PlayerOne: Egocentric World Simulator

Yuanpeng Tu[1,2] * **Hao Luo**[2,3] **Xi Chen**[1] **Xiang Bai**[4] **Fan Wang**[2] **Hengshuang Zhao**[1,†]
[1]HKU [2]DAMO Academy, Alibaba Group [3]Hupan Lab [4]HUST
https://playerone.github.io

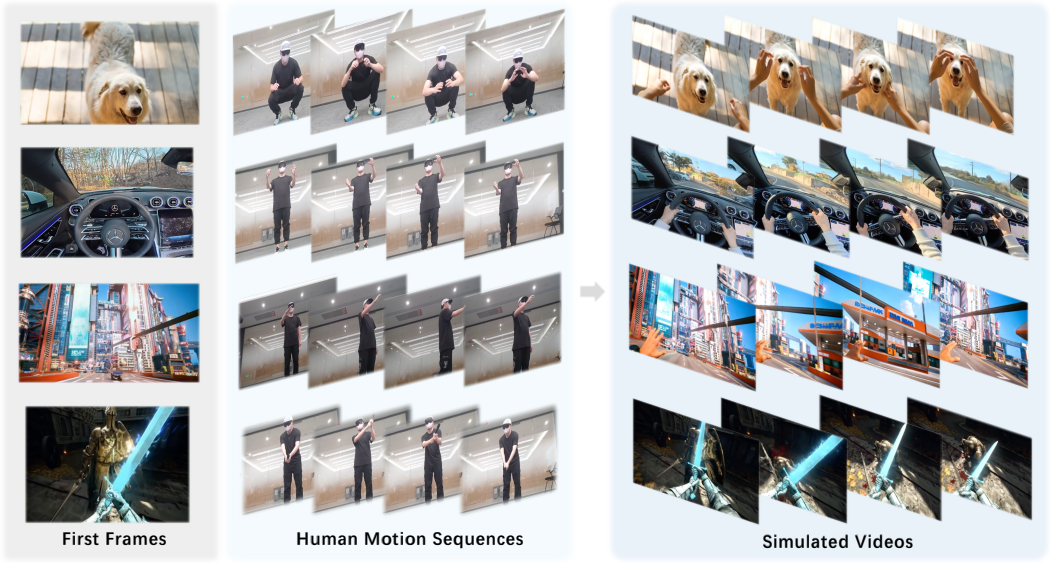

Figure 1: **Simulated videos of our PlayerOne**. Given an egocentric image as the scene to be explored, we can simulate egocentric immersive videos that are accurately aligned with the user's motion sequence captured by an exocentric camera. All the users have been anonymized and action videos are shot with the front camera.

Abstract

We introduce PlayerOne, the first egocentric realistic world simulator, facilitating immersive and unrestricted exploration within vividly dynamic environments. Given an egocentric scene image from the user, PlayerOne can accurately construct the corresponding world and generate egocentric videos that are strictly aligned with the real-scene human motion of the user captured by an exocentric camera. PlayerOne is trained in a coarse-to-fine pipeline that first performs pretraining on large-scale egocentric text-video pairs for coarse-level egocentric understanding, followed by finetuning on synchronous motion-video data extracted from egocentric-exocentric video datasets with our automatic construction pipeline. Besides, considering the varying importance of different components, we design a part-disentangled motion injection scheme, enabling precise control of part-level movements. In addition, we devise a joint reconstruction framework that progressively models both the 4D scene and video frames, ensuring scene consistency in the long-form video generation. Experimental results demonstrate its great generalization ability in precise control of varying human movements and world-consistent modeling of diverse scenarios. It marks the first endeavor into egocentric real-world simulation and can pave the way for the community to delve into fresh frontiers of world modeling and its diverse applications.

*Work during DAMO Academy internship. † Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

1 Introduction

World models [41, 33, 40, 14, 1, 39] have undergone extensive research due to their ability to model environmental dynamics and predict long-term outcomes. Recent breakthroughs in video diffusion models [35, 13, 20] have revolutionized this domain, enabling the synthesis of high-fidelity, action-conditioned simulations that forecast intricate future states. These advancements empower applications ranging from autonomous navigation in dynamic real-world environments to the creation of immersive, responsive virtual worlds in AAA game development. By bridging the gap between predictive modeling and interactive realism, world simulators are emerging as critical infrastructure for next-generation autonomous systems and game engines, particularly in scenarios requiring real-time adaptation to complex, evolving interactions.

Despite significant progress, this topic remains underexplored in existing research. Prior studies [10, 41, 7] predominantly focused on simulations within game-like environments, falling short of replicating realistic scenarios. Additionally, in their simulated environments, users are limited to performing predetermined actions (*i.e.*, directional movements). Operating within the confines of a constructed world restricts the execution of unrestricted movements as in real-world scenarios. While some initial efforts [23, 31, 1] have been made toward real-world simulation, they mainly contribute to world-consistent generation without human movement control. Consequently, users are reduced to passive spectators within the environment, rather than being active participants. This limitation significantly impacts the user experience, as it prevents the establishment of a genuine connection between the user and the simulated environment.

Faced with these challenges, we aim to design an egocentric world foundational framework that enables the user being a freeform adventurer. Given a user-provided egocentric image as the world to be explored, it can enable the user to perform unrestricted human movements real-time captured by an exocentric camera and consistent 4D scene modeling in the simulated world. Specifically, we propose the first realistic egocentric world simulator termed PlayerOne. Starting from a diffusion transformer (DiT) model [28], we first extract the latent of an egocentric user-input image. Meanwhile, we select the real-world human motions (*i.e.*, human pose or keypoints) as our motion representation. Considering the varying importance of different body parts in our task, the human motion sequence is partitioned into three groups (*i.e.*, head, hands, feet and body) and fed into our part-disentangled motion injection to generate latents that can enable precise part-wise control. Additionally, we developed a joint scene-frame reconstruction framework that can progressively complete scene point maps during the video generation process to enable scene-consistent generation. The DiT model takes the concatenation of the first frame latent, motion latent, video latent, and the point map latent as input and conducts noising and denoising on both the video and point map latent. Notably, the point map sequence is not required during inference, ensuring practical efficiency. Moreover, to overcome the absence of publicly available datasets, we curate required motion-video pairs from existing egocentric-exocentric datasets using an automated pipeline designed to filter and retain high-quality data. A coarse-to-fine training strategy is also designed to compensate for the data scarcity. The base model is fine-tuned on large-scale egocentric text-video data for coarse-level generation, then refined on our curated dataset to achieve precise motion control and scene modeling. Finally, we distill our trained model [43] to achieve real-time generation. By integrating these innovations, PlayerOne advances the field of dynamic world modeling. Our contributions are summarized as follows:

- We introduce PlayerOne, the first egocentric foundational simulator for realistic worlds, capable of generating video streams with precise control of highly free human motions and world consistency in real-time and exhibiting strong generalization in diverse scenarios.
- We design a novel part-disentangled motion injection scheme to enhance fine-grained motion alignment, where a joint scene-frame reconstruction framework is introduced to guarantee world-consistent modeling in long-term video generation as well.
- We construct an effective automatic dataset construction pipeline to extract high-quality motion-video pairs from existing egocentric-exocentric datasets, where a coarse-to-fine training scheme is also introduced to compensate for the data scarcity.

2 Related Work

Video generation. The rapid development of diffusion models [32, 16, 47, 3] has driven substantial advancements in video generation. Early researchers [11, 5] adapted existing text-to-image models to

enable text-to-video generation to compensate for the limited availability of high-quality video-text datasets. Subsequently, diffusion transformers based frameworks [42, 28, 20, 13, 35, 34] are proposed. When scaling-up training, they enable more highly realistic and temporally coherent generation results. Among them, HunyuanVideo [20] substitutes T5 with a Multimodal Large Language Model. LTX-Video [13] modifies the VAE decoder to handle the final denoising step and convert latents into pixels. Wan [35] introduces a full spatial-temporal attention to ensure computational efficiency.

World models. Existing world models [14, 40, 41, 7, 1, 39] can be roughly divided into two categories: 1) Agent learning targeted models, 2) World simulation models. For the former, they [14, 40, 39] aim at enhancing policy learning within simulated environments. Among them, Dreamer [14] and DayDreamer [40] solve long-horizon tasks from images purely by latent imagination. MuZero [39] runs the self-play of Monte Carlo tree search to build world models for Atari. Distinct from this direction, world simulation aims to model an environment by predicting the next state given the current state and action. These works focus on human interaction with neural networks through high-quality rendering, robust control, and strong domain generalization to real-world scenarios. With advances in video generation, high-quality world simulation with robust control has become feasible, leading to numerous works focusing on interactive world simulation [10, 31, 23, 41, 7, 1, 33, 6]. Among these works, WORLDMEM [41]. The Matrix [7] proposes the first world simulator capable of generating infinitely long real-scene video streams with real-time, responsive control. Matrix-Game [49] redefines video generation as an interactive process of exploration and creation. Cosmos [1] presents a general-purpose world model and a pre-training-then-post-training scheme. Aether [33] designs a unified framework with synergistic knowledge sharing across reconstruction, prediction, and planning objectives. However, these methods primarily focus on virtual game scenarios and are limited to specific directional actions, rather than facilitating high-degree-of-freedom motion control in real-world environments. To address these limitations, we target at developing a human motion driven realistic world simulator. Given an egocentric image, we can construct a real-scene world that immerses users as freeform adventurers with precise and unrestricted human motion control.

3 Method

In this section, we detail the methodology of PlayerOne. Sec. 3.1 introduces the relevant preliminaries and the overall pipeline. Sec. 3.2 presents the core of our proposed model, followed by Sec. 3.3 introducing our dataset construction and training strategy.

3.1 Overview

Video diffusion models [28, 17] consist of two key processes: a forward (noising) process and a reverse (denoising) process. The forward process gradually adds Gaussian noise, denoted as $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, to a clean latent sample $z_0 \in \mathbb{R}^{k \times c \times h \times w}$, where k, c, h, and w represent the dimensions of the video latents. This transforms z_0 into a noisy latent z_t. In the reverse process, a learned denoising model ϵ_θ progressively removes the noise from z_t to reconstruct the original latent representation.

As shown in Fig. 2, our method comprises two core modules: Part-disentangled Motion Injection (PMI) and Scene-frame Reconstruction (SR). In PMI, we use real-scene human motion as the motion condition to enable free action control for the user. The first frame is converted into z_{frame} via a 3D VAE encoder. The human motion sequence is split into three parts based on varying importance, and each part is fed into a 3D motion encoder to obtain latents. These motion latents are concatenated into $z_{motion} \in \mathcal{R}^{k \times 3 \times h \times w}$. To improve view alignment, we transform the head parameters of the human motion sequence into a camera sequence, which is then fed into a camera encoder. The output is added to the noised video latents z_{video} to inject view-change signals. In SR, we jointly reconstruct video frames and 4D scenes to ensure world-consistent generation in the context of long video generation. We render a point map sequence from the ground truth video and feed it into a point map encoder with an adapter to obtain $z_{point} \in \mathcal{R}^{k \times 64 \times h \times w}$. Finally, all latents and conditions are concatenated channel-wise. The training objective of our method can be expressed as:

$$\mathcal{L} = \mathbb{E}_{t \sim \mathcal{U}(0,1), \epsilon \sim \mathcal{N}(0,\mathbf{I})} \left[\left\| \epsilon - \epsilon_\theta \left(\mathbf{z}_t, t \right) \right\|_2^2 \right], \quad \text{where} \quad z_t = \alpha_t z_0 + \delta_t \epsilon \tag{1}$$

Where $t = 1, \ldots, T$, ${\alpha_t}^2 + {\delta_t}^2 = 1$. Since we only add noise to point map latents and video latents, thus $z_0 = z_{video} \otimes z_{point}$. $\otimes$ denotes the channel-wise concatenation operation, $\mathcal{U}(\cdot)$ represents a uniform distribution, and T denotes the denoising steps.

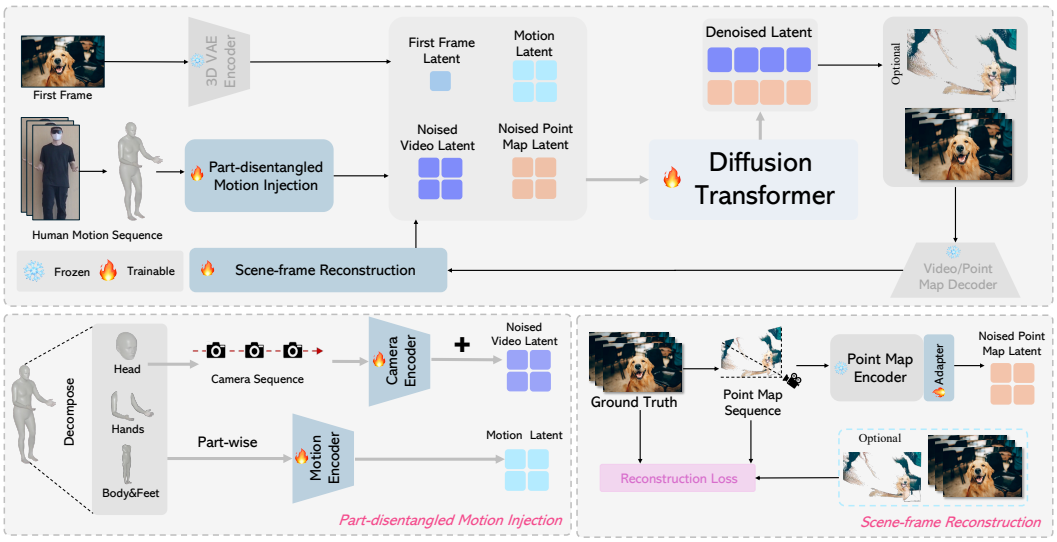

Figure 2: **Overall framework of our PlayerOne**. It begins by converting the egocentric first frame into visual tokens. The human motion sequence is split into groups and fed into the motion encoders respectively to generate part-wise motion latents, with the head parameters converted into a rotation-only camera sequence. This camera sequence is then encoded via a camera encoder, and its output is injected into noised video latents to improve view-change alignment. Next, we render a 4D scene point map sequence with the ground truth video, which is then processed by a point map encoder with an adapter to produce scene latents. Then we input the concatenation of these latents into the DiT Model and perform noising and denoising on both the video and scene latents to ensure world-consistent generation. Finally, the denoised latents are decoded by VAE decoders to produce the final results. Note that only the first frame and the human motion sequence are needed for inference.

3.2 Model Components

Part-disentangled motion injection. Prior studies [23, 31, 7, 49] typically utilize camera trajectories as motion conditions or are constrained to specific directional movements. These restrictions confine users to passive "observer" roles, preventing meaningful user interaction. In contrast, our approach empowers users to become active "participants" by adopting real-world human motion sequences (*i.e.*, human pose or keypoints) as motion conditions, allowing for more natural and unrestricted movement. However, our empirical analysis reveals that extracting latent representations holistically from human motion parameters complicates precise motion alignment. To address this challenge, we introduce a part-disentangled motion injection strategy that recognizes the distinct roles of various body parts. Specifically, hand movements are essential for interacting with objects in the environment, while the head plays a crucial role in maintaining egocentric perspective alignment. Accordingly, we categorize the human motion parameters into three groups: body and feet, hands, and head. Each group is processed through its own dedicated motion encoder, comprising eight layers of 3D convolutional networks, to extract the relevant latent features. This specialized processing ensures accurate and synchronized motion alignment. These latents are subsequently concatenated along the channel dimension to form the final part-aware motion latent representation $z_{motion} \in \mathcal{R}^{k\times 3\times h\times w}$.

To further enhance the egocentric view alignment, we solely transform the head parameters of the human motion sequence into a sequence of camera extrinsics with only rotation values. We zero out the translation values in the camera extrinsics, assuming the head parameters are at the camera coordinate system's origin. Specifically, suppose the head parameter $\mathbf{v} = (\theta_x, \theta_y, \theta_z)$, we first normalize the rotation axis as follows:

$$\mathbf{u} = \frac{\mathbf{v}}{\|\mathbf{v}\|}, \quad \theta = \|\mathbf{v}\| \tag{2}$$

Then we construct the rotation matrix as follows:

$$\mathbf{R} = \mathbf{I} + \sin\theta \cdot [\mathbf{u}]_\times + (1 - \cos\theta) \cdot [\mathbf{u}]_\times^2 \tag{3}$$

Where $\mathbf{u}_\times$ is the cross product matrix of $\mathbf{u}$, which can be denoted as follows:

$$[\mathbf{u}]_\times = \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{bmatrix} \tag{4}$$

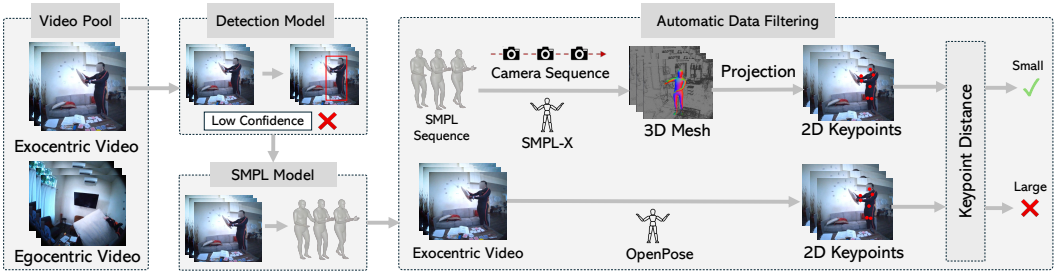

Figure 3: **The overall pipeline of the dataset construction**. By seamlessly integrating detection and human pose estimation models, we can extract motion-video pairs from existing egocentric-exocentric video datasets while retaining high-quality data through our automatic filtering scheme.

Then we use Plücker ray [45] to parameterize the camera extrinsics and then feed the output to an extra camera encoder, which shares a similar structure with the motion encoder. Then the latents from this encoder are added to the noised video latents to inject the view-change information.

Scene-frame reconstruction. While PMI enables precise control over egocentric perspective and motion, it does not guarantee scene consistency within the generated world. To address this limitation, we introduce a joint reconstruction framework that simultaneously models the 4D scene and video frames, ensuring scene coherence and continuity throughout the video. Specifically, it begins by employing CUT3R [36] to generate a point map for each frame based on ground truth video data, reconstructing the n-th frame's point map using information from frames 1 through n. These point maps are then compressed into latent representations using a specialized point map encoder [19]. To integrate these latents with video features, we implement an adapter composed of five 3D convolutional layers. This adapter aligns the point map latents with video latents and projects them into a shared latent space, facilitating seamless integration of motion and environmental data. Finally, we concatenate the latent representations from the first frame, the human motion sequence, the noised video latents, and corresponding noised point map latents. This comprehensive input is then fed into a diffusion transformer for denoising, resulting in a coherent and visually consistent world. Importantly, point maps are only required during the training phase. *During inference, the system simplifies the process by utilizing only the first frame and the corresponding human motion sequence to generate world-consistent videos.* This streamlined approach enhances generation efficiency while ensuring that the resulting environment remains stable and realistic throughout the entire video.

3.3 Training Strategy

Dataset preparation. The ideal training samples for our task are egocentric videos paired with corresponding motion sequences. However, no such dataset currently exists in publicly available repositories. As a substitute, we derive these data pairs from existing egocentric-exocentric video datasets through an automatic pipeline. Specifically, for each synchronized egocentric-exocentric video pair, we first employ SAM2 [30] to detect the largest person in the exocentric view. The background-removed exocentric video is then processed using SMPLest-X [44] to extract the SMPL parameters of the identified individual as the human motion. To enhance optimization stability, an L2 regularization prior is incorporated. We then evaluate the 2D reprojection consistency to filter out low-quality SMPL data. This involves generating a 3D mesh from the SMPL parameters using SMPLX [27], projecting the 3D joints onto the 2D image plane with the corresponding camera parameters, and extracting 2D key points via OpenPose [4]. The reprojection error is calculated by measuring the distance between the SMPL-projected 2D key points and those detected by OpenPose. Data pairs with reprojection errors in the top 10% are excluded, ensuring a final dataset of high-quality motion-video pairs. The refined SMPL parameters are decomposed into body and feet (66 dimensions), head orientation (3 dimensions), and hand articulation (45 dimensions per hand) components for each frame. These components are fed into their respective motion encoders. The dataset construction pipeline is illustrated in Fig. 3. As detailed in Tab. 1, our training dataset combines multiple publicly available datasets to ensure comprehensive coverage of diverse environmental contexts, action types, and intensity levels, thereby enhancing model generalization.

Coarse-to-fine training. Though we can extract high-quality motion-video training data with our automatic pipeline, the limited scale of this dataset is insufficient for training video generation models to produce high-quality egocentric videos. To address this, we harness the extensive egocentric text-video datasets (*i.e.*, Egovid-5M [38]). Specifically, we first fine-tune the baseline model using

Table 1: **Statistics of datasets** used for training our PlayerOne. "quality" particularly refer to the image resolution. "Ego-Exo" denotes whether the dataset contains egocentric-exocentric video pairs.

Dataset	EgoExo-4D [9]	Nymeria [24]	FT-HID [12]	EgoExo-Fitness [22]	Egovid-5M [38]
Size	740	1,200	38,364	1,276	5M
Resolution	1080p	1408p	1080p	1080p	1080p
Ego-Exo	✓	✓	✓	✓	×

Figure 4: **Investigation on coarse-to-fine training**. "Joint-Train" and "No Pretrain" denote training with both motion-video pairs and large-scale egocentric videos in a one-stage manner and training with only motion-video pairs respectively. The Wanx2.1 1.3B is adopted as the baseline.

LoRA on large-scale egocentric text-video data pairs, enabling egocentric video generation with coarse-level motion alignment. Then we freeze the trained LoRA and fine-tune the last six blocks of the model with our constructed high-quality dataset to enhance fine-grained human motion alignment and view-invariant scene modeling, which can effectively address the scarcity of pair-wise data. Finally, we adopt an asymmetric distillation strategy that supervises a causal student model with a bidirectional teacher [43] to achieve real-time generation and long-duration video synthesis.

4 Experiments

4.1 Experimental Setting

Implementation details. We choose Wanx2.1 1.3B [35] as the base generator. We set the LoRA rank and the update weight of the matrices as 128 and 4 respectively and initialize its weight following [35]. The inference step and the learning rate are set as 50 and 1×10^{-5} respectively, where the Adam optimizer and mixed-precision bf16 are adopted. The cfg of 7.5 is used. We train our model for 100,000 steps on 8 NVIDIA A100 GPUs with a batch size of 56 and sample resolution of 480×480. The generated video runs at eight frames per second, and we utilize 49 video frames (6 seconds) for training. After distillation, our method can achieve 8 FPS to generate the desired results. *All the action videos in this paper are shot with the front camera.*

Benchmark. Since there is no publicly available benchmark for our task, we construct a benchmark with 100 videos collected from Nymeria [25] dataset, which is not included for training. It consists of coarse-level motion descriptions for each sample and covers diverse realistic scenarios. Considering the information gap between the human motion sequence and the text, we further use Qwen2.5-VL [2] to enrich the caption to generate videos for the competitors for more fair comparisons.

Metrics. On our constructed benchmark, for evaluation of alignment with the given text descriptions, we calculate both CLIP-Score and DINO-Score, where PSNR, SSIM, and LPIPS [48] are employed to evaluate the video fidelity of the generated video. Besides, we calculate the frame consistency to evaluate the temporal coherence and consistency of the generated video frames over time. We further utilize a 3D hand pose estimation model [29] to estimate the hand pose of the generated videos and use the results of the ground truth video as the labels. Afterward, we follow [29] to calculate two metrics: (1) Mean Per-Joint Position Error (MPJPE): the L2 distance between the predicted and ground truth joints for each hand after subtracting the root joint. (2) Mean Relative-Root Position Error (MRRPE): the metric distance between the root joints of the left hand and right hand.

Figure 5: **Investigation on part-disentangled motion injection**. "ControlNet" denotes injecting motion latents with a ControlNet [47]. "Entangled" and "No Cam" denote inputting the whole motion sequence into a motion encoder without dividing into groups and removing the camera encoder respectively.

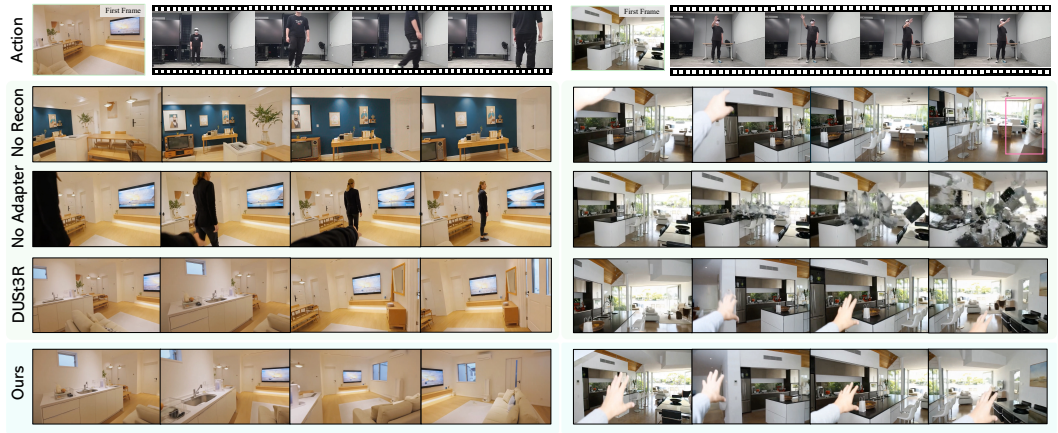

Figure 6: **Investigation on scene-frame reconstruction**. "No Recon"/"No Adapter" denote training without reconstruction/the adapter. "DUStR" is replacing CUT3R with DUStR for point map rendering.

4.2 Ablation Study

Investigation on coarse-to-fine training. We first evaluate several variants of our coarse-to-fine training scheme, as depicted in Fig. 4. Specifically, when inputting action descriptions into the baseline model without fine-tuning, the generated results exhibit noticeable flaws, such as hand distortions or the unexpected appearance of individuals. Similar issues can be observed when training with only motion-video pairs. We also explore jointly training with both large-scale egocentric videos and motion-video pairs. Specifically, when inputting egocentric videos, we set the motion latent values to zero and extract the latents of the text description to serve as the motion condition, where a balanced-sampling strategy is used as well. Despite this variant being capable of generating egocentric videos, it fails to produce results accurately aligned with the given human motion conditions. In contrast, our coarse-to-fine training scheme delivers much better outcomes compared to these variants.

Investigation on part-disentangled motion injection. Next, we conduct a detailed analysis of our PMI module. Specifically, three variants are included: ControlNet-based [47] motion injection, inputting motion sequences as a unified entity (the "Entangled" scheme), and removing our camera encoder. As shown in Fig. 5, the ControlNet-based scheme suffers from information loss, preventing it from producing results that accurately align with the specified motion conditions. Similarly, the entangled scheme demonstrates comparable shortcomings. Furthermore, removing the camera encoder leads to the model's inability to generate view-accurate alignments. As depicted in Fig. 5, this variant fails to produce the corresponding perspective change associated with crouching. Ultimately, our PMI module successfully generates outcomes that are both view-aligned and action-aligned.

Investigation on scene-frame reconstruction. Additionally, we conducted a detailed analysis of the SR module, exploring three variants: omitting reconstruction, removing the adapter within the SR

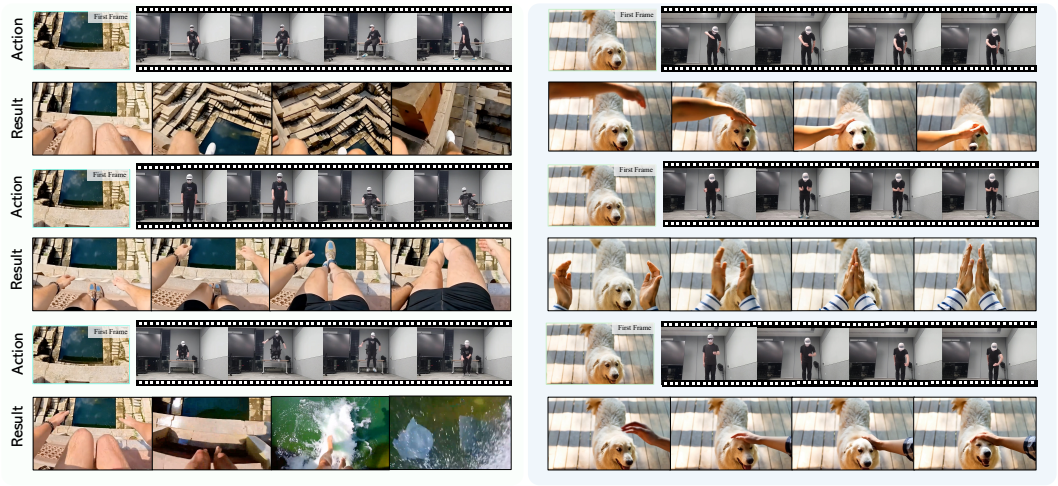

Figure 7: **Qualitative evaluation on the motion alignment**. We generate simulated videos based on the same first frame but different motion sequences. Results show that we can achieve accurate motion alignment.

Table 2: **Quantitative evaluation** on the components of PlayerOne. PlayerOne outperforms all these variants. "No Camera"/"Filtering" denote training without/with the camera encoder/data filtering.

	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR(↑)	FVD (↓)	LPIPS(↓)
Baseline	51.3	65.6	376.14	341.01	35.6	394.16	0.1421
+ Pretrain	56.6	74.4	258.05	232.17	41.2	301.32	0.1146
+ Pretrain&ControlNet	57.1	75.2	241.73	218.46	42.8	287.52	0.1103
+ Pretrain&Entangled	58.0	76.3	235.12	212.53	43.9	279.41	0.1060
+ Pretrain&PMI (No Camera)	60.7	79.8	183.25	196.35	45.6	257.04	0.0902
+ Pretrain&PMI	62.5	81.3	156.76	175.18	48.3	245.72	0.0839
+ Pretrain&PMI&Filtering	64.2	83.8	141.56	163.04	49.1	230.50	0.0782
+ Pretrain&PMI&Filtering&Recon(No Adapter)	62.7	81.6	176.23	180.10	47.3	240.17	0.0919
+ Pretrain&PMI&Filtering&Recon(DUSt3R)	67.5	87.7	129.08	152.22	52.2	228.20	0.0685
PlayerOne(ours)	**67.8**	**88.2**	**127.16**	**151.62**	**52.6**	**226.12**	**0.0663**

Table 3: **Quantitative comparison** between our PlayerOne and other works. Seven metrics are employed for the evaluation. PlayerOne outperforms these methods across all the metrics.

	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR(↑)	FVD (↓)	LPIPS(↓)
Aether [33]	38.0	64.2	415.70	431.05	38.1	397.40	0.1856
Cosmos(Diff-7B) [1]	45.3	70.3	301.92	324.12	43.7	346.09	0.1630
Cosmos(Diff-14B) [1]	51.6	79.7	256.73	253.06	47.8	302.17	0.1351
PlayerOne(ours)	**67.8**	**88.2**	**127.16**	**151.62**	**52.6**	**226.12**	**0.0663**

module, and substituting CUT3R [36] with DUStR [37] for point map rendering. As illustrated in Fig. 6, the absence of reconstruction results in the model's inability to generate consistently simulated results. Moreover, due to the distribution gap between latents of frames and point maps, training without the adapter leads to difficulty in loss convergence, causing noticeable distortions. Furthermore, after replacing CUT3R [36] with DUStR [37], our PlayerOne can also produce scene-consistent outputs, demonstrating its robustness to different point map rendering techniques.

Motion alignment. To verify the alignment capability with the given motion condition, we conduct experiments by generating world-simulated videos with the same first frame but different human motion sequences. Fig. 7 shows that our PlayerOne can accurately generate corresponding results according to different conditions and produce reasonable interactive changes.

Quantitative comparisons. We provide quantitative results on the core components of our PlayerOne in Tab. 2. All numerical results concur with the visualization outcomes. A significant performance improvement is observed when the model undergoes pre-training on large-scale egocentric text-video datasets. The introduction of PMI yields an additional accuracy boost, and it outperforms all of its variants. In addition, our designed filtering strategy maximizes performance as well by filtering noisy motion-video pairs. After removing the adapter, the performance suffers from a notable degradation due to the distribution gap between latents of video frames and point maps. By introducing our joint scene-frame reconstruction scheme, we achieve superior results across all metrics.

Figure 8: **Qualitative comparisons between our method and other competitors**. Our PlayerOne can achieve the best performance on both the motion alignment, video quality.

Table 4: **User study** on our PlayerOne and existing alternatives. "Quality", "Fidelity", "Smooth", and "Alignment" measure synthesis quality, object identity preservation, motion consistency, and alignment with the text descriptions, respectively. Each metric is rated from 1 (worst) to 4 (best).

	Quality (↑)	Fidelity (↑)	Smooth (↑)	Alignment (↑)
Aether [33]	1.32	1.30	1.31	1.34
Cosmos(Diff-7B) [1]	2.07	2.13	2.05	2.09
Cosmos(Diff-14B) [1]	3.02	2.94	2.98	2.71
PlayerOne(ours)	**3.59**	**3.63**	**3.65**	**3.86**

4.3 Comparison with State-of-the-arts

Quantitative comparison. Since there is no method sharing the same setting as ours, we selected two potential competitors for comparison: Cosmos [1] and Aether [33]. As shown in Tab. 3, our PlayerOne outperforms all the baselines by large margins, especially on the metrics of motion alignment. Notably, Cosmos [1] exhibits better generalization ability than Aether [33] by explicitly capturing general knowledge of real-world physics and natural behaviors. Besides qualitative results, we provide visualization comparisons in Fig. 8, where consistent superiority can be observed in diverse scenarios for both user interaction and world modeling.

User study. In Tab. 4, we report the comparison results of human preference rates. We let 20 annotators rate 25 groups of videos, where each group contains the generated video of each method and text description. And we provide detailed regulations to rate the results for scores of 1-4 from four views: "Quality", "Smooth", "Fidelity", "Alignment". "Quality" counts for whether the result is harmonized without considering fidelity. "Smooth" assesses the motion consistency across the video. "Fidelity" measures ID preservation and distortions within the video, while we use "Alignment" to measure the alignment with the given text descriptions. It can be noted that our model demonstrates significant superiority across all the metrics, especially for "Alignment", and "Smooth".

5 Conclusion

In conclusion, PlayerOne represents a significant advancement in interactive and realistic world modeling for video generation. Unlike conventional models that are restricted to particular game scenarios or actions, our PlayerOne can capture the complex dynamics of general-world environments and enable free motion control within the simulated world. By formulating world modeling as a joint process of videos and 4D scenes, our PlayerOne ensures coherent world generation and enhances motion and view alignment with the given conditions through part-disentangled motion injection. Experimental results demonstrate our superior performance across diverse scenarios.

Limitations. Despite the compelling outcomes, our performance in game scenarios is slightly inferior to realistic ones, likely due to the imbalanced distribution between realistic and game training data. It can be addressed by incorporating more game-scenario datasets in future research.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (No. 62441615, 62422606, 62201484) and DAMO Academy via DAMO Academy Research Intern Program.

References

[1] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv:2501.03575*, 2025. 2, 3, 8, 9, 21, 31

[2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv:2309.16609*, 2023. 6

[3] Andreas Blattmann, Robin Rombach, Kaan Oktay, and Björn Ommer. Retrieval-augmented diffusion models. In *NeurIPS*, 2022. 2

[4] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *TPAMI*, 2019. 5

[5] Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models. *arXiv:2305.13840*, 2023. 2

[6] Junhao Cheng, Yuying Ge, Yixiao Ge, Jing Liao, and Ying Shan. Animegamer: Infinite anime life simulation with next game state prediction. *arXiv:2504.01014*, 2025. 3

[7] Ruili Feng, Han Zhang, Zhantao Yang, Jie Xiao, Zhilei Shu, Zhiheng Liu, Andy Zheng, Yukun Huang, Yu Liu, and Hongyang Zhang. The matrix: Infinite-horizon world generation with real-time moving control. *arXiv:2412.03568*, 2024. 2, 3, 4, 21

[8] Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability, 2024. 25

[9] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *CVPR*, 2024. 6, 31

[10] Junliang Guo, Yang Ye, Tianyu He, Haoyu Wu, Yushu Jiang, Tim Pearce, and Jiang Bian. Mineworld: a real-time and open-source interactive world model on minecraft. *arXiv:2504.08388*, 2025. 2, 3, 21

[11] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. In *ICLR*, 2024. 2

[12] Zihui Guo, Yonghong Hou Hou, Pichao Wang, Zhimin Gao, Mingliang Xu, and Wanqing Li. Ft-hid: A large scale rgb-d dataset for first and third person human interaction analysis. *Neural Computing and Applications*, 2022. 6, 31

[13] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. Ltx-video: Realtime video latent diffusion. *arXiv:2501.00103*, 2024. 2, 3

[14] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv:1912.01603*, 2019. 2, 3

[15] Mariam Hassan, Sebastian Stapf, Ahmad Rahimi, Pedro M B Rezende, Yasaman Haghighi, David Brüggemann, Isinsu Katircioglu, Lin Zhang, Xiaoran Chen, Suman Saha, Marco Cannici, Elie Aljalbout, Botao Ye, Xi Wang, Aram Davtyan, Mathieu Salzmann, Davide Scaramuzza, Marc Pollefeys, Paolo Favaro, and Alexandre Alahi. Gem: A generalizable ego-vision multimodal world model for fine-grained ego-motion, object dynamics, and scene composition control, 2024. 25

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arxiv:2006.11239*, 2020. 2

[17] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. *arXiv*, 2022. 3

[18] Li Hu, Xin Gao, Peng Zhang, Ke Sun, Bang Zhang, and Liefeng Bo. Animate anyone: Consistent and controllable image-to-video synthesis for character animation, 2024. 26

[19] Zeren Jiang, Chuanxia Zheng, Iro Laina, Diane Larlus, and Andrea Vedaldi. Geo4d: Leveraging video generators for geometric 4d scene reconstruction. *arXiv:2504.07961*, 2025. 5

[20] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv:2412.03603*, 2024. 2, 3

[21] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with mast3r, 2024. 29, 31

[22] Yuan-Ming Li, Wei-Jin Huang, An-Lan Wang, Ling-An Zeng, Jing-Ke Meng, and Wei-Shi Zheng. Egoexo-fitness: towards egocentric and exocentric full-body action understanding. In *ECCV*, 2024. 6, 31

[23] Hanwen Liang, Junli Cao, Vidit Goel, Guocheng Qian, Sergei Korolev, Demetri Terzopoulos, Konstantinos N. Plataniotis, Sergey Tulyakov, and Jian Ren. Wonderland: Navigating 3d scenes from a single image. *arXiv:2412.12091*, 2024. 2, 3, 4, 21

[24] Lingni Ma, Yuting Ye, Fangzhou Hong, Vladimir Guzov, Yifeng Jiang, Rowan Postyeni, Luis Pesqueira, Alexander Gamino, Vijay Baiyya, Hyo Jin Kim, Kevin Bailey, David Soriano Fosas, C. Karen Liu, Ziwei Liu, Jakob Engel, Renzo De Nardi, and Richard Newcombe. Nymeria: A massive collection of multimodal egocentric daily motion in the wild. In *ECCV*, 2024. 6

[25] Lingni Ma, Yuting Ye, Fangzhou Hong, Vladimir Guzov, Yifeng Jiang, Rowan Postyeni, Luis Pesqueira, Alexander Gamino, Vijay Baiyya, Hyo Jin Kim, et al. Nymeria: A massive collection of multimodal egocentric daily motion in the wild. In *ECCV*, 2024. 6, 31

[26] Ze Ma, Daquan Zhou, Chun-Hsiao Yeh, Xue-She Wang, Xiuyu Li, Huanrui Yang, Zhen Dong, Kurt Keutzer, and Jiashi Feng. Magic-me: Identity-specific video customized diffusion, 2024. 26

[27] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, 2019. 5

[28] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv:2212.09748*, 2022. 2, 3

[29] Aditya Prakash, Ruisen Tu, Matthew Chang, and Saurabh Gupta. 3d hand pose estimation in everyday egocentric images. *arXiv:2312.06583*, 2024. 6

[30] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. In *ICLR*, 2025. 5

[31] 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. *arXiv:2503.03751*, 2025. 2, 3, 4, 21

[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *arXiv:2112.10752*, 2021. 2

[33] Aether Team, Haoyi Zhu, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Chunhua Shen, Jiangmiao Pang, and Tong He. Aether: Geometric-aware unified world modeling. *arXiv:2503.18945*, 2025. 2, 3, 8, 9, 31

[34] Genmo Team. Mochi 1. https://github.com/genmoai/models, 2024. 3

[35] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv:2503.20314*, 2025. 2, 3, 6, 31

[36] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. *arXiv:2501.12387*, 2025. 5, 8, 31

[37] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 8, 29

[38] Xiaofeng Wang, Kang Zhao, Feng Liu, Jiayu Wang, Guosheng Zhao, Xiaoyi Bao, Zheng Zhu, Yingya Zhang, and Xingang Wang. Egovid-5m: A large-scale video-action dataset for egocentric video generation. *arXiv:2411.08380*, 2024. 5, 6, 31

[39] Aurèle Hainaut Werner Duvaud. Muzero general: Open reimplementation of muzero. https://github.com/werner-duvaud/muzero-general, 2019. 2, 3

[40] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Ken Goldberg, and Pieter Abbeel. Daydreamer: World models for physical robot learning. In *CoRL*, 2022. 2, 3

[41] Zeqi Xiao, Yushi Lan, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. Worldmem: Long-term consistent world simulation with memory. *arXiv:2504.12369*, 2025. 2, 3, 21

[42] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformer. *arXiv:2410.10629*, 2024. 3

[43] Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *CVPR*, 2025. 2, 6

[44] Wanqi Yin, Zhongang Cai, Ruisi Wang, Ailing Zeng, Chen Wei, Qingping Sun, Haiyi Mei, Yanjun Wang, Hui En Pang, Mingyuan Zhang, Lei Zhang, Chen Change Loy, Atsushi Yamashita, Lei Yang, and Ziwei Liu. Smplest-x: Ultimate scaling for expressive human pose and shape estimation. *arXiv:2501.09782*, 2025. 5

[45] Jason Y Zhang, Amy Lin, Moneish Kumar, Tzu-Hsuan Yang, Deva Ramanan, and Shubham Tulsiani. Cameras as rays: Pose estimation via ray diffusion. In *ICLR*, 2024. 5

[46] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arxiv:2410.03825*, 2024. 29, 31

[47] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 2, 7

[48] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *arXiv:1801.03924*, 2018. 6

[49] Yifan Zhang, Chunli Peng, Boyang Wang, Puyi Wang, Qingcheng Zhu, Zedong Gao, Eric Li, Yang Liu, and Yahui Zhou. Matrix-game: Interactive world foundation model. *arXiv*, 2025. 3, 4

[50] Yuang Zhang, Jiaxi Gu, Li-Wen Wang, Han Wang, Junqi Cheng, Yuefeng Zhu, and Fangyuan Zou. Mimicmotion: High-quality human motion video generation with confidence-aware pose guidance, 2025. 26

[51] Jingkai Zhou, Benzhi Wang, Weihua Chen, Jingqi Bai, Dongyang Li, Aixi Zhang, Hao Xu, Mingyang Yang, and Fan Wang. Realisdance: Equip controllable character animation with realistic hands, 2024. 26

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to the contributions in Section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to the limitations in Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: Please refer to the assumptions and experimental results in Section 3 and Section 4.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please refer to the implementation details in Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Please refer to the supplementary material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to the implementation details in Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Please refer to the experimental results in Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please refer to the implementation details in Section 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Yes, it conforms to the requirements.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to the supplementary material.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The scope of our research does not involve data or models with a high risk for misuse. We utilize publicly available, non-sensitive datasets and models that do not require special safeguards for responsible release.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Please refer to the supplementary material.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: This question does not apply to our research as no new assets were introduced. Our study solely relies on existing, publicly available resources, and thus, no additional documentation for new assets is necessary.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This question does not apply to our research as it does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This question is not applicable to our research as no human subjects were involved.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: We only use the LLM for writing, editing, or formatting purposes.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

Appendix

Appendix Overview

This appendix complements the main paper with additional results, analyses, and implementation details:

- **Visualizations and reconstructed scenes.** Qualitative roll-outs and scene reconstructions in diverse scenarios (Figs. 9, 10, 11–13, 14).
- **Ablations on Nymeria.** Component-wise ablation and method comparison on the Nymeria-based benchmark (Tabs. 5, 6).
- **Backbone studies.** Robustness across different video backbones and capacity scaling from 1.3B to 14B parameters (Tabs. 7, 15).
- **Baselines adapted to our setting.** Comparisons to general-scene simulators (Cosmos-14B/7B, Aether) and to autonomous-driving systems (GEM, VISTA) under the same data and protocol (Tabs. 11, 8).
- **Data scale.** Effect of using only existing datasets vs. adding data constructed by our automatic pipeline (Tab. 9).
- **Motion injection schemes.** Comparison among alternative injection designs; our part-disentangled PMI is most effective (Tab. 10).
- **Long-horizon generation.** 24 s auto-regressive evaluation showing stable scene consistency and quality (Tab. 12).
- **Distillation & latency.** Quality before/after distillation (Tab. 13) and end-to-end latency ($\sim$119 ms/frame, $\sim$8.4 FPS).
- **Domain-wise analysis.** Separate results on game-like vs. real-world scenes (Tab. 14).
- **Granularity of PMI.** Impact of the number of motion parts in *Pretrain&PMI* (Tab. 16).
- **Rendering & filtering.** Robustness to point-map rendering backends (DUSt3R / MonST3R / MASt3R; Tab. 18) and study of data filtering ratios (Tab. 19).
- **3D consistency under viewpoint changes.** Multi-view reconstruction metrics (COLMAP) comparing "Ours" vs. "No Recon" (Tab. 17).
- **Implementation notes.** Clarifications of the training/inference pipeline (world/scene consistency is preserved at inference), superiority over prior video diffusion models, and detailed distillation settings.

A More Experimental Results

Framework comparisons with prior works. Fig. 10 illustrates the framework comparison between our PlayerOne and other competitors. Prior studies [10, 41, 7] have mainly concentrated on simulations within game-like environments, yet they often fail to accurately replicate real-world scenarios. Within these simulated environments, users are typically restricted to performing predefined actions, such as directional movements. This limitation confines user interactions to a constructed world, thereby restricting the execution of freeform movements akin to those in real-world settings. Existing realistic world simulators [23, 31, 1] often focus solely on world-consistent generation, lacking mechanisms for human movement control. As a result, users are relegated to passive observers, rather than active participants, within the environment. This significantly impacts the user experience by hindering the formation of a genuine connection with the simulated world. In contrast to these limitations, our approach enables freeform motion control for users, enhancing their interactive experience.

Evaluation on Nymeria. For more comprehensive evaluation, we have reconstructed our evaluation benchmark to provide a more comprehensive assessment.Specifically,we selected 100 videos featuring full-body actions from the test-sets of four different datasets: EgoExo-4D, Nymeria, FT-HID, EgoExo-Fitness. The results, as shown in Tab. 5 and 6, demonstrate that our method consistently outperforms both existing approaches and various ablated variants of our method.

Figure 9: **The reconstructed scene of our PlayerOne**. Our PlayerOnecan achieve relatively precise scene reconstruction by jointly modeling of video frames and scenes.

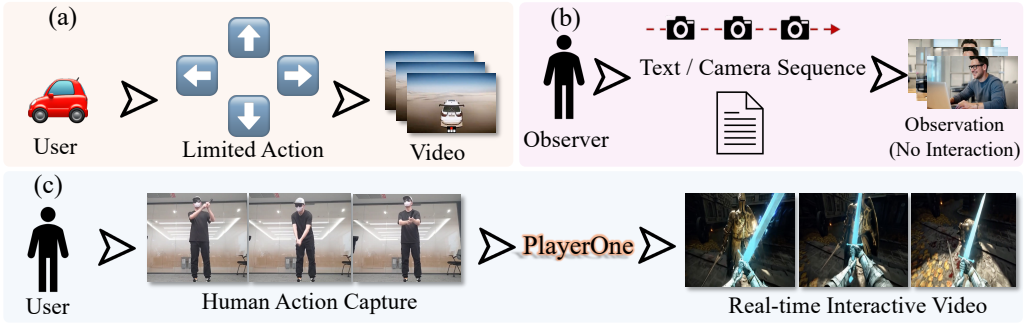

Figure 10: **Difference between our PlayerOneand prior works**. Our PlayerOnecan enable freeform movements in the simulated world and achieve great world consistency across diverse scenarios.

Investigation on the impact of different backbones. Here we conduct more analysis on the impact of backbones. As shown in Tab. 7, our method can achieve robust performance across all the backbones, showing great generalization ability.

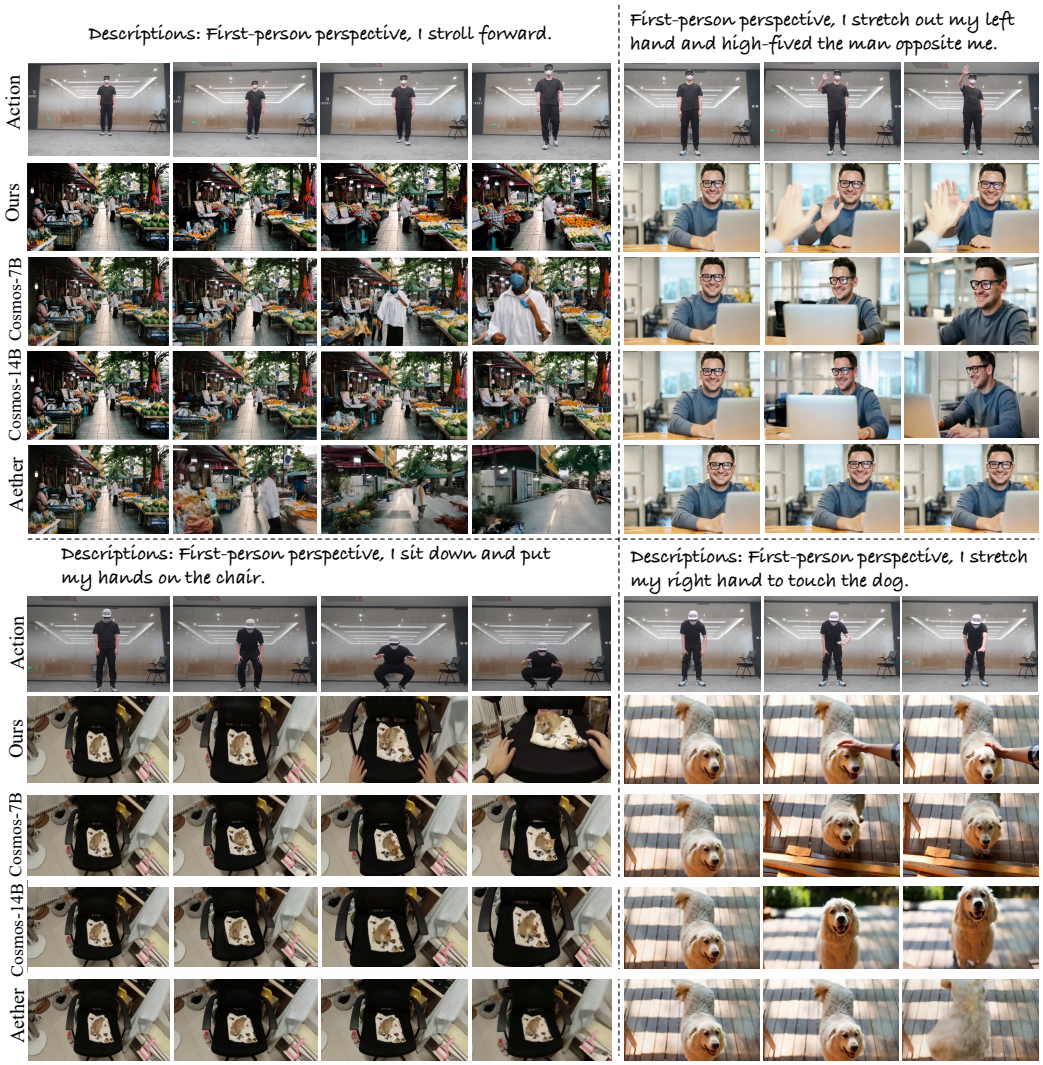

Figure 11: **More visualization results generated by our PlayerOne**. Our method demonstrates great superiority in both motion alignment and environmental interaction across different domains.

Table 5: **Quantitative evaluation** on the components of PlayerOne on the Nymeria dataset. "No Camera"/"Filtering" denote training without/with the camera encoder/data filtering.

Method	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
Baseline	48.72	62.13	402.31	370.27	33.57	425.15	0.1570
+ Pretrain	53.81	71.29	280.51	255.63	39.11	328.41	0.1265
+ Pretrain&ControlNet	54.28	72.31	264.10	241.91	40.16	314.27	0.1228
+ Pretrain&Entangled	55.24	73.54	258.88	234.16	41.23	308.12	0.1174
+ Pretrain&PMI (No Camera)	57.68	76.79	201.34	215.89	43.58	278.24	0.1017
+ Pretrain&PMI	59.57	78.56	170.01	191.43	46.02	266.73	0.0912
+ Pretrain&PMI&Filtering	61.74	81.11	153.17	178.64	47.37	249.33	0.0850
+ Pretrain&PMI&Filtering&Recon (No Adapter)	60.13	78.67	191.22	196.54	45.29	263.18	0.1023
+ Pretrain&PMI&Filtering&Recon (DUSt3R)	64.02	84.41	140.16	167.23	50.25	251.38	0.0788
PlayerOne (ours)	**64.87**	**85.79**	**137.91**	**162.74**	**50.72**	**247.97**	**0.0761**

Comparison with works in autonomous driving. Besides works in the manuscript, here we compare with works in autonomous driving as well. As shown in Tab. 8, we train these models with the same training data as ours and our method can achieve much better performance against them, demonstrating great superiority.

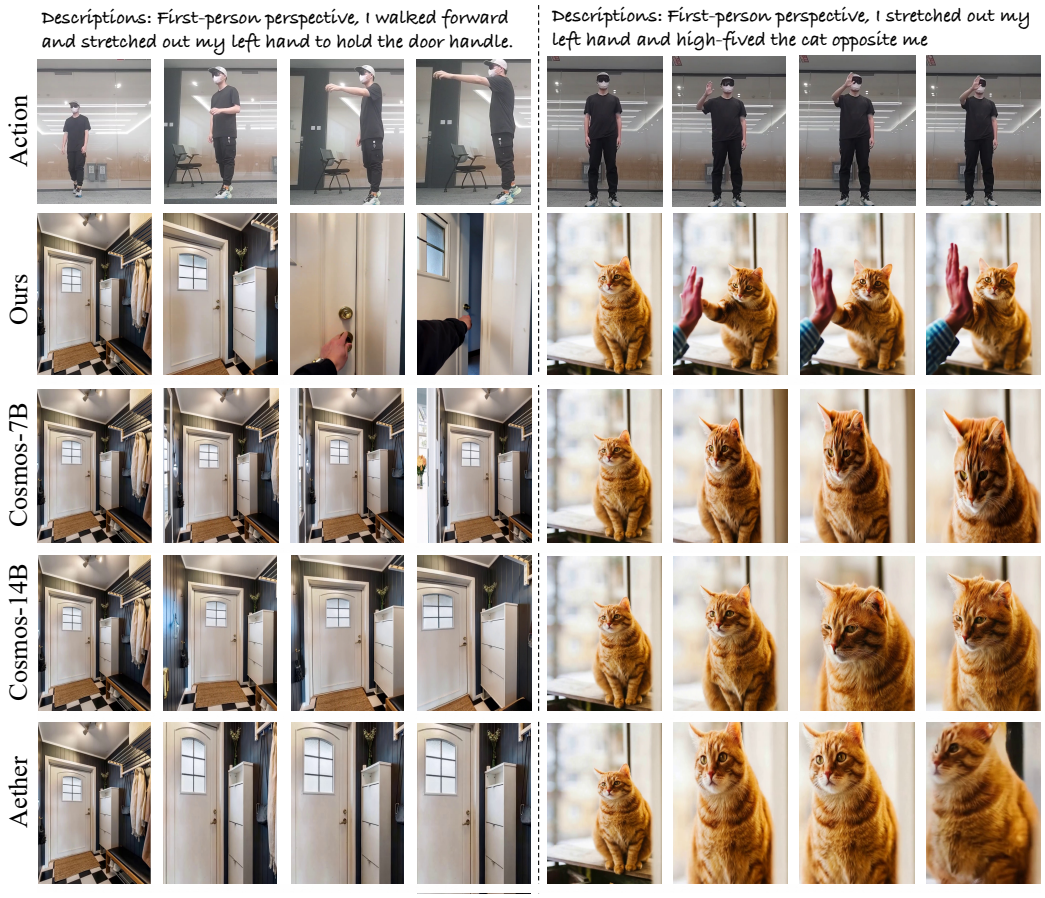

Figure 12: **More visualization results generated by our PlayerOne**. Our method demonstrates great superiority in both motion alignment and environmental interaction across different domains.

Figure 13: **More visualization results generated by our PlayerOne**. Our method demonstrates great superiority in both motion alignment and environmental interaction across different domains.

Limited v.s. Full training data. We validate the importance of our dataset construction pipeline by training with (i) only the existing datasets (*Limited-data*) and (ii) the union of existing datasets and our constructed data (*Full-data*). As shown in Tab. 9, using the full data yields clear, consistent improvements across all metrics: semantic alignment increases (DINO/CLIP: 56.4→67.8 / 76.7→88.2), geometric errors decrease (MPJPE/MRRPE: 182.42→127.16 / 202.35→151.62), and per-

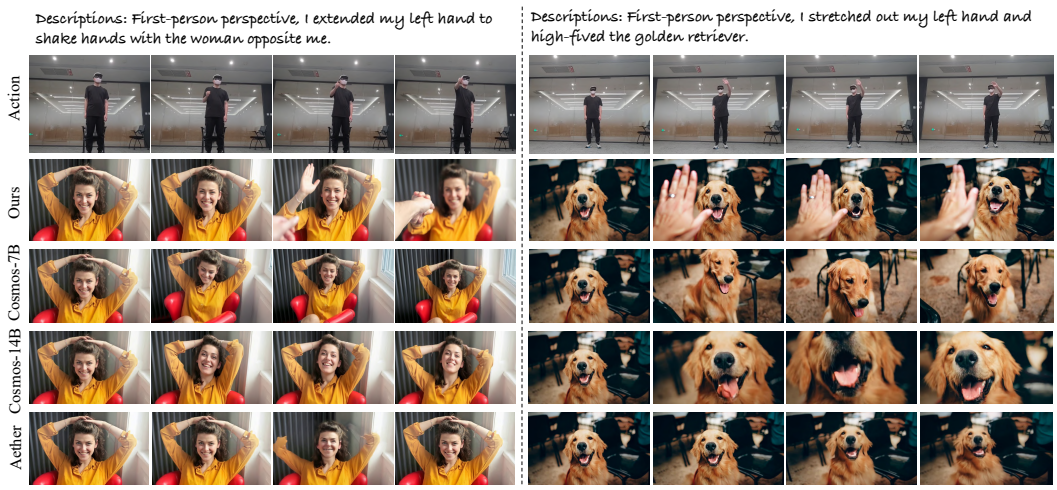

Figure 14: **More visualization results generated by our PlayerOne**. Our method demonstrates great superiority in both motion alignment and environmental interaction across different domains.

Table 6: **Quantitative comparison** between PlayerOne and other works on the benchmark constructed with the Nymeria dataset.

Method	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
Aether-5B	37.50	63.80	418.91	435.24	37.80	400.50	0.1881
Cosmos-7B	45.60	70.10	304.20	327.05	43.40	348.00	0.1617
Cosmos-14B	51.30	80.00	259.17	256.19	47.50	306.85	0.1374
PlayerOne (ours)	**64.80**	**85.70**	**137.91**	**162.74**	**50.72**	**247.97**	**0.0761**

Table 7: **Investigation on the impact of backbones** of PlayerOne.

Model	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
CogVideoX-5B	67.30	87.70	135.89	157.41	51.20	227.35	0.0724
Wan1.3B	67.80	88.20	127.16	151.62	52.60	226.12	0.0663
Wan14B	**69.10**	**90.70**	**109.34**	**124.20**	**56.00**	**182.55**	**0.0515**

Table 8: **Comparison with works in autonomous driving**. We train these models with the same training data to ensure fair comparisons.

Method	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
PlayerOne (ours)	**67.80**	**88.20**	**127.16**	**151.62**	**52.60**	**226.12**	**0.0663**
GEM [8]	40.10	62.70	405.32	391.21	38.20	378.40	0.1789
VISTA [15]	39.50	60.40	431.83	406.06	37.80	391.25	0.1857

Table 9: **Effect of data scale**. Data constructed with our automatic pipeline can significantly boost the performance.

Method	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
Limited-data	56.40	76.70	182.42	202.35	46.60	272.01	0.0827
Full-data	**67.80**	**88.20**	**127.16**	**151.62**	**52.60**	**226.12**	**0.0663**

ceptual/temporal quality improves (PSNR/FVD/LPIPS: 46.6→52.6 / 272.01→226.12 / 0.0827→0.0663). The gains are substantial (e.g., MPJPE -30.3%, MRRPE -25.1%), confirming that our constructed data broadens coverage of appearance and motion patterns and leads to stronger motion alignment and overall fidelity.

Table 10: **Comparison with different motion injection schemes**. Our proposed PMI can demonstrate much better performance against these schemes.

Method	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
PlayerOne (ours)	**67.80**	**88.20**	**127.16**	**151.62**	**52.60**	**226.12**	**0.0663**
RealisDance-Scheme [51]	62.30	83.90	137.44	161.20	50.90	246.25	0.0711
Magic-Me-Scheme [26]	61.60	82.80	139.10	163.42	50.60	250.71	0.0724
MimicMotion-Scheme [50]	61.10	82.20	140.34	164.55	50.30	252.66	0.0735
AnimateAnyone-Scheme [18]	60.80	81.80	141.22	165.37	50.10	254.10	0.0741

Table 11: **Comparison across other works** when using the same training data.

Method	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
PlayerOne (ours)	**67.80**	**88.20**	**127.16**	**151.62**	**52.60**	**226.12**	**0.0663**
Cosmos-14B	56.20	80.30	220.35	230.11	48.80	275.13	0.1076
Cosmos-7B	51.80	74.20	253.41	265.50	46.20	294.27	0.1252
Aether	46.50	68.00	291.20	310.33	42.90	334.68	0.1556

Comparison with different motion injection schemes. Here we conduct detailed comparisons with different motion injection schemes. As shown in Tab. 10, our motion injection scheme achieves the best results across all metrics, indicating superior motion alignment and visual fidelity. Compared with the strongest baseline (RealisDance-Scheme), we obtain higher semantic consistency (DINO/CLIP: +5.5/+4.3), lower geometric errors (MPJPE/MRRPE: -10.28/-9.58), and better perceptual/temporal quality (PSNR/FVD/LPIPS: +1.70/-20.13/-0.0048). These consistent gains suggest our design transfers motion more accurately while preserving identity and appearance, leading to the most reliable motion alignment among the evaluated schemes.

Transferring existing works to the same setting. We have thoroughly validated our approach by directly adapting several leading video diffusion models to the egocentric setting, with the same data and input motion injection schemes. Our results consistently show that these adapted baselines suffer from scene inconsistencies, poor alignment, and lower visual fidelity, while our method delivers stable, realistic, and precisely controlled egocentric video. This is further supported by both quantitative metrics and qualitative results provided in the manuscript. To further verify our superiority, we adapt the baselines in the main text (Cosmos-14B, Cosmos-7B, Aether) to our setting and apply our disentangled scheme with the same training data, our model still achieves notably better scene stability and action alignment as shown in Tab. 11.

Why separate processing contributes to precise alignment? Splitting the latent representation into three separate parts—head, hands, and body/feet—enables the model to more effectively capture the unique and often semi-independent motion characteristics of each region. Human body motion is inherently high-dimensional and highly articulated, with different parts frequently moving independently or exhibiting distinct dynamics. Encoding all motion information into a single, entangled latent can cause fine-grained cues—especially from subtle or fast-moving regions like hands or head—to be overshadowed by larger body movements, making precise alignment difficult. By disentangling the motion into part-specific latents, each body region's movement can be independently modeled and aligned with the corresponding input. This modular approach allows the model to better capture local nuances, improves compositionality, and enhances control over complex motions, resulting in higher motion fidelity and visual quality.

Latency between motion input and video output. After distillation, the measured end-to-end inference latency is approximately 119 milliseconds per generated-frame, corresponding to about 8.4 FPS. This latency is measured from the moment the motion input is received to the generation of the corresponding video frame, and includes all major processing steps.

Details of distillation. We strictly follow the Causvid distillation procedure in our implementation. During the distillation stage, we use the entire SMPL-Video paired dataset from the second phase of training. Specifically, we first generate 8,000 ODE pairs and train the student for 30,000 iterations using AdamW with a learning rate of 5×10^{-6}. We then continue training with our asymmetric

DMD loss for another 30,000 iterations using AdamW with a learning rate of 2×10^{-6}. A guidance scale of 3.5 is used, and we adopt the two time-scale update rule from DMD2 with a ratio of 5. The entire distillation process takes about 5 days on 32 A800 GPUs.

Details of training pipeline. We clarify that the *world/scene consistency* mechanism is *not* removed at inference. During training, a CUT3R-like reconstructor is used *only* to produce ground-truth point-map sequences corresponding to each video frame; these point maps supervise our scene-frame reconstruction loss and are further encoded to build noised point-map latents. In both training and inference, the *model inputs* are always the initial egocentric frame and the human-motion sequence, while the *prediction targets* are the video frames together with the point-map sequence.

Concretely, let the raw video be $V \in \mathbb{R}^{B\times K\times 3\times H\times W}$ and its point maps $P \in \mathbb{R}^{B\times K\times N\times 3}$. After VAE encoding, the video becomes $z^0_{\text{video}} \in \mathbb{R}^{B\times k\times c\times h\times w}$ with $h = H/8$, $w = W/8$, $k = K/4$. A 3D encoder plus an adapter maps the point maps to $z_{\text{point}} \in \mathbb{R}^{B\times k\times 64\times h\times w}$, matching the video latent's spatio-temporal shape. Both z_{video} and z_{point} are noised to obtain z'_{video} and z'_{point} for denoising training. The motion sequence, passed through our motion encoder, yields $z_{\text{motion}} \in \mathbb{R}^{B\times k\times 3\times h\times w}$. The first-frame latent $z_{\text{first}} \in \mathbb{R}^{B\times 1\times c\times h\times w}$ is repeated k times so that $z^k_{\text{first}} \in \mathbb{R}^{B\times k\times c\times h\times w}$. We then concatenate along channels:

$$z_{\text{input}} = \text{concat}\big(z^k_{\text{first}},\, z_{\text{motion}},\, z'_{\text{video}},\, z'_{\text{point}}\big) \in \mathbb{R}^{B\times k\times(3+64+2c)\times h\times w}.$$

A DiT backbone denoises z_{input} into z_{output} (same shape). Losses are computed between the predicted $(z^*_{\text{video}}, z^*_{\text{point}})$ and the ground-truth $(z_{\text{video}}, z_{\text{point}})$ latents via an MSE objective.

Inference. At test time there are no ground-truth point maps and thus no 3D point-map encoder is required. Instead, the network *jointly predicts* the video and its point-map latents autoregressively: both streams are initialized from noise and co-denoised frame by frame, conditioned on the initial frame and the motion sequence. Because the video and point-map streams are modeled with identical spatio-temporal layouts and are denoised in lockstep, the scene-consistency module remains fully active during inference—*no extra inputs are needed and no module is removed*. In short, training uses point maps to supervise and align the latent spaces; inference keeps the joint two-stream denoising (video + point maps) to maintain world consistency throughout generation.

Superiority over previous video diffusion models. Our approach advances egocentric video generation and simulation in several complementary ways that prior video diffusion systems do not address. **(i) Scene consistency and world modeling.** Instead of treating frames almost independently—which often causes background flicker, object drift, and abrupt scene changes—our method explicitly reconstructs and maintains a unified 3D world along the entire sequence. We jointly predict the video stream and a point–map stream under an explicit camera representation, so every generated frame is geometrically aligned to an evolving scene state; this yields stable, immersive first-person experiences over long autoregressive roll-outs. **(ii) Accurate and fine-grained first-person control.** Existing methods commonly rely on a single, entangled pose/control vector and therefore cannot model the different roles of head, hands, and feet in egocentric interaction. By employing part-disentangled motion injection, our system exposes independent, precise controls for head orientation (viewpoint/world alignment), hand motion (object interaction), and foot placement (navigation realism), enabling faithful user-driven control and correct coupling between the actor and the rendered environment. **(iii) Efficient and real-time generation.** Rather than invoking heavy 3D encoders at inference, we align modalities in the latent space using lightweight adapters inside the scene–frame reconstruction pathway, and we deploy a distilled backbone for decoding. This design removes bulky inference-time modules and substantially reduces compute/memory cost, enabling high-quality egocentric video at interactive rates suitable for VR/AR and gaming scenarios. **(iv) Comprehensive validation.** We adapt several strong video-diffusion baselines to the egocentric setting under a unified protocol and evaluate them on a reconstructed 100-video benchmark and an extended 24 s horizon. Across perceptual (DINO/CLIP), geometric (MPJPE/MRRPE), and temporal/fidelity (PSNR/FVD/LPIPS) metrics, our model consistently outperforms adapted baselines, confirming that the gains come from principled world modeling and fine-grained control rather than from model size alone. Collectively, these ingredients make PlayerOne a practical, accurate, and controllable egocentric world simulator, rather than a generic short-clip generator.

Limitation analysis on our model structure. From the view of the model, one current limitation of our model is the lack of explicit physical interaction modeling between the human body and the

Table 12: **Quantitative evaluation on long video generation**. Our proposed component demonstrates consistent performance boosts.

Method	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
Baseline	48.38	60.25	418.66	385.23	32.82	443.77	0.1687
+ Pretrain	53.74	69.30	293.45	270.84	38.19	341.66	0.1331
+ Pretrain&ControlNet	54.61	70.98	275.18	256.24	39.08	325.13	0.1279
+ Pretrain&Entangled	55.21	71.82	268.22	249.91	40.48	317.42	0.1229
+ Pretrain&PMI (No Camera)	57.36	74.48	210.98	229.54	42.62	286.93	0.1063
+ Pretrain&PMI	58.90	76.62	178.44	204.22	44.98	275.47	0.0955
+ Pretrain&PMI&Filtering	61.13	78.94	162.10	190.07	46.27	258.54	0.0898
+ Pretrain&PMI&Filtering&Recon (No Adapter)	59.78	76.02	200.04	209.34	44.22	272.60	0.1077
+ Pretrain&PMI&Filtering&Recon (DUSt3R)	63.23	82.62	148.08	178.91	49.13	260.95	0.0837
PlayerOne (ours)	**63.92**	**84.03**	**145.26**	**173.35**	**49.70**	**257.62**	**0.0816**

Table 13: **Effect of distillation**. Our method can achieve similar performance before/after distillation.

Model	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
Distilled	67.80	88.20	127.16	151.62	52.60	226.12	0.0663
Non-distilled	**68.20**	**88.70**	**125.20**	**149.40**	**53.00**	**221.85**	**0.0647**

Table 14: **Performance evaluation by domain**. Performance on game data is slightly inferior to the real one due to the inherent imbalance in training data.

Domain	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
Game	65.40	85.70	136.80	160.41	50.90	248.95	0.0705
Real	**67.80**	**88.20**	**127.16**	**151.62**	**52.60**	**226.12**	**0.0663**

environment. Structurally, our framework conditions video generation on SMPL motion sequences and a static point cloud reconstruction of the scene, but does not incorporate physical constraints or interaction modules such as collision detection, physics engines, or contact reasoning. Consequently, our model may generate unrealistic results, such as hands or body parts penetrating objects, or a lack of proper occlusion and collision feedback. We acknowledge this limitation and consider it a promising direction for future work.

Evaluation on long video generation. Our method is not limited to generating only short (e.g., 6-second) videos. After distillation with Causvid, our model can generate long videos in an auto-regressive fashion, producing each frame sequentially conditioned on previous outputs on-the-fly, rather than simply concatenating multiple short clips. This sequential generation, together with our current point cloud encoding approach, allows us to maintain strong scene consistency throughout long video sequences. To verify the superiority, we have extended our benchmark by increasing the video length to 24 seconds and evaluated our method under this longer horizon. The results in Tab. 12 show that our model maintains comparable performance and visual quality, demonstrating its ability to generate consistent long-term video sequences.

Impact of distillation. Here we present the performance gap before and after distillation. The results in Tab. 13 show that there exists a negligible performance gap.

Domain-wise evaluation. We evaluate our method separately on game and real-world data. As shown in Tab. 14, performance on game scenes is slightly lower: DINO/CLIP drop from 67.8/88.2 (Real) to 65.4/85.7 (Game), geometric errors increase (MPJPE/MRRPE: 127.16/151.62 $\rightarrow$ 136.80/160.41), and visual quality also dips (PSNR/FVD/LPIPS: 52.6/226.12/0.0663 $\rightarrow$ 50.9/248.95/0.0705). We attribute this modest gap to the training set containing more real-world data, which biases the model toward real distributions and leaves less coverage for game-specific appearance and motion patterns.

Investigation on the impact of backbone capacity. We compare a smaller (1.3B) and a stronger (14B) backbone to assess capacity scaling. As shown in Tab. 15, the 14B model brings consistent gains: semantic alignment improves (DINO/CLIP: 67.8$\rightarrow$69.1 / 88.2$\rightarrow$90.7), geometric errors drop

Table 15: **Ablation study on using a stronger backbone**. It can be observed that our method can achieve more superior performance.

Model	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
1.3B	67.80	88.20	127.16	151.62	52.60	226.12	0.0663
14B	**69.10**	**90.70**	**109.34**	**124.20**	**56.00**	**182.55**	**0.0515**

Table 16: **Progressive components of Pretrain&PMI**.

Part-Number	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR (↑)	FVD (↓)	LPIPS (↓)
"+Pretrain&PMI"-1	58.00	76.30	235.12	212.53	43.90	279.41	0.1060
"+Pretrain&PMI"-2	60.50	79.10	185.00	190.30	46.20	263.00	0.0950
"+Pretrain&PMI"-3	62.50	81.30	156.76	175.18	48.30	245.72	0.0839
"+Pretrain&PMI"-4	62.70	81.40	155.10	174.80	48.40	244.60	0.0830
"+Pretrain&PMI"-5	62.80	81.50	154.80	174.50	48.40	244.10	0.0828
"+Pretrain&PMI"-6	**63.00**	**81.80**	**152.62**	**172.90**	**48.60**	**243.82**	**0.0819**

(MPJPE/MRRPE: 127.16→109.34 / 151.62→124.20), and perceptual/temporal quality increases (PSNR/FVD/LPIPS: 52.6→56.0 / 226.12→182.55 / 0.0663→0.0515). These results confirm that scaling the backbone substantially improves motion alignment and overall fidelity.

Investigation on the impact of component division. We ablate the component division of *Pretrain&PMI* by progressively enabling its parts. As shown in Tab. 16, increasing the parts from 1 to 6 steadily improves semantic alignment (DINO/CLIP), reduces geometric errors (MPJPE/MRRPE), and enhances reconstruction quality (PSNR/FVD/LPIPS). The gains, however, saturate after part-4/5: too few parts cannot fully achieve motion alignment, while too many yield only marginal improvements at the cost of notably higher memory consumption. Balancing accuracy and efficiency, a moderate number of parts is preferred.

More visualization results. Here we provide more simulated results in Fig. 11 and Fig. 12. In terms of first-person action alignment and world consistency, we have achieved outstanding results in both game and real-world scenarios. Additionally, we selected highly dynamic settings, such as driving scenes, where our method successfully models the world with high accuracy while maintaining excellent video fluidity. More visualization results can be referred in the submitted video.

Visualization of the scene reconstruction. This section presents a comprehensive visualization of the reconstructed scenes using our PlayerOne. As illustrated in Fig. 9, our approach adeptly reconstructs both scenes and video frames through a progressive methodology. This ensures not only inter-frame coherence but also overarching scene consistency across a diverse array of scenarios. Specifically, the method seamlessly integrates temporal and spatial elements to maintain visual congruity, even in complex environments. The robustness of our technique is further reflected in its capacity to adapt to varying scene dynamics and compositions, thereby offering a reliable framework for generating high-quality, consistent video outputs. Through these visualizations, the effectiveness of our PlayerOnein achieving smooth and coherent reconstructions is clearly demonstrated, highlighting its potential applications in advanced graphical simulations and interactive environments.

Investigation on the impact of rendering. In addition to analyzing DUSt3R [37] within the manuscript, we extend our comparison by substituting the point map rendering technique with several alternative methods, including MonST3R [46] and MASt3R [21]. As detailed in Table 18, our approach exhibits remarkable generalization capabilities across the spectrum of rendering strategies. This robust performance underscores the versatility and adaptability of our method, making it highly effective in accommodating diverse rendering paradigms. By integrating various rendering methodologies, we showcase the method's extensive applicability and resilience in maintaining high-quality outputs, regardless of the specific techniques employed. The comparative analysis further reflects our method's potential for broad applicability in dynamic, real-world settings, demonstrating consistent, optimal performance in diverse operational contexts.

Table 17: **Investigation on the impact of reconstruction**.

	Chamfer (↓)	Overlap (↑)	Hausdorff (↓)
Ours	0.82	0.86	1.21
No Recon	1.94	0.53	2.82

Investigation on the impact of filtering. In this study, we examine how the filtering ratio affects model performance. As demonstrated in Table 19, increasing the filtering ratio leads to a decline in model performance, which can be attributed to insufficient data. Conversely, the absence of filtering also causes performance deterioration, primarily because noisy data is introduced during training, negatively impacting the accuracy of action alignment. Therefore, we have determined that a filtering ratio of 10% optimizes performance.

Visual and quantitative difference with the "No Adapter" and "No Recon" setting. The large performance gap for 'no adapter' comes from a mismatch in the feature spaces: our point map encoder is fixed, and its latent space is not directly aligned with the VAE-encoded video latent space. When these are simply concatenated without a dedicated adapter, the model struggles to reconcile the different statistics and semantics between the two types of latents, resulting in degraded video quality and scene breakdown. Our adapter module learns to map point map features into the appropriate latent space for effective fusion with video latents, thus improving both fidelity and scene consistency. This design is crucial for effective information transfer between geometry and appearance. For 'no recon', it is primarily due to two reasons:

- Strong Pretraining: Our model is first pretrained on a large-scale egocentric video dataset. This pretraining stage may equips the model with a weak ability to model scene dynamics and maintain a certain level of spatial consistency, even before introducing explicit scene-frame reconstruction.
- Lack of Direct Metrics: Most commonly used metrics (e.g., DINO, CLIP, PSNR, FVD, LPIPS) are not designed to specifically measure scene (world) consistency across frames. These metrics primarily evaluate frame-wise visual similarity, semantics, or overall diversity, but do not directly capture temporal geometric consistency, which is the core advantage of our reconstruction module.

To more objectively evaluate scene consistency under large viewpoint changes, we perform multi-view 3D reconstruction (using COLMAP) on generated videos. We then compare the overlap ratio and Chamfer Distance between the reconstructed point clouds for different methods as shown in Tab. 17. Our method yields significantly higher point cloud consistency and structural stability, demonstrating superior 3D scene coherence across time.

Definition of evaluation criteria. For user study, we evaluate the video quality from four views:

- Quality: Defined as the overall visual harmony and realism of the video, regardless of whether the generated content is perfectly true to the input. Annotators were asked: "Does the generated video look natural, coherent, and free from jarring artifacts?"
- Smoothness: Measures how consistently the motion flows across frames, without abrupt transitions or temporal artifacts. Annotators were prompted: "Are the motions and camera transitions fluid and continuous throughout the video?"
- Fidelity: Refers to the preservation of the subject's appearance (identity, clothing, background) and the absence of distortions or glitches. Annotators were asked: "Does the person look like the initial frame and is there minimal distortion?"
- Alignment: Defined as how closely the generated motion matches the intended action described in the motion condition. For this criterion, annotators viewed both the generated video and the motion condition signal (stick-figure or pose sequence) side-by-side for all 100 videos, and rated: "Does the generated video accurately follow the given pose/motion signal in timing, type of action, and spatial positioning?"

Table 18: **Investigation on the impact of different rendering methods**. Our PlayerOne shows great robustness against different rendering methods.

	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR(↑)	FVD (↓)	LPIPS(↓)
CUT3R [36]	67.8	88.2	127.16	163.62	50.3	236.12	0.0663
MonST3R [46]	67.1	88.4	127.68	164.90	49.8	235.09	0.0771
MASt3R [21]	67.4	87.8	127.35	163.06	50.1	240.10	0.0724

Table 19: **Investigation on the impact of different filtering ratios**.

Ratio	DINO-Score (↑)	CLIP-Score (↑)	MPJPE (↓)	MRRPE (↓)	PSNR(↑)	FVD (↓)	LPIPS(↓)
0	64.2	85.0	123.50	158.10	47.5	228.50	0.0587
5	65.4	86.5	125.00	160.20	48.7	230.75	0.0600
10	**67.8**	**88.2**	**127.16**	**163.62**	**50.3**	**236.12**	**0.0663**
15	66.0	87.2	126.00	162.00	49.8	234.00	0.0640

B Broader Impact

The proposed PlayerOne for video generation, designed to facilitate freeform human motion control within environments created from user-provided images while producing world-consistent videos, demonstrates considerable potential across diverse domains. It is particularly adept at generating engaging and dynamic educational content, thereby fostering experiential learning through interactive simulations. Moreover, the model optimizes the production of high-quality, consistent visual content for films, television, and online media, dramatically reducing both production time and costs. It also enables the creation of interactive narratives, allowing users to influence the storyline through their interactions within the generated environments, thus enhancing user engagement and narrative immersion. Beyond these applications, the world model serves as a valuable tool for research on human behavior and interactions within controlled virtual settings, offering insights for fields such as psychology, sociology, and human-computer interaction. By integrating these capabilities, the proposed world model not only amplifies existing applications but also paves the way for novel research and development, significantly contributing to technological progress and societal advancement.

C Limitations & Discussion

While significant strides have been made in egocentric interaction and coherent world modeling, certain limitations persist. Despite the compelling outcomes, the performance in game scenarios is somewhat diminished compared to realistic scenarios, likely due to the disproportionate amount of realistic training data available. Moreover, in highly dynamic scenes, predictions may falter, reflecting the constraints inherent in the current base model. Future research endeavors could potentially overcome these challenges by investigating novel action representations, incorporating an expanded dataset for game scenarios, and adopting a more robust base model.

D License of assets

Datasets (Apache 2.0 License) Nymeria [25]/FT-HID [12]/EgoExo-Fitness [22] (Creative Commons Attribution 4.0 International), EgoExo4D [9]/Egovid-5M [38](MIT License).

Codes The official repository of Aether [33] (MIT License), the official repository of Cosmos [1] (Apache 2.0 License), the official repository of Wan2.1 [35] (Apache 2.0 License).