

Gen Through the Lens: Camera-Controllable Video Generation via Cross-Coordinate Conditioning Fusion

Shuai Li^{*}, Jinxu Xiang^{*}, Xinyuan Li, Hanlei Guo, Xiyang Tan, Alan Zhao, Xiaotong Zhao, Ran Zhang[†]
Tencent Video Solar System

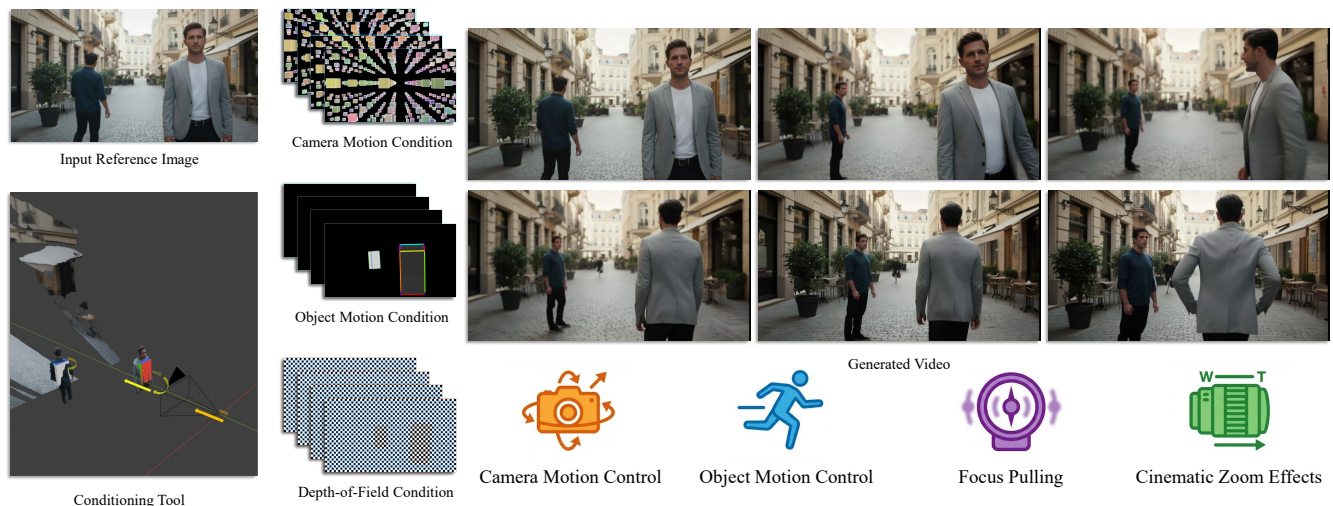


Figure 1: Overview of Gen Through the Lens, the first image-to-video generation framework that jointly enables controllable camera motion, object motion, and depth-of-field effects. Starting from a single input image, our model supports cinematic camera trajectories, zoom and focus transitions, and stable object dynamics in a physically consistent manner. The lower left shows a modeling software environment for editing camera trajectories, object motion, and depth-of-field information. The second column shows the three complementary conditioning signals, and the right side shows the generated video results, with the model’s key controllable capabilities summarized below.

Abstract

We present Gen Through the Lens, a camera-controllable image-to-video generation framework that synthesizes videos from a given reference image while enabling realistic and physically consistent camera operations. Unlike previous models that treat camera motion as abstract conditioning or rely on a single coordinate system, our approach explicitly decomposes and fuses control signals across three coordinate spaces: the camera coordinate (modeled by intrinsic parameters), the world coordinate (represented by spatial geometry hints), and the object coordinate (described by 3D bounding boxes of target subjects). Through cross-coordinate conditioning fusion, these heterogeneous cues are unified to guide the generative process, enabling precise control of camera motion, focus, and framing while preserving temporal and spatial coherence. Extensive experiments demonstrate that our method produces controllable and visually consistent results across diverse camera operations, outperforming existing approaches in both realism and editability.

1 Introduction

Recent advances in video generation have achieved impressive visual realism, yet providing precise and physically meaningful control over camera behavior remains a fundamental challenge.

Most existing generative models either treat camera motion as an implicit byproduct of appearance statistics or encode it using low-level signals such as optical flow or relative pose embeddings, making it difficult to specify complex camera operations such as metric-scale motion, depth-of-field variation, or coupled changes in camera intrinsics and extrinsics. A key difficulty lies in the heterogeneous nature of camera-related information, which simultaneously involves intrinsic parameters (e.g., focal length and distortion), extrinsic motion in 3D space, and interactions with scene geometry and object motion. Collapsing these factors into a single coordinate system or abstract latent codes obscures their physical meaning and limits controllability, particularly for advanced cinematic effects such as depth-of-field control or the Hitchcock zoom [10].

In this work, we present **Gen Through the Lens** (GTL), an image-to-video generation framework that enables explicit and physically grounded control over camera motion, object motion, and depth-of-field by decomposing and fusing camera-related signals across multiple coordinate spaces. Our approach introduces a cross-coordinate conditioning formulation that jointly models (i) the *camera coordinate*, capturing intrinsic parameters and lens effects, (ii) the *world coordinate*, encoding metric-scale camera motion and spatial geometry, and (iii) the *object coordinate*, representing object motion through 6D bounding boxes parameterized by 3D

^{*}Equal contribution. [†] Corresponding author.

translation and 3D rotation. By preserving the physical meaning of each coordinate space and explicitly fusing them during generation, our framework enables precise and interpretable control over camera motion, focus, and framing.

Building upon this formulation, we design three complementary conditioning signals corresponding to the three coordinate spaces: (1) a *camera motion condition*, which visualizes metric-scale camera extrinsics and intrinsics in the world coordinate space and enables explicit control over camera translation, rotation, and zoom; (2) an *object motion condition*, which represents foreground dynamics using 3D bounding boxes with 6D pose, thereby enabling disentangled control over object translation and rotation; and (3) a *depth-of-field condition*, which encodes focus depth and defocus behavior using layered depth cues, allowing explicit and time-varying control of bokeh effects. These conditions are geometrically consistent and mutually complementary, jointly enabling controllable video synthesis with coherent camera motion, stable object dynamics, and physically meaningful depth-of-field behavior.

However, effectively leveraging physically grounded conditions requires accurate and metrically consistent camera supervision, which is difficult to obtain from unconstrained videos due to scale ambiguity, temporal instability, and intrinsic-extrinsic inconsistencies. To address this challenge, we introduce a metric-scale dataset labeling pipeline that combines relative camera pose estimation with monocular metric depth and robust scale recovery, constructing a unified metric camera space for reliable camera-controllable supervision.

Extensive quantitative evaluations demonstrate that our method significantly improves camera controllability, geometric consistency, and motion stability compared to state-of-the-art approaches. In addition, qualitative results showcase compelling cinematic applications, including controllable depth-of-field effects and stable synthesis of the Hitchcock zoom. Together, these results highlight the importance of explicit, cross-coordinate camera modeling for controllable and physically consistent video generation.

2 Related Work

2.1 Camera-Controllable Video Generation

Camera motion control in generative video has evolved from coarse motion cues to explicit trajectory conditioning. Early text-based approaches such as AnimateDiff [7] and MotionDirector [42] typically fine-tune LoRA modules for a small set of motion patterns, providing convenient but imprecise control. Recent trajectory-guided methods instead condition video diffusion on camera parameters. MotionCtrl [36] conditions generation on raw camera parameters repeated over the spatial grid, while CameraCtrl [8] and CamCo [38] encode camera parameters as pixel-wise Plücker ray embeddings to obtain more discriminative viewpoint control. Several works lift the reference image to 3D and render proxy views for conditioning, such as ViewCrafter [40] and I2VControl-Camera [5], improving spatial coherence under viewpoint changes. To better handle long-range motion and cross-frame geometry, CamI2V [44] introduces epipolar attention mechanisms, and RealCam-I2V [15] further targets real-world usability by aligning trajectories to metric scale. Beyond pure conditioning, GEN3C [23] strengthens controllability

by maintaining a 3D cache rendered along user-specified trajectories, which also improves temporal 3D consistency. Despite these advances, most trajectory-based controllers still rely on approximate geometry or relative-scale supervision and treat the camera as a simplified projection model, which can lead to scale drift, limited robustness under large rotations or occlusions, and weak support for optical phenomena beyond pose changes.

2.2 Object-Controllable Video Generation

Object control in generative video has progressed from coarse region constraints to explicit trajectory-driven motion specification. Early approaches primarily rely on spatial conditions to anchor object placement but provide limited control over object dynamics. Recent works instead adopt drag-style interaction, where sparse point trajectories convey user intent and are injected into diffusion models through explicit motion representations. DragNUWA [39] introduces joint text-image-trajectory conditioning for open-domain object manipulation, while ImageConductor [16] decouples camera and object motions via separate LoRA branches for more precise interactive control. Many methods use optical flow as an intermediate representation: sparse trajectories are converted to sparse or dense flow to guide temporal attention or feature warping, as in Motion-I2V [26] and MOFA-Video [20]. Beyond flow, Tora [41] encodes trajectories into a trajectory map and integrates them into a scalable diffusion-transformer pipeline via a trajectory extractor and fusion module, and DragAnything [37] uses an entity representation with mask-based feature reorganization to improve object-level coherence under dragging. Despite improved controllability, these methods remain sensitive to trajectory placement and depth or occlusion ambiguities, can introduce local distortions under large shape variations, and often suffer from control conflicts when object motion and global camera motion are specified simultaneously.

2.3 Collaborative Motion Control

Collaborative motion control targets video generation with simultaneous camera motion and local object dynamics, where naive 2D motion controls often lead to local conflicts under complex camera movements. Early approaches unify camera and object motion in a single 2D representation. AnimateAnything [14] converts both motions into a unified optical-flow field, but entangles global camera-induced displacement with object-centric motion, limiting geometric interpretability. Subsequent works adopt point-based motion control derived from 2D perception signals. MotionPrompting [6] and I2VControl-Camera [5] leverage dense point embeddings to guide motion generation, but collaborative control remains rooted in 2D cues and is typically specified at region or segmentation-mask granularity, restricting fine-grained camera-object interaction modeling. Perception-as-Control [2] introduces a 3D-aware motion representation with separate encoders for camera and object control to reduce local conflicts. Despite improved motion harmony, collaborative approaches still depend on upstream perception quality, simplified 3D surrogates may miss motion factors, and overall controllability remains constrained by the underlying generator.

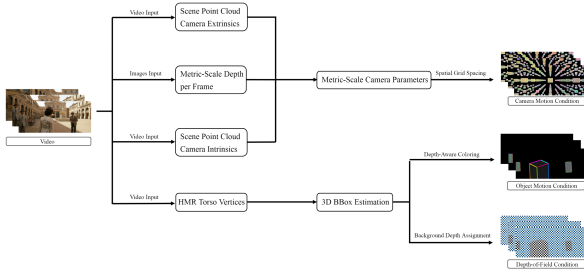


Figure 2: Overview of the data processing pipeline. Recovered metric-scale camera parameters and human-centered 3D geometry are converted into geometry-aware conditions for camera motion, object motion, and depth-of-field control.

3 Method

In this section, we present a metric-scale supervision and conditioning pipeline that first recovers semantic, camera, object, and depth-related supervision from real-world videos, and then constructs geometry-aware conditioning signals for controllable video generation. Together, these components enable disentangled yet coherent control over camera motion, object motion, depth-of-field, and zoom. The overall data processing workflow is illustrated in Fig. 2. Detailed implementation of each component is provided in Appendix Sec. A.

3.1 Label Prompt

We use InternVL2.5 [3] to generate dense textual descriptions for each video, which serve as semantic supervision during training. To avoid information leakage and ensure disentanglement between camera control and content description, we explicitly constrain the prompt to exclude camera-related information, such as camera motion, viewpoint, and shooting angles. The prompt is designed to encourage temporally coherent, detailed, and factually accurate descriptions that focus on scene content and object dynamics.

3.2 Camera Modeling

Metric-scale camera extrinsics and accurate camera intrinsics are essential for injecting reliable *camera motion condition* into camera-controllable generative models. Wan2.1 [29] is trained on a RealEstate-style dataset [45] that effectively normalizes camera trajectories under a fixed foreground depth assumption; when this assumption is violated at inference, the generated motion exhibits noticeable scale inconsistencies. To address this issue, we build a labeling pipeline that recovers metric-scale camera extrinsics and intrinsics for each video, forming a unified *metric camera space*.

Accurate camera-controllable video generation requires camera extrinsics and intrinsics that are both geometrically reliable and consistent under metric scale. However, existing methods typically specialize in only part of this problem, such as temporally stable camera motion, per-frame metric depth, or accurate intrinsic estimation. We therefore combine three complementary approaches—Pi3 [35], Moge2 [33], and MapAnything [13]—to recover metric-consistent camera parameters. Specifically, Pi3 provides temporally stable camera extrinsics up to an unknown global scale, Moge2 predicts

per-frame metric depth, and MapAnything provides reliable camera intrinsics.

We first estimate a robust global scale factor from Pi3 and Moge2 using pairwise 3D distance ratios across the video, which resolves the scale ambiguity in camera extrinsics. We then align the camera intrinsics predicted by MapAnything to the same metric space using a consistent scaling strategy, ensuring geometrically accurate projection across frames. The resulting metric-scale camera parameters are used as supervision for training. Quantitative evaluation of the recovered camera parameters is reported in Table 1.

Based on the recovered parameters, we construct a *camera motion condition* using an intermediate world-coordinate representation rather than raw Plücker embeddings [21] or per-frame point clouds. We normalize camera poses to the first frame and represent camera trajectories using a structured grid in world coordinates, which is rendered with the recovered intrinsics to ensure geometric consistency across frames. This representation encodes camera motion in a structured and controllable form.

3.3 3D Bounding Box Modeling

Controlling object motion solely from 2D image coordinates inevitably discards spatial information, including depth ordering under occlusion and object orientation. This ambiguity motivates the need for an explicit *object motion condition*.

Following Cinemaster [32], we represent object motion using 3D bounding boxes in full 6D, including 3D translation and 3D rotation. We focus on human subjects, as their geometry and motion can be reliably reconstructed from monocular videos using off-the-shelf Human Mesh Recovery (HMR) methods, enabling scalable automatic annotation. We employ TokenHMR [4] for identity-consistent tracking, and fit bounding boxes only to the rigid torso vertices of the recovered SMPL mesh [17]. This design mitigates large shape variations caused by articulated limbs or head motion (e.g., bending or sitting), resulting in more stable bounding boxes and temporally consistent 6D object motion supervision.

We then convert the recovered 3D boxes into an *object motion condition* that provides explicit local spatial cues. Complementing the camera motion condition, which defines the global metric space, the 3D bounding box condition indicates precise screen-space occupancy, relative depth ordering, and orientation. The condition is rendered as a projected 3D bounding box with depth-aware filling and consistent edge coloring, enabling the model to infer object rotation and placement without explicit angle representations. As shown in Fig. 5, although the model is trained only on humans, the learned object motion condition generalizes well to generic non-human objects.

3.4 Depth-of-Field Modeling

To model depth-of-field effects, we construct *depth-of-field condition* signals using recovered metric depth and a physically motivated bokeh operator following AkiRa [34], where blur is determined by the deviation between pixel depth and the focal plane. We use the same operator for both bokeh-augmented data generation and condition construction, ensuring consistency between training signals and supervision. The recovered metric camera space enables flexible focus modeling. By specifying time-varying focus depths,

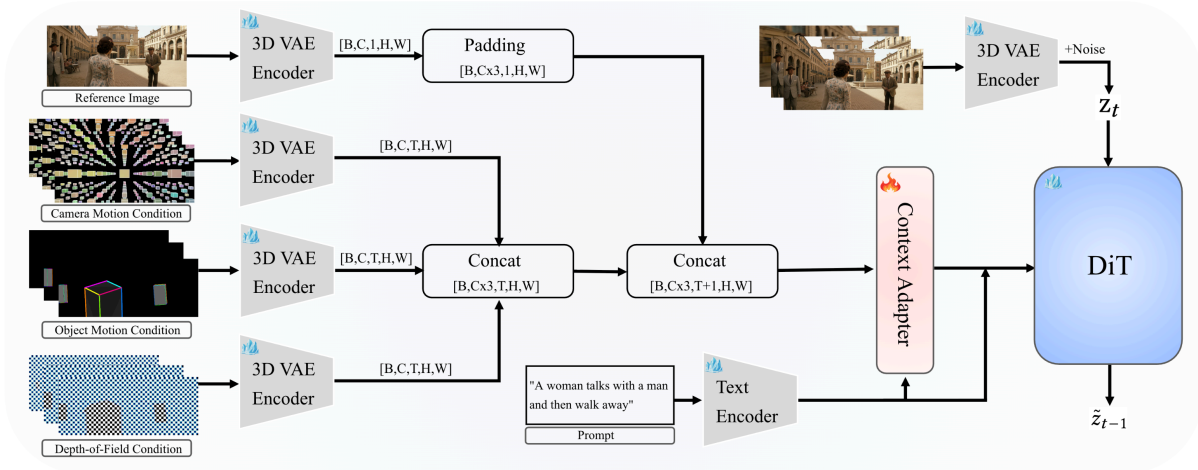


Figure 3: Overview of the proposed framework. The model jointly conditions video generation on camera motion, object motion, and depth-of-field signals across camera, world, and object coordinate spaces.

Method	Cam Trans. Err. ↓	Cam Rot. Err. (°) ↓
VGGT	41.25	20.34
Pi3	40.36	17.21
MapAnything	12.39	17.43
Ours	11.74	17.11

Table 1: Metric-scale camera pose accuracy on the MultiCamVideo Dataset with ground-truth camera annotations. Methods that estimate camera poses only up to scale (VGGT and Pi3) exhibit large errors on sequences with significant camera motion. Our method recovers metrically consistent camera parameters and achieves lower errors, validating the effectiveness of the proposed pipeline.

we can synthesize diverse behaviors such as smooth focus transitions and subject tracking. To improve temporal stability, we apply a boundary-aware smoothing step to mitigate artifacts near large depth discontinuities.

For conditioning, we adopt a layered representation instead of directly encoding pixel-wise focus distance. We approximate unknown background depth using a small set of canonical depth planes and combine them with foreground depth derived from 3D bounding boxes. The resulting representation captures the interaction between focus distance and scene depth while avoiding reliance on accurate full-scene depth estimation.

3.5 Zoom Video Augmentation

In practical cinematography, zoom is an essential component of camera behavior, yet recovering time-varying focal lengths directly from unconstrained videos is highly unreliable, and such sequences are difficult to curate at scale. We therefore synthesize zoom effects through data augmentation. Specifically, we model focal-length changes by applying a temporal center crop with randomly sampled crop factors and zoom intervals, and resizing the result back to the native resolution to simulate push-in and pull-out shots while keeping the final image resolution fixed. Once the zoom parameters

are sampled for a video, the same transformation is consistently applied to all conditioning streams—including the camera motion condition video, the object motion condition video, and the depth-of-field condition video—to preserve pixel-wise alignment.

4 Experiments

4.1 Training Paradigm

We curate a dataset with significant camera and object dynamics from DynPose-100K [24], MultiCamVideo [1], BEDLAM2.0 [27], and internal data, totaling 110k samples with camera trajectories and 3D bounding box annotations. We further construct a zoom dataset (8k) and a bokeh dataset (15k). All training videos are unified to 81 frames at 480p resolution.

We build GTL 14B on Wan2.1-14B [29], freezing the encoder, decoder, and DiT, and training only the Context Adapter. Training follows a three-stage curriculum from camera to object and finally to bokeh control. We initialize from partial VACE DiT weights [11]. Stage 1 trains camera control using trajectory annotations; Stage 2 adds 3D bounding boxes and zoom data for object motion and zoom; Stage 3 introduces bokeh data for joint control. The three stages run for 11k, 13k, and 7k iterations with batch size 8 and learning rate 5×10^{-5} , totaling approximately 288 GPU hours on 8 NVIDIA H800 GPUs. An overview of the model and training pipeline is shown in Fig. 3.

4.2 Evaluation Metrics

We present a comprehensive evaluation of our method across multiple complementary aspects, including camera controllability, geometric consistency, motion fidelity, and visual quality. For metrics that rely on extracted signals, we apply the same extraction pipeline to both the real input videos and the generated videos, and compare the recovered features, using explicit ground truth when available and real-video signals as pseudo-ground-truth otherwise. Brief metric descriptions are given below, with full details provided in Appendix B.

Method	FID ↓	FVD ↓	Qwen VQA Score ↑	Cam Trans. Err. ↓	Cam Rot. Err. (°) ↓	mIoU ↑	BBox Center Err. (%) ↓	BBox Rot. Err. (°) ↓	Traj. Err. (%) ↓
PAC	70.17	172.51	0.56	18.15	8.77	0.52	6.37	41.83	8.68
RC-I2V	55.12	98.76	0.70	9.86	4.63	0.45*	7.42*	47.34*	7.29
GEN3C	64.24	114.33	0.62	11.38	5.02	0.47*	8.24*	46.22*	8.39
GTL 14B	53.71	86.54	0.81	8.59	4.89	0.61	4.40	21.73	6.37

Table 2: Quantitative comparison with state-of-the-art methods. Lower is better (↓) for error-based metrics and latent space distance, while higher is better (↑) for overlap and semantic metrics. Screen-space errors are reported as percentages of the image diagonal. Metrics marked with * should be interpreted with caution, as RealCam-I2V and GEN3C do not support explicit control of object motion or target object positioning, which may bias object-related metrics toward worse values.

Camera motion accuracy. Camera controllability is evaluated using two metrics: (i) camera translation error, which quantifies relative deviation of the camera trajectory in 3D space; and (ii) camera rotation error, which measures orientation accuracy using geodesic distance on $SO(3)$. All camera trajectories are normalized to the first frame to ensure a consistent reference across sequences.

Object-level geometric stability. To assess foreground consistency under camera motion, we report the mean Intersection-over-Union (mIoU) measured by SAM2 [22] between predicted and reference foreground masks. In addition, we evaluate 3D bounding box pose accuracy using both screen-space center deviation and 3D rotational error, which measure how accurately the generated videos follow the object controls.

Trajectory-based motion accuracy. We assess motion fidelity using a trajectory consistency error computed from point tracks extracted by CoTracker [12]. This metric measures the alignment between induced 2D motion and the reference video by comparing point trajectories across frames, and reflects the combined effects of camera and object motion. When evaluated over longer trajectories, it becomes sensitive to accumulated drift, serving as an indicator of long-term motion stability.

Visual quality and semantic alignment. We report Fréchet Inception Distance (FID) [9] and Fréchet Video Distance (FVD) [28] to evaluate perceptual realism and temporal coherence. To assess semantic alignment with text prompts, we additionally report the Qwen VQA Score [31], which measures the consistency between generated videos and input prompts using a vision-language model.

4.3 Evaluation Datasets

We evaluate our method on a held-out test set consisting of 1k video clips drawn from DynPose-100K, the MultiCamVideo Dataset, BED-LAM2.0, and our proprietary internal data. Together, these datasets cover a broad spectrum of foreground occupancy ratios, camera motion complexity, and data sources, including real-world daily-life recordings, cinematic videos, and synthesized scene videos. All evaluation videos are disjoint from the training set. Videos are processed using the same metric-scale annotation pipeline described in Sec. 3, producing camera parameters, foreground masks, and 3D bounding box poses. This unified annotation enables consistent and fair comparison across different methods under identical evaluation conditions. Unless otherwise specified, all quantitative results reported later are computed on this evaluation set.

4.4 Camera Pose Estimation Accuracy

Subsequent camera-related evaluations rely on metric-scale camera poses extracted from generated videos. To ensure that reported errors reflect generation quality rather than estimation inaccuracies, we first assess the accuracy of our camera pose extraction pipeline. We compare our method with representative approaches, including VGGT [30], Pi3 [35], and MapAnything [13], using camera translation and rotation errors. As shown in Table 1, our method achieves improved metric-scale accuracy, validating the effectiveness of the proposed pipeline for reliable supervision. We further provide qualitative validation by visualizing the extracted camera motion condition alongside the input videos. As illustrated in Fig. 4, the recovered motion faithfully reflects scene structure and camera dynamics, with consistent changes in position and orientation, making it directly interpretable. An ablation study comparing training with camera parameters from MapAnything and our extraction pipeline is provided in Appendix Table 5.

4.5 Comparison with State-of-the-Art

Having established the accuracy of our camera pose extraction pipeline, we evaluate the proposed framework GTL 14B against representative state-of-the-art methods, including Perception-as-Control (PAC) [2], RealCam-I2V [15], and GEN3C [23]. Quantitative comparisons across camera motion accuracy, object-level stability, trajectory consistency, and visual quality are reported in Table 2. The inference hyperparameters for GTL 14B are provided in Appendix Sec. C.

As summarized in Table 2, GTL 14B achieves lower camera translation and rotation errors, indicating more accurate camera motion recovery under metric scale. With explicit 3D bounding box conditioning, the model better regulates object translation and rotation, reducing object-centric errors. Joint camera-object control further improves motion stability and trajectory consistency, suggesting that explicit geometry-aware conditions provide stronger motion supervision than implicit image-space guidance alone. Perceptual and semantic metrics further show that these geometric improvements do not come at the cost of visual quality.

Recent works such as CamCloneMaster [18] require a reference video to specify camera motion. In our setting, this would require using the ground-truth video as input, which introduces an unfair advantage. We therefore exclude it from quantitative evaluation and report qualitative results in Fig. 4. We also include results from Seedance 2.0 [25]. Due to its less explicit separation between reference and target generation, it often transfers appearance details

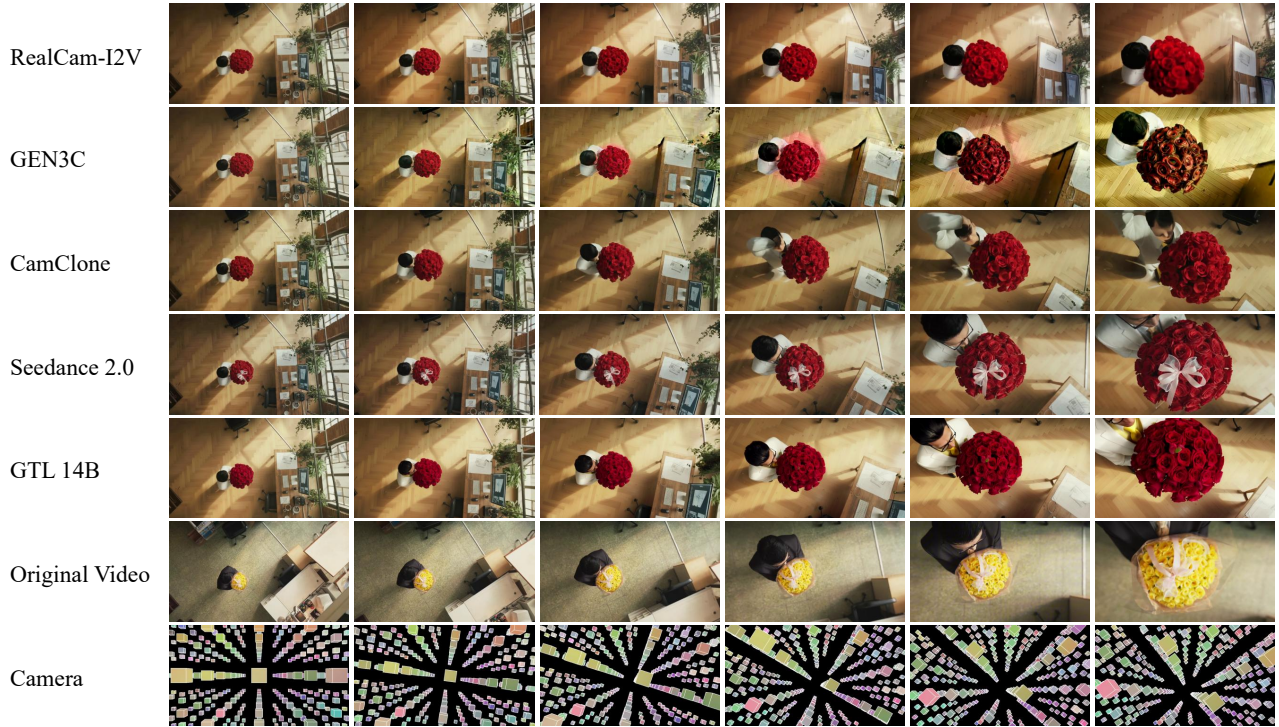


Figure 4: Qualitative comparison of camera controllability with state-of-the-art methods. Gen3C and RealCam-I2V use identical camera motion extracted from the original video, with the first frame as input. CamCloneMaster and Seedance 2.0 use the original video as the camera reference. Our Camera condition encodes recovered extrinsics and intrinsics. For clarity, text prompts are omitted in this figure and provided in the Appendix Sec. E.

from the reference video (e.g., the knot on the flower), leading to inconsistencies with the intended control signal.

The qualitative results in the following figures corroborate these findings. Fig. 4 shows that, when using only camera motion conditions, our method produces stable and geometrically consistent camera behavior under challenging camera trajectories. Fig. 5 further demonstrates that jointly applying camera and object motion conditions yields consistent camera motion and well-controlled object dynamics. Despite being trained primarily on human-centered motion supervision, the learned object motion condition generalizes effectively to non-human objects and complex scenes. Additional qualitative results are shown in Fig. 9.

To further validate disentanglement between camera and object motion, we visualize a controlled scenario in Fig. 6. Keeping the object fixed in screen space under camera motion requires the object to move together with the camera in 3D. This demonstrates that our model disentangles apparent 2D motion from physical 3D object motion, enabling independent control of camera behavior and object dynamics.

4.6 Ablation Study

To isolate the effect of individual design choices, we conduct ablation studies using a smaller Wan2.1-1.3B model trained on the same dataset for 30K steps under identical settings. Although the 1.3B

model exhibits slightly lower visual fidelity than the 14B model, it provides a reliable and efficient proxy for comparing relative performance across different conditioning designs. The ablation studies focus on relative metric differences rather than absolute generation quality, which are largely unaffected by the backbone scale. Additional ablation studies are provided in Appendix Sec. D. **Effect of Camera Condition Representation.** We study the impact of different camera condition representations on camera controllability. As an alternative representation, we consider a Plücker ray map [21], where camera motion is encoded using dense per-pixel rays parameterized by direction and moment. All variants use identical camera extrinsics and intrinsics as input, differing only in the representation of the camera condition. We evaluate controllability using camera translation and rotation errors, as reported in Table 3. The results show that our geometry-aware cubic grid representation provides more stable and accurate camera control than ray-based representations.

Effect of 3D Bounding Box Conditioning. We further analyze the role of explicit 3D bounding box conditioning in object motion control, as shown in Table 4. Besides our proposed rotation-aware 3D bounding box representation, we consider a simplified baseline using only binary foreground masks as object-level conditions. The mask-based representation provides only 2D spatial extent and lacks explicit depth ordering and rotational cues. Both variants are

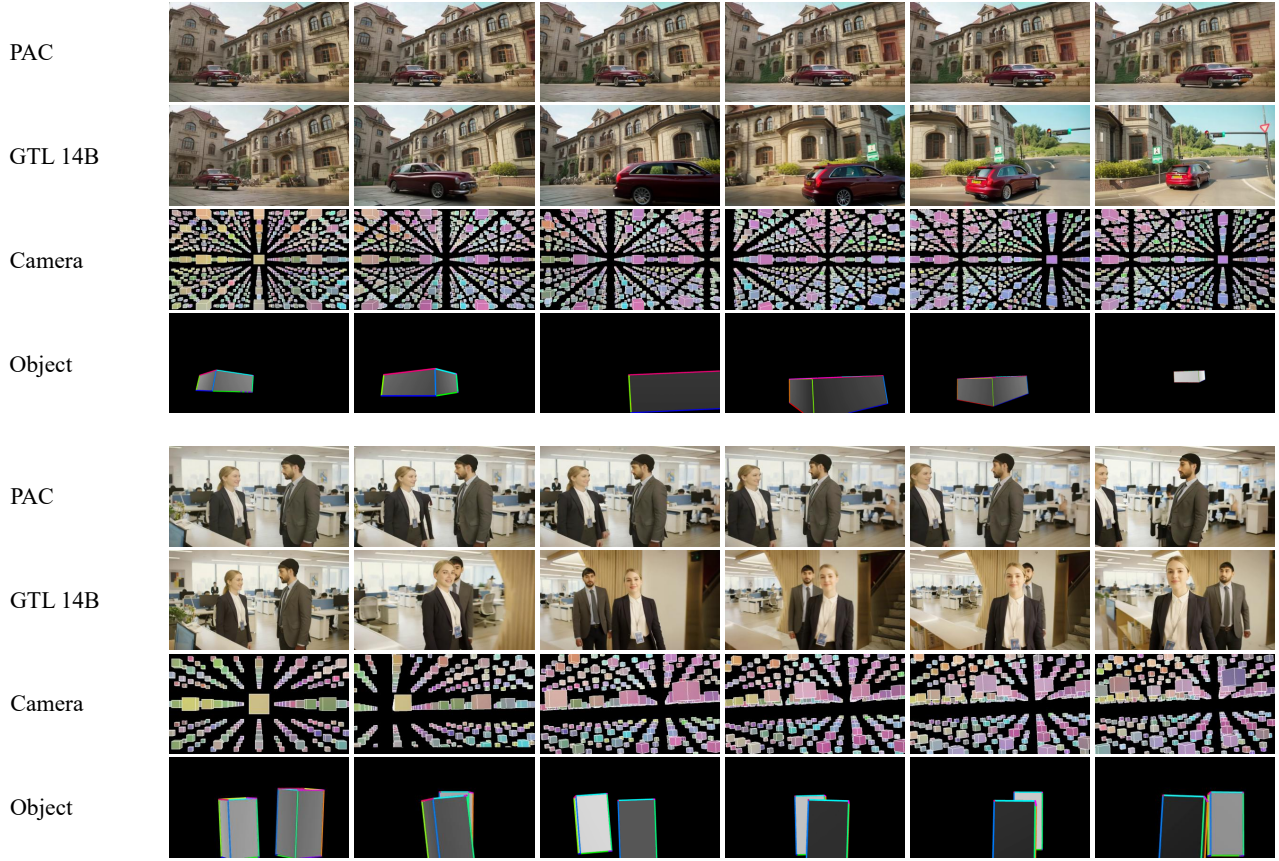


Figure 5: Qualitative comparison of object motion control. We compare our method with Perception-as-Control under aligned camera conditions and comparable object motion controls, focusing on the controllability and stability of foreground object motion. For clarity, text prompts are omitted in this figure and provided in the Appendix Sec. E.

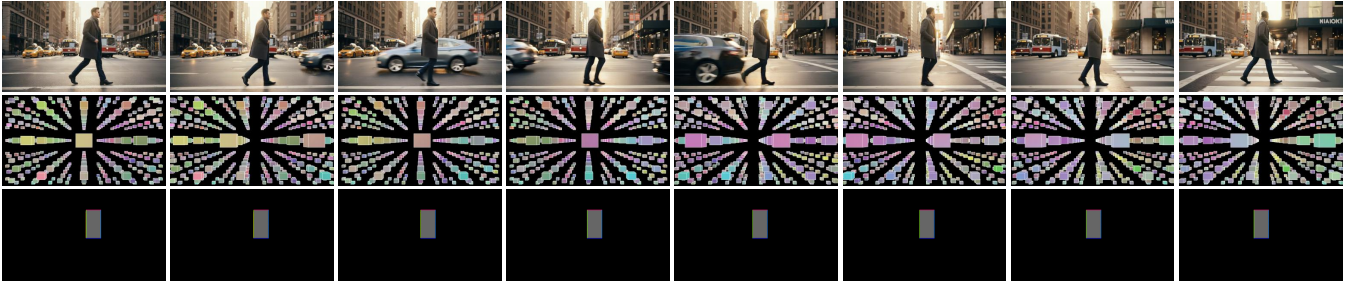


Figure 6: Disentanglement between camera and object motion. Keeping the object fixed in screen space under camera motion requires the object to move together with the camera in 3D. The prompts used here avoid specifying object motion and are provided in Appendix Sec. E.

trained with identical camera motion and depth-of-field conditions. We evaluate object controllability using bounding box center error, bounding box rotation error, and trajectory consistency measured by CoTracker. Results demonstrate that explicit 3D bounding box conditioning significantly improves rotational stability and long-term motion consistency.

5 Applications

Our geometry-aware conditioning enables expressive cinematic effects by explicitly disentangling camera extrinsics, intrinsics, and depth-of-field. We present two representative applications—controllable bokeh and the Hitchcock zoom—to illustrate its practical advantages.

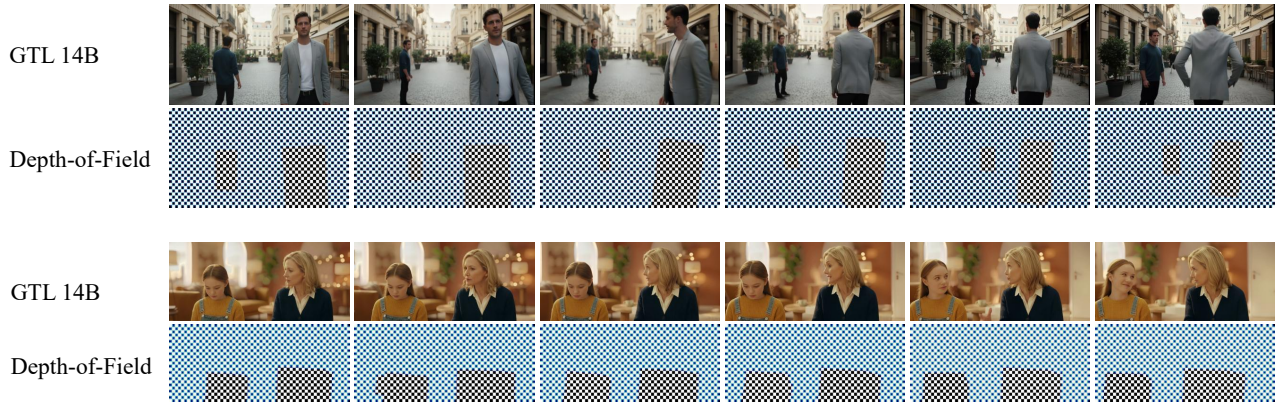


Figure 7: Visualization of depth-of-field conditions and generated results. Our representation enables controllable bokeh strength, time-varying focus shifts, and stable depth-of-field under camera and object motion. Note that focus cues are encoded through subtle pixel-level variations, which may not be visually salient. The corresponding text prompts are provided in Appendix Sec. E.

Camera Condition	Cam Trans. Err. ↓	Cam Rot. Err. (°) ↓
Ray Map (No Grid)	19.26	8.98
Color Cubic Grid (GTL 1.3B)	10.15	5.07

Table 3: Ablation studies on geometry-aware conditioning. Comparison of camera condition representations.

Object Condition	BBox Center Err. (%) ↓	BBox Rot. Err. (°) ↓	Traj. Err. (%) ↓
2D Mask	5.94	45.15	8.37
3D BBox (GTL 1.3B)	6.13	27.96	7.03

Table 4: Ablation studies on geometry-aware conditioning. Comparison of object motion conditioning.

5.1 Controllable Depth-of-Field and Bokeh

Our framework enables explicit and physically meaningful control over depth-of-field effects in generated videos. By leveraging metric depth information recovered in the metric camera space, we can synthesize a wide range of bokeh behaviors with a time-varying focus depth. In particular, the focal plane can be dynamically adjusted over time, allowing smooth focus transitions as well as focus-follow effects that track a moving subject.

Figure 7 shows that bokeh in our model is driven by the explicit depth-of-field condition, where both the relative depth to the focal plane and the bokeh strength are integrated into the conditioning video to drive the resulting blur. Because the focus depth is specified as a continuous, time-varying signal, our method naturally supports zoom-dependent and temporally varying focus behaviors commonly observed in real cinematography, as illustrated by the second example.

5.2 Hitchcock Zoom

The Hitchcock zoom [10] is a classic cinematic effect in which the camera translates in space while adjusting focal length in the opposite direction, preserving the apparent size of the foreground subject while dramatically altering background perspective. This

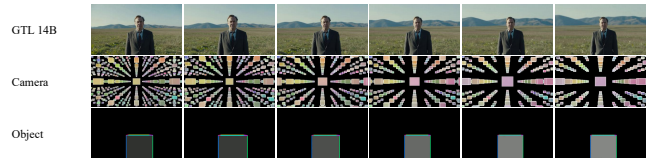


Figure 8: Hitchcock zoom effect synthesized by our model. In this example, the camera moves backward by 3 meters while the zoom is adjusted accordingly to approximately preserve the projected size of the subject in the image, as indicated by the object motion condition video. The corresponding text prompts are provided in Appendix Sec. E.

coupling of camera translation and zoom poses a significant challenge for generative models, as it requires accurate disentanglement and coordinated control of camera extrinsics and intrinsics.

As illustrated in Fig. 8, our framework explicitly encodes camera translation and focal length variation in the camera condition. Consequently, even when forward camera motion and inverse zoom produce visually similar image content, they correspond to distinct patterns in our camera condition, enabling stable and faithful synthesis of the Hitchcock zoom.

6 Conclusion

In this paper, we present Gen Through the Lens, a camera-controllable image-to-video generation framework that enables explicit and physically grounded control over camera motion, object motion, and depth-of-field effects from a single reference image. By decomposing camera-related signals into camera, world, and object coordinate spaces and unifying them through cross-coordinate conditioning fusion, our approach preserves the physical meaning of each control signal while enabling coherent, disentangled, and cinematographically meaningful video generation. Extensive experiments demonstrate consistent improvements over prior methods

in camera accuracy, motion stability, and editability, without compromising visual quality. Overall, this work highlights the importance of explicit multi-coordinate camera modeling for controllable and physically meaningful video synthesis, and provides a flexible foundation for camera-aware generative tools in cinematic content creation and film production.

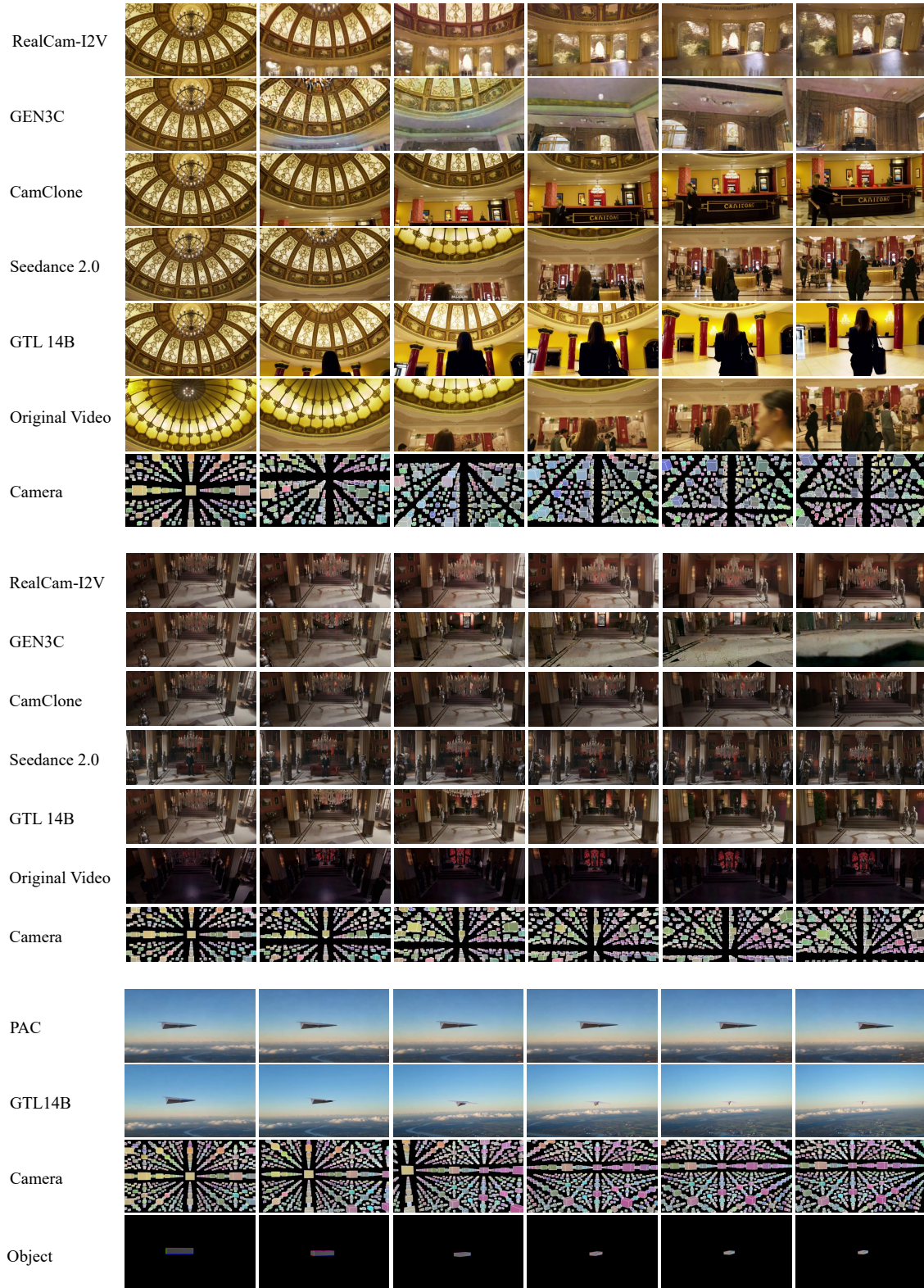


Figure 9: Additional qualitative results corresponding to the experiments. Text prompts are omitted in this figure and provided in the Appendix Sec. E.

References

- [1] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. 2025. ReCamMaster: Camera-Controlled Generative Rendering from a Single Video. *arXiv preprint arXiv:2503.11647* (2025).
- [2] Yingjie Chen, Yifang Men, Yuan Yao, Miaomiao Cui, and Liefeng Bo. 2025. Perception-as-Control: Fine-grained Controllable Image Animation with 3D-aware Motion Representation. *arXiv preprint arXiv:2501.05020* (2025).
- [3] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. 2024. Expanding Performance Boundaries of Open-Source Multimodal Models with Model, Data, and Test-Time Scaling. *arXiv preprint arXiv:2412.05271* (2024).
- [4] Sai Kumar Dwivedi, Yu Sun, Priyanka Patel, Yao Feng, and Michael J Black. 2024. TokenHMR: Advancing Human Mesh Recovery with a Tokenized Pose Representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 1323–1333.
- [5] Wanquan Feng, Jiawei Liu, Pengqi Tu, Tianhao Qi, Mingzhen Sun, Tianxiang Ma, Songtao Zhao, Siyu Zhou, and Qian He. 2024. I2VControl-Camera: Precise Video Camera Control with Adjustable Motion Strength. *arXiv preprint arXiv:2411.06525* (2024).
- [6] Daniel Geng, Charles Herrmann, Junhwa Hur, Forrester Cole, Serena Zhang, Tobias Pfaff, Tatiana Lopez-Guevara, Carl Doersch, Yusuf Aytar, Michael Rubinstein, Chen Sun, Oliver Wang, Andrew Owens, and Deqing Sun. 2024. Motion Prompting: Controlling Video Generation with Motion Trajectories. *arXiv preprint arXiv:2412.02700* (2024).
- [7] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. 2023. AnimateDiff: Animate Your Personalized Text-to-Image Diffusion Models without Specific Tuning. *arXiv preprint arXiv:2307.04725* (2023).
- [8] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. 2024. CameraCtrl: Enabling Camera Control for Text-to-Video Generation. *arXiv preprint arXiv:2404.02101* (2024).
- [9] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *Advances in neural information processing systems* 30 (2017).
- [10] Alfred Hitchcock. 1958. Vertigo. Paramount Pictures.
- [11] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. 2025. VACE: All-in-One Video Creation and Editing. *arXiv:2503.07598 [cs.CV]* <https://arxiv.org/abs/2503.07598>
- [12] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. 2024. CoTracker: It is Better to Track Together. In *European conference on computer vision*. Springer, 18–35.
- [13] Nikhil Keetha, Norman Müller, Johannes Schöninger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zaus, Ethan Weber, Nelson Antunes, et al. 2025. MapAnything: Universal Feed-Forward Metric 3D Reconstruction. *arXiv preprint arXiv:2509.13414* (2025).
- [14] Guojun Lei, Chi Wang, Hong Li, Rong Zhang, Yikai Wang, and Weiwei Wu. 2024. AnimateAnything: Consistent and Controllable Animation for Video Generation. *arXiv preprint arXiv:2411.10836* (2024).
- [15] Teng Li, Guangcong Zheng, Rui Jiang, Shuigen Zhan, Tao Wu, Yehao Lu, Yining Lin, Chuanyun Deng, Yapan Xiong, Min Chen, et al. 2025. RealCam-I2V: Real-World Image-to-Video Generation with Interactive Complex Camera Control. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 28785–28796.
- [16] Yaowei Li, Xintao Wang, Zhaoyang Zhang, Zhouxia Wang, Ziyang Yuan, Liangbin Xie, Ying Shan, and Yuexian Zou. 2025. Image Conductor: Precision Control for Interactive Video Synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 5031–5038.
- [17] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. 2023. SMPL: A Skinned Multi-Person Linear Model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 851–866.
- [18] Yawen Luo, Xiaoyu Shi, Jianhong Bai, Menghan Xia, Tianfan Xue, Xintao Wang, Pengfei Wan, Di Zhang, and Kun Gai. 2025. CamCloneMaster: Enabling Reference-based Camera Control for Video Generation. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*. 1–10.
- [19] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *ACM Trans. Graph.* 41, 4, Article 102 (July 2022), 15 pages. doi:10.1145/3528223.3530127
- [20] Muyao Niu, Xiaodong Cun, Xintao Wang, Yong Zhang, Ying Shan, and Yinqiang Zheng. 2024. MOFA-Video: Controllable Image Animation via Generative Motion Field Adaptions in Frozen Image-to-Video Diffusion Model. In *European Conference on Computer Vision*. Springer, 111–128.
- [21] Julius Plücker. 1865. XVII. On a New Geometry of Space. *Philosophical Transactions of the Royal Society of London* 155 (1865), 725–791.
- [22] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. 2024. SAM 2: Segment Anything in Images and Videos. *arXiv preprint arXiv:2408.00714* (2024).
- [23] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. 2025. GEN3C: 3D-Informed World-Consistent Video Generation with Precise Camera Control. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 6121–6132.
- [24] Chris Rockwell, Joseph Tung, Tsung-Yi Lin, Ming-Yu Liu, David F Fouhey, and Chen-Hsuan Lin. 2025. Dynamic Camera Poses and Where to Find Them. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 12444–12455.
- [25] Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, et al. 2026. Seedance 2.0: Advancing Video Generation for World Complexity. *arXiv preprint arXiv:2604.14148* (2026).
- [26] Xiaoyu Shi, Zhaoyang Huang, Fu-Yun Wang, Weikang Bian, Dasong Li, Yi Zhang, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, et al. 2024. Motion-I2V: Consistent and Controllable Image-to-Video Generation with Explicit Motion Modeling. In *ACM SIGGRAPH 2024 Conference Papers*. 1–11.
- [27] Joachim Tesch, Giorgio Becherini, Prerana Achar, Anastasios Yiannakidis, Muhammed Kocabas, Priyanka Patel, and Michael J Black. 2025. BEDLAM2.0: Synthetic Humans and Cameras in Motion. *arXiv preprint arXiv:2511.14394* (2025).
- [28] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. 2019. FVD: A New Metric for Video Generation. In *ICLR Workshop on Deep Generative Models for Highly Structured Data*.
- [29] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwei Yu, Haiming Zhao, Jianxiao Yang, et al. 2025. Wan: Open and Advanced Large-Scale Video Generative Models. *arXiv preprint arXiv:2503.20314* (2025).
- [30] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. 2025. VGGT: Visual Geometry Grounded Transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 5294–5306.
- [31] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024. Qwen2-VL: Enhancing Vision-Language Model's Perception of the World at Any Resolution. *arXiv preprint arXiv:2409.12191* (2024).
- [32] Qinghe Wang, Yawen Luo, Xiaoyu Shi, Xu Jia, Huchuan Lu, Tianfan Xue, Xintao Wang, Pengfei Wan, Di Zhang, and Kun Gai. 2025. CineMaster: A 3D-Aware and Controllable Framework for Cinematic Text-to-Video Generation. In *ACM SIGGRAPH 2025 Conference Papers*. 1–10.
- [33] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jialong Yang. 2025. MoGe-2: Accurate Monocular Geometry with Metric Scale and Sharp Details. *arXiv preprint arXiv:2507.02546* (2025).
- [34] Xi Wang, Robin Courant, Marc Christie, and Vicky Kalogeiton. 2025. AKiRa: Augmentation Kit on Rays for optical video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2609–2619.
- [35] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. 2025. Pi3: Permutation-Equivariant Visual Geometry Learning. *arXiv preprint arXiv:2507.13347* (2025).
- [36] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. 2024. MotionCtrl: A Unified and Flexible Motion Controller for Video Generation. In *ACM SIGGRAPH 2024 Conference Papers*. 1–11.
- [37] Weijia Wu, Zhuang Li, Yuchao Gu, Rui Zhao, Yefei He, David Junhao Zhang, Mike Zheng Shou, Yan Li, Tingting Gao, and Di Zhang. 2024. DragAnything: Motion Control for Anything using Entity Representation. *arXiv:2403.07420 [cs.CV]*
- [38] Dejia Xu, Wei Nie, Chao Liu, Sifei Liu, Jan Kautz, Zhangyang Wang, and Arash Vahdat. 2024. CamCo: Camera-Controllable 3D-Consistent Image-to-Video Generation. *arXiv preprint arXiv:2406.02509* (2024).
- [39] Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. 2023. DragNUWA: Fine-grained Control in Video Generation by Integrating Text, Image, and Trajectory. *arXiv preprint arXiv:2308.08089* (2023).
- [40] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. 2024. ViewCrafter: Taming Video Diffusion Models for High-Fidelity Novel View Synthesis. *arXiv preprint arXiv:2409.02048* (2024).
- [41] Zhenghao Zhang, Junchao Liao, Menghao Li, Zuozhuo Dai, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. 2025. Tora: Trajectory-oriented Diffusion Transformer for Video Generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2063–2073.
- [42] Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jia-Wei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. 2024. MotionDirector: Motion Customization of Text-to-Video Diffusion Models. In *European Conference on Computer Vision*. Springer, 273–290.
- [43] Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. 2023. UniPC: A Unified Predictor-Corrector Framework for Fast Sampling of Diffusion Models.

arXiv:2302.04867 [cs.LG] <https://arxiv.org/abs/2302.04867>

- [44] Guangcong Zheng, Teng Li, Rui Jiang, Yehao Lu, Tao Wu, and Xi Li. 2024. CamI2V: Camera-Controlled Image-to-Video Diffusion Model. *arXiv preprint arXiv:2410.15957* (2024).
- [45] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyfe, and Noah Snavely. 2018. Stereo Magnification: Learning View Synthesis Using Multiplane Images. *arXiv preprint arXiv:1805.09817* (2018).

A Additional Details on Geometry-Aware Conditioning

This appendix provides implementation details for the label prompt, camera motion condition, object motion condition, depth-of-field condition, and zoom augmentation.

A.1 Label Prompt Details

To ensure disentanglement between camera control and content description, we explicitly constrain the captions to exclude any camera related information. We use the following prompt for dense video caption generation with InternVL2.5:

You are a professional video description master, your task is to carefully observe the input video, and give a detailed and dense description of the video, please follow the instructions below: 1. It is required to pay attention to the timing of the video, reflecting the development process of the events in the video over time and the details. 2. The output content should not include camera-related information, such as camera shooting angles and camera movements. 3. Require the output to capture as much video content as possible, including details, considering both completeness and specificity. It should provide comprehensive information that is tied to specific elements of the video. 4. The output description is required to be accurate and coherent, without misinformation and misunderstanding. The output must be factually accurate, coherent in content description, and consistent with the video content. 5. Do not give the info about camera such as camera shooting Angle and camera Movement. The Output should no more than 225 words.

A.2 Camera Modeling and Camera Motion Condition

Metric-Scale Camera Recovery. We combine Pi3 [35] and Moge2 [33] to recover metric-scale camera motion. Pi3 provides temporally stable camera poses and scene structure up to an unknown global scale, while Moge2 predicts per-frame metric depth with limited temporal consistency.

To estimate the global scale, we randomly sample point pairs $(\mathbf{x}_a, \mathbf{x}_b)$ from reliable image regions, back-project them into 3D using Pi3 and Moge2, and compute distance ratios $d_k^{\text{Moge}}/d_k^{\text{Pi3}}$. We discard the top and bottom 20% of ratios both within each frame and globally across the video, and average the remaining values to obtain a single video-level scale factor s .

After scaling Pi3 translations and associated 3D structure, we apply a Gaussian temporal filter with a window size of 0.5 seconds independently to translation and rotation sequences.

For camera intrinsics, we use MapAnything [13] as an initial estimator. Its intrinsic scale is aligned to the recovered metric camera space using the same pairwise-distance scaling strategy, and further adjusted to account for rendering resolution.

Camera Motion Condition. We discretize the surrounding 3D space using a regular grid of axis-aligned cubes. Cubes are placed at uniform intervals of 0.8 meters along the x , y , and z axes in world coordinates, and each cube has an edge length of 0.15 meters. Only cubes within a distance range of $[1.5, 10]$ meters from the camera center are considered visible. When more than 300 cubes are visible, we render only the 300 closest cubes.

All camera poses are normalized to the first frame. The visible cubes are rendered using the recovered camera intrinsics and extrinsics.

To encode spatial position, each cube is assigned a color from a periodic spatial color field with a period of 10 meters along each axis. Specifically, we randomly sample $N = 269$ anchor points $\{(\mathbf{p}_i, \mathbf{c}_i)\}_{i=1}^N$ in a unit cube $[0, 1]^3$, where $\mathbf{p}_i \in \mathbb{R}^3$ satisfy $\|\mathbf{p}_i - \mathbf{p}_j\|_2 > 0.12$ for all $i \neq j$ under periodic boundary conditions, and $\mathbf{c}_i \in [0, 255]^3$ are RGB colors. The anchor set is tiled periodically in $[-1, 2]$ along each axis.

For a 3D query point $\mathbf{x}$, we find its $k = 10$ nearest anchors with distances $\{d_i\}_{i=1}^k$ and compute the interpolated color as

$$w_i = \frac{1}{d_i + \varepsilon} - \frac{1}{d_{\max} + \varepsilon}, \quad \tilde{w}_i = \frac{w_i}{\sum_{j=1}^k w_j}, \quad (1)$$

$$\mathbf{c}(\mathbf{x}) = \sum_{i=1}^k \tilde{w}_i \mathbf{c}_i. \quad (2)$$

We approximate this dense color field using a multi-grid hash encoding [19] with a lightweight MLP, which is then used to paint all cubes efficiently.

A.3 3D Bounding Box Modeling and Object Motion Condition

We use TokenHMR [4] to recover identity-consistent human meshes across frames. For each tracked individual, we extract torso vertices from the recovered SMPL mesh [17] and fit a tight 3D bounding box.

The object motion condition is rendered as a projected 3D bounding box on a black background. Pixels inside the projected box are assigned grayscale values in the range $[25, 255]$ using per-video relative depth normalization, while background pixels remain at 0.

During training, if no object motion is present, we use an all-black video as the object motion condition.

During inference, we use the same per-video depth normalization rule as in training. The near bound is set to the minimum of 1 meter and the closest distance from the camera to any point on the object bounding box. The far bound is set to the maximum of 10 meters and the farthest distance from the camera to the bounding box. Depth values inside the box are normalized to this range and mapped to grayscale intensities.

To encode orientation, the 12 edges of the 3D bounding box are rendered using distinct colors sampled uniformly from the hue channel of HSV space. The color assignment is fixed across frames and the dataset.

A.4 Depth-of-Field Modeling and Depth-of-Field Condition

We use the same physically motivated bokeh operator for both bokeh video construction and depth-of-field condition generation. Given a per-frame focus depth, the operator computes blur radius from the depth deviation to the focal plane. To stabilize blur near large depth discontinuities, we identify regions with abrupt depth changes and apply spatially adaptive smoothing to the blur-radius field.

The depth-of-field condition adopts a three-channel layered representation based on a synthetic checkerboard image. All channels share the same initial black-and-white checkerboard pattern, where the image width is divided into 30 vertical segments and the height is adjusted to maintain approximately square cells.

Each channel corresponds to a canonical background depth in world coordinates, fixed at 5 meters, 15 meters, and 25 meters. For each frame, foreground pixels inside the projected 3D bounding box are assigned the object’s metric depth, while background pixels retain the channel-specific depth.

The bokeh operator is applied independently to each depth map, producing three grayscale videos that are stacked into RGB channels to form the final depth-of-field condition video. Because blur depends jointly on depth and bokeh strength, the resulting condition exhibits color variation across channels.

During training, if no bokeh effect is present, we use the original unblurred checkerboard as the condition. During inference, the same procedure can be applied with manually specified bounding boxes to enable user-controlled depth-of-field effects.

A.5 Zoom Augmentation

We simulate zoom using temporal center cropping followed by resizing to the original resolution. Crop factors and zoom intervals are sampled randomly. The same zoom transformation is applied consistently to the camera motion condition video, object motion condition video, and depth-of-field condition video to preserve pixel-wise alignment.

B Evaluation Metrics

This appendix provides detailed definitions of all evaluation metrics used in the main paper.

B.1 Camera Translation Error

To quantify camera path accuracy in a scale-invariant manner, we compute the average relative translation error between predicted and ground-truth camera positions. All camera trajectories are first normalized with respect to the camera pose at frame $t = 0$, such that translations $\mathbf{t}$ represent relative displacements from the initial camera position, and rotations $\mathbf{R}$ are defined relative to the initial orientation.

The translation error is defined as:

$$\mathcal{E}_{\text{trans}} = \frac{1}{T} \sum_{t=1}^T \frac{\|\mathbf{t}_t^{\text{pred}} - \mathbf{t}_t^{\text{gt}}\|_2}{\max(\max_{1 \leq \tau \leq T} \|\mathbf{t}_\tau^{\text{gt}}\|_2, 1)} * 100, \quad (3)$$

The error metric is summarized as a single scalar value.

B.2 Camera Rotation Error

Camera orientation accuracy is evaluated using the geodesic distance between predicted and ground-truth rotations:

$$\mathcal{E}_{\text{rot}} = \frac{1}{T} \sum_{t=1}^T \arccos \left(\frac{\text{trace}(\mathbf{R}_t^{\text{pred}} \mathbf{R}_t^{\text{gt}\top}) - 1}{2} \right), \quad (4)$$

where $\mathbf{R}_t^{\text{pred}}$ and $\mathbf{R}_t^{\text{gt}}$ denote rotation matrices at frame t . The error is reported in degrees.

B.3 Mask Intersection-over-Union (mIoU)

Foreground spatial consistency is evaluated using mean Intersection-over-Union (mIoU):

$$\text{mIoU} = \frac{1}{T} \sum_{t=1}^T \frac{|M_t^{\text{pred}} \cap M_t^{\text{gt}}|}{|M_t^{\text{pred}} \cup M_t^{\text{gt}}|}, \quad (5)$$

where M_t^{pred} and M_t^{gt} denote predicted and reference foreground masks at frame t , obtained using SAM2.

B.4 3D Bounding Box Pose Error

We evaluate object-level geometric stability using both translation and rotation errors of the 3D bounding box. We employ a greedy algorithm to match M trajectory pairs ($M = 2$ in our setting), computing the error exclusively on frames $t \in \mathcal{T}_{\text{comm}}$ where M bounding boxes are all detected.

The screen-space center deviation is defined as:

$$\mathcal{E}_{\text{bbox-trans}} = \frac{1}{M \cdot |\mathcal{T}_{\text{comm}}|} \sum_{m=1}^M \sum_{t \in \mathcal{T}_{\text{comm}}} \frac{\|\mathbf{c}_{t,m}^{\text{pred}} - \mathbf{c}_{t,m}^{\text{gt}}\|_2}{\sqrt{H^2 + W^2}}, \quad (6)$$

where $\mathbf{c}_t$ is the projected 2D center of the bounding box, and H, W are image height and width.

Similarly, bounding box rotation error is measured using geodesic distance:

$$\mathcal{E}_{\text{bbox-rot}} = \frac{1}{M \cdot |\mathcal{T}_{\text{comm}}|} \sum_{m=1}^M \sum_{t \in \mathcal{T}_{\text{comm}}} \arccos \left(\frac{\text{trace}(\mathbf{R}_{\text{bbox},t,m}^{\text{pred}} \mathbf{R}_{\text{bbox},t,m}^{\text{gt}\top}) - 1}{2} \right). \quad (7)$$

B.5 Trajectory Consistency Error

Long-term motion fidelity is evaluated using trajectory consistency error computed from point tracks extracted by CoTracker:

$$\mathcal{E}_{\text{traj}} = \frac{1}{NT} \sum_{i=1}^N \sum_{t=1}^T \frac{\|\mathbf{p}_{i,t}^{\text{pred}} - \mathbf{p}_{i,t}^{\text{gt}}\|_2}{\sqrt{H^2 + W^2}}, \quad (8)$$

where $\mathbf{p}_{i,t}$ denotes the 2D position of point i at frame t .

B.6 Visual Quality and Semantic Alignment

We evaluate perceptual quality using Fréchet Inception Distance (FID) and Fréchet Video Distance (FVD). Semantic alignment between generated videos and input prompts is evaluated using the Qwen VQA Score, computed by querying a vision-language model on sampled frames and averaging the confidence scores.

C Inference hyperparameters

During inference, we used the unipc [43] sampling method, with the number of steps set to 20. The classifier-free guidance scale is set to 6.

Camera Extraction Method	Cam Trans. Err. ↓	Cam Rot. Err. (°) ↓
MapAnything	13.53	6.84
GTL	10.15	5.07

Table 5: Ablation study on camera extraction pipeline. We compare camera parameters directly estimated by MapAnything with those produced by our pipeline under the same training setting. The results show that our camera extraction pipeline provides more reliable supervision, leading to improved camera translation and rotation accuracy.

D Additional Ablation Study

To evaluate the effectiveness of our camera extraction pipeline, we replace our metric-scale camera parameters with those directly estimated by MapAnything. We then train a GTL-1.3B model using these parameters as-is, while keeping all other components and training settings identical to the main model.

We observe that MapAnything produces camera parameters with inconsistent intrinsic across frames, leading to noticeable instability in the resulting camera motion condition. This inconsistency introduces additional difficulty during training and degrades controllability.

As shown in Table 5, the model trained with our camera pipeline achieves significantly better performance in camera-related metrics compared to the MapAnything baseline, demonstrating the superiority of our camera extraction pipeline for controllable camera modeling.

E Text Prompts for Experiments

We provide the text prompts corresponding to all qualitative results for reproducibility. For fairness, all methods share the same prompts, except for camera-related descriptions, which are ignored by our model but used by baseline methods when applicable. Camera-related descriptions are marked with [Camera].

For Seedance 2.0, we use a different prompt format that incorporates both the input image and reference camera motion. Specifically, the prompt follows the template: “Use the input image as the first frame, follow only the camera motion from the reference video, and generate a new video described as: <PROMPT>.”, where <PROMPT> corresponds to the descriptions listed below.

Camera Experiments (Fig. 4).

- During the day, sunlight streamed in through the windows. Indoors, a man in a white suit, yellow tie, and glasses, with his hair meticulously styled, held a large bouquet of red roses with a serious expression. [Camera: The camera began with an overhead shot from above the ceiling, then rotated downwards, finally focusing on the pink ribbon decorating the rose bouquet.]

Bounding Box Experiments (Fig. 5).

- During the daytime, in a European-style residential complex, a red sedan slowly drives away from the complex’s gate, traveling along a gray road. The complex is surrounded by two-story European-style residential buildings. A row of bicycles is parked to the left of the building entrance, and a

motorcycle is parked to the right. [Camera: The camera follows the car as it moves to the right, and when the car reaches the complex gate, the camera stops moving and begins to pan to the right until the car drives out of the complex gate.]

- The video begins with a medium shot of a bright, modern office lobby. In the foreground is a professional woman with her hair tied back. Standing slightly behind her is a man wearing a black beanie. Initially, the woman has a warm and professional smile, while the man’s expression is more reserved. The woman walks forward, always facing the camera, with the man following closely behind. The woman guides the man through the office space. As they move forward, the camera smoothly moves backward, keeping them centered in the frame. More environmental details gradually become visible, including a marble elevator and staircase in the background.

Disentanglement Experiments (Fig. 6).

- A man on a city street, surrounded by buildings.

Depth-of-Field Experiments (Fig. 7).

- This video clip documents an encounter between two men on a European street. The scene opens with a medium shot. In the foreground, on the right, a man is looking at the camera. In the background, another man is walking away down the street. The man in the foreground turns around and begins to walk forward. As he walks forward, the man he was following stops, turns around to face him, and stands there expressionlessly.
- A woman is talking to a girl.

Hitchcock Experiments (Fig. 8).

- A man standing on a green grassy field, with majestic mountains in the background under a dim, overcast sky.

Additional Experiments (Fig. 9).

- A woman with long brown hair, wearing a black suit and carrying a large black shoulder bag, walks slowly towards the hotel reception desk with her back to the camera. The lobby is predominantly yellow, with crystal chandeliers hanging from the ceiling, and red load-bearing columns with gold accents symmetrically positioned on either side of the reception desk. [Camera: Starting from the rooftop, the camera moves downward into the interior, then smoothly pushes forward through the hall]
- Two rows of armored warriors stand motionless on either side. In the center of the main reception area sits a man in a suit, exuding an imposing presence. [Camera: The camera starts slightly facing the ground, then levels to a forward view and continues a steady forward push]
- A paper airplane floated in the air, slowly drifting away into the distance. [Camera: The camera remains stationary and rotates to the right]