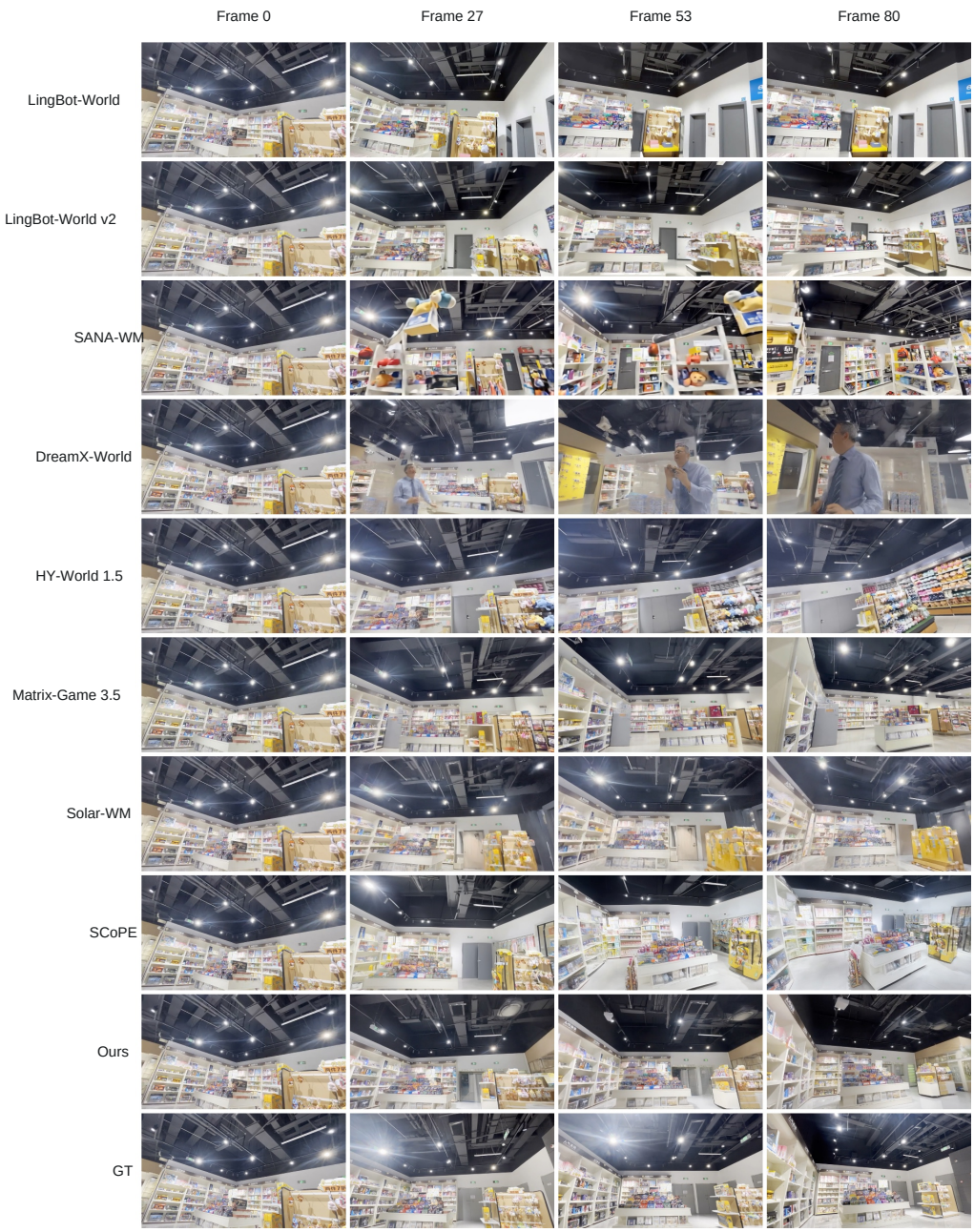

Figure 4: All navigation baselines on a further DL3DV clip at four time steps.

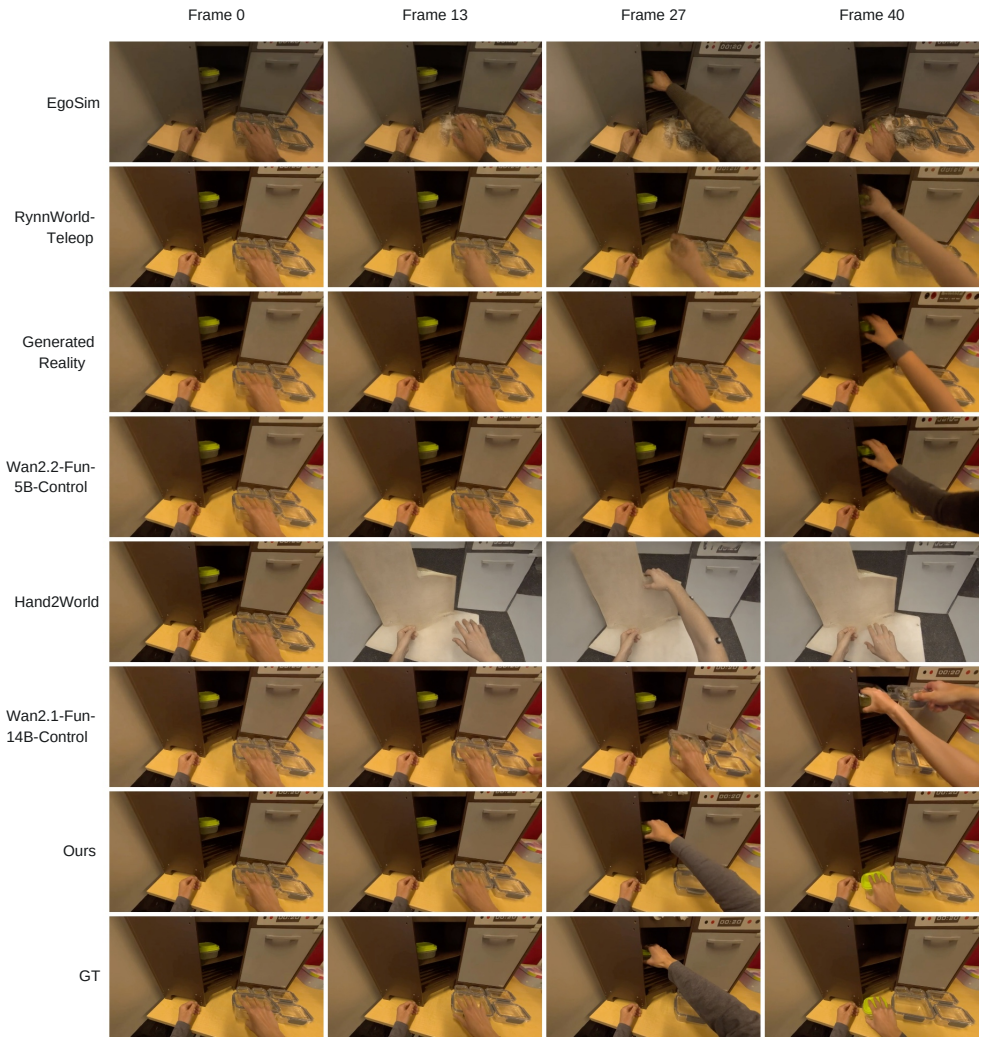

Figure 5: All manipulation baselines on a further EgoDex clip at four time steps.

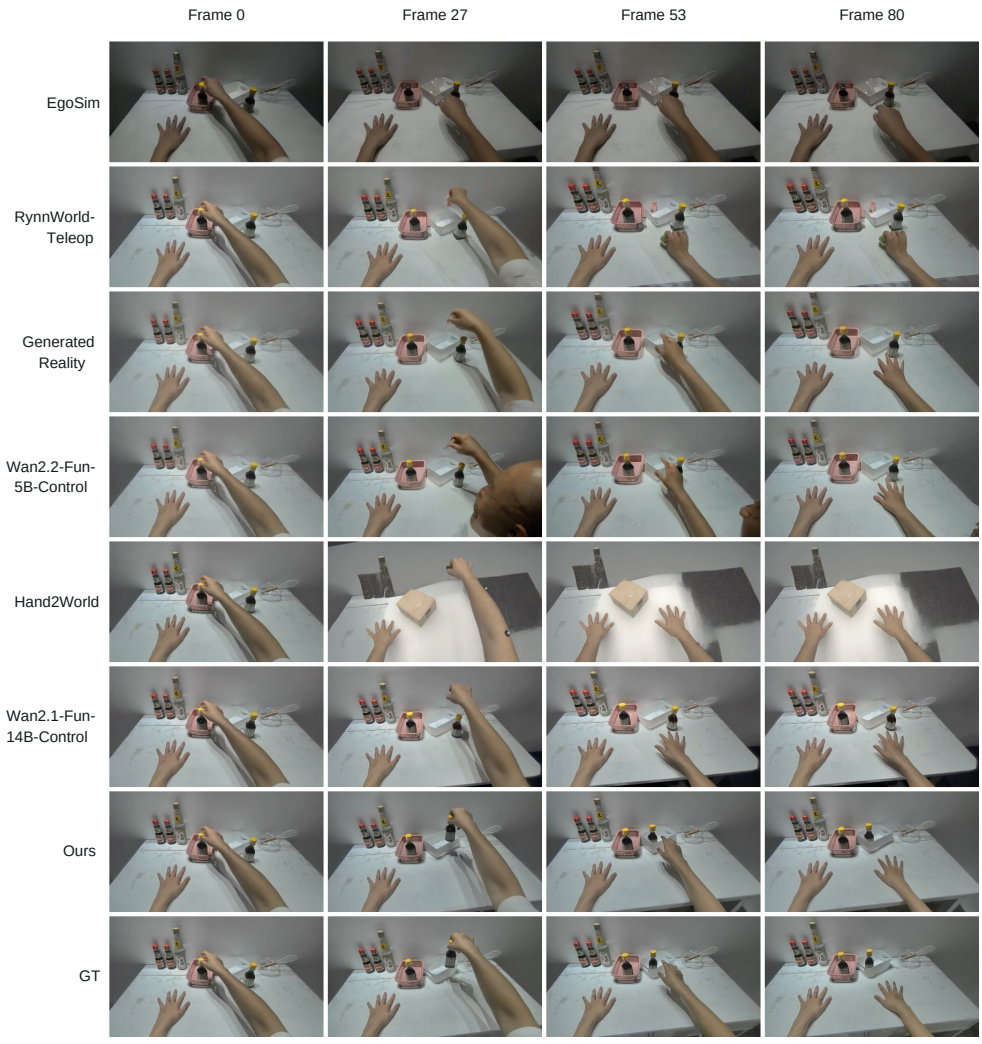

Figure 6: All manipulation baselines on a further KAI-Ego clip at four time steps.

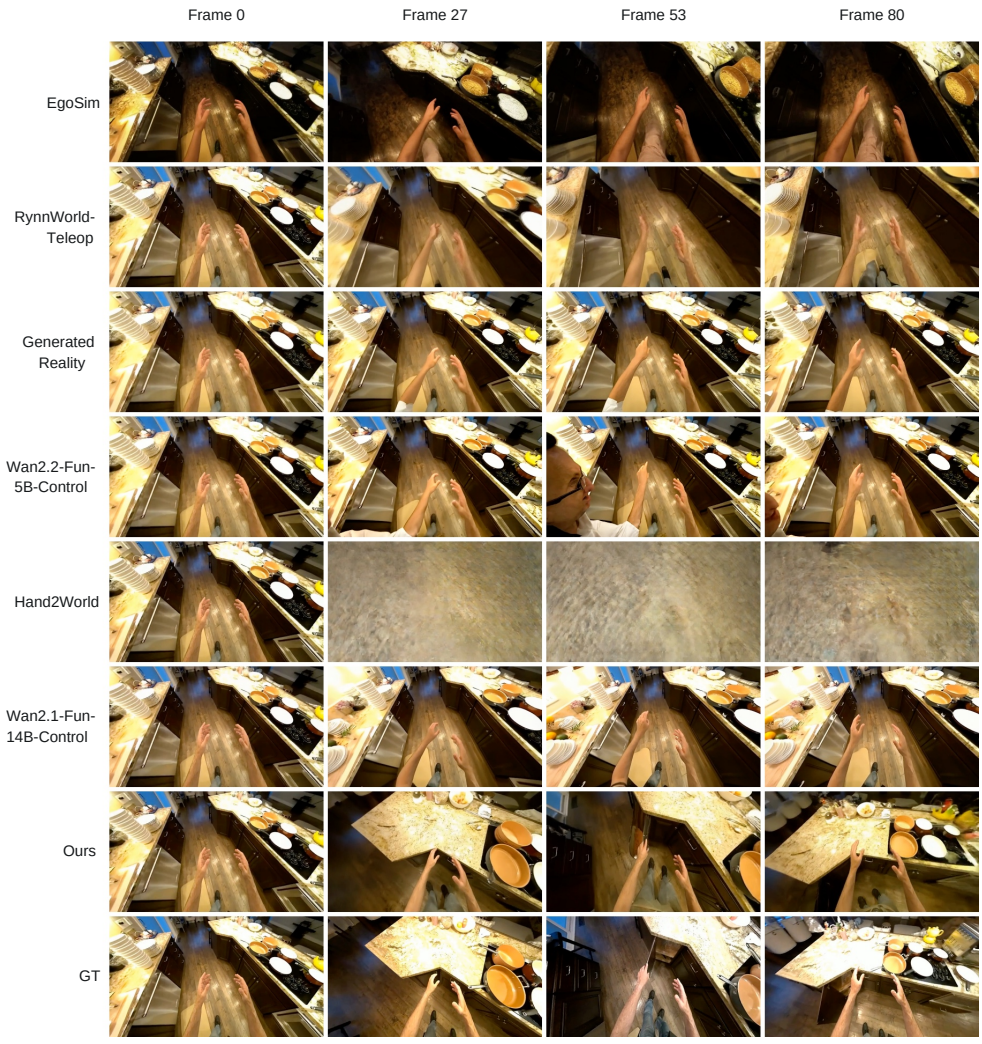

Figure 7: All manipulation baselines on the EgoVerse clip at four time steps.

F Discussion and Limitations

MM-World renders what the observer does and leaves the world's response to the generator, a division that lets one interface serve both actions but also defines the model's limits. The renders are only as accurate as the geometry behind them. The scene render inherits the errors of the estimated first-frame depth, which affect appearance more than camera control (Sec. 5.4); latent-space scene representations (Wang et al., 2026a) and joint generation of geometry and video (World Labs Team, 2026) would reduce reliance on external rendering and reconstruction, respectively. The hand render is aligned with the image rather than metrically with the scene (Sec. 3.2), so it conveys the hands' distance only through apparent size and leaves hand–scene occlusion to the generator; metric alignment would allow depth testing between the two renders. The world's response, in turn, is only as plausible as the training data allow: as with video generators in general (Motamed et al., 2026), its physical consistency is not guaranteed, and deformable objects and liquids remain challenging for MM-World. Our annotation pipeline requires only RGB video (Sec. 4.1), so the training data can be extended to such interactions.