TriMotion: Modality-Agnostic Camera Control for Video Generation

Seunghyun Shin[1,2*], Jifei Song[2], Wooseok Jeon[3], Hae-Gon Jeon[3†], and Jiankang Deng[4†]

[1] GIST AI Graduate School, South Korea
[2] Huawei Darwin Research Center, United Kingdom
[3] Department of Artificial Intelligence, Yonsei University, South Korea
[4] Imperial College London, United Kingdom
Project page: `https://seunghyuns98.github.io/TriMotion`

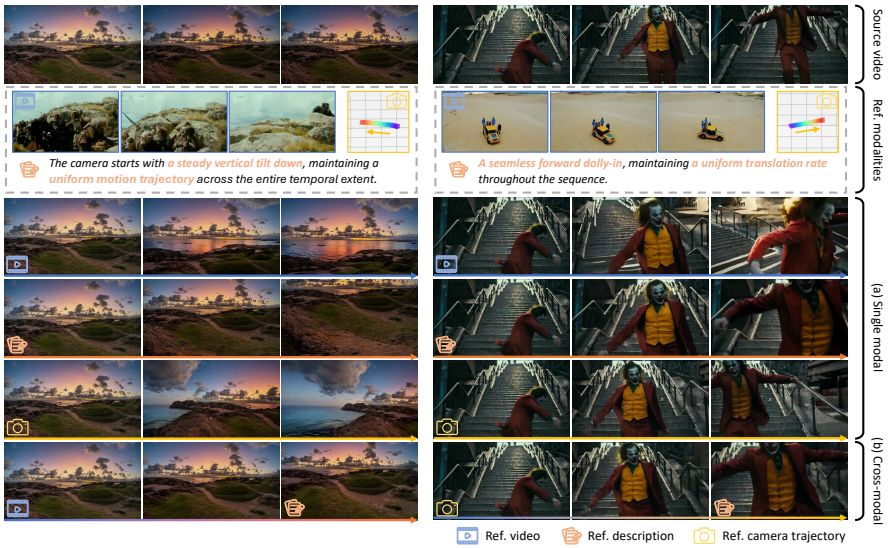

Fig. 1: Camera-controlled video generation results of TriMotion. (a) TriMotion enables precise camera control from three reference modalities. (b) It also supports cross-modal camera control, allowing camera motion to be transferred across modalities.

Abstract. Camera motion control is essential for directing viewpoint changes in generative systems. However, existing methods typically condition the generation process on a single specific modality, such as explicit pose trajectories or reference videos, limiting their ability to support heterogeneous user inputs. To address this limitation, we present *TriMotion*, a modality-agnostic framework for camera-controlled video generation that maps video, pose, and text inputs, describing the same camera trajectory into a shared motion embedding space. Learning such a space requires synchronized supervision across modalities. Therefore, we build

* Work done during an internship at Huawei Darwin Research Center.
† Corresponding authors.

the *Motion Triplet Dataset* by extending a Multi-Cam Video Dataset with geometry-grounded motion descriptions derived from camera extrinsics. We further introduce a latent motion consistency objective that leverages the motion embedding space to encourage the generated video to follow the target camera trajectory directly in latent space, avoiding the cost of pixel-space decoding. Extensive experiments show that TriMotion generates high-quality videos that accurately follow the target camera trajectories across all three modalities. Beyond standard generation, the shared motion embedding space also enables flexible applications such as sequential motion composition and cross-modal motion interpolation.

Keywords: Camera Control · Video Diffusion · Multi Modal Alignment

1 Introduction

Camera movement has long been regarded as a fundamental technique for organizing cinematic space and directing viewer attention [1]. By modulating viewpoint over time, filmmakers shape atmosphere and emotional emphasis within a scene. For example, in the film *Vertigo*, the dolly zoom distorts spatial perception to externalize the protagonist's psychological instability, a technique frequently analyzed in film studies for its affective and perceptual impact [2]. Beyond traditional cinema, controlling camera motion has become critical in modern generative systems, where viewpoint dynamics strongly influence realism, narrative coherence, and user intent realization.

Driven by the rapid advancement of conditional video diffusion models [3–10], recent approaches have actively explored various conditioning signals to direct camera movement. However, existing camera-control methods are often restricted to a single specific input reference modality. Pose-conditioned methods [11–23] require explicit camera trajectories in the geometric manner which is not intuitive for general users and difficult to specify accurately. Reference-video-based approaches [24,25] implicitly encode motion patterns, yet they lack explicit trajectory control and offer limited flexibility. Text descriptions, while intuitive, remain underexplored for camera control due to their lack of explicit temporal structures and geometric constraints, which makes it difficult to ensure precise and physically consistent trajectory execution. Although each modality offers advantages in terms of accessibility and geometric precision, existing methods typically restrict users to a single modality, limiting the flexibility of generative systems. This motivates a shared motion representation that bridges heterogeneous inputs and enables consistent camera motion control across modalities.

To this end, we propose TriMotion, a unified framework for camera-controlled video generation. Trimotion maps diverse control signals including video, pose, and text inputs, into a single continuous motion representation. By aligning equivalent camera trajectories across modalities, we preserve both temporal dynamics and physical consistency. This shared representation enables consistent camera controls in latent video diffusion models regardless of the input modality.

Since training such a cross-modal space inherently demands synchronized, multi-modal supervision, we curate the Motion Triplet Dataset. Leveraging

a Large Language Model (LLM)-based pipeline, we translate explicit camera extrinsics from the Multi-Cam Video Dataset [15] into geometry-grounded motion descriptions. This process pairs multi-view video data with geometry-grounded text, providing the essential foundation for aligning divergent modalities without sacrificing geometric fidelity.

Built on the Motion Triplet Dataset, TriMotion consists of two main components: First, we learn a unified motion representation using global contrastive alignment, temporal token matching, and pose regression, so that the representation captures both trajectory semantics and camera geometry. Second, we introduce a latent motion consistency objective implemented through a motion embedding predictor, which regularizes the diffusion backbone directly in latent space without repeated pixel-space decoding. A dedicated motion embedding predictor ensures geometric alignment entirely in the latent domain, thereby improving controllability while bypassing traditional decoding bottlenecks.

Empirical evaluations confirm that TriMotion not only establishes new state-of-the-art benchmarks for camera-controllable video generation but also pioneers novel interaction paradigms, successfully enabling sequential motion composition and cross-modal motion interpolation. (See Fig. 1).

2 Related Work

Camera-Controllable Video Generation Camera-controllable video generation aims to follow a user-specified camera trajectory while preserving temporal and geometric consistency. Existing methods can be broadly grouped into three families. First, pose-conditioned methods condition pretrained video generators [3, 4, 8, 9, 26] on explicit camera trajectories. For example, MotionCtrl [11] directly incorporates camera extrinsics into temporal modules, while methods such as CameraCtrl [13], VD3D [16], AC3D [17], and I2VControl-Camera [27] adopt advanced geometric parameterizations like Plücker rays or point trajectories to improve control. Second, geometry-aware image-to-video methods enhance cross-view consistency through 3D reasoning. CamCo [28] and CamI2V [14] incorporate epipolar-constrained attention, whereas RealCam-I2V [29] further leverages metric-depth-based scene reconstruction and an interactive 3D interface to design camera trajectories on real images. Finally, reference-based methods avoid explicit camera parameters by transferring motion from exemplar videos. MotionMaster [25] and MotionClone [30] extract motion cues from temporal attention maps to guide generation.

Camera-controlled Video-to-Video Generation Camera-controlled video-to-video generation aims to re-render an input clip under a new viewpoint while preserving scene identity, appearance, and dynamics. Existing methods can be categorized into two directions. The first direction uses conditioning-based control, where generative models are guided by target camera parameters or motion cues. GCD [20] introduces synthetic multi-view training, while ReCamMaster [15] improves robustness on real videos through stronger video conditioning and a large synchronized multi-camera dataset. DaS [19] uses 3D tracking videos as control signals, GS-DiT [31] builds pseudo-4D Gaussian fields from dense 3D point

tracking. Trajectory Attention [32] injects pixel trajectories through a dedicated attention branch. CamCloneMaster [24] transfers camera motion from a reference video through token concatenation, supporting both Image-to-Video (I2V) and Video-to-Video (V2V) without camera parameters. Another approach follows warping-based view transformation, where source content is first projected or warped into target viewpoints using estimated geometry and then refined through a generative model. ReCapture [21] first produces anchor videos and refines them with masked video fine-tuning, while TrajectoryCrafter [18], Vid-CamEdit [22], and Reangle-A-Video [23] use point clouds or estimated geometry to guide diffusion-based completion under novel views.

TriMotion's Approach Unlike task-specific or modality-restricted approaches, TriMotion learns a shared motion embedding space across video, pose, and text inputs. This unified approach is enabled by the *Motion Triplet Dataset*, which provides synchronized supervision across the three modalities, enabling flexible and consistent camera control across different input types.

3 Motion Triplet Dataset

The Motion Triplet Dataset is built upon the Multi-Cam Video Dataset [15] which contains 136K high-fidelity Unreal Engine 5 [33] videos spanning 13.6K dynamic scenes across 40 diverse 3D environments, with 122K unique camera trajectories and accurate ground-truth camera extrinsics. While this dataset provides synchronized videos and poses, it lacks textual descriptions of camera motion. Accordingly, we design an automated geometry-grounded captioning pipeline that converts raw camera trajectories into natural language.

Directly feeding raw camera pose sequences to a Large Language Model (LLM) is often suboptimal, since pose trajectories are continuous numerical signals with temporal dependencies, whereas LLMs are optimized for discrete language tokens. Recent works highlight modality gaps when applying LLMs to numerical time-series data and show that reliable temporal numerical reasoning remains challenging [34, 35]. To mitigate this, we first convert them into relative trajectories anchored at the first frame, and compute frame-wise translation and rotation changes given per-frame camera extrinsics. Small variations below fixed thresholds are treated as stationary, while significant motions are mapped to canonical camera operations such as Dolly, Pan, and Tilt, then organized into temporally ordered motion phases. This produces a compact symbolic sequence that preserves the essential structure of the trajectory.

We use these symbolic sequences to prompt Qwen3-4B-Instruct [36] to generate two levels of motion descriptions: a short summary of the overall trajectory and a detailed paragraph describing temporal transitions and speed changes. The model is instructed to distinguish simultaneous from sequential motion and to avoid explicit metric values, matching the qualitative style of realistic user prompts. The resulting Motion Triplet Dataset provides aligned supervision over videos, camera trajectories, and motion descriptions for learning a unified motion representation. We further validate the quality of the generated descriptions through a user study in Sec. 5.6.

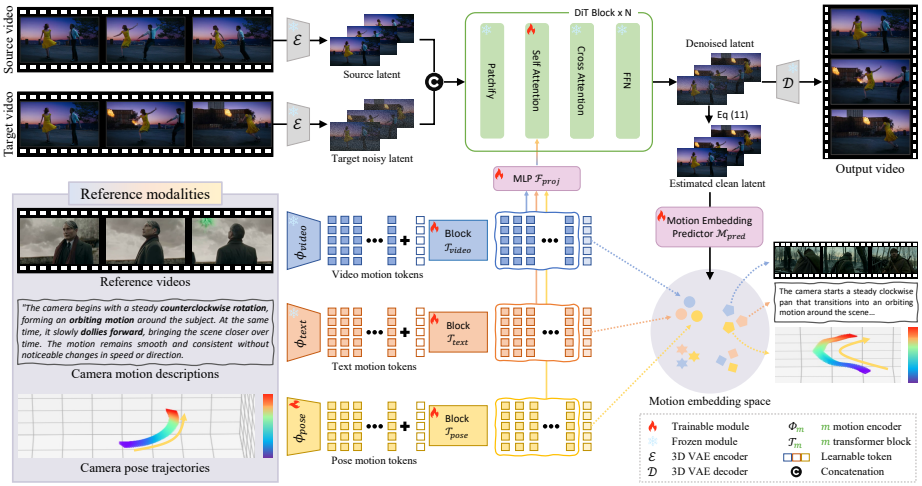

Fig. 2: Overview of TriMotion. TriMotion maps video, pose, and text motion inputs into a unified motion embedding space and uses the resulting embedding to condition the latent video diffusion backbone for camera-controlled video generation.

4 Method

In this section, we describe TriMotion, a framework for cross-modal camera motion control built on the latent video diffusion transformer [9]. We first introduce the diffusion backbone and motion conditioning setup (Sec. 4.1). We then describe two key components of the framework: a Unified Motion Embedding Space that aligns video, text, and pose inputs in a shared representation (Sec. 4.2), and a Latent Motion Consistency constraint that preserves the target camera trajectory during generation (Sec. 4.3). Finally, we present the overall training objective and strategy (Sec. 4.4). An overview is shown in Fig. 2.

4.1 Diffusion Backbone and Motion Conditioning

Given a source video $x_{\text{src}} \in \mathbb{R}^{F\times C\times H\times W}$ and a target video $x_{\text{tgt}} \in \mathbb{R}^{F\times C\times H\times W}$, a 3D Variational Autoencoder (VAE) encoder $\mathcal{E}$ compresses them into latent representations

$$\mathbf{z}_{\text{src}} = \mathcal{E}(x_{\text{src}}), \qquad \mathbf{z}_0 = \mathcal{E}(x_{\text{tgt}}), \tag{1}$$

where the temporal and spatial dimensions are downsampled by factors of 4 and 8, respectively. A decoder $\mathcal{D}$ reconstructs the video such that $x \approx \mathcal{D}(\mathbf{z})$.

Following rectified flow training [37–39], we sample $\epsilon \sim \mathcal{N}(0, I)$ and define a noisy target latent as:

$$\mathbf{z}_t = (1-t)\mathbf{z}_0 + t\epsilon, \quad t \in [0,1]. \tag{2}$$

The diffusion input is then formed by concatenating the clean source latent and the noisy target latent along the temporal dimension: $\tilde{\mathbf{z}}_t = [\mathbf{z}_{\text{src}}; \mathbf{z}_t]$.

In addition to the latent input, the denoiser is conditioned on a global appearance description y, the first source frame I, and a target motion embedding sequence $\mathbf{e}_m$ from the unified motion space described in Sec. 4.2. The description and first frame are encoded by a frozen T5 encoder [40] and a frozen CLIP image encoder [41], respectively. To incorporate target motion, we inject $\mathbf{e}_m$ into each transformer block through a block-specific projection MLP $\mathcal{F}_{\text{proj}}$ and residual addition:

$$\mathbf{h}_{\text{in}} = \mathbf{h} + \mathcal{F}_{\text{proj}}(\mathbf{e}_m), \tag{3}$$

where $\mathcal{F}_{\text{proj}}$ maps the motion embedding to the hidden feature layout of the diffusion backbone. This residual motion injection steers the temporal dynamics of the denoising process while preserving pretrained spatial priors.

The denoiser $\mathbf{v}_\theta$ is optimized to predict the target velocity $\mathbf{u}_t(\mathbf{z}_0, \epsilon) = \epsilon - \mathbf{z}_0$:

$$\mathcal{L}_{\text{denoise}} = \mathbb{E}_{t,\mathbf{z}_{\text{src}},\mathbf{z}_0,\epsilon,y,I,\mathbf{e}_m}\left[\|\mathbf{v}_\theta(\tilde{\mathbf{z}}_t, t, y, I, \mathbf{e}_m) - \mathbf{u}_t(\mathbf{z}_0, \epsilon)\|_2^2\right]. \tag{4}$$

This design allows the model to preserve scene appearance from the source video while generating the target video under the desired camera motion. To support both I2V and V2V within a unified formulation, we construct the source latent as a pseudo-video latent whose first frame is obtained by encoding the input image, while the remaining latent frames are zero-filled for I2V.

4.2 Unified Motion Embedding Space

To enable modality-agnostic camera control, we map video, text, and pose inputs into a shared motion embedding space that captures both global trajectory intent and temporal camera dynamics. Given an input from modality $m \in \{\text{video}, \text{text}, \text{pose}\}$, a modality-specific encoder Φ_m produces a sequence of N motion tokens $\mathbf{h}_m \in \mathbb{R}^{N\times D}$, where N is a fixed token length shared across modalities and D denotes the embedding dimension. We prepend a learnable *global motion token* $\mathbf{h}_g \in \mathbb{R}^{1\times D}$ to aggregate trajectory-level information. The concatenated sequence is processed by a lightweight temporal Transformer $\mathcal{T}_m$:

$$\mathbf{e}_m = \mathcal{T}_m([\mathbf{h}_g; \mathbf{h}_m]), \quad \mathbf{e}_m \in \mathbb{R}^{(N+1)\times D} \tag{5}$$

The first output token $\mathbf{e}_m^{(0)}$ encodes the global motion intent, while the remaining tokens $\mathbf{e}_m^{(1:N)}$ represent temporally ordered camera dynamics.

Video Motion Encoder Φ_{video}. To extract camera motion cues from video, we adopt the feature aggregation module of VGGT [42]. The encoder patchifies input frames and appends a learnable camera token to each frame sequence. Through Alternating-Attention blocks that interleave frame-wise and global self-attention, multi-view 3D geometric information is aggregated into these camera tokens. We use the resulting camera tokens as the motion token sequence $\mathbf{h}_{\text{video}}$.

Text Motion Encoder Φ_{text}. Unlike camera motion, which unfolds over time, text provides a static description without explicit temporal structure. We encode the sentence using a frozen T5 encoder [40] to obtain contextualized text tokens. To lift this representation into a temporal motion sequence, we introduce N learnable motion queries that cross-attend to the text tokens via multi-head attention. The resulting query features form the motion token sequence $\mathbf{h}_{\text{text}}$, which represents temporal camera dynamics inferred from language.

Pose Motion Encoder Φ_{pose}. We represent the camera pose at each frame as a flattened 3×4 camera extrinsic matrix. To project these geometrically explicit parameters into the shared D-dimensional embedding space, we apply a frame-wise multi-layer perceptron (MLP) with GELU activations. This yields the motion token sequence $\mathbf{h}_{\text{pose}}$ while preserving the original trajectory structure.

To align equivalent camera trajectories across modalities, we train the motion encoders with a composite objective consisting of global alignment, temporal synchronization, and geometric fidelity regularization.

Global Alignment. We align the global motion tokens $\mathbf{e}_m^{(0)}$ across modalities using an InfoNCE objective computed over modality pairs:

$$\mathcal{L}_{\text{NCE}} = \sum_{(u,v)\in\mathcal{P}} \frac{1}{B} \sum_{i=1}^{B} -\log \frac{\exp\left(\text{sim}(\mathbf{u}_i^{(0)}, \mathbf{v}_i^{(0)})/\tau\right)}{\sum_{j=1}^{B} \exp\left(\text{sim}(\mathbf{u}_i^{(0)}, \mathbf{v}_j^{(0)})/\tau\right)}, \tag{6}$$

where $\mathcal{P} = \{(\mathbf{e}_{\text{video}}, \mathbf{e}_{\text{text}}), (\mathbf{e}_{\text{video}}, \mathbf{e}_{\text{pose}}), (\mathbf{e}_{\text{text}}, \mathbf{e}_{\text{pose}})\}$, B is the batch size, τ is a temperature parameter, and $\text{sim}(\cdot,\cdot)$ denotes cosine similarity. In practice, we symmetrize the loss by swapping the anchor and target modalities.

Temporal Synchronization. To enforce fine-grained temporal alignment, we minimize the cosine distance between corresponding temporal tokens:

$$\mathcal{L}_{\text{temp}} = \sum_{(u,v)\in\mathcal{P}} \frac{1}{B \cdot N} \sum_{i=1}^{B} \sum_{k=1}^{N} \left(1 - \text{sim}(\mathbf{u}_i^{(k)}, \mathbf{v}_i^{(k)})\right). \tag{7}$$

Geometric fidelity regularization. Contrastive objectives are primarily driven by relative similarity on normalized embeddings, which may weaken absolute geometric cues in the learned space [43]. To retain physically meaningful camera motion, we attach a shared pose regressor that predicts camera extrinsics from each modality embedding:

$$\mathcal{L}_{\text{pose}} = \sum_{m\in\{\text{video,text,pose}\}} \frac{1}{B} \sum_{i=1}^{B} \|\hat{\mathbf{p}}_{m,i} - \mathbf{p}_{\text{GT},i}\|_1, \tag{8}$$

where $\hat{\mathbf{p}}_{m,i}$ denotes the pose predicted from modality m for the i-th sample, and $\mathbf{p}_{\text{GT},i}$ is the corresponding ground-truth camera pose. This regularization encourages the motion embeddings to preserve consistent 3D camera geometry.

The overall objective for learning the unified motion space is

$$\mathcal{L}_{\text{align}} = \mathcal{L}_{\text{NCE}} + \lambda_t \mathcal{L}_{\text{temp}} + \lambda_p \mathcal{L}_{\text{pose}}, \tag{9}$$

where λ_t and λ_p balance the different loss terms.

4.3 Latent Motion Consistency

Although the target motion embedding guides generation, it does not by itself guarantee strict geometric consistency in the generated trajectory. To improve trajectory fidelity, we introduce a Motion Embedding Predictor $\mathcal{M}_{\text{pred}}$, which consists of 3D convolutions and a temporal Transformer encoder. Given a clean latent $\mathbf{z}_0$, the predictor estimates a motion embedding sequence: $\hat{\mathbf{e}} = \mathcal{M}_{\text{pred}}(\mathbf{z}_0)$.

We first train the predictor using a dual-granularity cosine similarity loss:

$$\mathcal{L}_{\text{motion}} = \left(1 - \text{sim}(\hat{\mathbf{e}}^{(0)}, \mathbf{e}^{(0)}_{\text{video}})\right) + \lambda_h \frac{1}{N} \sum_{k=1}^{N} \left(1 - \text{sim}(\hat{\mathbf{e}}^{(k)}, \mathbf{e}^{(k)}_{\text{video}})\right), \tag{10}$$

where λ_h balances global and token-wise alignment. The first term aligns the global motion intent, while the second enforces frame-wise temporal consistency.

Since the diffusion backbone may yield imperfect estimates of the clean latent, we improve robustness by injecting mild diffusion noise during predictor training, sampling from lower diffusion timesteps. After convergence, $\mathcal{M}_{\text{pred}}$ is frozen and used during diffusion backbone training.

During diffusion training, we reconstruct the clean target latent $\hat{\mathbf{z}}_0$ from the noisy target latent using the predicted velocity $\mathbf{v}_\theta(\tilde{\mathbf{z}}_t, t, y, I, e_m)$:

$$\hat{\mathbf{z}}_0 = \mathbf{z}_t - t\, \mathbf{v}_\theta(\tilde{\mathbf{z}}_t, t, y, I, \mathbf{e}_m). \tag{11}$$

We then feed $\hat{\mathbf{z}}_0$ into the frozen predictor and apply the same motion consistency loss with respect to the target motion embedding $\mathbf{e}_m$. This latent-space constraint explicitly encourages the generated trajectory to match the target motion without requiring pixel-space reconstruction or decoding of high-resolution video frames.

4.4 Training Objective and Strategy

We train our generative backbone by jointly optimizing the denoising loss and the latent motion consistency loss. Given the motion embedding predicted from the reconstructed clean latent, $\hat{\mathbf{e}}_{\text{gen}} = \mathcal{M}_{\text{pred}}(\hat{\mathbf{z}}_0)$, we enforce alignment with the target motion embedding $\mathbf{e}_m$ using $\mathcal{L}_{\text{motion}}$. The overall objective is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{denoise}} + \lambda_m \mathcal{L}_{\text{motion}}(\hat{\mathbf{e}}_{\text{gen}}, \mathbf{e}_m), \tag{12}$$

where λ_m controls a trade-off between generative fidelity and motion consistency.

Following the efficient fine-tuning paradigm of CamCloneMaster [24], we update only the 3D spatial-temporal attention layers and the block-specific projection MLPs $\mathcal{F}_{\text{proj}}$ within the diffusion backbone. This preserves the pretrained generative prior while efficiently adapting the model to motion-conditioned video generation. To support both I2V and V2V within a single framework, we use balanced joint training. At each iteration, the model receives either a full source sequence (V2V) or a single-frame source condition (I2V) with equal probability. This strategy encourages robust performance across both tasks without bias toward a particular source format.

Table 1: Quantitative results for I2V & V2V tasks. (Best: **Bold**, Second best: Underline)

Method	Target Modality	Visual Quality				Dynamic Quality		Motion Accuracy			Source Preservation	
		FVD ↓	FID ↓	CLIP Score ↑	Vbench Visual ↑	CLIP-F ↑	Vbench Dynamic ↑	E_{rot} ↓	E_{trans} ↓	CamMC ↓	FVD-V ↓	CLIP-V ↑
Image-to-Video (I2V)												
CamI2V [14]	Pose	361.95	39.56	29.32	0.5655	0.9834	0.8688	1.1890	3.9492	4.5648	-	-
MotionClone [30]	Video	601.77	50.00	28.72	0.5826	0.9843	0.7887	2.1734	6.4202	7.4974	-	-
CamCloneMaster [24]	Video	**312.13**	39.68	29.00	0.5555	0.9823	0.9042	1.5235	4.0527	4.9585	-	-
TriMotion	Text	348.63	40.23	28.41	0.5594	**0.9844**	**0.9249**	1.6875	4.0148	5.0989	-	-
TriMotion	Video	337.09	39.68	29.23	0.5591	0.9841	0.9128	1.1710	3.8465	4.4117	-	-
TriMotion	Pose	317.85	**39.11**	**29.37**	**0.5620**	0.9843	0.9168	**1.0113**	**3.8441**	**4.3087**	-	-
Video-to-Video (V2V)												
DaS [19]	Pose	274.22	38.21	29.29	0.4953	0.9744	0.8893	2.7558	7.7572	9.2675	270.65	0.8204
TrajectoryCrafter [18]	Pose	299.96	39.51	29.40	0.5680	0.9764	0.9026	**0.6954**	3.2621	**3.5711**	267.18	0.8794
ReCamMaster [15]	Pose	253.65	38.77	29.13	0.5696	0.9883	0.9277	1.3184	3.6559	4.3611	233.58	0.8833
CamCloneMaster [15]	Video	241.38	38.64	29.24	0.5725	0.9878	0.9235	1.5040	3.8987	4.7964	195.89	0.8791
TriMotion	Text	237.87	38.65	29.10	**0.5820**	0.9890	0.9345	1.6030	3.7525	4.7778	186.73	0.9041
TriMotion	Video	254.55	38.86	29.28	0.5728	0.9882	0.9258	1.1795	3.6468	4.2384	216.55	0.8851
TriMotion	Pose	**221.59**	**38.16**	**29.43**	0.5767	**0.9895**	**0.9372**	0.9488	**3.2356**	3.6797	**172.25**	**0.9071**

5 Experiments

5.1 Experimental Setup

Implementation Details. We train our method on the Motion Triplet Dataset described in Sec. 3, with 1% of the data held out for validation. The unified motion space is trained for 100 epochs, the Motion Embedding Predictor for 10 epochs, and the diffusion backbone for 10K iterations. All models are trained on 4 NVIDIA H200 GPUs using the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 0.01, and a learning rate of 1×10^{-4}.

Evaluation Set. Our evaluation set consists of 500 source videos from Koala-36M [44] and 500 video-pose pairs from RealEstate10K [45]. Koala-36M provides diverse real-world videos with scene-level text descriptions, while RealEstate10K provides videos with corresponding per-frame camera poses. For samples with missing motion descriptions, we generate them from pose trajectories following the pipeline in Sec. 3, enabling evaluation across all three input modalities.

Evaluation Metrics. We evaluate performance from three perspectives: visual quality, dynamic quality, and camera motion accuracy. For visual quality, we report FVD [46], FID [47], CLIP Score, and the Image Quality and Aesthetic Quality metrics from VBench [48], which assess visual clarity and perceptual appeal. For dynamic quality, we report CLIP-F [15] together with VBench metrics including Motion Smoothness, Temporal Flickering, and Subject/Background Consistency. For camera motion accuracy, we estimate camera poses using MegaSaM [49] and compute Translation Error (E_{trans}), Rotation Error (E_{rot}), and Camera Motion Consistency (CamMC) [14]. For V2V, we further report FVD-V [50] and CLIP-V to measure temporal quality and source-content preservation.

5.2 Comparison with State-of-the-Art Methods

Baseline Models. We compare TriMotion with state-of-the-art camera-controlled video generation methods. For I2V, we compare against CamI2V [14], MotionClone [30], and CamCloneMaster [24]. CamI2V conditions generation on explicit camera information with geometry-aware attention, while MotionClone is a

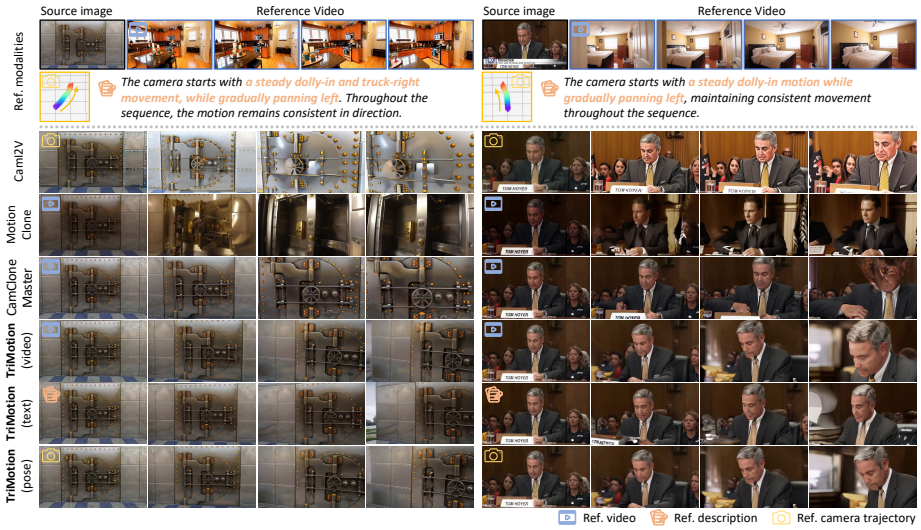

Fig. 3: Qualitative comparison for Camera-controlled I2V generation results.

training-free framework that transfers motion directly from a reference video through temporal attention cues. CamCloneMaster also transfers camera motion from a reference video, but does so by jointly processing reference and target tokens in a unified attention framework. For V2V, we compare against DaS [19], TrajectoryCrafter [18], ReCamMaster [15], and CamCloneMaster [24]. DaS uses 3D tracking videos as motion guidance, TrajectoryCrafter relies on explicit geometry and rendered point-cloud cues, ReCamMaster conditions generation on target camera trajectories together with source video features. All methods are evaluated using their official implementations and pretrained models.

Quantitative Comparison. As shown in Tab. 1, TriMotion achieves the best overall balance between visual quality, temporal stability, and camera-motion accuracy across both I2V and V2V settings. The pose-conditioned variant performs best on most metrics, while the video- and text-conditioned variants remain competitive, suggesting that the shared motion space provides an effective control signal across all three modalities. We attribute this advantage to the unified motion embedding space and the latent motion consistency loss, which together reduce motion drift during training. While TrajectoryCrafter achieves competitive camera accuracy via explicit 3D point-cloud guidance, such rendering often introduces occlusion artifacts and texture degradation, undermining perceptual quality and temporal stability under challenging viewpoint changes.

Qualitative Comparison. Fig. 3 and Fig. 4 present qualitative comparisons on I2V and V2V, respectively. Across all input modalities, TriMotion generates cleaner videos with more stable motion and follows the target camera trajectory more faithfully. CamI2V and ReCamMaster generally follow the target pose reasonably well, but still exhibit visible artifacts in challenging cases. MotionClone and CamCloneMaster capture the coarse motion trend, yet often drift from the desired trajectory. TrajectoryCrafter achieves strong camera control, but large

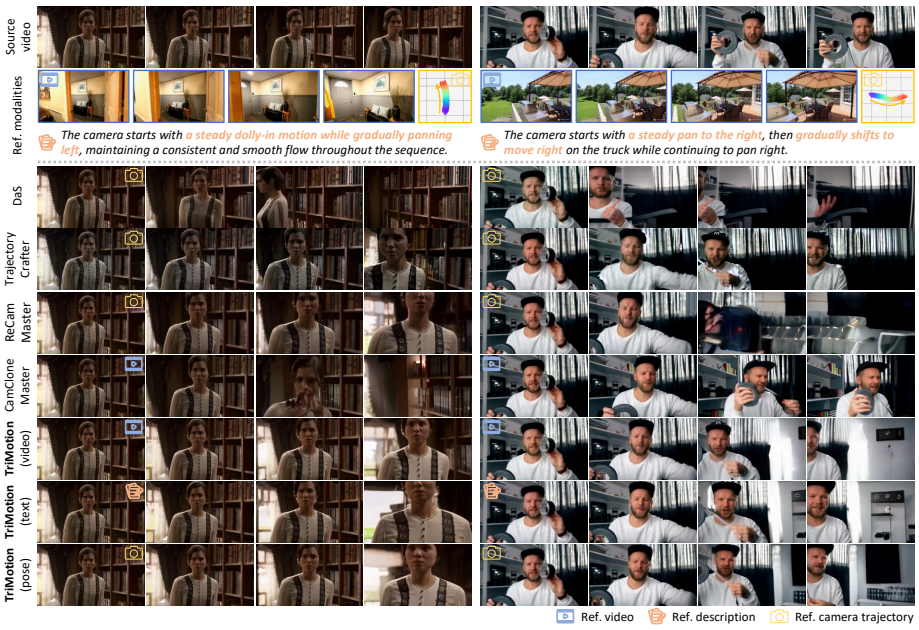

Fig. 4: Qualitative comparison for Camera-controlled V2V generation results.

Table 2: Quantitative results of our ablation study on conditioning design and latent motion consistency. (Best: **Bold**, Second best: Underline)

Method	Visual Quality				Dynamic Quality		Motion Accuracy			Source Preservation	
	FVD ↓	FID ↓	CLIP Score ↑	Vbench Visual ↑	CLIP-F ↑	Vbench Dynamic ↑	E_{rot} ↓	E_{trans} ↓	CamMC ↓	FVD-V ↓	CLIP-V ↑
Concatenation	**193.56**	39.75	**29.83**	**0.5777**	**0.9914**	**0.9461**	2.1250	10.5327	11.2641	**116.34**	**0.9214**
Subtraction	265.53	**37.12**	29.80	0.5714	0.9866	0.9259	2.8396	9.6904	10.9317	227.69	0.8903
w/o $\mathcal{L}_{\text{motion}}$	249.85	38.48	29.18	0.5683	0.9883	0.9255	1.2307	4.2986	4.8660	214.42	0.8828
TriMotion	221.59	38.16	29.43	0.5767	0.9895	0.9372	**0.9488**	**3.2356**	**3.6797**	172.25	0.9071

viewpoint changes reveal noticeable artifacts, while DaS often struggles to preserve the source video content. Overall, TriMotion follows compound and long-horizon camera motions more faithfully while maintaining better scene structure and temporal coherence.

5.3 Ablation Study

To validate our conditioning design and latent motion consistency, we perform ablations in the V2V setting with pose conditioning, since alternative variants require a source motion embedding that is not naturally available in I2V.

As shown in Tab. 2, concatenation improves several visual metrics but degrades camera-motion accuracy, suggesting that jointly providing source and target embeddings introduces competing motion cues and biases the model toward source preservation. By contrast, target-only conditioning provides the cleanest motion signal and the best overall trade-off. Using the difference vector is less effective, likely because it removes absolute motion structure and retains only relative displacement.

Table 3: Quantitative results of the analysis on the unified motion embedding space. Positive and Negative refer to the average cosine similarity between positive and negative pairs, respectively. (Best: **Bold**, Second best: Underline)

Metric	Cross Modal Alignment			Metric	Geometric Information				
	Video ↔ Pose	Video ↔ Text	Text ↔Pose		VGGT [42]	MegaSAM [49]	Pose	Video	Text
Positive	0.9605	0.8887	0.9121	E_{rot} ↓	1.6226	**0.1478**	0.4505	0.9972	1.4300
Negative	0.0317	0.0256	0.0243	E_{trans} ↓	**0.4459**	0.7003	0.5530	1.7211	2.9489
Recall@1	98.50 %	98.00 %	99.50 %	CamMC ↓	**0.4966**	0.7713	0.6603	1.8707	3.8359

We further ablate the latent motion consistency loss by training with and without $\mathcal{L}_{\text{motion}}$. Removing this loss consistently degrades camera accuracy and slightly weakens visual and temporal quality, indicating that motion conditioning alone is insufficient and that the proposed constraint improves trajectory fidelity by aligning the predicted clean latent with the target motion embedding.

5.4 Analysis on Unified Motion Embedding Space

We analyze the unified motion space from two perspectives: cross-modal alignment and geometric fidelity. For cross-modal alignment, we use the validation split of the Motion Triplet Dataset, where modality pairs from the same trajectory are treated as positives and pairs from different trajectories within the same scene are treated as negatives. We report cosine similarity and Recall@1 across modalities. As shown in Tab. 3, the learned embeddings exhibit a clear separation between positive and negative pairs and achieve strong retrieval performance, indicating that they capture motion patterns rather than scene identity.

We further evaluate geometric fidelity through pose regressor in Sec. 4.2 on the evaluation set from RealEstate10K. The results show that the learned embeddings preserve meaningful geometric information and remain competitive with strong feed-forward pose estimation models, including VGGT [42] and MegaSaM [49]. Notably, even text embeddings can regress camera poses reasonably well, suggesting that the language representation also captures underlying motion geometry.

5.5 Applications: Cross-Modal Motion Composition

A key advantage of the unified motion embedding space is that motion representations from different modalities become directly composable. Since video, pose, and text inputs are aligned in the same motion space, their embeddings can be manipulated at inference time to enable flexible camera control beyond single-modality conditioning.

Given a pair of motion embeddings $\mathbf{e}_a$ and $\mathbf{e}_b$ from different modalities, we explore two composition settings. First, we study sequential motion composition by concatenating the two motion sequences, where the second motion is offset by the final state of the first to ensure continuity. Second, we study cross-modal motion interpolation by linearly interpolating between $\mathbf{e}_a$ and $\mathbf{e}_b$, which produces camera motions that smoothly blend the two inputs. As shown in Fig. 5, the unified motion space supports both coherent multi-stage trajectories and smooth interpolation without retraining or manual trajectory design.

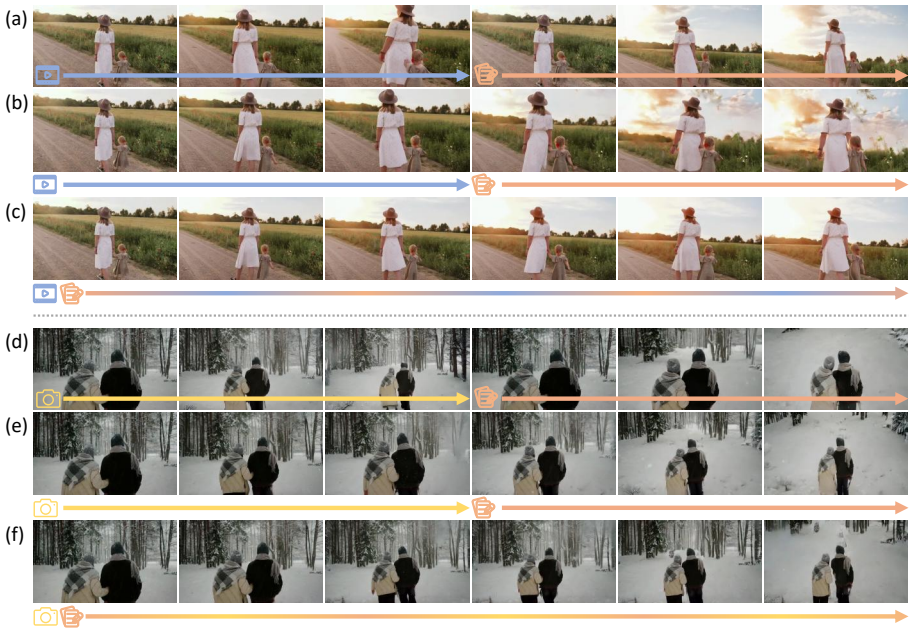

Fig. 5: Applications to cross-modal motion composition for camera-controlled V2V generation. (a) and (d) show camera-controlled V2V results from a single modality, while (b) and (e) illustrate sequential composition, and (c) and (f) illustrate linear interpolation between motion embeddings.

5.6 User Study

We conduct a user study on Amazon Mechanical Turk (MTurk) [51] with 20 participants to evaluate three aspects of our framework: (1) the quality of the motion descriptions in the Motion Triplet Dataset, (2) the effectiveness of cross-modal motion composition, and (3) human preference on camera-motion following and perceptual quality. All evaluation items are randomly drawn from the corresponding set and presented in randomized order.

Motion-description quality. We evaluate 30 video-text pairs from the Motion Triplet Dataset, considering the short and detailed descriptions separately. For each pair, participants rate how well the text matches the camera motion in the video on a five point scale, where 1 and 5 indicate a poor match and an excellent match, respectively. The results in Tab. 4 indicate that the proposed geometry-grounded captioning pipeline reliably captures the underlying camera trajectories.

Cross-modal motion composition. We further assess the composability of the unified motion space using 10 sequential-composition examples and 10 interpolation examples. For each result, users rate on a five-point scale whether the generated video clearly executes the two motions in sequence in the sequential-composition setting or reflects both input motions in the interpolation setting. As

Table 4: The result of user study. We report the mean and standard deviation for description quality and cross-modal composition.

Description Quality	Short Description		Long Description		Cross-modal	Sequential	Interpolation	
Score	3.94 ± 0.23		4.11 ± 0.22		Score	3.97 ± 0.23	3.88 ± 0.27	
I2V	-	CamI2V	MotionClone	CamClone Master	TriMotion Text	TriMotion Video	TriMotion Pose	
Artifact ↓	-	26.0%	63.5%	18.5%	17.5%	15.5%	**12.5%**	
Motion Accuracy ↑	-	78.5%	41.5%	67.5%	81.0%	82.5%	**86.5%**	
V2V	DaS	Trajectory Crafter	ReCam Master	CamClone Master	TriMotion Text	TriMotion Video	TriMotion Pose	
Artifact ↓	44.5%	20.5%	11.0%	21.5%	15.5%	**8.5%**	9.5%	
Motion Accuracy ↑	43.5%	69.0%	80.0%	61.5%	74.5%	**89.5%**	86.5%	

shown in Tab. 4, the learned motion space supports flexible cross-modal motion composition.

Human evaluation of generated videos. For generation quality, we consider 10 examples from each of the I2V and V2V settings. Participants are first shown a reference clip illustrating the target camera motion and are then presented with videos generated by TriMotion and the baseline methods similar to previous works [52–54]. They complete two evaluation tasks independently by selecting all videos that (1) contain noticeable visual artifacts and (2) faithfully follow the target camera motion. We report the resulting selection rates in Tab. 4, where lower is better for artifact selection and higher is better for motion-following selection. The results show that TriMotion generates high-quality videos while following the target camera motion more faithfully than prior methods.

6 Conclusion

We present *TriMotion*, a unified framework for cross-modal camera motion control in video generation. By aligning video, pose, and text in a shared motion embedding space, TriMotion enables flexible and consistent camera control across heterogeneous inputs. We further introduce a latent motion consistency constraint that improves trajectory fidelity during generation. Experiments on both I2V and V2V show that TriMotion achieves a strong balance among perceptual quality, temporal coherence, and camera-motion accuracy. In addition, the shared motion space supports practical applications such as sequential motion composition and cross-modal motion interpolation, highlighting the flexibility of the proposed framework.

Limitations and Future Work. Although TriMotion effectively transfers smooth camera motions, extreme trajectories can still produce boundary outpainting artifacts when large unseen regions must be synthesized. A promising direction for future work is to integrate our framework with explicit 3D background outpainting or temporal extrapolation modules to improve robustness under large geometric transformations. It would also be interesting to extend the framework beyond camera motion and jointly model cinematic attributes [55, 56] such as color grading, which could enable richer transfer of filmic style and mood together with camera dynamics.

Acknowledgment This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(RS-2024-00338439) and the InnoCORE program of the Ministry of Science and ICT (N10250156).

References

1. David Bordwell, Kristin Thompson, and Jeff Smith. *Film art: An introduction*, volume 7. McGraw-Hill New York, 2008.
2. Robin Wood. *Hitchcock's films revisited*. Columbia University Press, 2002.
3. Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022.
4. Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.
5. 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 *CVPR*, pages 22563–22575, 2023.
6. Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, et al. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024.
7. Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pages 399–417. Springer, 2024.
8. 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.
9. Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.
10. Wooseok Jeon, Seunghyun Shin, Dongmin Shin, and Hae-Gon Jeon. Motion prior distillation in time reversal sampling for generative inbetweening. In *The Fourteenth International Conference on Learning Representations*, 2026.
11. 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.

12. 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.
13. 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.
14. 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.
15. Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. Recammaster: Camera-controlled generative rendering from a single video. In *ICCV*, 2025.
16. 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.
17. 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.
18. Mark Yu, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 100–111, 2025.
19. 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.
20. Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. In *European Conference on Computer Vision (ECCV)*, 2024.
21. David Junhao Zhang, Roni Paiss, Shiran Zada, Nikhil Karnad, David E. Jacobs, Yael Pritch, Inbar Mosseri, Mike Zheng Shou, Neal Wadhwa, and Nataniel Ruiz. Recapture: Generative video camera controls for user-provided videos using masked video fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
22. Junyoung Seo, Jisang Han, Jaewoo Jung, Siyoon Jin, JoungBin Lee, Takuya Narihira, Kazumi Fukuda, Takashi Shibuya, Donghoon Ahn, Shoukang Hu, Seungryong Kim, and Yuki Mitsufuji. Vid-camedit: Video camera trajectory editing with generative rendering from estimated geometry. *arXiv preprint arXiv:2506.13697*, 2025.
23. Hyeonho Jeong, Suhyeon Lee, and Jong Chul Ye. Reangle-a-video: 4d video generation as video-to-video translation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.
24. Yawen Luo, Xiaoyu Shi, Jianhong Bai, Menghan Xia, Tianfan Xue, Xintao Wang, Pengfei Wan, Di Zhang, and Kun Gai. Camclonemaster: Enabling reference-based camera control for video generation. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pages 1–10, 2025.

25. Teng Hu, Jiangning Zhang, Ran Yi, Yating Wang, Hongrui Huang, Jieyu Weng, Yabiao Wang, and Lizhuang Ma. Motionmaster: Training-free camera motion transfer for video generation. *arXiv preprint arXiv:2404.15789*, 2024.
26. Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7310–7320, 2024.
27. 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.
28. 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.
29. Teng Li, Guangcong Zheng, Rui Jiang, Shuigen Zhan, Tao Wu, Yehao Lu, Yining Lin, and Xi Li. Realcam-i2v: Real-world image-to-video generation with interactive complex camera control. *arXiv preprint arXiv:2502.10059*, 2025.
30. Pengyang Ling, Jiazi Bu, Pan Zhang, Xiaoyi Dong, Yuhang Zang, Tong Wu, Huaian Chen, Jiaqi Wang, and Yi Jin. Motionclone: Training-free motion cloning for controllable video generation. *arXiv preprint arXiv:2406.05338*, 2024.
31. Weikang Bian, Zhaoyang Huang, Xiaoyu Shi, Yijin Li, Fu-Yun Wang, and Hongsheng Li. Gs-dit: Advancing video generation with dynamic 3d gaussian fields through efficient dense 3d point tracking. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21717–21727, 2025.
32. 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.
33. Epic Games. Unreal Engine 5. `https://www.unrealengine.com/en-US/unreal-engine-5`. Accessed: 2026-02-28.
34. Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.
35. Mizuki Arai, Tatsuya Ishigaki, Masayuki Kawarada, Yusuke Miyao, Hiroya Takamura, and Ichiro Kobayashi. Evaluating llms' ability to understand numerical time series for text generation. In *Proceedings of the 18th International Natural Language Generation Conference*, pages 232–248, 2025.
36. An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.
37. Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.
38. Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.
39. Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.
40. Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer

learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.
41. Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.
42. Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025.
43. Sharut Gupta, Joshua Robinson, Derek Lim, Soledad Villar, and Stefanie Jegelka. Structuring representation geometry with rotationally equivariant contrastive learning. *arXiv preprint arXiv:2306.13924*, 2023.
44. Qiuheng Wang, Yukai Shi, Jiarong Ou, Rui Chen, Ke Lin, Jiahao Wang, Boyuan Jiang, Haotian Yang, Mingwu Zheng, Xin Tao, et al. Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8428–8437, 2025.
45. Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.
46. Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.
47. Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.
48. Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024.
49. Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast and robust structure and motion from casual dynamic videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10486–10496, 2025.
50. Yiming Xie, Chun-Han Yao, Vikram Voleti, Huaizu Jiang, and Varun Jampani. Sv4d: Dynamic 3d content generation with multi-frame and multi-view consistency. *arXiv preprint arXiv:2407.17470*, 2024.
51. Kevin Crowston. Amazon mechanical turk: A research tool for organizations and information systems scholars. In *Shaping the Future of ICT Research. Methods and Approaches: IFIP WG 8.2, Working Conference, Tampa, FL, USA, December 13-14, 2012. Proceedings*, 2012.
52. Seonmi Park, Inhwan Bae, Seunghyun Shin, and Hae-Gon Jeon. Kinetic typography diffusion model. In *European Conference on Computer Vision*, pages 166–185. Springer, 2024.
53. Chanhui Lee, Seunghyun Shin, Donggyu Choi, Hae-gon Jeon, and Jeany Son. Universal image immunization against diffusion-based image editing via semantic injection. *arXiv preprint arXiv:2602.14679*, 2026.

54. Wooseok Jeon, Seungho Park, Seunghyun Shin, Sangeyl Lee, Hyeonho Jeong, and Hae-Gon Jeon. Rebalancing reference frame dominance to improve motion in image-to-video models. *arXiv preprint arXiv:2605.19398*, 2026.
55. Seunghyun Shin, Dongmin Shin, Jisu Shin, Hae-Gon Jeon, and Joon-Young Lee. Video color grading via look-up table generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19141–19152, 2025.
56. Seunghyun Shin, Jisu Shin, Jihwan Bae, Inwook Shim, and Hae-Gon Jeon. Close imitation of expert retouching for black-and-white photography. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25037–25046, June 2024.