VerseCrafter: Dynamic Realistic Video World Model with 4D Geometric Control

Sixiao Zheng[1,2] , Minghao Yin[3], Wenbo Hu[4†], Xiaoyu Li[4], Ying Shan[4], Yanwei Fu[1,2†]
[1]Fudan University [2]Shanghai Innovation Institute [3]HKU [4]ARC Lab, Tencent PCG

Project Page: https://sixiaozheng.github.io/VerseCrafter_page/

Figure 1. **VerseCrafter** enables precise control of camera motion and multi-object motion via a 4D Geometric Control representation built from a static background point cloud and per-object 3D Gaussian trajectories, producing videos that better follow the desired motion than Yume [63] and Uni3C [11] and more closely match the ground-truth video.

Abstract

*Video world models aim to simulate dynamic, real-world environments, yet existing methods struggle to provide unified and precise control over camera and multi-object motion, as videos inherently capture dynamics in the projected 2D image plane. To bridge this gap, we introduce **VerseCrafter**, a geometry-driven video world model that generates dynamic, realistic videos from a unified 4D geometric world state. Our approach is centered on a novel **4D Geometric Control** representation, which encodes the world state as a static background point cloud and per-object 3D Gaussian trajectories. This representation captures each object's motion path and probabilistic 3D occupancy over time, providing a flexible, category-agnostic alternative to rigid bounding boxes and parametric models. We render 4D Geometric Control into 4D control maps for a pre-trained video diffusion model, enabling high-fidelity, view-consistent video generation that faithfully follows the specified dynamics. To enable training at scale, we develop an automatic data engine and construct **VerseControl4D**, a real-world dataset of 35K training samples with automatically derived prompts and rendered 4D control maps. Extensive experiments show that VerseCrafter achieves superior visual quality and more accurate control over camera and multi-object motion than prior methods.*

†Corresponding authors.

1. Introduction

Video world models learn to simulate world dynamics by generating future frame sequences conditioned on past observations and control signals, such as actions or camera trajectories [13, 30, 41, 48, 63]. They provide a unified interface for visual prediction [32], navigation [7], and manipulation [23]. However, the reliance on video introduces a fundamental challenge: while an ideal world model should simulate the full 4D spatiotemporal space to reflect our physical reality, videos inherently capture dynamics in the projected 2D image plane.

To bridge this gap, recent works introduce camera control into video generation through explicit 3D geometry [11, 117, 127], implicit pose embeddings [52], or learned movement embeddings [9, 12, 70]. However, these methods are often limited to static scenes or leave multi-object motion uncontrolled. Existing approaches typically rely on 2D cues such as point trajectories [98], optical flow [58], masks [125], or bounding boxes [91], which lack 3D awareness and often fail under large viewpoint changes. More advanced 3D-aware methods use depth maps [124], sparse 3D trajectories [15], 3D bounding boxes [94], or parametric human models like SMPL-X [11] to align camera and object motion in 3D space. Nevertheless, these control representations remain inadequate for modeling multi-object dynamics as a unified, compact, and editable 4D geometric scene state in a shared world coordinate frame. For instance, sparse trajectories are often noisy and incomplete, 3D bounding boxes impose rigid constraints ill-suited to natural objects, and SMPL-X representations are category-limited. Furthermore, several existing works focus on synthetic game environments [41, 111, 115], where precise annotations are available for training. However, controllable modeling of complex, realistic 4D scenes with multi-object motion remains underexplored.

Thus we propose *VerseCrafter*, a realistic, dynamic video world model that enables precise control of camera and multi-object motion within a unified 4D geometric world state, as shown in Fig. 1. At the core of VerseCrafter is our *4D Geometric Control* representation, which represents the scene state as a static background point cloud for scene geometry and per-object 3D Gaussian trajectories to capture object dynamics. Each 3D Gaussian trajectory models an object's probabilistic 3D occupancy over time: its mean defines the motion path, while its covariance captures the object's spatial extent and orientation. This probabilistic formulation provides a soft, flexible, and category-agnostic way to model diverse object shapes and motions, overcoming the limitations of rigid 3D bounding boxes or category-specific parametric models. Crucially, the background point cloud and per-object 3D Gaussian trajectories share a common world coordinate frame, enabling coherent and unified control over both camera and object motion.

By rendering 4D Geometric Control into multi-channel 4D control maps, we condition a frozen Wan2.1-14B video diffusion backbone [88] via a lightweight GeoAdapter, an adapter-style branch inspired by ControlNet [119]. This conditioning enables the generation of high-fidelity videos that faithfully reflect the explicit 4D geometric world state. Unlike 2D control signals, our 4D Geometric Control is inherently 3D-aware, making it naturally more view-consistent and robust to occlusions, and thus a more effective and reliable interface for video world modeling. Training VerseCrafter requires large-scale paired data of real-world videos and corresponding 4D geometric control. To this end, we construct *VerseControl4D*, a real-world video dataset with automatically derived prompts and rendered 4D control maps. This dataset supports large-scale training on diverse real-world videos.

Our contributions are threefold:

- We introduce a novel ***4D Geometric Control*** representation that unifies camera and multi-object motion in a shared world coordinate frame. By using 3D Gaussian trajectories, it provides a flexible and category-agnostic way to control object dynamics, overcoming the limitations of rigid, category-specific models.
- We present ***VerseCrafter***, a geometry-driven video world model that leverages 4D Geometric Control for precise control over camera and multi-object motion. This enables the generation of high-fidelity, view-consistent videos that accurately follow complex 4D controls.
- We construct ***VerseControl4D***, a real-world dataset with automatically derived prompts and rendered 4D control maps, with 35K training samples. This addresses a key data bottleneck and supports large-scale training on diverse real-world videos.

2. Related Works

Video World Models. World models learn environment dynamics from observations by predicting future states for simulation, planning, and control [30, 31, 48]. Early visual world models adopt recurrent and latent-variable architectures [16, 24, 29, 64, 68, 87], while recent approaches use transformer and diffusion backbones to roll out realistic videos conditioned on actions, text, or camera trajectories [1, 2, 6, 9, 12, 20, 36, 40, 45, 50, 70, 88, 103, 110, 115], and further extend temporal horizons with memories or long-sequence models [52, 73, 101]. Geometry-aware works such as DeepVerse [13], Voyager [41], and Yume [63] incorporate 3D geometry to support 4D video generation and world exploration, but are controlled via text, actions, or camera tokens and do not expose a compact, editable 4D geometric state for real-world multi-object dynamics. In contrast, VerseCrafter learns a geometry-driven mapping from 4D Geometric Control to dynamic, realistic videos, enabling disentangled control over camera and multi-object motion.

3D World Generation. Recent work leverages powerful 2D generative priors to synthesize explorable 3D environments from text, images, or videos [49, 121]. Early methods mainly focus on object-level or single-scene generation [19, 38, 74, 118, 120, 122], distilling image diffusion models [79] into NeRFs [65], implicit fields, meshes, or 3D Gaussian splats [44], or optimizing scene geometry from multi-view or panoramic observations [18, 80, 113, 114, 116]. More recent approaches scale up to navigable 3D worlds [7], combining depth estimation [108], camera-guided video diffusion, iterative inpainting, and panoramic inputs to construct room- or city-scale Gaussian scenes for exploration [14, 54, 60, 61, 81, 86, 92, 111, 129]. However, these pipelines largely model static, synthetic-like scenes and provide limited explicit control over real-world multi-object dynamics. In contrast, VerseCrafter operates on real-world videos and represents the scene with a static background point cloud and per-object 3D Gaussian trajectories, forming an explicit 4D geometric scene state for geometry-consistent dynamic video generation.

Controllable Video Generation. Controllable video generation aims to steer camera and object motion via conditioning signals. Camera-controlled models [3, 4, 34, 46, 53, 84, 106, 126] such as MotionCtrl [98] and CameraCtrl [33] inject camera extrinsics, Plücker-style encodings, or 3D priors [11, 21, 28, 39, 75, 78, 99, 104, 117, 124, 127] into video diffusion models for viewpoint control, but mostly assume static or weakly dynamic scenes. Object motion [10, 25, 35, 51, 55, 57, 62, 66, 67, 69, 76, 82, 83, 85, 89, 90, 96, 97, 100, 107, 109, 112, 123, 128] is typically controlled using 2D cues (bounding boxes, masks, trajectories, strokes, optical flow) as in Boximator [91], DragAnything [102], and MotionCanvas [105], or with more 3D-aware signals such as depth maps, sparse 3D trajectories, 3D boxes, or SMPL-X bodies in I2V3D [124], Uni3C [11], CineMaster [94], Perception-as-Control [15], and LongVie [26]. While these methods improve controllability, 2D controls remain view-dependent and fragile under large camera changes, and many 3D controls are category-specific, rigid, or tied to reconstruction-heavy pipelines. Recent approaches [15, 22, 27, 58, 94, 98, 105, 109, 127] begin to jointly control camera and object motion, but their control spaces are still fragmented rather than a unified, compact world state. VerseCrafter instead introduces *4D Geometric Control*: a compact, category-agnostic 4D geometric scene state where a static background point cloud and per-object 3D Gaussian trajectories in a shared world coordinate frame jointly drive camera and multi-object motion.

3. Method

We propose **VerseCrafter**, a geometry-driven video world model that generates dynamic, realistic videos from an explicit 4D geometric scene state while enabling disentangled control over camera and multi-object motion. Our framework has two key components: (i) a unified *4D Geometric Control* representation (Sec. 3.1), which represents the 4D geometric scene state in a shared world coordinate frame, and (ii) a lightweight *GeoAdapter* (Sec. 3.2), which injects encoded 4D control maps into a frozen Wan2.1-14B backbone while preserving its strong visual prior. Given an input image and a prompt, we construct 4D Geometric Control as a static background point cloud and per-object 3D Gaussian trajectories, specify a camera trajectory in the shared frame, render them into 4D control maps, and feed these maps into GeoAdapter to generate dynamic, realistic videos.

3.1. 4D Geometric Control

We represent each scene as an explicit 4D geometric scene state, which we term *4D Geometric Control*. This editable representation consists of a static background point cloud P^{bg} and per-object 3D Gaussian trajectories $\{\mathcal{G}_o^t\}$, all defined in a shared world coordinate frame.

Background point cloud. As shown in Fig. 2, we start from the input image, estimate monocular depth and camera intrinsics $\mathbf{K}$ using MoGe-2 [95], and obtain object masks $\{M_o\}$ with Grounded SAM2 [77], where the user selects one or more objects to control via text prompts or clicks. We take the input view as the reference world coordinate frame, so that the reference camera pose is given by $\mathbf{R}_1 = \mathbf{I}$ and $\mathbf{t}_1 = \mathbf{0}$. Each pixel $\mathbf{u} = (u, v, 1)^\top$ with depth $D_1(\mathbf{u})$ is then back-projected as

$$\mathbf{p}(\mathbf{u}) = \mathbf{R}_1^\top \big(D_1(\mathbf{u})\mathbf{K}^{-1}\mathbf{u} - \mathbf{t}_1\big). \tag{1}$$

We use the object masks to partition the reconstructed point cloud into per-object point clouds

$$P_o = \big\{\mathbf{x}_{o,k} \,\big|\, \mathbf{x}_{o,k} = \mathbf{p}(\mathbf{u}_k),\ \mathbf{u}_k \in M_o\big\}, \tag{2}$$

and a static background point cloud

$$P^{\text{bg}} = \big\{\mathbf{p}(\mathbf{u}) \,\big|\, \mathbf{u} \notin \bigcup_o M_o\big\} = \{\mathbf{p}_i\}_{i=1}^{N_{\text{bg}}}. \tag{3}$$

During generation, the background at frame t is rendered from P^{bg} under the camera pose, so viewpoint changes are realized as rigid camera motion in a fixed 3D world rather than by hallucinating a new background at every frame.

3D Gaussian trajectories. A single 3D Gaussian $\mathcal{G}_o(\mathbf{x}) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_o, \boldsymbol{\Sigma}_o)$ in the world coordinate frame compactly encodes an object's position (through $\boldsymbol{\mu}_o$), approximate shape and size (through the eigenvalues of $\boldsymbol{\Sigma}_o$), and orientation (through the eigenvectors of $\boldsymbol{\Sigma}_o$). *A 3D Gaussian trajectory for an object o* is then defined as a sequence of Gaussians

$$\{\mathcal{G}_o^t\}_{t=1}^T, \quad \mathcal{G}_o^t(\mathbf{x}) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_o^t, \boldsymbol{\Sigma}_o^t), \tag{4}$$

whose means $\{\boldsymbol{\mu}_o^t\}$ trace the motion path in 3D, while the covariances $\{\boldsymbol{\Sigma}_o^t\}$ capture how the object's spatial extent

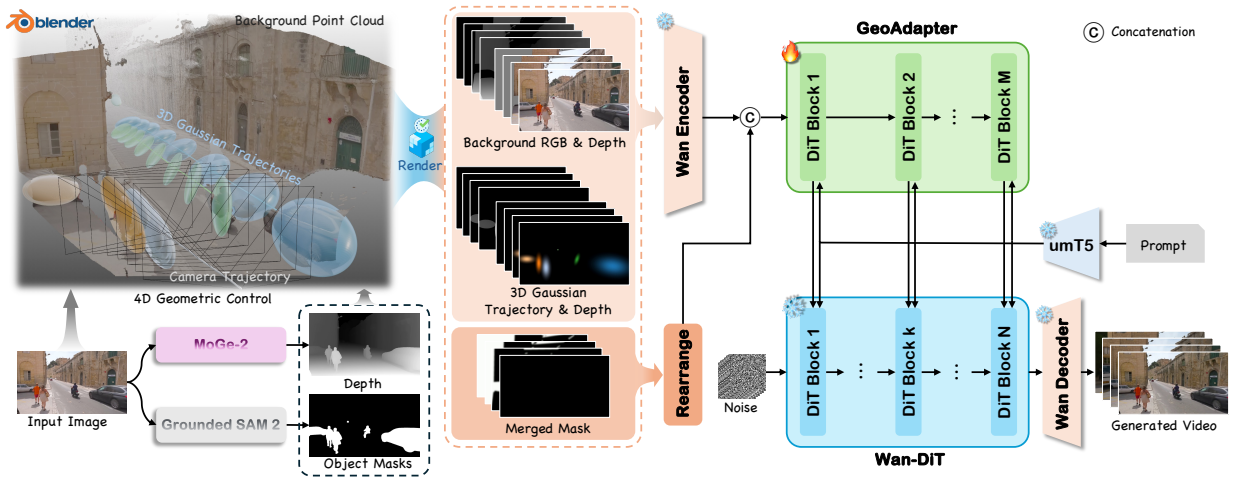

Figure 2. **Framework of VerseCrafter.** Given an input image and a text prompt, we estimate depth and obtain user-specified object masks to construct *4D Geometric Control* consisting of a static background point cloud and per-object 3D Gaussian trajectories in a shared world coordinate frame. A camera trajectory is specified in the shared frame, and together with the 4D Geometric Control, rendered into per-frame background RGB/depth, 3D Gaussian trajectory RGB/depth, and a soft merged mask, forming multi-channel 4D control maps. The 4D control maps are encoded and fed into the proposed GeoAdapter, which conditions a frozen Wan2.1-14B backbone together with text embeddings from umT5, enabling geometry-consistent video generation with precise control over camera and multi-object motion.

and orientation evolve over time. This probabilistic formulation describes the object's 3D occupancy in a soft, continuous manner and yields a compact control space that is more flexible than rigid 3D bounding boxes and more category-agnostic than parametric body models.

To initialize the trajectory for each controllable object o, we fit a full-covariance Gaussian to its point cloud P_o obtained in the previous step:

$$\boldsymbol{\mu}_o = \frac{1}{N_o}\sum_k \mathbf{x}_{o,k}, \boldsymbol{\Sigma}_o = \frac{1}{N_o}\sum_k (\mathbf{x}_{o,k}-\boldsymbol{\mu}_o)(\mathbf{x}_{o,k}-\boldsymbol{\mu}_o)^\top, \tag{5}$$

which gives an initial Gaussian $\mathcal{G}_o(\mathbf{x})$

The low-dimensional parameters $\{\boldsymbol{\mu}_o^t, \boldsymbol{\Sigma}_o^t\}$ naturally support flexible, user-driven editing. In practice, we convert each $\mathcal{G}_o^t$ into an ellipsoid mesh for visualization in a 3D editor such as Blender, and let the user specify or refine the trajectory by dragging and keyframing this ellipsoid in world coordinate space. The edited poses and shapes are mapped back to $\{\boldsymbol{\mu}_o^t, \boldsymbol{\Sigma}_o^t\}$ as control signals. The ellipsoids are only a user interface; all conditioning maps used by model are rendered directly from the underlying 3D Gaussians.

Rendering 4D control maps. Given our 4D Geometric Control, we render per-frame 4D control maps in the target camera views. For each frame t, we generate three types of maps: (i) background RGB/depth, RGB_t^{bg} and $\text{Depth}_t^{\text{bg}}$, by projecting the static background point cloud P^{bg} under the camera pose $(\mathbf{R}_t, \mathbf{t}_t)$; (ii) 3D Gaussian trajectory RGB/depth, $\text{RGB}_t^{\text{traj}}$ and $\text{Depth}_t^{\text{traj}}$, by projecting the per-object Gaussians $\{\mathcal{G}_o^t\}$ into soft elliptical footprints and taking depth from the corresponding ellipsoid surfaces; and (iii) a soft merged mask M_t, used as a control mask, that indicates regions where the diffusion model should synthesize or overwrite content, obtained by inverting background visibility and merging it with the projected 3D Gaussian footprints, followed by Gaussian smoothing. For the first frame $t = 1$, we replace RGB_1^{bg} with input image and set $M_1 = 0$, so that the first frame is preserved and only future frames are modified. Background and 3D Gaussian maps share the same 4D geometric scene state but are rendered through *decoupled* channels, disentangling camera motion from object motion while preserving geometric consistency.

3.2. VerseCrafter Architecture

Backbone. We adopt Wan2.1-14B [88] as a frozen latent video diffusion backbone with a Wan Encoder, a Wan-DiT denoiser and a Wan Decoder. VerseCrafter treats Wan2.1 as a generic video prior: we do not change its architecture or weights, and instead attach a lightweight geometric adapter that conditions the backbone on rendered 4D control maps.

GeoAdapter. We take the rendered background maps and 3D Gaussian trajectory maps, RGB^{bg}, Depth^{bg}, RGB^{traj}, $\text{Depth}^{\text{traj}}$, together with the soft merged mask M. The four RGB/depth maps are encoded by the same Wan Encoder, while M is reshaped and interpolated to the latent resolution, following the practice in [42, 88]. The encoded RGB/depth maps and the processed mask are concatenated channel-wise to form a spatio-temporal geometry tensor. GeoAdapter is a lightweight DiT-style branch that operates on this geometry tensor. It shares the same hidden dimensionality as the Wan-DiT blocks, but uses far fewer layers. We interleave GeoAdapter blocks with the frozen Wan-DiT: every k-th DiT block in Wan-DiT is paired with a GeoAd-

apter block whose output is linearly projected and added to the corresponding DiT block as a residual modulation. Text prompts are encoded by umT5 [17] into text embeddings, which are injected into both Wan-DiT and GeoAdapter blocks through the same text-conditioning interfaces. This adapter-based conditioning injects 4D geometric information into Wan 2.1 with only a small number of extra parameters, while keeping all backbone weights fixed.

Inference. At inference time, VerseCrafter supports camera-only, object-only, and joint control within a unified framework. For *camera-only* control, we render the background control maps, while setting the 3D Gaussian trajectory RGB/depth maps to zero. The merged mask may still be nonzero due to viewpoint changes. For *object-only* control, we keep the camera pose fixed, render static background control maps from P^{bg}, and render the 3D Gaussian trajectory maps to control object motion. For *joint* control, both background and trajectory control maps are rendered from the same 4D geometric scene state, enabling coordinated and geometry-consistent control over camera and multi-object motion.

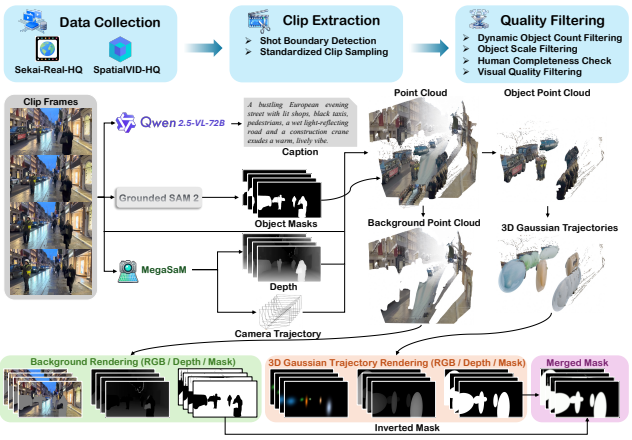

Figure 3. **Construction pipeline of VerseControl4D.** Starting from Sekai-Real-HQ and SpatialVID-HQ, we extract 81-frame clips and apply quality filtering. For each retained clip, Qwen2.5-VL-72B, Grounded-SAM2, and MegaSAM provide captions, object masks, depth, and camera trajectory, which are lifted into background/object point clouds, from which 3D Gaussian trajectories are fitted, and then rendered into background/trajectory maps and a soft merged mask that constitute our 4D control maps.

4. VerseControl4D Dataset

To train and evaluate VerseCrafter on complex real-world scenes with 4D Geometric Control, we construct **VerseControl4D**, a real-world dataset with automatically derived prompts and rendered 4D control maps. As shown in Fig. 3, VerseControl4D is built through four stages: data collection, clip extraction, quality filtering, and data annotation.

Data collection. VerseControl4D is built from two recent world-exploration datasets, Sekai-Real-HQ [56] and SpatialVID-HQ [93], which provide long in-the-wild videos with diverse outdoor and urban scenes, camera poses, and captions, but lack object-motion labels. We use their high-resolution videos as the raw video pool for constructing our 4D geometric control annotations.

Clip extraction. We apply PySceneDetect to detect shots in the videos. For each shot longer than 81 frames, we uniformly sample an 81-frame sub-clip and discard shorter shots, matching the default temporal length used by the Wan2.1 backbone.

Quality filtering. We apply an object-centric filtering pipeline to retain clips with clean geometry and controllable foreground. Using Grounded-SAM2 with prompts such as *"person . human . car . animal"*, we first obtain object masks on the first frame, and keep only clips whose controllable object count lies in $[1, 6]$. We then discard clips where any object mask covers more than 20% of the image area. For human instances, we further remove clips whose masks touch image borders or whose aspect ratios fall outside $[2, 4]$, as these typically correspond to severely truncated pedestrians. Finally, we apply visual-quality filtering based on aesthetic and luminance scores to exclude blurry or over-/under-exposed clips, yielding a set of visually clean, geometrically reliable videos.

Data annotation. For each retained clip, we automatically generate a text prompt and rendered 4D control maps. We first generate a descriptive caption using Qwen2.5-VL-72B [5], which serves as text prompt. For geometry, we adopt MegaSAM as the base pipeline and replace its monocular and metric depth modules with MoGe-2 [95] and UniDepth V2 [72], respectively, to obtain more accurate and temporally consistent depth. Given the video frames, the estimated depth maps, and the camera trajectory, we reconstruct a 3D point cloud for each frame. Applying Grounded-SAM2 object masks to the per-frame point clouds yields per-object point clouds and a static background point cloud P^{bg}, as described in Sec. 3.1. For each object, we then fit per-frame 3D Gaussians and form a 3D Gaussian trajectory $\{\mathcal{G}_o^t\}$. Finally, we render the 4D Geometric Control into model-ready 4D control maps. The static background point cloud is rendered under the camera trajectory to obtain background RGB/depth maps. The 3D Gaussian trajectories are rendered to obtain 3D Gaussian trajectories RGB/depth maps. We then invert the background mask and merge it with the 3D Gaussian trajectories mask to produce a soft merged mask that marks regions where the video diffusion model should synthesize content.

In total, VerseControl4D contains 35,000 training samples and 1,000 validation samples. In the training set, about 26% of samples are sourced from Sekai-Real-HQ and 74% from SpatialVID-HQ, while 20% of the samples depict static scenes, encouraging VerseCrafter to learn both

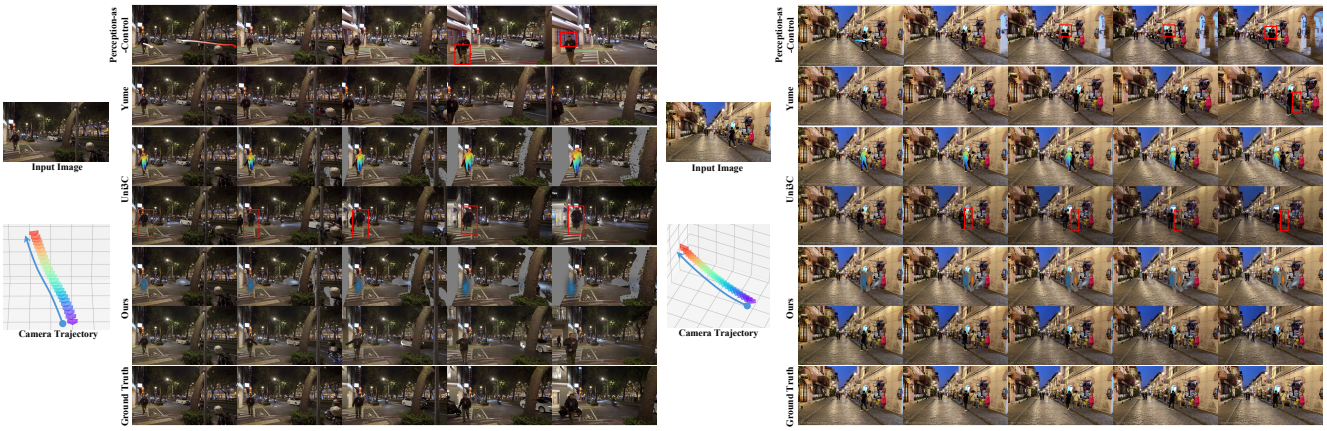

Figure 4. **Qualitative comparison of joint camera and object motion control.** Perception-as-Control and Uni3C exhibit noticeable human deformation, while Yume roughly follows the text-described motion but lacks precise camera control. Uni3C is also limited to single human. In contrast, VerseCrafter more faithfully follows both the camera trajectory and multi-object motion while maintaining sharp appearance and geometrically consistent backgrounds.

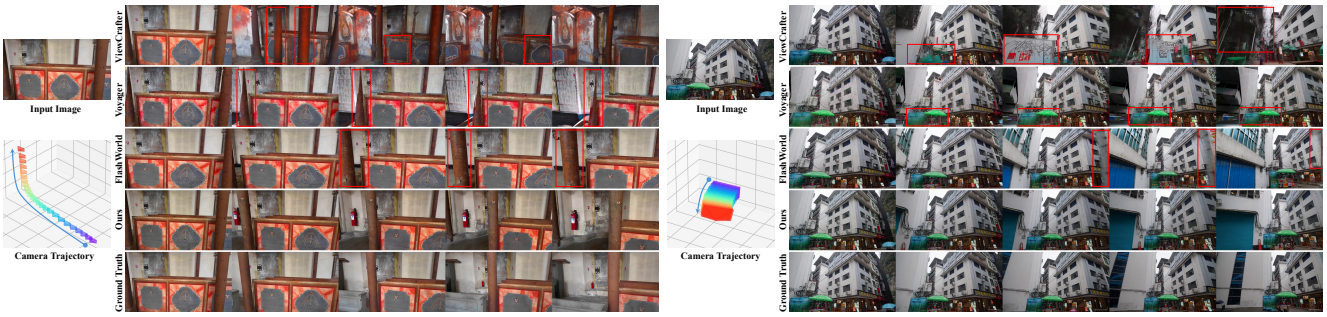

Figure 5. **Qualitative comparison of camera-only motion control on static scenes.** ViewCrafter and Voyager exhibit distorted facades, drifting structures, or inaccurate camera motion, while FlashWorld tends to produce blurred scene boundaries and imprecise camera motion. In contrast, VerseCrafter better follows the target camera trajectory while preserving sharp details and globally consistent 3D geometry.

camera-only world exploration and coupled camera–object dynamics. The validation set includes 250 static-scene samples to specifically assess camera-only control.

5. Experiments

Implementation Details. We build VerseCrafter upon the Wan2.1 T2V-14B model. The Wan backbone is kept frozen, and only GeoAdapter is updated. Each GeoAdapter block is initialized from the weights of its paired DiT block in Wan-DiT to stabilize training, and we set $k = 5$ so that every 5th DiT block in Wan-DiT is paired with a GeoAdapter block. We use the Adam optimizer with a learning rate of 2e-5, a constant learning rate schedule with 100 warmup steps. All experiments are conducted on 16 96-GB GPUs with a global batch size of 16. Training is performed in two stages: we first train for 2,500 iterations on 480P clips, and then fine-tune the same model for another 2,500 iterations on 720P clips. The total wall-clock training time is about 380 hours. We adopt classifier-free guidance during training by randomly dropping the text condition with probability 0.1. At inference time, we use 50 denoising steps and a classifier-free guidance scale of 5.0. Generating an 81-frame 720P video clip on 8 96-GB GPU takes about 1152 seconds, with a peak per-GPU memory usage of about 90 GB.

Evaluation Metrics. We evaluate overall video quality using VBench-I2V. For camera control, we follow CameraCtrl [33] and report rotation error (RotErr) and translation error (TransErr). For object-motion control, we adopt ObjMC proposed in MotionCtrl [98]. Given a generated video, we apply the same geometry annotation pipeline used for VerseControl4D to estimate its camera trajectory and 3D Gaussian trajectories, and compare them with the corresponding ground-truth trajectories from our dataset. ObjMC is computed as the average Euclidean distance between the estimated and ground-truth 3D Gaussian means over all controlled objects and frames.

5.1. Joint Camera and Object Motion Control

We first evaluate joint control of camera and object motion on VerseControl4D. As shown in Table 1, VerseCrafter

Table 1. **Joint camera and object motion control on VerseControl4D.** We report VBench-I2V scores and 3D control metrics (RotErr, TransErr, ObjMC). VerseCrafter achieves the best overall video quality and the most accurate joint control of camera and object motion.

	Overall Score ↑	Imaging Quality ↑	Aesthetic Quality ↑	Dynamic Degree ↑	Motion Smoothness ↑	Background Consistency ↑	Subject Consistency ↑	I2V Background ↑	I2V Subject ↑	RotErr ↓	TransErr ↓	ObjMC ↓
Perception-as-Control [15]	83.66	66.81	53.34	73.91	96.89	93.19	94.02	96.35	94.78	5.006	8.767	6.556
Yume [63]	85.47	71.16	52.39	72.24	**98.96**	95.66	96.43	98.51	98.39	7.560	8.735	7.959
Uni3C [11]	83.55	68.06	53.16	66.09	98.94	93.74	94.19	97.19	97.05	1.361	7.731	5.883
Ours	**88.10**	**72.70**	**57.49**	**86.26**	98.79	**95.69**	**96.48**	**98.76**	**98.65**	**0.890**	**3.103**	**2.507**

Table 2. **Camera-only motion control on static scenes.** On the static subset of VerseControl4D, we report VBench-I2V scores and camera control metrics (RotErr, TransErr). VerseCrafter achieves the best overall visual quality while substantially reducing camera pose errors.

	Overall Score ↑	Imaging Quality ↑	Aesthetic Quality ↑	Dynamic Degree ↑	Motion Smoothness ↑	Background Consistency ↑	Subject Consistency ↑	I2V Background ↑	I2V Subject ↑	RotErr ↓	TransErr ↓
ViewCrafter [117]	84.04	69.56	55.52	68.02	97.86	92.09	94.25	97.70	97.29	2.101	9.868
Voyager [41]	78.12	55.48	49.80	65.34	**99.39**	92.31	91.55	86.02	85.03	3.557	3.880
FlashWorld [54]	85.33	71.68	**58.74**	73.46	98.35	94.27	92.47	95.38	98.32	1.792	3.257
Ours	**86.80**	**74.57**	54.78	**80.34**	97.62	**94.88**	**95.55**	**97.86**	**98.79**	**0.650**	**2.587**

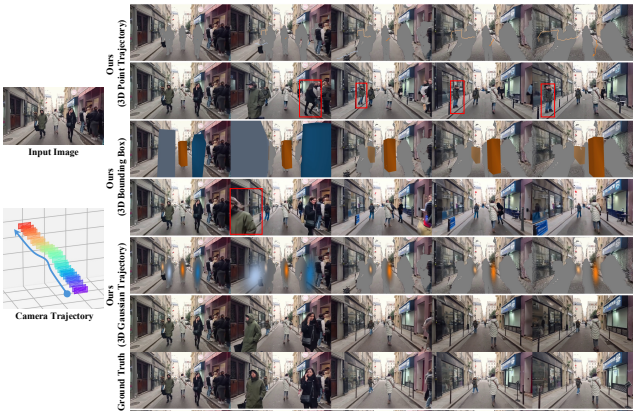

Figure 6. **Ablation on 3D representations for object motion control.** We compare object control using *3D point trajectory* (top), *3D bounding box* (middle), and *3D Gaussian trajectory* (bottom). 3D point trajectory and 3D bounding box often cause scale drift and misaligned motion (red boxes), whereas 3D Gaussian trajectory better follows the intended object motion while preserving plausible shapes and background interactions.

Figure 7. **Ablation on depth-aware control.** We compare VerseCrafter without depth inputs (*Ours (w/o depth)*, top) and with RGB+depth inputs (middle) under the same camera trajectory. Without depth, the model often produces incorrect foreground-background ordering, e.g., lampposts are pulled in front of distant buildings, and occlusion boundaries drift over time (red boxes). With RGB+depth, the model recovers consistent parallax and occlusion, producing geometry much closer to the ground truth.

achieves the best VBench-I2V scores among all compared methods, with clear gains in Overall Score, Imaging Quality, Aesthetic Quality, and both subject and background consistency. On 3D control metrics, VerseCrafter substantially reduces rotation, translation, and object-motion errors compared with the best-performing baseline, reflecting much tighter alignment with the target 4D geometric control. Qualitative comparisons in Fig. 4 further highlight these differences: Perception-as-Control and Uni3C exhibit noticeable human deformation, while Yume roughly follows the text-described motion but lacks precise camera control. Uni3C, relying on SMPL-X, is limited to single-person motion and struggles with other categories such as vehicles. In contrast, VerseCrafter more faithfully follows both the camera trajectory and 3D Gaussian trajectories while maintaining sharp appearance and geometrically consistent backgrounds.

5.2. Camera-Only Motion Control

We evaluate camera-only control on the static-scene subset of VerseControl4D, where objects remain stationary and only the camera moves. As shown in Table 2, VerseCrafter achieves the best VBench-I2V performance among all compared methods, with consistent gains in Overall Score, Imaging Quality, and both subject and background consistency, while maintaining motion smoothness comparable to prior methods. On 3D camera metrics, VerseCrafter substantially reduces rotation and translation errors relative to the best-performing baseline, indicating that it follows the target camera trajectory much more faithfully in static scenes. Qualitative comparisons in Fig. 5 further confirm these trends: ViewCrafter and Voyager exhibit distorted facades, drifting structures, or inaccurate camera motion, while FlashWorld tends to produce blurred scene boundaries and imprecise camera motion. In contrast, Ver-

Table 3. **Ablation study on 3D representation, depth, and decoupled controls.** We compare different variants of VerseCrafter using VBench-I2V and 3D control metrics (RotErr, TransErr, ObjMC). Our full model with 3D Gaussian trajectories, depth-aware rendering, and decoupled background/foreground controls achieves the best visual quality and the most accurate camera and object motion control.

	Overall Score ↑	Imaging Quality ↑	Aesthetic Quality ↑	Dynamic Degree ↑	Motion Smoothness ↑	Background Consistency ↑	Subject Consistency ↑	I2V Background ↑	I2V Subject ↑	RotErr ↓	TransErr ↓	ObjMC ↓
Ours (3D Bounding Box)	85.45	69.23	55.70	78.57	98.70	92.92	93.27	97.74	97.48	1.350	3.805	4.520
Ours (3D Point Trajectory)	85.57	70.29	55.27	78.23	98.63	94.00	92.75	97.85	97.55	1.298	3.281	6.896
Ours (w/o depth)	85.64	70.19	55.00	80.60	98.66	92.07	92.83	98.07	97.69	1.177	3.900	4.929
Ours (BG & FG Merged)	85.72	69.19	54.86	83.72	98.65	91.15	92.86	97.93	97.41	1.080	3.803	3.726
Ours	**88.10**	**72.70**	**57.49**	**86.26**	**98.79**	**95.69**	**96.48**	**98.76**	**98.65**	**0.890**	**3.103**	**2.507**

Figure 8. **Ablation on decoupled background and foreground controls.** We compare a variant that merges background and foreground controls into a single map (*Ours (BG & FG Merged)*, top) with our default decoupled design (middle). When the controls are merged, object motion control degrades significantly (red boxes), whereas the decoupled design better preserves the static background and produces more accurate and stable object motion.

seCrafter preserves straight structures, stable depth relationships, and an appearance closer to the ground-truth video, evidencing precise camera control in a static scene.

5.3. Ablation Study

We conduct ablations to analyze three key design choices in VerseCrafter: (i) the 3D representations for object motion, (ii) the use of depth in control maps, and (iii) the decoupling of background and foreground controls. All variants share the same training data, backbone, and optimization settings; only the control design is changed.

3D representations for object motion. To isolate the effect of our motion representation, we derive two alternatives from per-object 3D Gaussian trajectories: (1) an oriented **3D bounding box** whose axes follow the Gaussian's principal directions and whose side lengths are scaled by its principal spreads; and (2) a **3D point trajectory** that retains only the Gaussian centroid. The rest of the pipeline is unchanged: we rasterize cuboids (for boxes) or tiny disks/spheres (for points) instead of Gaussian ellipses. As reported in Table 3, replacing Gaussians with boxes slightly hurts both visual quality and control accuracy, while point trajectories give the weakest object-motion consistency. Qualitatively (Fig. 6), points and boxes often cause scale drift and misaligned motion, whereas 3D Gaussian trajectories better follow the intended object trajectories and preserve plausible object shapes.

Depth-aware Control. To evaluate the effect of depth, we remove depth channel from the background and trajectory controls ("Ours (w/o depth)" in Table 3). This variant yields a lower Overall Score and significantly worse 3D control (higher RotErr and ObjMC values). As shown in Fig. 7, without depth, the model produces incorrect foreground-background ordering: vertical structures like streetlights appear beside shelves in the foreground, while buildings that should be behind the character are positioned elsewhere, and occlusion boundaries drift over time. With RGB+depth, VerseCrafter recovers more consistent parallax and occlusion, producing geometry much closer to the ground truth.

Decoupled vs. merged controls. We further compare our decoupled design with a variant that merges the background and 3D Gaussian trajectory maps into a single control stream (*Ours (BG & FG Merged)* in Table 3). Although this variant still benefits from the explicit 4D geometric scene state, it consistently underperforms the full model, with a particularly noticeable drop in object-motion accuracy. As shown in Fig. 8, merging the controls leads to clear degradation in object motion control. In contrast, keeping decoupled design preserves static geometry while producing more precise and stable object motion, which is crucial for accurate and geometry-consistent control.

6. Conclusion

We present **VerseCrafter**, a geometry-driven video world model built upon an explicit **4D Geometric Control**, represented by a static background point cloud and per-object 3D Gaussian trajectories in a shared world coordinate frame. Coupled with GeoAdapter, which conditions a frozen Wan2.1 backbone on rendered 4D control maps, this design enables high-fidelity video generation with precise, disentangled control over camera and multi-object motion. To support training and evaluation, we construct **VerseControl4D**, a real-world dataset with automatically derived prompts and rendered 4D control maps, comprising 35K training samples. Experiments and ablations show that VerseCrafter delivers superior visual quality and more accurate joint camera and object motion than existing controllable video generators and video world models, highlighting 4D Geometric Control as a promising interface for future work on dynamic world simulation and editing.

Acknowledgments. The paper is supported by Shanghai Municipal Science Technology Major Project (2025SHZDZX025G02).

A. Preliminary: Video Diffusion Models

Modern video diffusion models operate in a compact latent space learned by a spatio-temporal VAE. Given a video $x \in \mathbb{R}^{T\times H\times W\times 3}$, the encoder E maps it to latent features $z_0 = E(x) \in \mathbb{R}^{T'\times C\times H'\times W'}$, on which the generative process is defined [8, 79]. A standard forward diffusion process gradually perturbs z_0 into noisy variables z_t via

$$q(z_t \mid z_0) = \sqrt{\alpha_t}\, z_0 + \sqrt{1-\alpha_t}\, \epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \tag{6}$$

and a denoiser ϵ_θ is trained to predict the noise under time step t and conditioning signal c (e.g., text prompts or reference frames) as

$$\mathcal{L}_{\text{diff}}(\theta) = \mathbb{E}_{z_0,t,\epsilon}\big[\|\epsilon_\theta(z_t, t, c) - \epsilon\|_2^2\big], \tag{7}$$

following the DDPM formulation [37]. Recent video generators further adopt continuous-time flow matching. Given clean latents z_0 and a Gaussian samples z_1, one defines linear interpolants $z_\tau = (1-\tau)z_0 + \tau z_1$ with $\tau \in [0, 1]$ and learns a velocity field v_θ by

$$\mathcal{L}_{\text{flow}}(\theta) = \mathbb{E}_{z_0,\tau,\epsilon}\big[\|v_\theta(z_\tau, \tau, c) - (z_1 - z_0)\|_2^2\big], \tag{8}$$

as in flow-matching and related ODE-based generative formulations [43, 59]. These objectives are typically implemented with Diffusion Transformers (DiT), which operate on spatio-temporal latent tokens and inject (t, c) through attention [71], forming the backbone of current foundation video generators.

Wan2.1 instantiates the above latent video diffusion / flow-matching paradigm with a 3D VAE and a DiT-based denoiser, together with rich multi-modal conditioning trained on large-scale, diverse video–text data [88]. In VerseCrafter, we adopt Wan2.1-14B as a frozen latent video diffusion backbone and treat it as a generic video prior. Specifically, we keep the Wan Encoder, Wan-DiT, and Wan Decoder unchanged, and attach a lightweight geometry-aware control interface, namely GeoAdapter, to selected Wan-DiT blocks. The detailed architecture of VerseCrafter is provided in the Sec. B.

B. Model Architecture Details

VerseCrafter is built on top of the Wan2.1 T2V-14B backbone [88], a latent video diffusion / flow-matching model with a 3D VAE (Wan Encoder and Wan Decoder) and a DiT-based denoiser (Wan-DiT). As shown in Fig. 9, we keep the Wan2.1 backbone frozen and introduce a geometry-aware conditioning pathway with a lightweight GeoAdapter that conditions selected Wan-DiT blocks on rendered 4D control maps. Table 4 summarizes the input resolution, number of Wan-DiT layers, hidden dimension, GeoAdapter injection pattern, and fine-tuning configuration of VerseCrafter.

Table 4. **Model configuration of VerseCrafter.** Settings include the final output resolution, number of Wan-DiT layers, GeoAdapter injection blocks, pre-trained backbone, and training schedule.

	VerseCrafter
Final resolution	720P
Num. Wan-DiT layers	40
GeoAdapter injection blocks	$[0, 5, 10, 15, 20, 25, 30, 35]$
Pre-trained backbone	Wan2.1 T2V-14B
Hidden dimension	5120
Batch size	16
Training schedule	2,500 it. @480P + 2,500 it. @720P

Geometry encoding and tokenization. For each frame t, we render background RGB/depth RGB_t^{bg}, $\text{Depth}_t^{\text{bg}}$, 3D Gaussian trajectory RGB/depth $\text{RGB}_t^{\text{traj}}$, $\text{Depth}_t^{\text{traj}}$, and a soft merged mask M_t that marks regions where the diffusion model should synthesize or overwrite content. For t=1, we replace RGB_1^{bg} with the input image and set M_1=0. The four RGB/depth maps are encoded by the frozen Wan Encoder to obtain latent features at the VAE resolution, while the mask $M \in \mathbb{R}^{1\times T\times H\times W}$ is rearranged to align with the latent grid of Wan Encoder (the "Rearrange" module in Fig. 9). Let s_t, s_h, and s_w denote the temporal and spatial strides of Wan Encoder (we use s_t=4 and s_h=s_w=8). Following the practice in [42, 88], we drop the singleton channel dimension, split the spatial dimensions into $s_h \times s_w$ sub-cells, and fold these sub-cells into the channel dimension via a reshape–permute operation, yielding a tensor of shape $C_M \times T \times H' \times W'$ with C_M=$s_h s_w$, H'=H/s_h, and W'=W/s_w. We then downsample the temporal dimension using nearest-neighbor interpolation to match the latent depth $T' = (T + s_t - 1)/s_t$, producing $\hat{M} \in \mathbb{R}^{C_M\times T'\times H'\times W'}$. Finally, $\hat{M}$ is concatenated channel-wise with the encoded background and 3D Gaussian trajectory latents to form a unified spatio-temporal geometry feature $\mathcal{G} \in \mathbb{R}^{T'\times H'\times W'\times C_{\mathcal{G}}}$. Following Wan-DiT, we partition $\mathcal{G}$ into non-overlapping 3D patches and linearly project each patch into a token embedding, yielding a sequence of geometry tokens $\mathbf{g} \in \mathbb{R}^{L\times D}$, where $L = T'H'W'$ is the number of spatio-temporal patches and D matches the hidden width of Wan-DiT. Because we use identical strides, positional encodings, and patch sizes, the geometry tokens are spatially and temporally aligned with the latent video tokens processed by Wan-DiT.

GeoAdapter integration. GeoAdapter is a lightweight DiT-style branch that operates on geometry tokens $\mathbf{g}$. It shares the same hidden dimensionality and positional

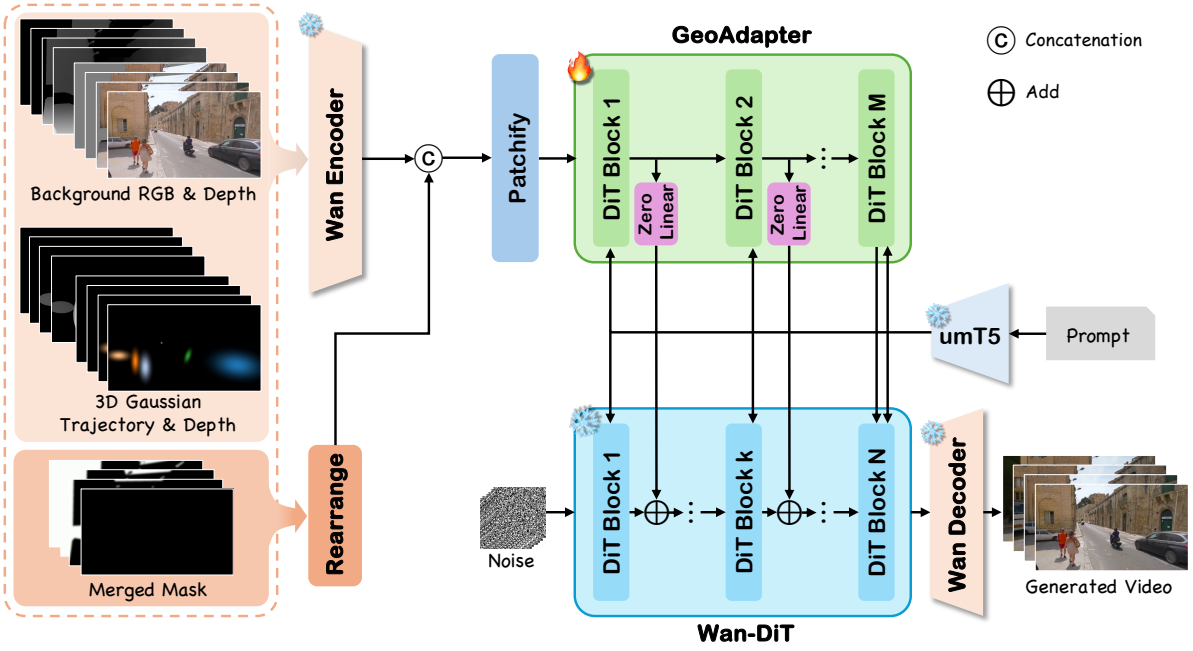

Figure 9. **Detailed architecture of VerseCrafter.** Background RGB & depth maps and 3D Gaussian trajectory RGB & depth maps are first encoded by the frozen Wan Encoder. The soft merged mask is rearranged into latent-aligned channels, and all geometry latents are then concatenated along the channel dimension to form a unified spatio-temporal geometry feature. This feature is patchified into tokens and processed by the GeoAdapter branch. At selected Wan-DiT blocks, GeoAdapter outputs are passed through zero-initialized linear layers and added to the backbone tokens as residual modulations, enabling geometry-consistent control over camera motion and multi-object motion.

encodings as Wan-DiT, but contains far fewer layers. Let $\{\mathcal{B}_1, \ldots, \mathcal{B}_N\}$ denote N Wan-DiT blocks, and let $\{\mathcal{G}_1, \ldots, \mathcal{G}_M\}$ denote M GeoAdapter blocks. We attach GeoAdapter as a residual modulation branch to a subset of Wan-DiT blocks. Concretely, we choose a stride k and inject GeoAdapter after every k-th Wan-DiT block; see Table 4 for the exact injection pattern and configuration. For each Wan-DiT block $\mathcal{B}_n$ whose index n belongs to the injection set, with input tokens $\mathbf{x}_n \in \mathbb{R}^{L\times D}$ and geometry tokens $\mathbf{g}$, we add a geometry-conditioned residual of the form

$$\mathbf{x}_{n+1} = \mathcal{B}_n(\mathbf{x}_n) + \mathcal{G}_m(\mathbf{g})\,\mathbf{W}_0^{(m)}, \tag{9}$$

where $\mathcal{G}_m$ is the corresponding GeoAdapter block and $\mathbf{W}_0^{(m)} \in \mathbb{R}^{D\times D}$ is its output projection. Each GeoAdapter block is initialized from the weights of its paired Wan-DiT block for stable training, while $\mathbf{W}_0^{(m)}$ is initialized to zero. As a result, VerseCrafter behaves identically to the original Wan2.1 backbone at the beginning of training. During fine-tuning, $\mathbf{W}_0^{(m)}$ gradually learns to inject geometry information through residual modulation, in the spirit of zero-initialized adapter designs such as ControlNet [119].

Table 5. **VerseControl4D data split and scene-type statistics.** We report the number of samples from each source dataset and split. *Dynamic scenes* contain coupled camera motion and foreground object motion, while *static scenes* have negligible object motion and are used for camera-only evaluation.

Split	Sekai-Real-HQ	SpatialVID-HQ	
	Dynamic Scenes	Static Scenes	Dynamic Scenes
Train	9,071	7,000	18,929
Validation	468	250	282

C. VerseControl4D Dataset Details

We construct **VerseControl4D**, a large-scale real-world video dataset with automatically derived prompts and rendered 4D control maps. As described in the main paper, VerseControl4D is built through four stages: data collection, clip extraction, quality filtering, and data annotation. The rendered 4D control maps comprise background RGB/depth maps, 3D Gaussian trajectory RGB/depth maps, and a soft merged mask.

VerseControl4D contains 35,000 training samples and 1,000 validation samples. Table 5 summarizes the data distribution by source dataset and scene type. Overall, 26% of the samples come from Sekai-Real-HQ and 74% from

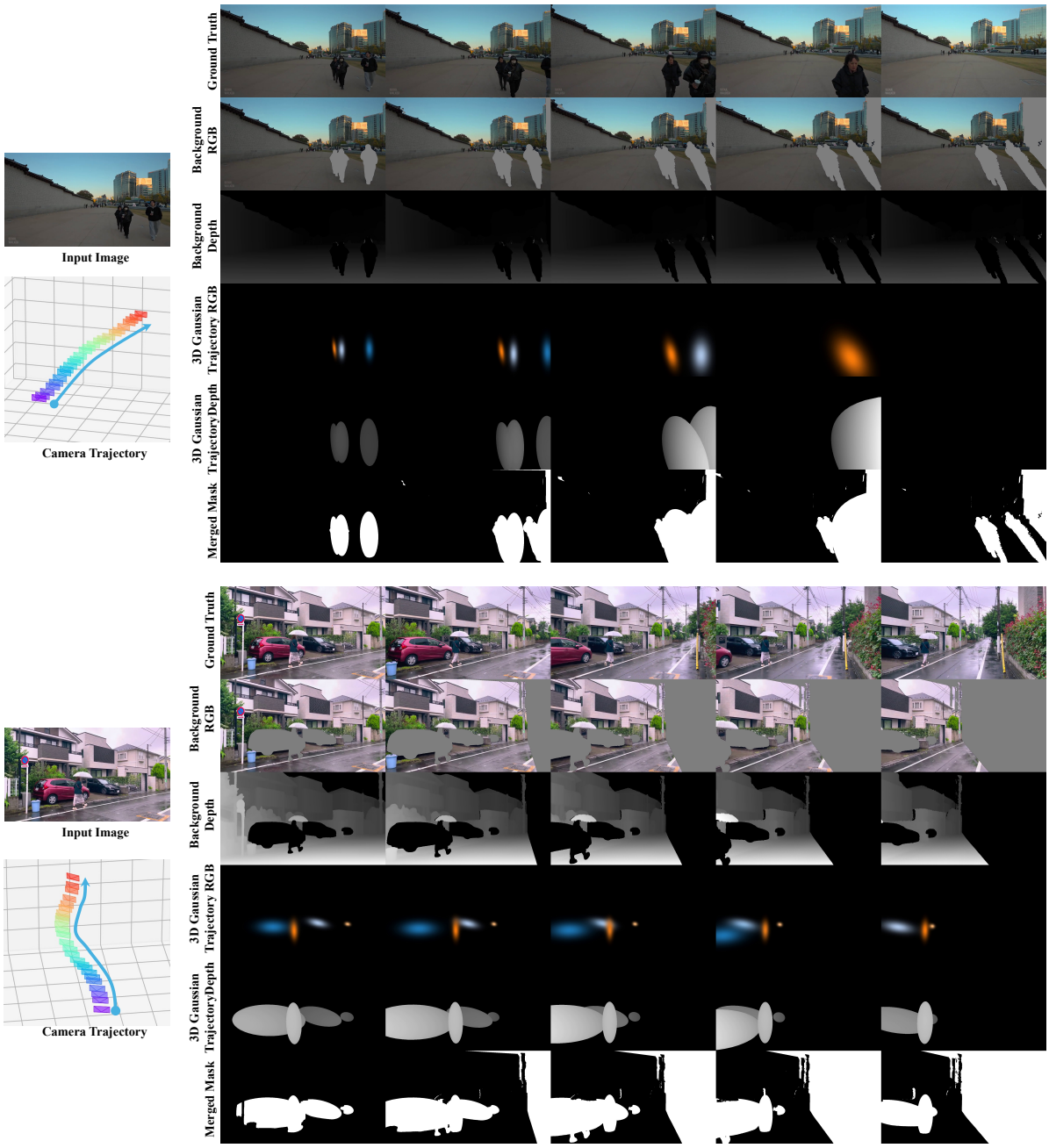

Figure 10. **VerseControl4D dataset examples.** For each clip, we visualize the input image and target camera trajectory (left), followed by several frames of ground-truth video and our rendered control signals (right): background RGB/depth, 3D Gaussian trajectory RGB/depth for controlled objects, and the final merged mask. These signals are automatically derived by our annotation pipeline in main paper.

SpatialVID-HQ, reflecting their complementary scene coverage. To support both camera-only world exploration and joint camera-object control, VerseControl4D includes *dynamic scenes* (clips with salient foreground object motion together with camera motion) and *static scenes* (clips with negligible object motion and only camera motion). About 20% of the training samples are *static scenes*, and the validation set includes 250 static-scene samples for dedicated camera-only evaluation. Representative samples and their rendered 4D control signals are shown in Fig. 10.

D. Evaluation Metrics

D.1. VBench-I2V

We evaluate image-to-video generation quality using the VBench Image-to-Video (I2V) evaluation suite, denoted as

VBench-I2V. For each generated clip, we follow the official VBench-I2V protocol: the conditioning image and its corresponding generated video are fed into the evaluation pipeline, which computes a set of learned, human-aligned metrics that jointly capture video-image consistency and perceptual video quality. In our experiments, we report the following eight VBench-I2V dimensions, and define the *Overall Score* as the simple arithmetic mean of these eight normalized scores, where higher values indicate better performance:

- **Imaging Quality.** This metric measures low-level image fidelity, including sharpness and the absence of artifacts such as blur, noise, or overexposure. VBench uses an image quality predictor (e.g., MUSIQ) and averages its scores across frames to obtain a video-level imaging-quality score.
- **Aesthetic Quality.** This dimension assesses the artistic and aesthetic appeal of individual frames, including composition, color harmony, and realism. VBench applies an aesthetic quality predictor (e.g., the LAION aesthetic model) to each frame and averages the predictions over the clip.
- **Dynamic Degree.** This metric quantifies how dynamic the generated video is. Optical flow magnitudes (e.g., estimated by RAFT) are used to measure the amount of motion, and the score reflects whether the model produces sufficiently active (non-static) content.
- **Motion Smoothness.** This metric evaluates whether subject and camera motion evolve smoothly and follows reasonable physical dynamics. VBench leverages a pretrained video frame interpolation prior to assess how well intermediate motion can be interpolated, with smoother and more physically plausible motion receiving higher scores.
- **Background Consistency.** This dimension measures the temporal stability of background layout and texture. Frame-level features (e.g., CLIP) are compared across time; large feature variations indicate flickering or unstable backgrounds and lead to lower scores.
- **Subject Consistency.** This dimension evaluates the temporal consistency of the foreground subject *within* the video, regardless of the input image. VBench computes subject-region features across frames and measures their similarity over time to penalize identity drift or sudden appearance changes.
- **I2V Background (Video–Image Background Consistency).** This metric evaluates how well the global background in the video matches the background in the input image, especially for scene-centric inputs. VBench uses background-sensitive features (e.g., DreamSim) and aggregates image–frame and inter-frame similarities into a single background consistency score.
- **I2V Subject (Video–Image Subject Consistency).** This metric measures how well the main subject in the generated video matches the subject in the input image. VBench extracts high-level visual features (e.g., DINO) from the conditioning image and from each video frame, and combines image–frame similarities with inter-frame similarities into a weighted average subject consistency score.

Formally, given these eight per-dimension scores $\{s_k\}_{k=1}^{8}$ returned by VBench-I2V for a video, we define

$$\text{Overall Score} = \frac{1}{8}\sum_{k=1}^{8} s_k, \tag{10}$$

which is the value reported as "Overall Score" in the main paper.

D.2. Rotation Error (RotErr)

To measure how well the generated camera motion follows the ground-truth camera trajectory, we adopt the camera-alignment metric from CameraCtrl [33]. For each generated video, we estimate its camera trajectory using the same geometry-annotation pipeline used for VerseControl4D, yielding rotation matrices $\{\mathbf{R}^j_{\text{gen}}\}_{j=1}^{n}$ and translation vectors $\{\mathbf{T}^j_{\text{gen}}\}_{j=1}^{n}$, where n is the number of frames. Let $\{\mathbf{R}^j_{\text{gt}}\}_{j=1}^{n}$ denote the corresponding ground-truth rotation matrices. The rotation error is computed by comparing the ground-truth and generated rotation matrices at each frame:

$$\text{RotErr} = \sum_{j=1}^{n} \arccos\left(\frac{\text{tr}\big(\mathbf{R}^j_{\text{gen}}{\mathbf{R}^j_{\text{gt}}}^{\top}\big) - 1}{2}\right), \tag{11}$$

where $\text{tr}(\cdot)$ denotes the matrix trace. Lower RotErr indicates better alignment between the generated and ground-truth camera orientations.

D.3. Translation Error (TransErr)

We also evaluate the accuracy of the generated camera positions. Let $\{\mathbf{T}^j_{\text{gt}}\}_{j=1}^{n}$ and $\{\mathbf{T}^j_{\text{gen}}\}_{j=1}^{n}$ be the ground-truth and generated camera translation vectors for a video with n frames. Following CameraCtrl [33], the translation error is defined as the sum of per-frame Euclidean distances between the translation vectors:

$$\text{TransErr} = \sum_{j=1}^{n} \big\|\mathbf{T}^j_{\text{gt}} - \mathbf{T}^j_{\text{gen}}\big\|_2, \tag{12}$$

Lower TransErr indicates that the generated camera positions more closely match the ground-truth camera positions.

D.4. Object Motion Control (ObjMC)

For object-motion control, we follow the ObjMC metric proposed in MotionCtrl [98] and extend it to the multi-

Table 6. **Memory–time trade-off under different inference settings.** We report peak per-GPU memory and diffusion inference time for the 50-step setting. FSDP reduces peak memory from 90 GB to 70 GB with negligible runtime change, and FSDP + CPU offload further reduces it to 57 GB (36.7% reduction) with only a small increase in diffusion inference time.

Inference setting	Peak GPU memory (GB)	Memory reduction (%)	Diffusion inference time (s)
Baseline	90	0.0	866
Baseline + FSDP	70	22.2	870
Baseline + FSDP + CPU offload	57	36.7	880

Table 7. **Stage-wise end-to-end inference latency and cacheability.** We report the runtime breakdown for generating an 81-frame 720P video on 8×96GB GPUs. 4D geometric scene state is reusable across repeated edits of the same scene, and model loading is a one-time startup cost, whereas 4D control rendering and diffusion sampling must be rerun when the edited controls change.

Stage	Time (s) ↓	Cacheable?
4D Geometric State Construction	∼23	✓
4D Control Maps Rendering	∼60	✗
Diffusion Sampling		
Diffusion Model Loading	∼203	✓
Diffusion Inference	∼866 (50 steps)	✗
Diffusion Inference	∼715 (30 steps)	✗

object setting under our 3D Gaussian trajectory representation. Given a generated video, we run the same geometry annotation pipeline used for VerseControl4D to estimate per-object 3D Gaussian trajectories, and compare them with the corresponding ground-truth trajectories from our dataset.

Let N_{gt} and N_{pred} denote the numbers of ground-truth and predicted controlled objects in a sample, and let T be the number of frames. For each ground-truth object $o \in \{1, \dots, N_{\text{gt}}\}$ and frame $t \in \{1, \dots, T\}$, we denote the ground-truth 3D Gaussian mean by $\boldsymbol{\mu}_o^{(t)} \in \mathbb{R}^3$ and the estimated mean from the generated video by $\hat{\boldsymbol{\mu}}_k^{(t)} \in \mathbb{R}^3$ for a predicted object k.

Multi-object matching. Since N_{gt} and N_{pred} may differ, we first define the trajectory distance between a ground-truth object o and a predicted object k as the average Euclidean distance between their 3D Gaussian means over time:

$$d(o,k) = \frac{1}{T} \sum_{t=1}^{T} \left\| \hat{\boldsymbol{\mu}}_k^{(t)} - \boldsymbol{\mu}_o^{(t)} \right\|_2. \tag{13}$$

We then build a cost matrix $\mathbf{C} \in \mathbb{R}^{N_{\text{gt}} \times N_{\text{pred}}}$ with entries $C_{ok} = d(o,k)$. To handle unmatched objects, we pad this matrix with dummy rows and columns and fill them with a constant penalty λ (set to 10.0 m in our experiments). Finally, we apply the Hungarian algorithm [47] to obtain an optimal one-to-one matching between ground-truth and predicted trajectories. This step assigns each ground-truth object either to a predicted trajectory or to a dummy entry when no suitable match exists.

ObjMC score. Given the optimal matching, we define the per-object trajectory error for a ground-truth object o as

$$d_o = \begin{cases} d(o,k) & \text{if } o \text{ is matched to a predicted object } k, \\ \lambda & \text{if } o \text{ is unmatched,} \end{cases} \tag{14}$$

and compute the final ObjMC score as the average over all ground-truth controlled objects:

$$\text{ObjMC} = \frac{1}{N_{\text{gt}}} \sum_{o=1}^{N_{\text{gt}}} d_o. \tag{15}$$

Lower ObjMC indicates more accurate multi-object 3D motion control, and the unmatched penalty λ penalizes missed objects under the one-to-one matching formulation.

E. Additional Qualitative Results

We provide additional qualitative comparisons on VerseControl4D, following the same evaluation settings and baselines as in the main paper. Fig. 11 and Fig. 12 present *dynamic scenes* with joint camera and multi-object motion control. Perception-as-Control and Uni3C often exhibit noticeable object deformation, while Yume roughly follows the text-described motion but lacks precise camera control. Uni3C is also limited to a single human and does not generalize well to diverse multi-object scenarios. In contrast, VerseCrafter more faithfully follows both the camera trajectory and multi-object motion while maintaining sharp appearance and geometrically consistent backgrounds.

Fig. 13 and Fig. 14 present *static scenes* for camera-only motion control. ViewCrafter, Voyager and FlashWorld often exhibit distorted facades, drifting structures, or inaccurate camera motion. In contrast, VerseCrafter better follows the target camera trajectory while preserving sharp details and globally consistent 3D geometry. These additional examples further demonstrate VerseCrafter's robustness under real-world 4D Geometric Control in both dynamic and static settings.

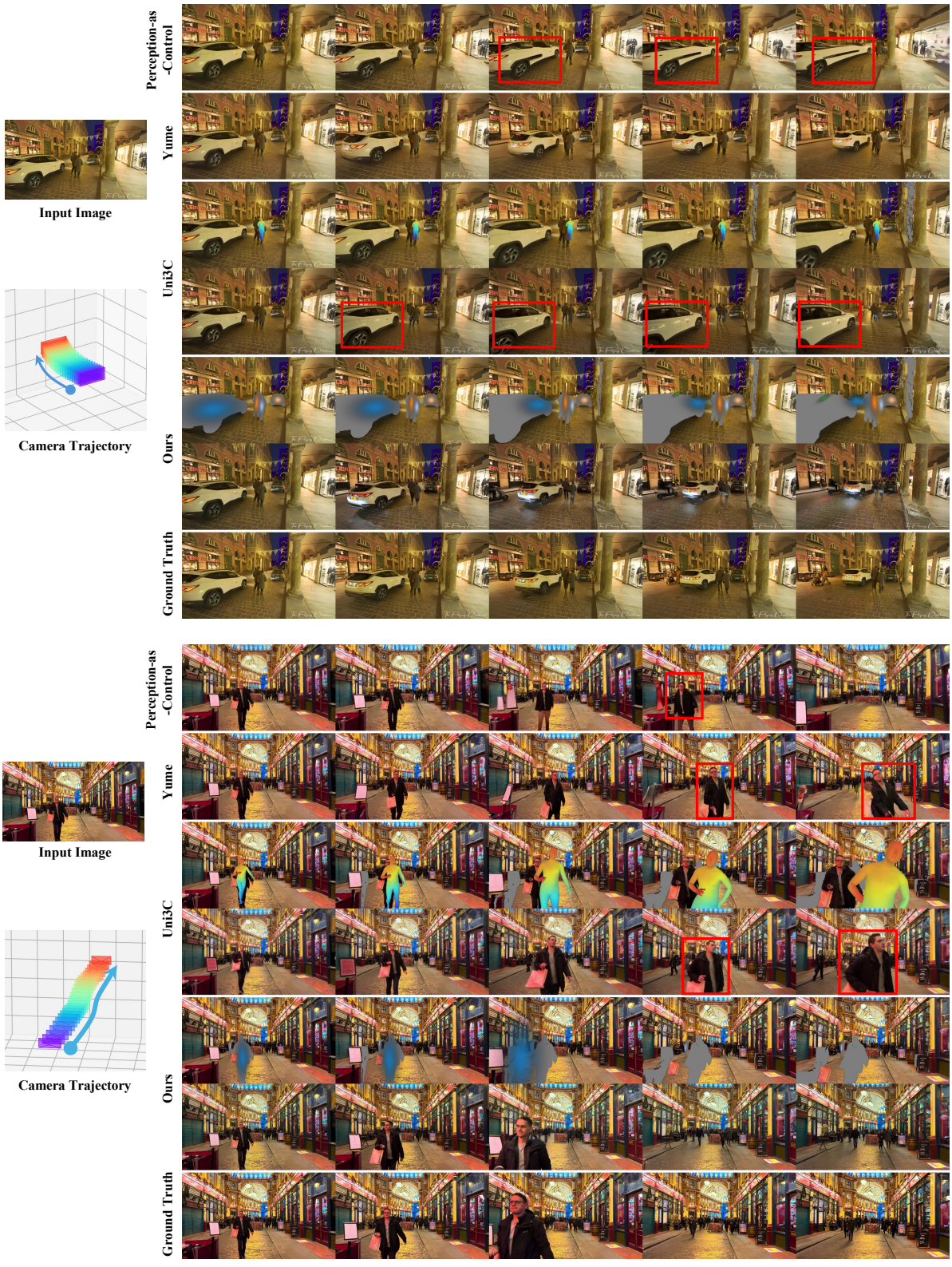

Figure 11. **Additional qualitative comparison of joint camera and object motion control.** Perception-as-Control and Uni3C exhibit noticeable object deformation, while Yume roughly follows the text-described motion but lacks precise camera control. Uni3C is also limited to a single human. In contrast, VerseCrafter more faithfully follows both the camera trajectory and multi-object motion while maintaining sharp appearance and geometrically consistent backgrounds.

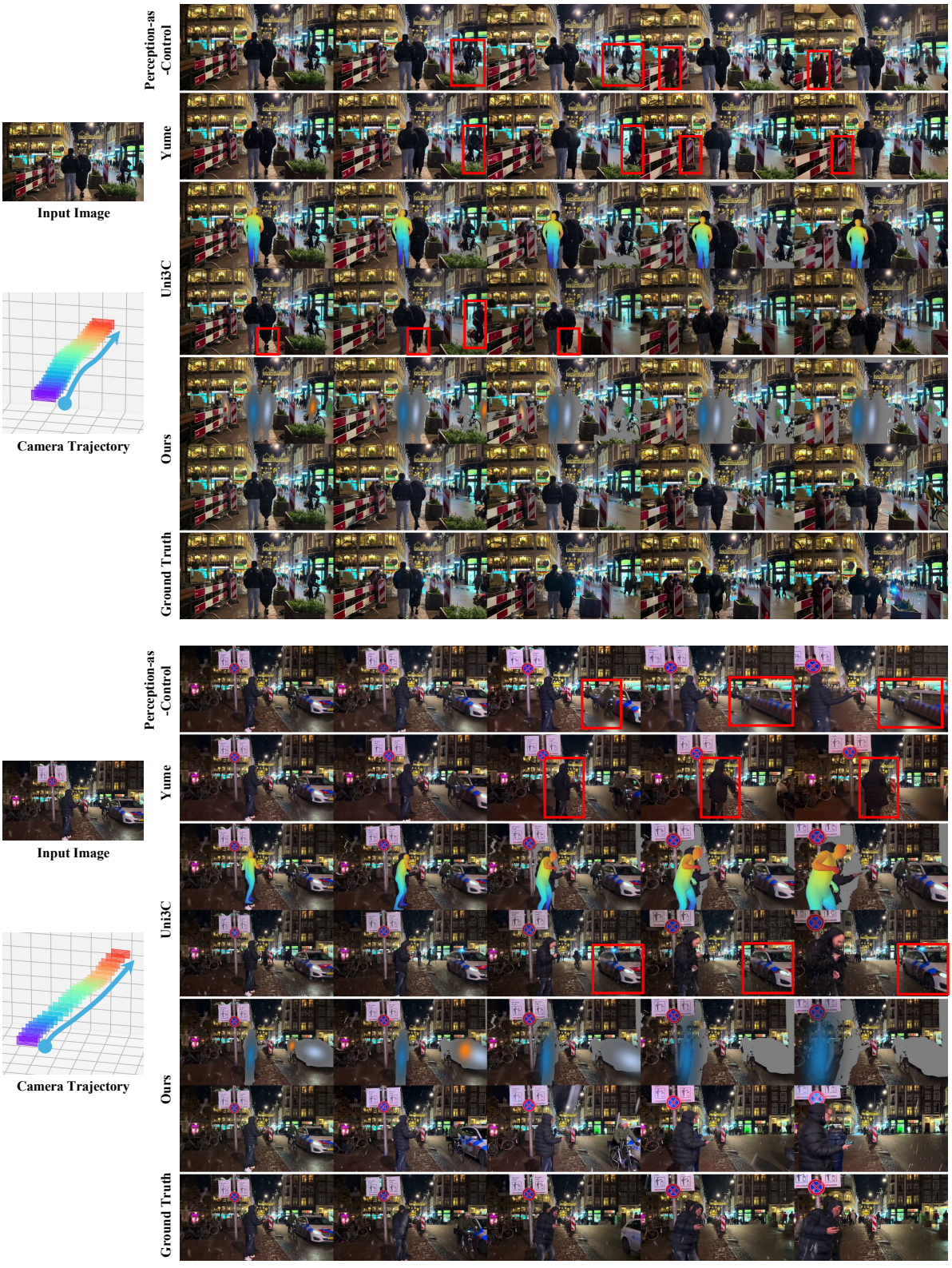

Figure 12. **Additional qualitative comparison of joint camera and object motion control.** Perception-as-Control and Uni3C exhibit noticeable object deformation, while Yume roughly follows the text-described motion but lacks precise camera control. Uni3C is also limited to a single human. In contrast, VerseCrafter more faithfully follows both the camera trajectory and multi-object motion while maintaining sharp appearance and geometrically consistent backgrounds.

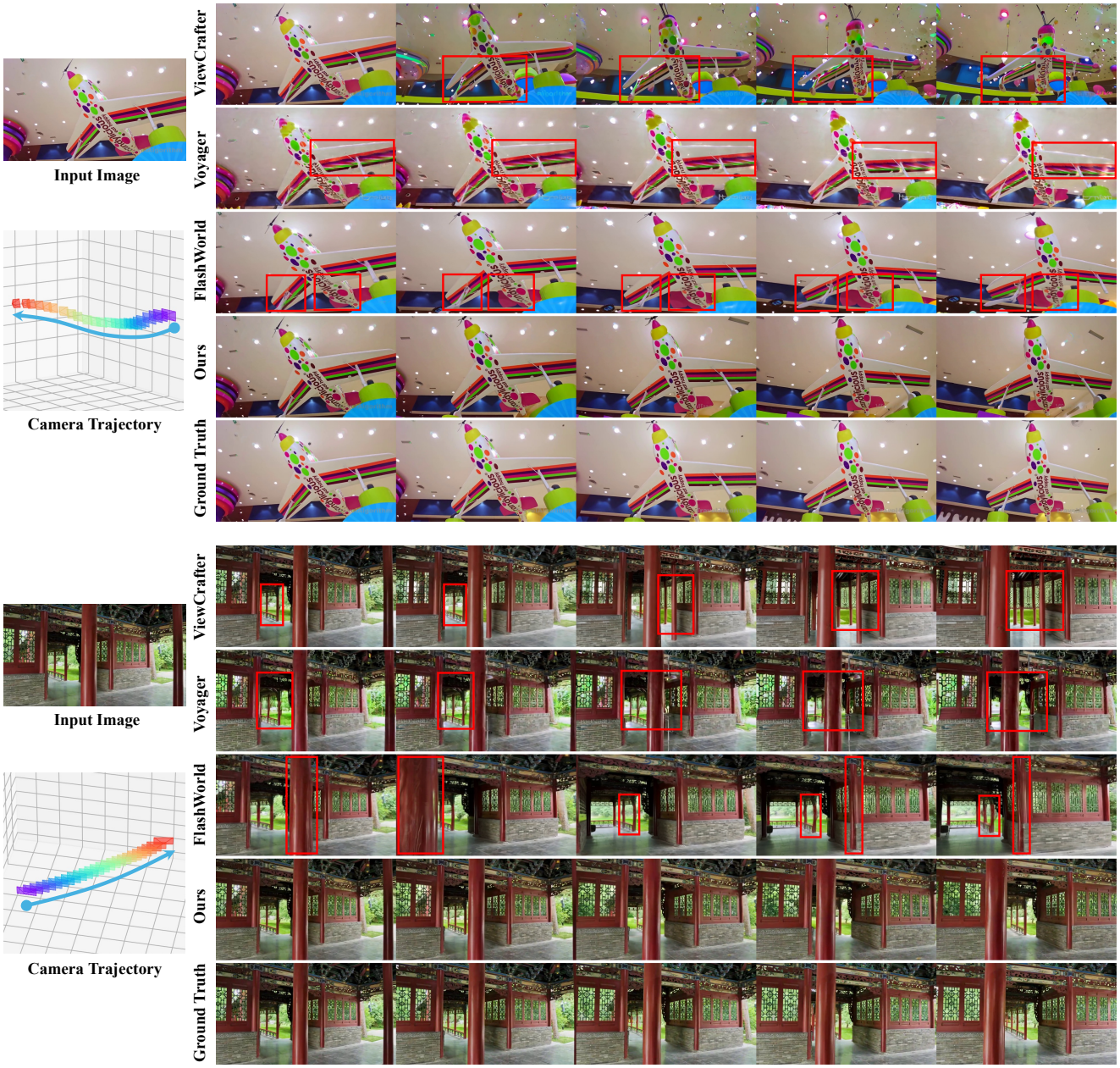

Figure 13. **Additional qualitative comparison of camera-only motion control on static scenes.** ViewCrafter, Voyager and FlashWorld exhibit distorted facades, drifting structures, or inaccurate camera motion. In contrast, VerseCrafter better follows the target camera trajectory while preserving sharp details and globally consistent 3D geometry.

F. Inference Efficiency and Memory Usage

We further analyze the inference cost of VerseCrafter. For generating an 81-frame 720P video on 8×96GB GPUs, Table 7 shows that diffusion inference is the dominant bottleneck, while 4D geometric state construction is cacheable across repeated edits of the same scene and diffusion model loading is a one-time startup cost. Accordingly, the per-edit latency is substantially reduced for subsequent edits, and can be further lowered by using fewer denoising steps.

Table 6 summarizes the memory and runtime trade-off under different inference settings. FSDP substantially reduces peak per-GPU memory with negligible runtime overhead, and FSDP + CPU offload further lowers memory at only a small additional cost. These results suggest that the current practical bottleneck is diffusion inference rather than 4D geometric state construction.

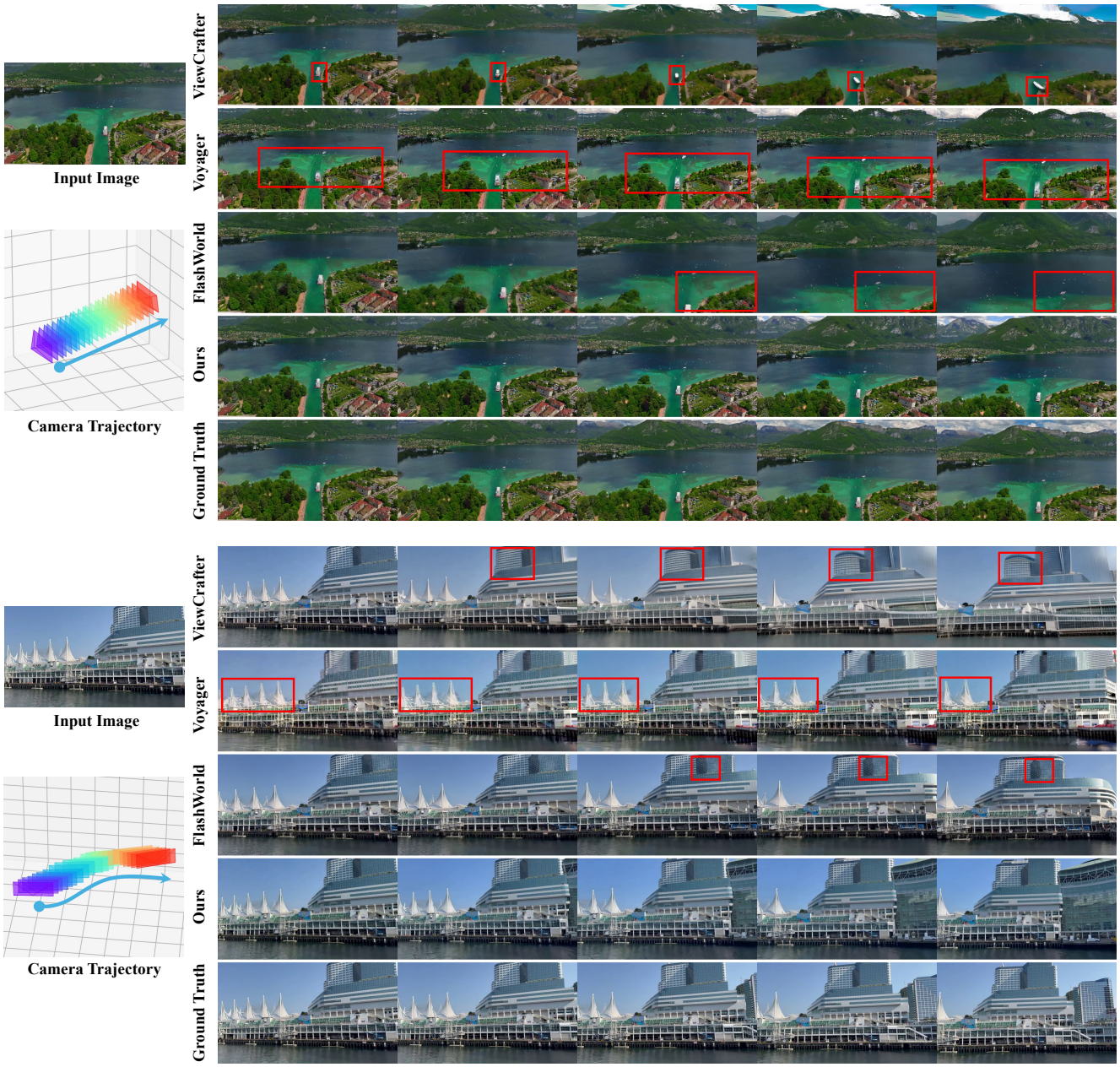

Figure 14. **Additional qualitative comparison of camera-only motion control on static scenes.** ViewCrafter, Voyager and FlashWorld exhibit distorted facades, drifting structures, or inaccurate camera motion. In contrast, VerseCrafter better follows the target camera trajectory while preserving sharp details and globally consistent 3D geometry.

G. Limitations and Future Work

Despite the encouraging results, VerseCrafter still has several limitations that suggest promising directions for future work.

First, our current object representation provides only *ellipsoid-level* control through a single 3D Gaussian per object, which limits fine-grained pose and part-level articulation, especially for human-like or near-symmetric objects. More expressive object representations, such as multiple Gaussians per object or articulated 3D structures, may improve fine-grained orientation and pose control.

Second, our background point cloud is reconstructed from the first frame and serves as a mostly static geometric scaffold, which limits controllability for highly non-rigid and texture-dominant scene dynamics such as waterfalls. Incorporating explicit dynamic background representations or temporally evolving scene geometry may improve con-

trollability in such cases.

Third, although VerseCrafter enforces 4D geometric consistency through explicit camera control and 3D Gaussian trajectory control, it does not impose explicit physical constraints during generation. Integrating stronger physics priors, such as collision-aware losses, contact constraints, ground constraints, or differentiable physics guidance, could improve physical realism and controllability in complex interactions.

Finally, VerseCrafter remains computationally expensive at high resolution and long temporal horizons because it conditions a large frozen video diffusion backbone and renders multi-channel 4D controls for all frames. Future work may explore more efficient backbones, distilled sampling, cached control encoding, and streaming or long-video synthesis to enable faster and longer world rollouts.

References

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

[2] Hassan Abu Alhaija, Jose Alvarez, Maciej Bala, Tiffany Cai, Tianshi Cao, Liz Cha, Joshua Chen, Mike Chen, Francesco Ferroni, Sanja Fidler, et al. Cosmos-transfer1: Conditional world generation with adaptive multimodal control. *arXiv preprint arXiv:2503.14492*, 2025. 2

[3] Sherwin Bahmani, Ivan Skorokhodov, Aliaksandr Siarohin, Willi Menapace, Guocheng Qian, Michael Vasilkovsky, Hsin-Ying Lee, Chaoyang Wang, Jiaxu Zou, Andrea Tagliasacchi, et al. Vd3d: Taming large video diffusion transformers for 3d camera control. *arXiv preprint arXiv:2407.12781*, 2024. 3

[4] Sherwin Bahmani, Ivan Skorokhodov, Guocheng Qian, Aliaksandr Siarohin, Willi Menapace, Andrea Tagliasacchi, David B Lindell, and Sergey Tulyakov. Ac3d: Analyzing and improving 3d camera control in video diffusion transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22875–22889, 2025. 3

[5] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 5

[6] Philip J. Ball, Jakob Bauer, Frank Belletti, Bethanie Brownfield, Ariel Ephrat, Shlomi Fruchter, Agrim Gupta, Kristian Holsheimer, Aleksander Holynski, Jiri Hron, Christos Kaplanis, Marjorie Limont, Matt McGill, Yanko Oliveira, Jack Parker-Holder, Frank Perbet, Guy Scully, Jeremy Shar, Stephen Spencer, Omer Tov, Ruben Villegas, Emma Wang, Jessica Yung, Cip Baetu, Jordi Berbel, David Bridson, Jake Bruce, Gavin Buttimore, Sarah Chakera, Bilva Chandra, Paul Collins, Alex Cullum, Bogdan Damoc, Vibha Dasagi, Maxime Gazeau, Charles Gbadamosi, Woohyun Han, Ed Hirst, Ashyana Kachra, Lucie Kerley, Kristian Kjems, Eva Knoepfel, Vika Koriakin, Jessica Lo, Cong Lu, Zeb Mehring, Alex Moufarek, Henna Nandwani, Valeria Oliveira, Fabio Pardo, Jane Park, Andrew Pierson, Ben Poole, Helen Ran, Tim Salimans, Manuel Sanchez, Igor Saprykin, Amy Shen, Sailesh Sidhwani, Duncan Smith, Joe Stanton, Hamish Tomlinson, Dimple Vijaykumar, Luyu Wang, Piers Wingfield, Nat Wong, Keyang Xu, Christopher Yew, Nick Young, Vadim Zubov, Douglas Eck, Dumitru Erhan, Koray Kavukcuoglu, Demis Hassabis, Zoubin Gharamani, Raia Hadsell, Aäron van den Oord, Inbar Mosseri, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 3: A new frontier for world models. 2025. 2

[7] Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15791–15801, 2025. 2, 3

[8] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22563–22575, 2023. 9

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

[10] Ryan Burgert, Yuancheng Xu, Wenqi Xian, Oliver Pilarski, Pascal Clausen, Mingming He, Li Ma, Yitong Deng, Lingxiao Li, Mohsen Mousavi, et al. Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13–23, 2025. 3

[11] Chenjie Cao, Jingkai Zhou, Shikai Li, Jingyun Liang, Chaohui Yu, Fan Wang, Xiangyang Xue, and Yanwei Fu. Uni3c: Unifying precisely 3d-enhanced camera and human motion controls for video generation. *arXiv preprint arXiv:2504.14899*, 2025. 1, 2, 3, 7

[12] Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. Gamegen-x: Interactive open-world game video generation. *arXiv preprint arXiv:2411.00769*, 2024. 2

[13] Junyi Chen, Haoyi Zhu, Xianglong He, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Zhoujie Fu, Jiangmiao Pang, et al. Deepverse: 4d autoregressive video generation as a world model. *arXiv preprint arXiv:2506.01103*, 2025. 2

[14] Luxi Chen, Zihan Zhou, Min Zhao, Yikai Wang, Ge Zhang, Wenhao Huang, Hao Sun, Ji-Rong Wen, and Chongxuan Li. Flexworld: Progressively expanding 3d scenes for flexiable-view synthesis. *arXiv preprint arXiv:2503.13265*, 2025. 3

[15] Yingjie Chen, Yifang Men, Yuan Yao, Miaomiao Cui, and Liefeng Bo. Perception-as-control: Fine-grained controllable image animation with 3d-aware motion representation. *arXiv preprint arXiv:2501.05020*, 2025. 2, 3, 7

[16] Silvia Chiappa, Sébastien Racaniere, Daan Wierstra, and Shakir Mohamed. Recurrent environment simulators. *arXiv preprint arXiv:1704.02254*, 2017. 2

[17] Hyung Won Chung, Noah Constant, Xavier Garcia, Adam Roberts, Yi Tay, Sharan Narang, and Orhan Firat. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. *arXiv preprint arXiv:2304.09151*, 2023. 5

[18] Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. *arXiv preprint arXiv:2311.13384*, 2023. 3

[19] Dana Cohen-Bar, Elad Richardson, Gal Metzer, Raja Giryes, and Daniel Cohen-Or. Set-the-scene: Global-local training for generating controllable nerf scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2920–2929, 2023. 3

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

[21] Wanquan Feng, Jiawei Liu, Pengqi Tu, Tianhao Qi, Mingzhen Sun, Tianxiang Ma, Songtao Zhao, Siyu Zhou, and Qian He. I2vcontrol-camera: Precise video camera control with adjustable motion strength. *arXiv preprint arXiv:2411.06525*, 2024. 3

[22] Wanquan Feng, Tianhao Qi, Jiawei Liu, Mingzhen Sun, Pengqi Tu, Tianxiang Ma, Fei Dai, Songtao Zhao, Siyu Zhou, and Qian He. I2vcontrol: Disentangled and unified video motion synthesis control. *arXiv preprint arXiv:2411.17765*, 2024. 3

[23] Stefano Ferraro, Pietro Mazzaglia, Tim Verbelen, and Bart Dhoedt. Focus: object-centric world models for robotic manipulation. *Frontiers in Neurorobotics*, 19:1585386, 2025. 2

[24] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *Advances in neural information processing systems*, 29, 2016. 2

[25] Xiao Fu, Xian Liu, Xintao Wang, Sida Peng, Menghan Xia, Xiaoyu Shi, Ziyang Yuan, Pengfei Wan, Di Zhang, and Dahua Lin. 3dtrajmaster: Mastering 3d trajectory for multi-entity motion in video generation. *arXiv preprint arXiv:2412.07759*, 2024. 3

[26] Jianxiong Gao, Zhaoxi Chen, Xian Liu, Jianfeng Feng, Chenyang Si, Yanwei Fu, Yu Qiao, and Ziwei Liu. Longvie: Multimodal-guided controllable ultra-long video generation. *arXiv preprint arXiv:2508.03694*, 2025. 3

[27] Daniel Geng, Charles Herrmann, Junhwa Hur, Forrester Cole, Serena Zhang, Tobias Pfaff, Tatiana Lopez-Guevara, Carl Doersch, Yusuf Aytar, Michael Rubinstein, et al. Motion prompting: Controlling video generation with motion trajectories. *arXiv preprint arXiv:2412.02700*, 2024. 3

[28] Zekai Gu, Rui Yan, Jiahao Lu, Peng Li, Zhiyang Dou, Chenyang Si, Zhen Dong, Qifeng Liu, Cheng Lin, Ziwei Liu, et al. Diffusion as shader: 3d-aware video diffusion for versatile video generation control. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–12, 2025. 3

[29] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. *Advances in neural information processing systems*, 31, 2018. 2

[30] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3), 2018. 2

[31] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pages 2555–2565. PMLR, 2019. 2

[32] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023. 2

[33] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024. 3, 6, 12

[34] Hao He, Ceyuan Yang, Shanchuan Lin, Yinghao Xu, Meng Wei, Liangke Gui, Qi Zhao, Gordon Wetzstein, Lu Jiang, and Hongsheng Li. Cameractrl ii: Dynamic scene exploration via camera-controlled video diffusion models. *arXiv preprint arXiv:2503.10592*, 2025. 3

[35] Xuehai He, Shuohang Wang, Jianwei Yang, Xiaoxia Wu, Yiping Wang, Kuan Wang, Zheng Zhan, Olatunji Ruwase, Yelong Shen, and Xin Eric Wang. Mojito: Motion trajectory and intensity control for video generation. *arXiv preprint arXiv:2412.08948*, 2024. 3

[36] Xianglong He, Chunli Peng, Zexiang Liu, Boyang Wang, Yifan Zhang, Qi Cui, Fei Kang, Biao Jiang, Mengyin An, Yangyang Ren, et al. Matrix-game 2.0: An open-source, real-time, and streaming interactive world model. *arXiv preprint arXiv:2508.13009*, 2025. 2

[37] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 9

[38] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 3

[39] Chen Hou, Guoqiang Wei, Yan Zeng, and Zhibo Chen. Training-free camera control for video generation. *arXiv preprint arXiv:2406.10126*, 2024. 3

[40] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023. 2

[41] Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson WH Lau, Wangmeng Zuo, et al. Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. *arXiv preprint arXiv:2506.04225*, 2025. 2, 7

[42] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025. 4, 9

[43] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative

models. *Advances in neural information processing systems*, 35:26565–26577, 2022. 9

[44] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 3

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

[46] Zhengfei Kuang, Shengqu Cai, Hao He, Yinghao Xu, Hongsheng Li, Leonidas J Guibas, and Gordon Wetzstein. Collaborative video diffusion: Consistent multi-video generation with camera control. *Advances in Neural Information Processing Systems*, 37:16240–16271, 2024. 3

[47] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 13

[48] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1): 1–62, 2022. 2

[49] Haoran Li, Haolin Shi, Wenli Zhang, Wenjun Wu, Yong Liao, Lin Wang, Lik-hang Lee, and Peng Yuan Zhou. Dreamscene: 3d gaussian-based text-to-3d scene generation via formation pattern sampling. In *European Conference on Computer Vision*, pages 214–230. Springer, 2024. 3

[50] Jiaqi Li, Junshu Tang, Zhiyong Xu, Longhuang Wu, Yuan Zhou, Shuai Shao, Tianbao Yu, Zhiguo Cao, and Qinglin Lu. Hunyuan-gamecraft: High-dynamic interactive game video generation with hybrid history condition. *arXiv preprint arXiv:2506.17201*, 2025. 2

[51] Quanhao Li, Zhen Xing, Rui Wang, Hui Zhang, Qi Dai, and Zuxuan Wu. Magicmotion: Controllable video generation with dense-to-sparse trajectory guidance. *arXiv preprint arXiv:2503.16421*, 2025. 3

[52] Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. Vmem: Consistent interactive video scene generation with surfel-indexed view memory. *arXiv preprint arXiv:2506.18903*, 2025. 2

[53] Teng Li, Guangcong Zheng, Rui Jiang, Tao Wu, Yehao Lu, Yining Lin, Xi Li, et al. Realcam-i2v: Real-world image-to-video generation with interactive complex camera control. *arXiv preprint arXiv:2502.10059*, 2025. 3

[54] Xinyang Li, Tengfei Wang, Zixiao Gu, Shengchuan Zhang, Chunchao Guo, and Liujuan Cao. Flashworld: High-quality 3d scene generation within seconds. *arXiv preprint arXiv:2510.13678*, 2025. 3, 7

[55] Yaowei Li, Xintao Wang, Zhaoyang Zhang, Zhouxia Wang, Ziyang Yuan, Liangbin Xie, Yuexian Zou, and Ying Shan. Image conductor: Precision control for interactive video synthesis. *arXiv preprint arXiv:2406.15339*, 2024. 3

[56] Zhen Li, Chuanhao Li, Xiaofeng Mao, Shaoheng Lin, Ming Li, Shitian Zhao, Zhaopan Xu, Xinyue Li, Yukang Feng, Jianwen Sun, et al. Sekai: A video dataset towards world exploration. *arXiv preprint arXiv:2506.15675*, 2025. 5

[57] Jingyun Liang, Jingkai Zhou, Shikai Li, Chenjie Cao, Lei Sun, Yichen Qian, Weihua Chen, and Fan Wang. Realismotion: Decomposed human motion control and video generation in the world space. *arXiv preprint arXiv:2508.08588*, 2025. 3

[58] Xinyao Liao, Xianfang Zeng, Liao Wang, Gang Yu, Guosheng Lin, and Chi Zhang. Motionagent: Fine-grained controllable video generation via motion field agent. *arXiv preprint arXiv:2502.03207*, 2025. 2, 3

[59] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 9

[60] Yifan Liu, Zhiyuan Min, Zhenwei Wang, Junta Wu, Tengfei Wang, Yixuan Yuan, Yawei Luo, and Chunchao Guo. Worldmirror: Universal 3d world reconstruction with any-prior prompting. *arXiv preprint arXiv:2510.10726*, 2025. 3

[61] Taiming Lu, Tianmin Shu, Junfei Xiao, Luoxin Ye, Jiahao Wang, Cheng Peng, Chen Wei, Daniel Khashabi, Rama Chellappa, Alan Yuille, et al. Genex: Generating an explorable world. *arXiv preprint arXiv:2412.09624*, 2024. 3

[62] Wan-Duo Kurt Ma, John P Lewis, and W Bastiaan Kleijn. Trailblazer: Trajectory control for diffusion-based video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024. 3

[63] Xiaofeng Mao, Shaoheng Lin, Zhen Li, Chuanhao Li, Wenshuo Peng, Tong He, Jiangmiao Pang, Mingmin Chi, Yu Qiao, and Kaipeng Zhang. Yume: An interactive world generation model. *arXiv preprint arXiv:2507.17744*, 2025. 1, 2, 7

[64] Willi Menapace, Stephane Lathuiliere, Sergey Tulyakov, Aliaksandr Siarohin, and Elisa Ricci. Playable video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10061–10070, 2021. 2

[65] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3

[66] Chong Mou, Mingdeng Cao, Xintao Wang, Zhaoyang Zhang, Ying Shan, and Jian Zhang. Revideo: Remake a video with motion and content control. *arXiv preprint arXiv:2405.13865*, 2024. 3

[67] Muyao Niu, Xiaodong Cun, Xintao Wang, Yong Zhang, Ying Shan, and Yinqiang Zheng. Mofa-video: Controllable image animation via generative motion field adaptions in frozen image-to-video diffusion model. In *European Conference on Computer Vision*, pages 111–128. Springer, 2025. 3

[68] Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. *Advances in neural information processing systems*, 28, 2015. 2

[69] Karran Pandey, Matheus Gadelha, Yannick Hold-Geoffroy, Karan Singh, Niloy J Mitra, and Paul Guerrero. Motion modes: What could happen next? *arXiv preprint arXiv:2412.00148*, 2024. 3

[70] Jack Parker-Holder, Philip Ball, Jake Bruce, Vibhavari Dasagi, Kristian Holsheimer, Christos Kaplanis, Alexandre Moufarek, Guy Scully, Jeremy Shar, Jimmy Shi, Stephen Spencer, Jessica Yung, Michael Dennis, Sultan Kenjeyev, Shangbang Long, Vlad Mnih, Harris Chan, Maxime Gazeau, Bonnie Li, Fabio Pardo, Luyu Wang, Lei Zhang, Frederic Besse, Tim Harley, Anna Mitenkova, Jane Wang, Jeff Clune, Demis Hassabis, Raia Hadsell, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 2: A large-scale foundation world model. 2024. 2

[71] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 9

[72] Luigi Piccinelli, Christos Sakaridis, Yung-Hsu Yang, Mattia Segu, Siyuan Li, Wim Abbeloos, and Luc Van Gool. Unidepthv2: Universal monocular metric depth estimation made simpler. *arXiv preprint arXiv:2502.20110*, 2025. 5

[73] Ryan Po, Yotam Nitzan, Richard Zhang, Berlin Chen, Tri Dao, Eli Shechtman, Gordon Wetzstein, and Xun Huang. Long-context state-space video world models. *arXiv preprint arXiv:2505.20171*, 2025. 2

[74] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 3

[75] Stefan Popov, Amit Raj, Michael Krainin, Yuanzhen Li, William T Freeman, and Michael Rubinstein. Camctrl3d: Single-image scene exploration with precise 3d camera control. *arXiv preprint arXiv:2501.06006*, 2025. 3

[76] Haonan Qiu, Zhaoxi Chen, Zhouxia Wang, Yingqing He, Menghan Xia, and Ziwei Liu. Freetraj: Tuning-free trajectory control in video diffusion models. *arXiv preprint arXiv:2406.16863*, 2024. 3

[77] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024. 3

[78] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6121–6132, 2025. 3

[79] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 3, 9

[80] Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, et al. Zeronvs: Zero-shot 360-degree view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9420–9429, 2024. 3

[81] Manuel-Andreas Schneider, Lukas Höllein, and Matthias Nießner. Worldexplorer: Towards generating fully navigable 3d scenes. *arXiv preprint arXiv:2506.01799*, 2025. 3

[82] Xiaoyu Shi, Zhaoyang Huang, Fu-Yun Wang, Weikang Bian, Dasong Li, Yi Zhang, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, et al. Motion-i2v: Consistent and controllable image-to-video generation with explicit motion modeling. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 3

[83] Xincheng Shuai, Henghui Ding, Zhenyuan Qin, Hao Luo, Xingjun Ma, and Dacheng Tao. Free-form motion control: A synthetic video generation dataset with controllable camera and object motions. *arXiv preprint arXiv:2501.01425*, 2025. 3

[84] Wenqiang Sun, Shuo Chen, Fangfu Liu, Zilong Chen, Yueqi Duan, Jun Zhang, and Yikai Wang. Dimensionx: Create any 3d and 4d scenes from a single image with controllable video diffusion. *arXiv preprint arXiv:2411.04928*, 2024. 3

[85] Maham Tanveer, Yang Zhou, Simon Niklaus, Ali Mahdavi Amiri, Hao Zhang, Krishna Kumar Singh, and Nanxuan Zhao. Motionbridge: Dynamic video inbetweening with flexible controls. *arXiv preprint arXiv:2412.13190*, 2024. 3

[86] HunyuanWorld Team, Zhenwei Wang, Yuhao Liu, Junta Wu, Zixiao Gu, Haoyuan Wang, Xuhui Zuo, Tianyu Huang, Wenhuan Li, Sheng Zhang, et al. Hunyuanworld 1.0: Generating immersive, explorable, and interactive 3d worlds from words or pixels. *arXiv preprint arXiv:2507.21809*, 2025. 3

[87] Ruben Villegas, Arkanath Pathak, Harini Kannan, Dumitru Erhan, Quoc V Le, and Honglak Lee. High fidelity video prediction with large stochastic recurrent neural networks. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[88] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 2, 4, 9

[89] Zhang Wan, Sheng Tang, Jiawei Wei, Ruize Zhang, and Juan Cao. Dragentity: Trajectory guided video generation using entity and positional relationships. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 108–116, 2024. 3

[90] Hanlin Wang, Hao Ouyang, Qiuyu Wang, Wen Wang, Ka Leong Cheng, Qifeng Chen, Yujun Shen, and Limin Wang. Levitor: 3d trajectory oriented image-to-video synthesis. *arXiv preprint arXiv:2412.15214*, 2024. 3

[91] Jiawei Wang, Yuchen Zhang, Jiaxin Zou, Yan Zeng, Guoqiang Wei, Liping Yuan, and Hang Li. Boximator: Generating rich and controllable motions for video synthesis. *arXiv preprint arXiv:2402.01566*, 2024. 2, 3

[92] Jiahao Wang, Luoxin Ye, TaiMing Lu, Junfei Xiao, Jiahan Zhang, Yuxiang Guo, Xijun Liu, Rama Chellappa, Cheng Peng, Alan Yuille, et al. Evoworld: Evolving panoramic world generation with explicit 3d memory. *arXiv preprint arXiv:2510.01183*, 2025. 3

[93] Jiahao Wang, Yufeng Yuan, Rujie Zheng, Youtian Lin, Jian Gao, Lin-Zhuo Chen, Yajie Bao, Yi Zhang, Chang Zeng,

Yanxi Zhou, et al. Spatialvid: A large-scale video dataset with spatial annotations. *arXiv preprint arXiv:2509.09676*, 2025. 5

[94] Qinghe Wang, Yawen Luo, Xiaoyu Shi, Xu Jia, Huchuan Lu, Tianfan Xue, Xintao Wang, Pengfei Wan, Di Zhang, and Kun Gai. Cinemaster: A 3d-aware and controllable framework for cinematic text-to-video generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–10, 2025. 2, 3

[95] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *arXiv preprint arXiv:2507.02546*, 2025. 3, 5

[96] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability. *Advances in Neural Information Processing Systems*, 36:7594–7611, 2023. 3

[97] Zhouxia Wang, Yushi Lan, Shangchen Zhou, and Chen Change Loy. Objctrl-2.5 d: Training-free object control with camera poses. *arXiv preprint arXiv:2412.07721*, 2024. 3

[98] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2, 3, 6, 12

[99] Zun Wang, Jaemin Cho, Jialu Li, Han Lin, Jaehong Yoon, Yue Zhang, and Mohit Bansal. Epic: Efficient video camera control learning with precise anchor-video guidance. *arXiv preprint arXiv:2505.21876*, 2025. 3

[100] Jianzong Wu, Xiangtai Li, Yanhong Zeng, Jiangning Zhang, Qianyu Zhou, Yining Li, Yunhai Tong, and Kai Chen. Motionbooth: Motion-aware customized text-to-video generation. *Advances in Neural Information Processing Systems*, 37:34322–34348, 2024. 3

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

[102] Weijia Wu, Zhuang Li, Yuchao Gu, Rui Zhao, Yefei He, David Junhao Zhang, Mike Zheng Shou, Yan Li, Tingting Gao, and Di Zhang. Draganything: Motion control for anything using entity representation. In *European Conference on Computer Vision*, pages 331–348. Springer, 2025. 3

[103] Jiannan Xiang, Guangyi Liu, Yi Gu, Qiyue Gao, Yuting Ning, Yuheng Zha, Zeyu Feng, Tianhua Tao, Shibo Hao, Yemin Shi, et al. Pandora: Towards general world model with natural language actions and video states. *arXiv preprint arXiv:2406.09455*, 2024. 2

[104] Zeqi Xiao, Wenqi Ouyang, Yifan Zhou, Shuai Yang, Lei Yang, Jianlou Si, and Xingang Pan. Trajectory attention for fine-grained video motion control. *arXiv preprint arXiv:2411.19324*, 2024. 3

[105] Jinbo Xing, Long Mai, Cusuh Ham, Jiahui Huang, Aniruddha Mahapatra, Chi-Wing Fu, Tien-Tsin Wong, and Feng Liu. Motioncanvas: Cinematic shot design with controllable image-to-video generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–11, 2025. 3

[106] Dejia Xu, Weili Nie, Chao Liu, Sifei Liu, Jan Kautz, Zhangyang Wang, and Arash Vahdat. Camco: Camera-controllable 3d-consistent image-to-video generation. *arXiv preprint arXiv:2406.02509*, 2024. 3

[107] Tianshuo Xu, Zhifei Chen, Leyi Wu, Hao Lu, Yuying Chen, Lihui Jiang, Bingbing Liu, and Yingcong Chen. Motion dreamer: Realizing physically coherent video generation through scene-aware motion reasoning. *arXiv preprint arXiv:2412.00547*, 2024. 3

[108] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024. 3

[109] Shiyuan Yang, Liang Hou, Haibin Huang, Chongyang Ma, Pengfei Wan, Di Zhang, Xiaodong Chen, and Jing Liao. Direct-a-video: Customized video generation with user-directed camera movement and object motion. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–12, 2024. 3

[110] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 2

[111] Zhongqi Yang, Wenhang Ge, Yuqi Li, Jiaqi Chen, Haoyuan Li, Mengyin An, Fei Kang, Hua Xue, Baixin Xu, Yuyang Yin, et al. Matrix-3d: Omnidirectional explorable 3d world generation. *arXiv preprint arXiv:2508.08086*, 2025. 2, 3

[112] Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. *arXiv preprint arXiv:2308.08089*, 2023. 3

[113] Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6658–6667, 2024. 3

[114] Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5916–5926, 2025. 3

[115] Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos. *arXiv preprint arXiv:2501.08325*, 2025. 2

[116] Jason J Yu, Fereshteh Forghani, Konstantinos G Derpanis, and Marcus A Brubaker. Long-term photometric consistent novel view synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7094–7104, 2023. 3

[117] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying

Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024. 2, 3, 7

[118] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pages 1–19. Springer, 2024. 3

[119] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023. 2, 10

[120] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024. 3

[121] Qihang Zhang, Chaoyang Wang, Aliaksandr Siarohin, Peiye Zhuang, Yinghao Xu, Ceyuan Yang, Dahua Lin, Bolei Zhou, Sergey Tulyakov, and Hsin-Ying Lee. Towards text-guided 3d scene composition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6829–6838, 2024. 3

[122] Shengjun Zhang, Jinzhao Li, Xin Fei, Hao Liu, and Yueqi Duan. Scene splatter: Momentum 3d scene generation from single image with video diffusion model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6089–6098, 2025. 3

[123] Zhenghao Zhang, Junchao Liao, Menghao Li, Zuozhuo Dai, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. Tora: Trajectory-oriented diffusion transformer for video generation. *arXiv preprint arXiv:2407.21705*, 2024. 3

[124] Zhiyuan Zhang, Dongdong Chen, and Jing Liao. I2v3d: Controllable image-to-video generation with 3d guidance. *arXiv preprint arXiv:2503.09733*, 2025. 2, 3

[125] Zhongwei Zhang, Fuchen Long, Zhaofan Qiu, Yingwei Pan, Wu Liu, Ting Yao, and Tao Mei. Motionpro: A precise motion controller for image-to-video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 27957–27967, 2025. 2

[126] Guangcong Zheng, Teng Li, Rui Jiang, Yehao Lu, Tao Wu, and Xi Li. Cami2v: Camera-controlled image-to-video diffusion model. *arXiv preprint arXiv:2410.15957*, 2024. 3

[127] Sixiao Zheng, Zimian Peng, Yanpeng Zhou, Yi Zhu, Hang Xu, Xiangru Huang, and Yanwei Fu. Vidcraft3: Camera, object, and lighting control for image-to-video generation. *arXiv preprint arXiv:2502.07531*, 2025. 2, 3

[128] Haitao Zhou, Chuang Wang, Rui Nie, Jinlin Liu, Dongdong Yu, Qian Yu, and Changhu Wang. Trackgo: A flexible and efficient method for controllable video generation. *arXiv preprint arXiv:2408.11475*, 2024. 3

[129] Haoyi Zhu, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Chunhua Shen, Jiangmiao Pang, and Tong He. Aether: Geometric-aware unified world modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8535–8546, 2025. 3