MoRight: Motion Control Done Right

Shaowei Liu[1,2*], Xuanchi Ren[1], Tianchang Shen[1], Huan Ling[1], Saurabh Gupta[2], Shenlong Wang[2], Sanja Fidler[1], and Jun Gao[1]

[1] NVIDIA
[2] University of Illinois Urbana-Champaign
[*]Work done during an internship at NVIDIA.
https://research.nvidia.com/labs/sil/projects/moright

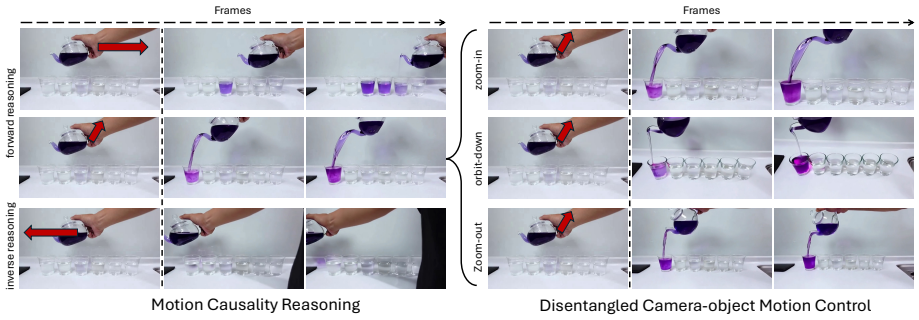

Fig. 1: Given a single input image, our method enables controllable interactive motion generation with motion causality reasoning. *Left:* Users can provide active motion (*e.g.* action of hand) to drive scene dynamics (*forward reasoning*) or specify desired passive outcomes (*e.g.* trajectory of teapot) and recover plausible driving actions (*inverse reasoning*). *Right:* The model further enables disentangled control of object motion and camera viewpoint, allowing users to explore the scene with custom views and motions.

Generating motion-controlled videos—where user-specified actions drive physically plausible scene dynamics under freely chosen viewpoints—demands two capabilities: (1) *disentangled motion control*, allowing users to separately control the object motion and adjust camera viewpoint; and (2) *motion causality*, *i.e.*, generative action–consequence dependency, ensuring that user-driven actions trigger coherent reactions from other objects rather than merely displacing pixels. Existing methods fall short on both fronts: they entangle camera and object motion into a single tracking signal and treat motion as kinematic displacement without modeling causal relationships between object motion. We introduce MoRight, a unified framework that addresses both limitations through disentangled motion modeling. Object motion is specified in a canonical static-view and transferred to an arbitrary target camera viewpoint via temporal cross-view attention, enabling disentangled camera and object control. We further decompose motion into *active* (user-driven) and *passive* (consequence) components, training the model to learn motion causality from data. At inference, users can either supply active motion and MoRight predicts consequences (*forward reasoning*), or specify desired passive outcomes and MoRight recovers plausible driving actions (*inverse reasoning*), all while freely adjusting the camera viewpoint. Experiments on three benchmarks demonstrate state-of-the-art performance in generation quality, motion controllability, and interaction awareness.

1 Introduction

Humans interact with the physical world as active agents: we move our viewpoint, manipulate objects, and reason about how actions lead to consequences. Yet existing video generation models lack this unified capability [7, 12, 33, 66, 80]. Bridging this gap is essential for applications that demand interactive visual reasoning, from embodied AI agents [6, 70, 77] that must anticipate action outcomes [22, 26, 28], to world models [8, 31, 34, 79, 82] that simulate physical interactions, and to immersive content creation [11, 21, 44, 49, 50, 75, 86] where users freely navigate and manipulate scenes. A desirable video generation system should therefore offer joint controllability over both camera and object motion—generating visually coherent frames under arbitrary viewpoint changes while producing causally consistent scene dynamics driven by user-specified actions.

Existing controllable video generation methods [1, 10, 15, 23, 67, 68, 73] work as *renderers*: given displacements for *all* pixels, they generate a visually realistic video that adheres to the displacements. These approaches have two key limitations in practice. First, they entangle camera and object motion, making joint control ambiguous because viewpoint changes alter pixel trajectories. Second, they interpret user-specified motion as simple kinematic displacement and largely ignore its consequences. Models thus follow trajectories rather than reason about causal relationships between objects. Here, we use *causal reasoning* in a generative sense, referring to action-consequence dependencies rather than structured causal inference. In reality, actions have consequences: lifting a teapot may make water pour. Specifying all objects' motion through prompts is often impractical. Without modeling such action-consequence relationships, motion-controlled generation cannot capture the structure of real-world interactions.

To overcome these challenges, we introduce MoRight, a unified framework for video generation with disentangled camera–object motion control and motion causality reasoning as shown in Fig. 1. Given a reference image, user-specified motion trajectories, and target viewpoints, MoRight generates videos where objects follow the desired motion and the scene is rendered from the specified cameras. For disentangled camera-object motion control, our key insight is that specifying the motion of objects under camera changes is inherently difficult. We therefore introduce a **dual-stream motion formulation**. The first branch models and generates object motion in the source image plane under a canonical static viewpoint, allowing users to easily specify dynamic trajectories. The second branch represents the target camera motion and transfers object dynamics from the canonical branch via temporal cross-view attention. This *cross-view motion transfer* enables independent control of camera and object motion while maintaining coherent scene dynamics. For motion causality reasoning, we achieve this by decomposing object motion into two categories during training: *active motion*, representing user-driven actions, and *passive motion*, representing their causal outcomes. By conditioning on either active or passive motion signals, the video model learns to generate all the scene dynamics, capturing action–consequence relationships within the scene. This yields two complementary capabilities: for-

ward reasoning of scene evolution from user actions, and inverse reasoning of plausible actions that produce a desired outcome.

We evaluate MoRight on three benchmarks covering diverse interaction scenarios. Results show that MoRight outperforms existing methods in generation quality, motion controllability, and interaction awareness, validating the effectiveness of disentangled motion control and causal motion reasoning.

In summary, our contributions are threefold. (1) We propose a disentangled framework for camera and object motion control, enabling users to draw motion trajectories directly in the image plane while freely adjusting viewpoints to generate coherent videos. (2) We empower video generation models with motion reasoning capability by modeling action–consequence relationships, allowing user-driven motions to meaningfully interact with the environment and produce consistent scene dynamics. (3) We demonstrate that MoRight supports both forward and inverse reasoning: given active motion inputs, it predicts future scene evolution; given desired passive outcomes, it recovers plausible actions that achieve them. Together, these advances establish a unified framework for controllable and reasoning-aware video generation.

2 Related Work

2.1 Motion-Controlled Video Generation

Controllable video generation has been studied across a spectrum of motion granularity, from coarse region-level signals such as bounding boxes [54,68,76], sparse keypoint tracks [41, 73, 75, 83], optical flow fields [38, 42, 62, 63, 87], and dense per-pixel trajectories [15,23]. Despite steady progress, trajectory-based methods share two fundamental limitations. First, because trajectories are defined in pixel space, they inevitably entangle object and camera motion: any viewpoint change alters all trajectories, making joint control ill-posed without explicit foreground–background decomposition. Second, producing physically plausible motion typically demands carefully crafted trajectories from dedicated motion-generation models [11,49,50,53,55,62] or laborious manual annotation. MoRight overcomes both issues by disentangling camera and object motion by design, and by accepting lightweight inputs—simple strokes or sparse tracklets—that the model completes into coherent, interaction-aware dynamics.

2.2 Camera–Object Motion Disentanglement

Separating camera motion from scene dynamics [35, 40, 43, 81, 88] is a long-standing challenge in controllable generation [23, 64, 71, 72] and video understanding [36,51]. Existing methods [14,25,62,87] attempt to decouple the two by treating them as independent control signals, yet they typically rely on privileged information such as per-frame depth [14, 25], 3D object trajectories [17, 29, 39], or foreground–background segmentation masks [46, 56, 62, 87], and pre-warp all signals to their anticipated future locations. These assumptions require the full video sequence or 3D motion to be known in advance, limiting image-to-video

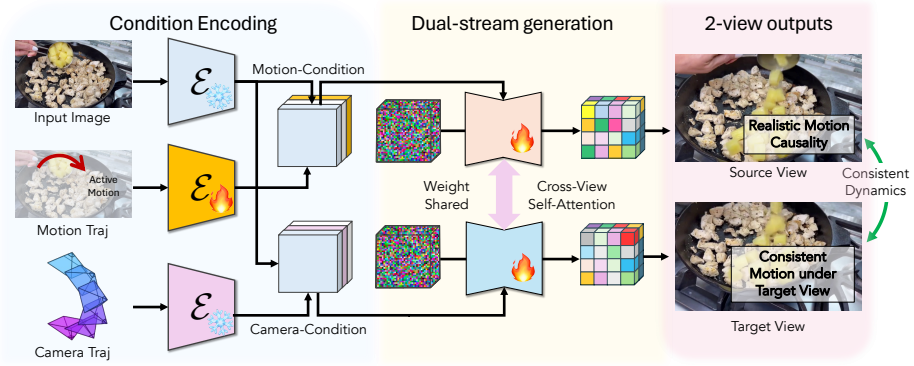

Fig. 2: Model architecture. Our model uses a shared-weight dual-stream architecture to disentangle object and camera motion. The canonical stream encodes trajectories with a track encoder and learns motion in a fixed canonical view, while the target stream encodes camera poses through a camera encoder. The resulting motion and camera conditions are injected into each attention block. Cross-view self-attention links the streams, transferring canonical-view motion to the target view for disentangled camera-object motion generation.

settings where only one reference frame is available. MoRight instead uses a canonical static-view branch for object dynamics and transfers them to arbitrary target viewpoints via cross-view attention, without explicit 3D supervision or precomputed scene decomposition.

2.3 Interaction and Causal Reasoning in Video Generation

A complementary line of work moves beyond kinematic animation toward causally grounded video generation. Some methods use external physics engines [11, 37, 44, 50, 55] or explicit action representations such as force vectors [24] to model specific phenomena, e.g., fluid flow or rigid-body collisions. While effective in constrained domains, they are tailored to particular interactions and require simulation in the loop, limiting generality. Others delegate causal reasoning to vision-language or large language models [45, 57, 74, 78], which first predict outcomes in text and then transfer them to video generators through intermediate representations such as flow fields [53, 55], edge maps [53], or depth [53, 84]. This two-stage pipeline is prone to error propagation, as imprecise textual predictions are further degraded during cross-modal conversion, yielding spatially inaccurate dynamics. MoRight sidesteps both limitations by learning latent action–response structure directly from video data. Decomposing motion into active (user-driven) and passive (consequence) components enables pixel-level cause-effect reasoning in a single forward pass, supporting both forward consequence prediction and inverse action inference.

3 Approach

Given a single image I, we aim to generate a T-frame video $\mathbf{x} \in \mathbb{R}^{T\times H\times W\times 3}$ that follows user-defined object motion, represented by pixel trajectories $\mathcal{T}$, and

a specified camera motion sequence $C_{i\,i=1}^{T}$. The generated video should model motion causality and produce coherent scene dynamics. To achieve this, we first present our approach for disentangled camera and object motion control in Sec. 3.1 and Fig. 2, and then introduce motion causality modeling in Sec. 3.2. Sec. 3.3 describes our training data curation pipeline, and training details along with the inference pipeline are provided in Sec. 3.4.

3.1 Disentangled Camera-Object Motion Control

Most existing approaches adopt pixel-wise trajectories as motion control signals. However, such representations inherently entangle object motion with viewpoint changes. The video model must implicitly reason how an object moves, and the camera changes, without explicit geometric cues, when conditioned on these signals. Our key insight is that object motion is intrinsically unambiguous when expressed in a canonical camera. Building on this observation, we decouple the motion signal from the viewpoint transformation through a dual-stream generation framework [2, 3]. The first stream synthesizes a canonical video in a static camera, where object motion can be directly and faithfully controlled. The second stream generates the target video with both camera and object motion. The two streams can interact with each other through self-attention as shown in Fig. 2. By jointly denoising both streams, the model learns to transfer motion cues from canonical space to arbitrary camera poses, where the canonical stream serves as an anchor for motion control. This design resolves motion-camera entanglement at generation time while supporting heterogeneous supervision, including motion-only, camera-only, and fully coupled data Sec. 3.3.

Preliminaries. We build upon a DiT-based [58] latent video diffusion models [16, 66]. A pretrained VAE encoder first encodes the video into a latent space $\mathbf{z}_0 = \mathcal{E}(\mathbf{x}) \in \mathbb{R}^{\hat{T}\times\hat{H}\times\hat{W}\times d}$. The diffusion model is trained in this latent space via flow matching [48]. Specifically, we first sample a noise from a Gaussian distribution: $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$, and form $\mathbf{z}_t = (1-t)\,\mathbf{z}_0 + t\,\boldsymbol{\epsilon}$ for $t \in [0, 1]$. The DiT $\mathcal{G}_\theta$ is trained to regress the velocity:

$$\mathcal{L} = \mathbb{E}_{\mathbf{z}_0,\, t,\, \boldsymbol{\epsilon}} \left[\left\| \mathcal{G}_\theta(\mathbf{z}_t, t, \mathbf{c}) - (\boldsymbol{\epsilon} - \mathbf{z}_0) \right\|^2 \right], \tag{1}$$

where $\mathbf{c}$ denotes conditioning signals ($e.g.$, text). At inference, an ODE solver [89] integrates the learned velocity from a noise to a clean latent $\hat{\mathbf{z}_0}$, from which a decoder reconstructs the original video $\hat{\mathbf{x}} = \mathcal{D}(\hat{\mathbf{z}_0})$.

Dual-stream generation. The user first provides the object motion $\{\boldsymbol{\tau}_i^{\text{can}}\}_{i=1}^{T}$ in the canonical frame. The first stream generates the videos only with object motion, and the second stream generates the videos with both object and camera motion. Concretely, the two streams receive their individual conditions:

$$\mathbf{c}^{\text{can}} = \big\{I,\ C_1,\ \{\boldsymbol{\tau}_i^{\text{can}}\}\big\}, \qquad \mathbf{c}^{\text{tar}} = \big\{I,\ \{C_i\},\ \emptyset\big\},$$

where C_1 is the identity first-frame camera pose, and $\emptyset$ denotes empty motion.

In the following, we assume paired training data: a ground-truth video $\mathbf{x}^{\text{can}}$ with object motion only and a corresponding video $\mathbf{x}^{\text{tar}}$ with both object and

camera motion. Extensions to other data are described in Sec. 3.3 and Sec. 3.4. We add independent noise at the same timestep t to obtain $\mathbf{z}_t^{\text{can}}$ and $\mathbf{z}_t^{\text{tar}}$, concatenate the two latents along the temporal dimension, and jointly denoise them. We slightly modify the positional embedding to distinguish the streams, with details in the supplement. Thus, the streams share DiT weights and differ only in input and conditioning, exchanging information through self-attention. During inference, they are jointly denoised; we return the target-stream output $\mathbf{x} = \mathcal{D}(\hat{\mathbf{z}}_0^{\text{tar}})$, while the canonical stream acts as a "virtual" anchor.

Condition injection and motion transfer. We inject camera and motion conditions into the latents of DiT at every transformer block. Specifically:

Camera encoding. We follow Gen3C [85] and warp the first image I using the corresponding camera pose and estimated depth [47]. We then encode the warped frames via encoder $\mathcal{E}$ from VAE and obtain the latent $\mathbf{z}^{\text{cam}} \in \mathbb{R}^{\hat{T}\times\hat{H}\times\hat{W}\times d}$. For the canonical stream, we use the identity matrix for warping.

Motion encoding. Following [23], we build a per-pixel trajectory map where pixels along one trajectory share the same temporal-correspondence embedding. We then encode it via a lightweight encoder to obtain $\mathbf{e}^{\text{trk}} \in \mathbb{R}^{\hat{T}\times\hat{H}\times\hat{W}\times d}$. For the target stream, we simply set $\mathbf{e}^{\text{trk}} = \mathbf{0}$ since the condition is empty.

Condition injection. The camera and motion encodings are fused via learned linear projections and added into the latent feature at each transformer block. Let $\mathbf{f}$ denote the feature of one block:

$$\mathbf{f}^i \leftarrow \mathbf{f}^i + W_{\text{cam}}\,\mathbf{z}^{i,\text{cam}} + W_{\text{trk}}\,\mathbf{e}^{i,\text{trk}}, \quad i \in \{\text{can, tar}\}. \tag{2}$$

The features from the two streams are then concatenated and passed through the self-attention layer:

$$\big[\,\mathbf{f}^{\text{can}};\ \mathbf{f}^{\text{tar}}\,\big] := \text{SelfAttn}\big(\big[\,\mathbf{f}^{\text{can}};\ \mathbf{f}^{\text{tar}}\,\big]\big), \tag{3}$$

allowing target-view tokens to attend to motion-conditioned canonical tokens and vice versa, implicitly exchanging the motion information in latent space. This inject-then-synchronize repeats at every block, progressively transferring motion across views, as shown in Fig. 2.

3.2 Motion Causality Modeling

Disentangling the camera from motion alone is insufficient for realistic interactions: when a hand pushes a cup, the cup must slide; when a ball strikes a stack of blocks, the blocks must scatter. We term this as *motion causality*: the ability to reason plausible consequences from the given actions, and vice versa.

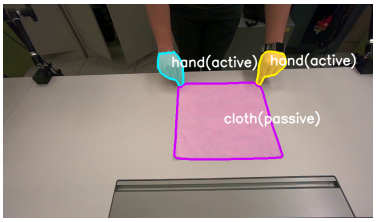

Fig. 3: Active vs. passive motion. The *active* object (hand) initiates the action, while the *passive* object (cloth) responds.

To model this, we decompose the motion tracks of all foreground objects $\mathcal{T} = \{\boldsymbol{\tau}_i\}_{i=1}^{T}$ into two complementary components as shown in Fig. 3: $\boldsymbol{\tau}_i = \boldsymbol{\tau}_i^{\text{act}} \cup \boldsymbol{\tau}_i^{\text{pas}}$, where

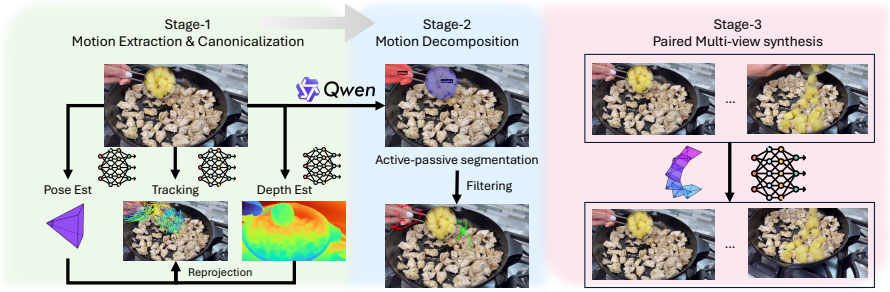

Fig. 4: Data curation pipeline. Foundation models [29,35,59] extract depth, camera poses, and tracks from raw videos. A VLM [4] segments tracks into active/passive regions. We further optionally use a video-to-video model [20] to generate paired videos with the same object motion but different camera motions.

$\boldsymbol{\tau}_i^{\text{act}}$ captures *active* motion, the intentional action applied to the scene (*e.g.*, a hand pushing), and $\boldsymbol{\tau}_i^{\text{pas}}$ captures *passive* motion, the resulting reaction of other objects (*e.g.*, the pushed object sliding). Such causality is critical for applications such as embodied AI [9, 18, 27].

The central mechanism to enable the causality modeling is through *motion dropout*. Specifically, during training, we randomly drop out one motion component from the input, and supervise the model on the full video containing both active and passive motion:

$$\tilde{\boldsymbol{\tau}}_i := \begin{cases} \boldsymbol{\tau}_i^{\text{act}}, & \xi < p, \\ \boldsymbol{\tau}_i^{\text{pas}}, & \text{otherwise}, \end{cases} \tag{4}$$

where ξ is sampled from a uniform distribution $\mathcal{U}(0,1)$, p is the dropout probability and $\tilde{\boldsymbol{\tau}}_i$ is used as tracking condition to video model.

During training, we do not distinguish active and passive motion in the input, relying on the model to infer the dropped component and generate plausible videos. This asymmetric supervision encourages learning action-consequence relationships instead of replaying provided trajectories.

At inference, the learned causality supports *forward reasoning* (action → reaction), where users specify an action and the model predicts consequences, and *inverse reasoning* (reaction → action), where users prescribe an outcome and the model synthesizes a plausible driving action. We demonstrate both in Sec. 4.4.

3.3 Training Data Curation

Curating data to train our dual-stream model is challenging since most real-world videos are single-view and always entangle camera and object motion, while our model requires paired videos depicting the same dynamics under different viewpoints. We first describe our data annotation pipeline to extract the pixel trajectories, camera poses, and active/passive motion, and provide details of our data curation for training afterwards. The overview of pipeline is shown in Fig. 4.

Motion extraction and canonicalization. Given a video $\mathbf{x}$, we estimate per-frame depth maps $\{D_i\}$, camera poses $\{C_i\}$, and intrinsics K using ViPE [35], and extract dense pixel trajectories $\mathcal{T} = \{\boldsymbol{\tau}_i\}_{i=1}^{T}$ with AllTracker [29]. Each trajectory is unprojected to 3D and reprojected into the first frame:

$$\boldsymbol{\tau}_i^{\text{can}} = \pi\big(K,\, C_0\, C_u^{-1}\, \pi^{-1}(K,\, \boldsymbol{\tau}_i,\, D_i)\big), \tag{5}$$

where π^{-1} lifts 2D points to 3D using depth and intrinsics, and π projects onto the image plane of I_0. We assume constant intrinsics across the video.

Active and passive motion decomposition. For the given video, we prompt a vision-language model (Qwen3 [4]) to identify the active and passive objects, then segment the first frame with SAM2 [59], yielding masks M^{act} and M^{pas}, for active and passive objects, respectively. Trajectories are assigned to each component by mask membership, producing $\boldsymbol{\tau}^{\text{can,act}}$ and $\boldsymbol{\tau}^{\text{can,pas}}$ for the motion dropout training in Eq. (4). We also generate per-video captions describing each motion component and only provide the caption of one component during training to prevent information leakage.

Synthetic data generation for paired two-view videos. We leverage a synthetic data generation pipeline to generate paired two-view videos for training. Specifically, we first curate the videos whose cameras are static by checking the displacement of camera poses estimated from ViPE [35]. We then synthesize corresponding moving-camera videos using a camera-control video-to-video model [19], providing supervision for the second stream. To increase camera diversity, we further augment the data with basic camera operations (*e.g.*, orbit, pan, zoom) as well as dynamic camera trajectories extracted from real videos.

Single-view real-world data for mixed-training. The generated paired videos inevitably contain visual artifacts. In our dual-stream mode, the abundant single-view real-world data can be leveraged for mixed-training. First, for the videos with static cameras (object motion only), we duplicate the video and treat it as the target video. In this case, the video model learns to exchange the motion condition from the first stream (with motion condition) into the target stream (without motion condition). Second, for the videos that exhibit both camera and object motion, we feed the condition into the video model as described in Sec. 3.1 and only supervise the second stream, leaving the loss at the first stream being zero. These two mixed-training strategies expose video models to real-world data with diverse camera and object motion, increasing the robustness and generalizability to various camera and motion configurations, while mitigating artifacts from synthetic data.

Rendered Graphics Data. We further incorporate synthetic data from SynCamMaster [3] to expose our model to more camera diversity.

3.4 Training and Inference

We train the DiT using the flow matching loss from Eq. (1), and apply two complementary dropout strategies to encourage the model learn the motion causality.

Multi-granularity motion dropout. Per Eq. (4), we randomly retain either $\boldsymbol{\tau}^{\text{act}}$ or $\boldsymbol{\tau}^{\text{pas}}$ to encourage causal reasoning. We further obtain multi-granularity

trajectories by averaging per-pixel trajectories within each patch. During training, we randomly select the granularity, enabling the model to capture both fine-grained pixel control and object-level manipulation.

Occlusion and track dropout. For the obtained trajectory, we randomly mask a subset of it to simulate occlusion and tracking failures that can happen during inference, improving robustness to missing or unreliable tracks.

Inference. At test time, users first specify motion by drawing sparse trajectories (simple curves or strokes on the first image) to indicate the desired direction and magnitude of movement, along with an optional text prompt and target camera poses $\{C_i\}_{i=1}^{T}$. We further perform occlusion-aware masking by approximating visibility ordering from the first-frame depth. The model then jointly denoises both streams, with the second stream being presented to the user.

4 Experiments

4.1 Implementation Details

We build upon the pretrained Wan2.1-14B [66] and fine-tune only the camera encoder, trajectory encoder, and self-attention layers. The trajectory embedding dim is 64, and the camera encoder uses 32 channels. We train the model with 15K iterations on 64 GPUs with a global batch size of 16, using AdamW [52] with a learning rate of 3×10^{-5} and weight decay 0.001. Trajectory dropout is set to 0.1 and text-conditioning dropout to 0.2.

Following the data curation pipeline in Sec. 3.3, we build our training data from large-scale public video datasets, including Panda-70M [13] and Wild-SDG-1M [35]. From these sources, we collect 76K static-view videos, from which we synthesize 43K paired dynamic-view videos using camera-controlled video-to-video generation, and 3.4K synthetic interaction videos from SyncMaster [3]. All videos are processed at 480p resolution. At inference, we sample 35 diffusion steps; generating one video takes approximately 15 minutes on a single A100 GPU. More implementation details are presented in supplement.

4.2 Experiment Settings

Evaluation metrics. We evaluate our model across four different aspects: *Video quality*: PSNR and SSIM against reference videos, and FID [32] and FVD [65] for distribution-level similarity. *Camera accuracy*: rotation and translation errors [2, 3, 30] between reference poses and poses estimated from generated videos using ViPE [35]; we report **median** errors across frames to mitigate estimation noise. *Motion accuracy*: end-point error (EPE) [15, 23], the ℓ_2 distance between ground-truth object tracks and predicted tracks extracted with AllTracker; we report the **median** EPE to reduce the impact of outlier tracks. *Motion realism*: Physical Commonsense (PC) and Semantic Adherence (SA) from VideoPhy [5], both 5-point scores normalized to $[0, 1]$. All evaluations are conducted at 480p resolution.

Evaluation Datasets. We evaluate on three datasets spanning diverse interaction scenarios. DynPose-100K [61] is an in-the-wild dataset with highly

Fig. 5: Disentangled camera-object control. MoRight controls object motion and camera viewpoint. Rows 1-3 fix the camera and vary object motion (rows 1-2: forward; row 3: inverse), while rows 4-6 fix object motion and vary camera motion.

dynamic camera motion; we manually select 50 videos exhibiting strong viewpoint changes and clear object interactions. WISA [69] is a large-scale physical-dynamics dataset; we select 50 videos from categories including collision, deformation, elasticity, liquid, and rigid-body motion. We further collect 50 real-world cooking videos, featuring complex hand-object interactions.

4.3 Disentangled Camera-Object Motion Control

Existing works on controllable video generation typically focus on either camera or object motion in isolation. Motion-conditioned baselines [15, 23, 67] typically receive privileged signals of both foreground and background tracks of all the pixel trajectories for control, while our method only uses reprojected trajectories defined on the canonical frame, without access to future-frame pixel trajectories. We compare our method with state-of-the-art baselines to evaluate the quality in motion control and camera control. While challenging, this evaluation allows us to demonstrate our model's unique ability to generate faithful motion from disentangled controls—a task that is fundamentally more difficult than baselines.

Baselines and setup. We compare with controllable video generation methods: base model Wan2.1 [66], Gen3C [60] for camera control, and motion-conditioned models MP [23], ATI [67], and WanMove [15] using dense pixel tracks. For fairness, we retrain Gen3C and MP in our setup, with all methods sharing the

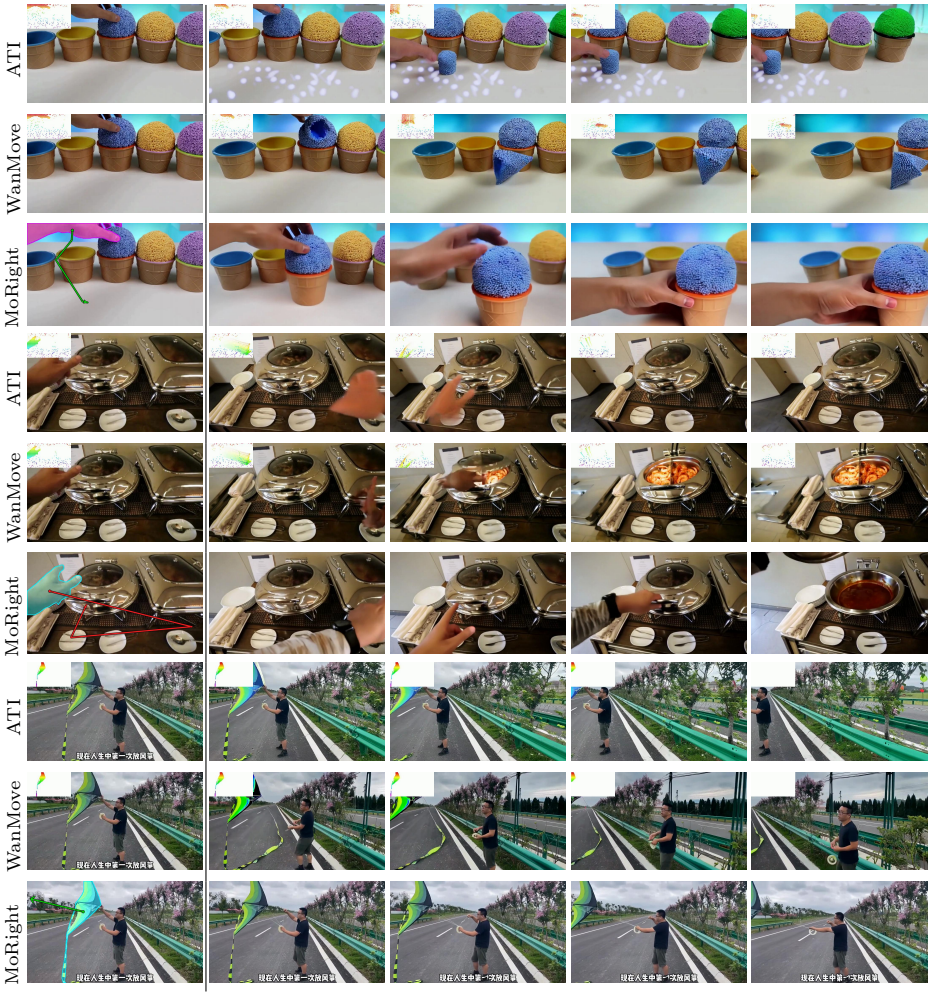

Fig. 6: Qualitative comparison of ATI [67], WanMove [15], and MoRight on interactive motion generation with camera control. All methods use the same input image. ATI and WanMove rely on pixel-aligned per-frame tracks (top-left), which entangle camera and object motion and require privileged future tracks. In contrast, MoRight uses only reprojected first-frame tracks. The first two rows show active motion reasoning, and the third row shows passive motion reasoning. Our model disentangles camera and object control and produces more coherent interactions.

Table 1: Controllable video generation on DynPose-100K [61] and Cooking. We compare models with camera and object motion control. Tracking-based methods (MP, ATI, WanMove) require privileged foreground/background tracks, while MoRight uses only first-frame reprojected trajectories and camera poses. Despite weaker inputs, MoRight achieves comparable visual quality and more accurate motion control. All methods use the Wan2.1-14B backbone; * denotes models reimplemented and trained by us. Best and second-best results are marked in **bold** and underline.

		DynPose-100K					Cooking				
Method	Full info	PSNR ↑	SSIM ↑	Rot ↓	Trans ↓	EPE ↓	PSNR ↑	SSIM ↑	Rot ↓	Trans ↓	EPE ↓
Wan2.1 [66]	×	11.23	0.435	-	-	–	14.23	0.527	-	-	–
Gen3c* [60]	×	12.45	0.507	5.46	4.09	-	15.37	**0.613**	**1.97**	**10.03**	-
MP* [23]	✓	11.72	0.455	6.76	6.04	7.56	15.68	0.564	2.50	12.24	**4.25**
ATI [67]	✓	13.18	0.493	5.62	6.54	8.43	15.93	0.582	4.25	16.94	5.87
WanMove [15]	✓	**13.91**	**0.521**	**4.12**	**3.56**	8.05	16.42	0.589	2.93	13.27	5.47
Ours	×	12.30	0.457	4.55	4.61	**7.64**	**16.44**	0.594	2.16	10.11	4.27

Wan2.1-14B backbone. We evaluate on DynPose-100K [61] and Cooking, using future-frame pixel-space EPE for motion accuracy.

Results. Quantitative results are provided in Tab. 1, with controllable results in Fig. 5, qualitative comparisons in Fig. 6. On DynPose-100K [61], WanMove [15] achieves the best overall numbers. Our method is slightly behind under highly dynamic camera motion, where errors in camera pose estimation and trajectory reprojection can degrade the input control signals. Nevertheless, MoRight attains comparable controllability to methods that rely on privileged future-frame tracking information and achieves the best EPE for object motion accuracy. We observe that ATI [67] and WanMove [15], which couple camera and object motion in a single tracking signal, tend to favor the dominant motion mode in highly dynamic settings—sometimes sacrificing camera accuracy or object tracking fidelity. On the Cooking benchmark, our method achieves the best overall performance in both visual quality and motion control accuracy. More controllable generation results and qualitative comparisons are provided in the supplement.

4.4 Motion Causality Modeling

Setup. We evaluate the causality of modeling motion on WISA [69] and Cooking, measuring both generation quality (FID, FVD) and motion realism (PC, SA). We compare with MP* [23], ATI [67], and WanMove [15], all following the same input protocol as Sec. 4.3. Every model receives *active motion* representing the user-specified action (e.g., a hand pushing an object); the goal is to generate plausible interaction outcomes. Baseline methods use their original prompts containing both motion descriptions and expected consequences. Our model receives only the active motion description, without specifying passive outcomes, and must infer the resulting interactions.

Results. Quantitative results are reported in Tab. 2. MoRight achieves the highest PC score on WISA, indicating strong physical commonsense, and the

Table 2: Interactive motion generation on WISA [69] and Cooking. We compare motion-conditioned video generation models on WISA and Cooking, evaluating video quality (FID, FVD) and motion realism (PC, SA). Prior methods require detailed motion captions with full interaction descriptions, while MoRight uses only a single active motion description yet achieves comparable quality with stronger physical commonsense reasoning. All methods use the Wan2.1-14B backbone for fair comparison; * denotes models reimplemented and trained by us. Best and second-best results are marked in **bold** and underline.

		WISA				Cooking			
Method	Full info	FID ↓	FVD ↓	PC ↑	SA ↑	FID ↓	FVD ↓	PC ↑	SA ↑
MP* [23]	✓	57.29	975.94	0.75	0.82	43.49	759.53	0.87	0.89
ATI [67]	✓	69.80	990.82	0.75	**0.83**	55.80	881.94	0.85	**0.90**
WanMove [15]	✓	61.34	1088.23	0.73	**0.83**	53.51	882.90	0.84	0.87
Ours	×	**52.95**	**876.03**	**0.76**	0.82	**39.94**	**730.46**	**0.88**	0.89

best video quality (FID, FVD) on both datasets. For SA, we rewrite the input prompt to remove passive motion descriptions and avoid information leakage; consequently, our score is slightly lower than methods that use full prompts containing both actions and outcomes, yet remains comparable. This confirms that MoRight's generations stay semantically aligned with intended outcomes while demonstrating genuine causal motion reasoning rather than relying on prompt-supplied answers. Qualitative examples of both reasoning modes—*forward* (action → reaction) and *inverse* (reaction → action)—are shown in Fig. 7. Our model generates plausible reactions when providing the active motion, and can reason the meaningful active motion when providing passive motion. More qualitative visualizations are shown in the supplement.

4.5 Human Perceptual Evaluation

We conduct a human study on 30 test examples to assess controllability, motion realism, and photorealism. Videos are shown in random order, and participants select the best result for each criterion, with ties and *None* allowed. After filtering unreliable responses, we obtain 330 evaluations per criterion from 11 participants. As shown in Fig. 8, our method is preferred in 53.5%, 54.6%, and 55.9% of cases for controllability, motion realism, and photorealism, outperforming ATI [67] and WanMove [15]. Notably, while these baselines use privileged full 3D trajectories, our method relies only on first-frame active trajectories, demonstrating the benefit of disentangled interaction reasoning.

4.6 Ablation Studies

We ablate model design, training strategies, and input conditions in Tab. 3.
Model design and training. *Cascaded pipeline* (row 1): a naive solution for disentangling camera-object motion is to first generate motion-controlled video under a static camera, followed by a Gen3C-style camera controller to move

Fig. 7: Causal interaction reasoning. In the first 3 rows, we provide active motion (*e.g.*, hand movement) as input, and the model infers the resulting passive motion (*e.g.*, cloth movement). In the last 3 rows, we provide passive motion (*e.g.*, ball movement), and the model infers the corresponding active motion (*e.g.*, human movement).

the camera. However, this approach introduces error accumulation between two stages, yielding larger control errors. *W/o fixed-view branch* (row 2): we only train with dynamic camera views and jointly encode the reprojected tracks and camera embeddings, removing the canonical-view anchor. The model struggles to disentangle camera and object motion, resulting in significantly worse camera and tracking accuracy. *W/o motion reasoning* (row 3): we disable active/passive decomposition during training. This approach increases FID/FVD and reduces PC, indicating degraded interaction quality. *W/o mixed supervision* (row 4): We only train the model on paired data. It slightly degrades camera accuracy, as the paired subset contains limited camera motion diversity.

Input conditions. We vary the motion input configuration to evaluate the robustness of our models. Specifically, we evaluate with coarse segment-level trajectories vs. fine-grained pixel tracks, as well as active vs. passive motion inputs. Performance remains stable across all settings, confirming that MoRight flexibly handles different motion granularities and types while maintaining strong controllability and causal reasoning capability.

4.7 Limitation

Despite promising results, our method has several limitations, as detailed in the supplement. It may produce incorrect interaction reasoning, unnatural motion

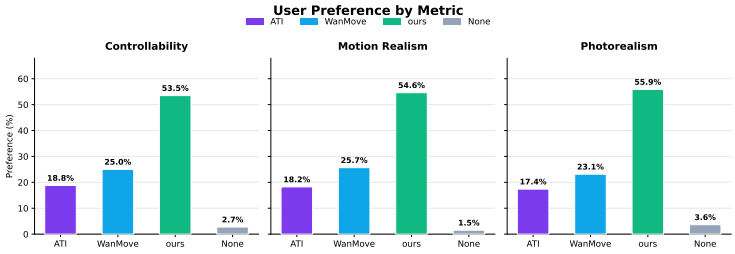

Fig. 8: Human perceptual evaluation. Across 330 responses from 11 participants, our method is preferred in controllability, motion realism, and photorealism, outperforming ATI [67] and WanMove [15].

Table 3: Ablation of motion controllability and reasoning on the Cooking benchmark. We ablate architectural choices, causal reasoning, hybrid training, and different motion input granularities. Our full model achieves the best overall performance across photometric quality, controllability, and motion realism, while remaining robust to different motion granularities and input conditions (active and passive).

	Photometric				Controllability			Motion	
Setting	FID ↓	FVD ↓	PSNR ↑	SSIM ↑	Rot ↓	Trans ↓	EPE ↓	PC ↑	SA ↑
cascaded	41.74	728.80	15.98	0.569	2.69	11.50	5.05	0.87	0.89
w/o fixed view	51.17	997.83	14.15	0.515	3.36	14.57	14.30	0.87	0.89
w/o reasoning	44.04	784.19	15.55	0.562	2.88	12.49	5.05	0.87	0.88
w/o mixed training	41.94	808.96	16.29	0.583	2.22	12.80	**4.09**	0.87	0.89
ours (coarse)	**39.83**	**725.88**	**16.45**	**0.594**	2.21	10.98	4.37	**0.88**	0.88
ours (passive)	44.20	838.67	15.99	0.588	2.21	11.04	7.27	0.87	0.88
ours (active)	39.94	730.46	16.44	0.594	**2.16**	**10.11**	4.27	**0.88**	0.89

under sparse trajectories, physical inconsistencies such as disappearing objects, or hallucinated content in later frames. The model also struggles with very complex or fast camera motion, where interaction dynamics may degrade.

5 Conclusion

We present MoRight, a unified framework for controllable, interaction-aware video generation. MoRight tackles entangled camera-object motion with a dual-stream design for independent trajectory and viewpoint control, and improves causal reasoning by decomposing motion into active and passive components to learn action-response dynamics. At inference, MoRight supports forward prediction from active motion and inverse reasoning from desired passive outcomes. Experiments on DynPose-100K, WISA, and Cooking demonstrate strong generation quality, motion control, and interaction awareness.

Acknowledgement We would like to thank Jiahui Huang, Zian Wang, Xiao Fu, and Chen-Hsuan Lin for their help and support with video data processing, data generation, and infrastructure.

References

1. Akkerman, R., Feng, H., Black, M.J., Tzionas, D., Abrevaya, V.F.: Interdyn: Controllable interactive dynamics with video diffusion models. In: CVPR. pp. 12467–12479 (2025)
2. Bai, J., Xia, M., Fu, X., Wang, X., Mu, L., Cao, J., Liu, Z., Hu, H., Bai, X., Wan, P., et al.: Recammaster: Camera-controlled generative rendering from a single video. arXiv preprint arXiv:2503.11647 (2025)
3. Bai, J., Xia, M., Wang, X., Yuan, Z., Fu, X., Liu, Z., Hu, H., Wan, P., Zhang, D.: Syncammaster: Synchronizing multi-camera video generation from diverse viewpoints. ICLR (2025)
4. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., Ge, W., Guo, Z., Huang, Q., Huang, J., Huang, F., Hui, B., Jiang, S., Li, Z., Li, M., Li, M., Li, K., Lin, Z., Lin, J., Liu, X., Liu, J., Liu, C., Liu, Y., Liu, D., Liu, S., Lu, D., Luo, R., Lv, C., Men, R., Meng, L., Ren, X., Ren, X., Song, S., Sun, Y., Tang, J., Tu, J., Wan, J., Wang, P., Wang, P., Wang, Q., Wang, Y., Xie, T., Xu, Y., Xu, H., Xu, J., Yang, Z., Yang, M., Yang, J., Yang, A., Yu, B., Zhang, F., Zhang, H., Zhang, X., Zheng, B., Zhong, H., Zhou, J., Zhou, F., Zhou, J., Zhu, Y., Zhu, K.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
5. Bansal, H., Lin, Z., Xie, T., Zong, Z., Yarom, M., Bitton, Y., Jiang, C., Sun, Y., Chang, K.W., Grover, A.: Videophy: Evaluating physical commonsense for video generation. arXiv preprint arXiv:2406.03520 (2024)
6. Bar, A., Zhou, G., Tran, D., Darrell, T., LeCun, Y.: Navigation world models. In: CVPR. pp. 15791–15801 (2025)
7. Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. In: CVPR. pp. 22563–22575 (2023)
8. Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., Ng, C., Wang, R., Ramesh, A.: Video generation models as world simulators. OpenAI technical reports (2024), https://openai.com/research/video-generation-models-as-world-simulators
9. Bu, Q., Cai, J., Chen, L., Cui, X., Ding, Y., Feng, S., Gao, S., He, X., Hu, X., Huang, X., et al.: Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. arXiv preprint arXiv:2503.06669 (2025)
10. Burgert, R., Xu, Y., Xian, W., Pilarski, O., Clausen, P., He, M., Ma, L., Deng, Y., Li, L., Mousavi, M., et al.: Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In: CVPR. pp. 13–23 (2025)
11. Chen, B., Jiang, H., Liu, S., Gupta, S., Li, Y., Zhao, H., Wang, S.: Physgen3d: Crafting a miniature interactive world from a single image. In: CVPR. pp. 6178–6189 (2025)
12. Chen, H., Zhang, Y., Cun, X., Xia, M., Wang, X., Weng, C., Shan, Y.: Videocrafter2: Overcoming data limitations for high-quality video diffusion models. arXiv preprint arXiv:2401.09047 (2024)
13. Chen, T.S., Siarohin, A., Menapace, W., Deyneka, E., Chao, H.w., Jeon, B.E., Fang, Y., Lee, H.Y., Ren, J., Yang, M.H., et al.: Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In: CVPR. pp. 13320–13331 (2024)
14. Chen, Y., Men, Y., Yao, Y., Cui, M., Bo, L.: Perception-as-control: Fine-grained controllable image animation with 3d-aware motion representation. In: ICCV. pp. 14380–14389 (2025)

15. Chu, R., He, Y., Chen, Z., Zhang, S., Xu, X., Xia, B., Wang, D., Yi, H., Liu, X., Zhao, H., et al.: Wan-move: Motion-controllable video generation via latent trajectory guidance. arXiv preprint arXiv:2512.08765 (2025)
16. Cosmos, T.: Cosmos world foundation model platform for physical ai. arXiv preprint arXiv:2501.03575 (2025)
17. Doersch, C., Yang, Y., Vecerik, M., Gokay, D., Gupta, A., Aytar, Y., Carreira, J., Zisserman, A.: Tapir: Tracking any point with per-frame initialization and temporal refinement. In: ICCV. pp. 10061–10072 (2023)
18. Finn, C., Levine, S.: Deep visual foresight for planning robot motion. In: IEEE Int. Conf. Robot. Autom. pp. 2786–2793. IEEE (2017)
19. Fu, X., Tang, S., Shi, M., Liu, X., Gu, J., Liu, M.Y., Lin, D., Lin, C.H.: Plenoptic video generation. arXiv preprint arXiv:2601.05239 (2025)
20. Fu, X., Tang, S., Shi, M., Liu, X., Gu, J., Liu, M.Y., Lin, D., Lin, C.H.: Plenoptic video generation. arXiv preprint arXiv:2601.05239 (2026)
21. Gao, Q., Xu, Q., Cao, Z., Mildenhall, B., Ma, W., Chen, L., Tang, D., Neumann, U.: Gaussianflow: Splatting Gaussian dynamics for 4D content creation. arXiv preprint arXiv:2403.12365 (2024)
22. Gao, S., Liang, W., Zheng, K., Malik, A., Ye, S., Yu, S., Tseng, W.C., Dong, Y., Mo, K., Lin, C.H., Ma, Q., Nah, S., Magne, L., Xiang, J., Xie, Y., Zheng, R., Niu, D., Tan, Y.L., Zentner, K., Kurian, G., Indupuru, S., Jannaty, P., Gu, J., Zhang, J., Malik, J., Abbeel, P., Liu, M.Y., Zhu, Y., Jang, J., Fan, L.J.: Dreamdojo: A generalist robot world model from large-scale human videos. arXiv preprint arXiv:2602.06949 (2026)
23. Geng, D., Herrmann, C., Hur, J., Cole, F., Zhang, S., Pfaff, T., Lopez-Guevara, T., Doersch, C., Aytar, Y., Rubinstein, M., Sun, C., Wang, O., Owens, A., Sun, D.: Motion prompting: Controlling video generation with motion trajectories. arXiv preprint arXiv:2412.02700 (2024)
24. Gillman, N., Herrmann, C., Freeman, M., Aggarwal, D., Luo, E., Sun, D., Sun, C.: Force prompting: Video generation models can learn and generalize physics-based control signals. arXiv preprint arXiv:2505.19386 (2025)
25. Gu, Z., Yan, R., Lu, J., Li, P., Dou, Z., Si, C., Dong, Z., Liu, Q., Lin, C., Liu, Z., et al.: Diffusion as shader: 3d-aware video diffusion for versatile video generation control. In: ACM SIGGRAPH (2025)
26. Hafner, D., Lillicrap, T., Ba, J., Norouzi, M.: Dream to control: Learning behaviors by latent imagination. arXiv preprint arXiv:1912.01603 (2019)
27. Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., Davidson, J.: Learning latent dynamics for planning from pixels. In: ICML. pp. 2555–2565. PMLR (2019)
28. Hafner, D., Pasukonis, J., Ba, J., Lillicrap, T.: Mastering diverse domains through world models. arXiv preprint arXiv:2301.04104 (2023)
29. Harley, A.W., You, Y., Sun, X., Zheng, Y., Raghuraman, N., Gu, Y., Liang, S., Chu, W.H., Dave, A., You, S., et al.: Alltracker: Efficient dense point tracking at high resolution. In: ICCV. pp. 5253–5262 (2025)
30. He, H., Xu, Y., Guo, Y., Wetzstein, G., Dai, B., Li, H., Yang, C.: Cameractrl: Enabling camera control for text-to-video generation. arXiv preprint arXiv:2404.02101 (2024)
31. He, X., Peng, C., Liu, Z., Wang, B., Zhang, Y., Cui, Q., Kang, F., Jiang, B., An, M., Ren, Y., et al.: Matrix-game 2.0: An open-source real-time and streaming interactive world model. arXiv preprint arXiv:2508.13009 (2025)

32. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: NeurIPS (2017)
33. Hong, W., Ding, M., Zheng, W., Liu, X., Tang, J.: Cogvideo: Large-scale pretraining for text-to-video generation via transformers. arXiv preprint arXiv:2205.15868 (2022)
34. Hong, Y., Mei, Y., Ge, C., Xu, Y., Zhou, Y., Bi, S., Hold-Geoffroy, Y., Roberts, M., Fisher, M., Shechtman, E., et al.: Relic: Interactive video world model with long-horizon memory. arXiv preprint arXiv:2512.04040 (2025)
35. Huang, J., Zhou, Q., Rabeti, H., Korovko, A., Ling, H., Ren, X., Shen, T., Gao, J., Slepichev, D., Lin, C.H., et al.: Vipe: Video pose engine for 3d geometric perception. arXiv preprint arXiv:2508.10934 (2025)
36. Huang, N., Zheng, W., Xu, C., Keutzer, K., Zhang, S., Kanazawa, A., Wang, Q.: Segment any motion in videos. In: CVPR. pp. 3406–3416 (2025)
37. Jiang, H., Hsu, H.Y., Zhang, K., Yu, H.N., Wang, S., Li, Y.: Phystwin: Physics-informed reconstruction and simulation of deformable objects from videos. In: ICCV. pp. 7219–7230 (2025)
38. Jin, W., Dai, Q., Luo, C., Baek, S.H., Cho, S.: Flovd: Optical flow meets video diffusion model for enhanced camera-controlled video synthesis. In: CVPR (2025)
39. Karaev, N., Rocco, I., Graham, B., Neverova, N., Vedaldi, A., Rupprecht, C.: Cotracker: It is better to track together. In: ECCV (2024)
40. Kopf, J., Rong, X., Huang, J.B.: Robust consistent video depth estimation. In: CVPR. pp. 1611–1621 (2021)
41. Li, Q., Xing, Z., Wang, R., Zhang, H., Dai, Q., Wu, Z.: Magicmotion: Controllable video generation with dense-to-sparse trajectory guidance. In: ICCV. pp. 12112–12123 (2025)
42. Li, Z., Niklaus, S., Snavely, N., Wang, O.: Neural scene flow fields for space-time view synthesis of dynamic scenes. In: CVPR (2021)
43. Li, Z., Tucker, R., Cole, F., Wang, Q., Jin, L., Ye, V., Kanazawa, A., Holynski, A., Snavely, N.: Megasam: Accurate, fast and robust structure and motion from casual dynamic videos. In: CVPR. pp. 10486–10496 (2025)
44. Li, Z., Yu, H.X., Liu, W., Yang, Y., Herrmann, C., Wetzstein, G., Wu, J.: Wonderplay: Dynamic 3d scene generation from a single image and actions. In: ICCV. pp. 9080–9090 (2025)
45. Lian, L., Shi, B., Yala, A., Darrell, T., Li, B.: Llm-grounded video diffusion models. arXiv preprint arXiv:2309.17444 (2023)
46. Liang, F., Wu, B., Wang, J., Yu, L., Li, K., Zhao, Y., Misra, I., Huang, J.B., Zhang, P., Vajda, P., et al.: Flowvid: Taming imperfect optical flows for consistent video-to-video synthesis. In: CVPR. pp. 8207–8216 (2024)
47. Lin, H., Chen, S., Liew, J., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: Recovering the visual space from any views. arXiv preprint arXiv:2511.10647 (2025)
48. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. arXiv preprint arXiv:2210.02747 (2022)
49. Liu, S., Guo, C., Zhou, B., Wang, J.: Ponimator: Unfolding interactive pose for versatile human-human interaction animation. In: ICCV. pp. 12068–12077 (2025)
50. Liu, S., Ren, Z., Gupta, S., Wang, S.: Physgen: Rigid-body physics-grounded image-to-video generation. In: ECCV. pp. 360–378. Springer (2024)
51. Liu, S., Yao, D.Y., Gupta, S., Wang, S.: Visual sync: Multi-camera synchronization via cross-view object motion. arXiv preprint arXiv:2512.02017 (2025)

52. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
53. Lv, J., Huang, Y., Yan, M., Huang, J., Liu, J., Liu, Y., Wen, Y., Chen, X., Chen, S.: Gpt4motion: Scripting physical motions in text-to-video generation via blender-oriented gpt planning. In: CVPR. pp. 1430–1440 (2024)
54. Ma, W.D.K., Lewis, J.P., Kleijn, W.B.: Trailblazer: Trajectory control for diffusion-based video generation. arXiv preprint arXiv:2401.00896 (2023)
55. Montanaro, A., Savant Aira, L., Aiello, E., Valsesia, D., Magli, E.: Motioncraft: Physics-based zero-shot video generation. NeurIPS **37**, 123155–123181 (2024)
56. Niu, M., Cun, X., Wang, X., Zhang, Y., Shan, Y., Zheng, Y.: Mofa-video: Controllable image animation via generative motion field adaptions in frozen image-to-video diffusion model. In: ECCV. pp. 111–128. Springer (2024)
57. Pan, C., Yaman, B., Nesti, T., Mallik, A., Allievi, A.G., Velipasalar, S., Ren, L.: Vlp: Vision language planning for autonomous driving. In: CVPR. pp. 14760–14769 (2024)
58. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV. pp. 4195–4205 (2023)
59. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714 (2024), https://arxiv.org/abs/2408.00714
60. Ren, X., Shen, T., Huang, J., Ling, H., Lu, Y., Nimier-David, M., Müller, T., Keller, A., Fidler, S., Gao, J.: Gen3c: 3d-informed world-consistent video generation with precise camera control. In: CVPR. pp. 6121–6132 (2025)
61. Rockwell, C., Tung, J., Lin, T.Y., Liu, M.Y., Fouhey, D.F., Lin, C.H.: Dynamic camera poses and where to find them. In: CVPR. pp. 12444–12455 (2025)
62. Shi, X., Huang, Z., Wang, F.Y., Bian, W., Li, D., Zhang, Y., Zhang, M., Cheung, K.C., See, S., Qin, H., et al.: Motion-i2v: Consistent and controllable image-to-video generation with explicit motion modeling. In: ACM SIGGRAPH. pp. 1–11 (2024)
63. Siarohin, A., Lathuilière, S., Tulyakov, S., Ricci, E., Sebe, N.: First order motion model for image animation. In: NeurIPS (2019)
64. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: Mocogan: Decomposing motion and content for video generation. In: CVPR (2018)
65. Unterthiner, T., van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., Gelly, S.: Towards accurate generative models of video: A new metric & challenges. arXiv preprint arXiv:1812.01717 (2018)
66. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
67. Wang, A., Huang, H., Fang, Z., Yang, Y., Ma, C.: Ati: Any trajectory instruction for controllable video generation. arXiv preprint **arXiv:2505.22944** (2025)
68. Wang, J., Zhang, Y., Zou, J., Zeng, Y., Wei, G., Yuan, L., Li, H.: Boximator: Generating rich and controllable motions for video synthesis. arXiv preprint arXiv:2402.01566 (2024)
69. Wang, J., Ma, A., Cao, K., Zheng, J., Zhang, Z., Feng, J., Liu, S., Ma, Y., Cheng, B., Leng, D., et al.: Wisa: World simulator assistant for physics-aware text-to-video generation. arXiv preprint arXiv:2503.08153 (2025)

70. Wang, X., Zhu, Z., Huang, G., Chen, X., Zhu, J., Lu, J.: Drivedreamer: Towards real-world-drive world models for autonomous driving. In: ECCV. pp. 55–72. Springer (2024)
71. Wang, Y., Bilinski, P., Bremond, F., Dantcheva, A.: G3AN: Disentangling appearance and motion for video generation. In: CVPR (2020)
72. Wang, Y., Bremond, F., Dantcheva, A.: Inmodegan: Interpretable motion decomposition generative adversarial network for video generation. arXiv preprint arXiv:2101.03049 (2021)
73. Wang, Z., Yuan, Z., Wang, X., Chen, T., Xia, M., Luo, P., Shan, Y.: Motionctrl: A unified and flexible motion controller for video generation. In: ACM SIGGRAPH (2024)
74. Wu, T.H., Lian, L., Gonzalez, J.E., Li, B., Darrell, T.: Self-correcting llm-controlled diffusion models. In: CVPR. pp. 6327–6336 (2024)
75. Wu, W., Li, Z., Gu, Y., Zhao, R., He, Y., Zhang, D.J., Shou, M.Z., Li, Y., Gao, T., Zhang, D.: Draganything: Motion control for anything using entity representation. In: ECCV (2024)
76. Xing, J., Mai, L., Ham, C., Huang, J., Mahapatra, A., Fu, C.W., Wong, T.T., Liu, F.: Motioncanvas: Cinematic shot design with controllable image-to-video generation. In: ACM SIGGRAPH (2025)
77. Yang, M., Du, Y., Ghasemipour, K., Tompson, J., Schuurmans, D., Abbeel, P.: Learning interactive real-world simulators. arXiv preprint arXiv:2310.06114 **1**(2), 6 (2023)
78. Yang, X., Li, B., Zhang, Y., Yin, Z., Bai, L., Ma, L., Wang, Z., Cai, J., Wong, T.T., Lu, H., et al.: Vlipp: Towards physically plausible video generation with vision and language informed physical prior. In: ICCV. pp. 12360–12370 (2025)
79. Yang, Z., Ge, W., Li, Y., Chen, J., Li, H., An, M., Kang, F., Xue, H., Xu, B., Yin, Y., et al.: Matrix-3d: Omnidirectional explorable 3d world generation. arXiv preprint arXiv:2508.08086 (2025)
80. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al.: Cogvideox: Text-to-video diffusion models with an expert transformer. arXiv preprint arXiv:2408.06072 (2024)
81. Yao, D.Y., Zhai, A.J., Wang, S.: Uni4d: Unifying visual foundation models for 4d modeling from a single video. In: CVPR. pp. 1116–1126 (2025)
82. Ye, S., Ge, Y., Zheng, K., Gao, S., Yu, S., Kurian, G., Indupuru, S., Tan, Y.L., Zhu, C., Xiang, J., et al.: World action models are zero-shot policies. arXiv preprint arXiv:2602.15922 (2026)
83. Yin, S., Wu, C., Liang, J., Shi, J., Li, H., Ming, G., Duan, N.: Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. arXiv preprint arXiv:2308.08089 (2023)
84. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: ICCV. pp. 3836–3847 (2023)
85. Zhang, Q., Wang, C., Siarohin, A., Zhuang, P., Xu, Y., Yang, C., Lin, D., Zhou, B., Tulyakov, S., Lee, H.Y.: SceneWiz3D: Towards text-guided 3D scene composition. In: CVPR (2024)
86. Zhang, T., Yu, H.X., Wu, R., Feng, B.Y., Zheng, C., Snavely, N., Wu, J., Freeman, W.T.: Physdreamer: Physics-based interaction with 3d objects via video generation. In: ECCV (2024)
87. Zhang, Z., Long, F., Qiu, Z., Pan, Y., Liu, W., Yao, T., Mei, T.: Motionpro: A precise motion controller for image-to-video generation. In: CVPR. pp. 27957–27967 (2025)

88. Zhang, Z., Cole, F., Li, Z., Rubinstein, M., Snavely, N., Freeman, W.T.: Structure and motion from casual videos. In: ECCV. pp. 20–37. Springer (2022)
89. Zhao, W., Bai, L., Rao, Y., Zhou, J., Lu, J.: Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. NeurIPS **36**, 49842–49869 (2023)