Camera Operator: Object-Grounded Camera Trajectory Generation from Text and 3D Bounding Box Sequences

Zhongyuan Hu
huzhongyyuan@gmail.com
Tsinghua University
Shenzhen, China

Yue Ma*
mayuefighting@gmail.com
Hong Kong University of Science and Technology
Hong Kong, China

Jiangming Wang
jiangmingwang643@gmail.com
Sun Yat-Sen University
Guangzhou, China

Ronghui Li
lrh22@mails.tsinghua.edu.cn
Tsinghua University
Shenzhen, China

Xiu Li†
li.xiu@sz.tsinghua.edu.cn
Tsinghua University
Shenzhen, China

Abstract

Camera trajectory design plays a crucial role in video production, as it determines how a target object is presented throughout a shot. In object-aware camera trajectory generation, the key challenge is not only to produce plausible camera motion, but also to maintain stable camera–object interaction as the target moves. Specifically, the trajectory should keep the object within a proper image region, preserve a stable and reasonable scale, and ensure the camera consistently faces the object. Existing methods improve controllability from text or point-based object cues, but they often fail to preserve this interaction over time. Language provides high-level camera-motion intent, while a time-varying 3D bounding box supplies frame-wise target geometry for preserving framing, visibility, and viewing direction. However, object location alone does not capture the target's spatial extent and orientation, and existing datasets lack aligned supervision for generating camera trajectories from both text and structured 3D target state. To address this, we establish target-object camera trajectory generation from text and 3D target state as a new benchmark task, and introduce BlockCam, a 41K-sequence benchmark with aligned textual motion descriptions, target-object 3D bounding box sequences, and camera trajectories collected from real and synthetic videos under continuous target visibility. We further propose Camera Operator, a geometry-conditioned flow-matching model whose intermediate clean trajectory estimate enables projection-space supervision of camera–object relations. We evaluate this benchmark using trajectory-alignment and distributional metrics as well as two geometric measures, In-Frame Ratio and Look-at Error, to assess target visibility and camera–target alignment. Experiments show that the full configuration strengthens relation preservation while maintaining strong text–trajectory alignment. Across progressively richer geometry-conditioned configurations, the full-box model outperforms its location-only counterpart, and a user study favors our results in terms of alignment and framing.

CCS Concepts

• **Computing methodologies** → **Animation**.

Keywords

Camera Trajectory Generation, Controllable Video Generation

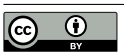
ACM Reference Format:
Zhongyuan Hu, Yue Ma, Jiangming Wang, Ronghui Li, and Xiu Li. 2026. Camera Operator: Object-Grounded Camera Trajectory Generation from Text and 3D Bounding Box Sequences. In *Proceedings of the 34th ACM International Conference on Multimedia (MM '26), November 10–14, 2026, Rio de Janeiro, Brazil.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3767308.3835459

1 Introduction

Camera trajectories are a core control signal [1, 11, 13, 14, 19, 27, 44, 45] in video production and controllable video generation, because they determine how a designated subject is presented throughout a shot. A particularly important setting is object-aware camera trajectory generation, where the camera must continuously follow a designated target object. This capability is useful for virtual cinematography and camera-controlled video synthesis, where explicit camera motion provides structured and editable control. However, the task is fundamentally challenging. Object-aware camera planning is not merely about producing a smooth 3D path or roughly following the target center. A usable shot must preserve a stable camera-object relation over time: the target should remain visible, occupy a reasonable image region, and stay consistently looked at as both the camera and the target move. In other words, even a smooth trajectory can still fail cinematographically if the target drifts out of frame, undergoes abrupt scale changes, or is no longer consistently viewed. We refer to these failure modes as object drift, composition drift, and gaze drift, respectively.

Related 3D research spans motion generation and joint camera–human modeling [12, 21, 48]. Recent controllable video generation methods use camera motion as an external condition for synthesizing videos [1, 13, 16, 45], demonstrating the value of explicit camera control, but they generally assume that a suitable trajectory is already available. Meanwhile, camera trajectory

*Project lead.
†Corresponding author.



MM '26, Rio de Janeiro, Brazil

ACM ISBN 979-8-4007-2213-4/2026/11
https://doi.org/10.1145/3767308.3835459

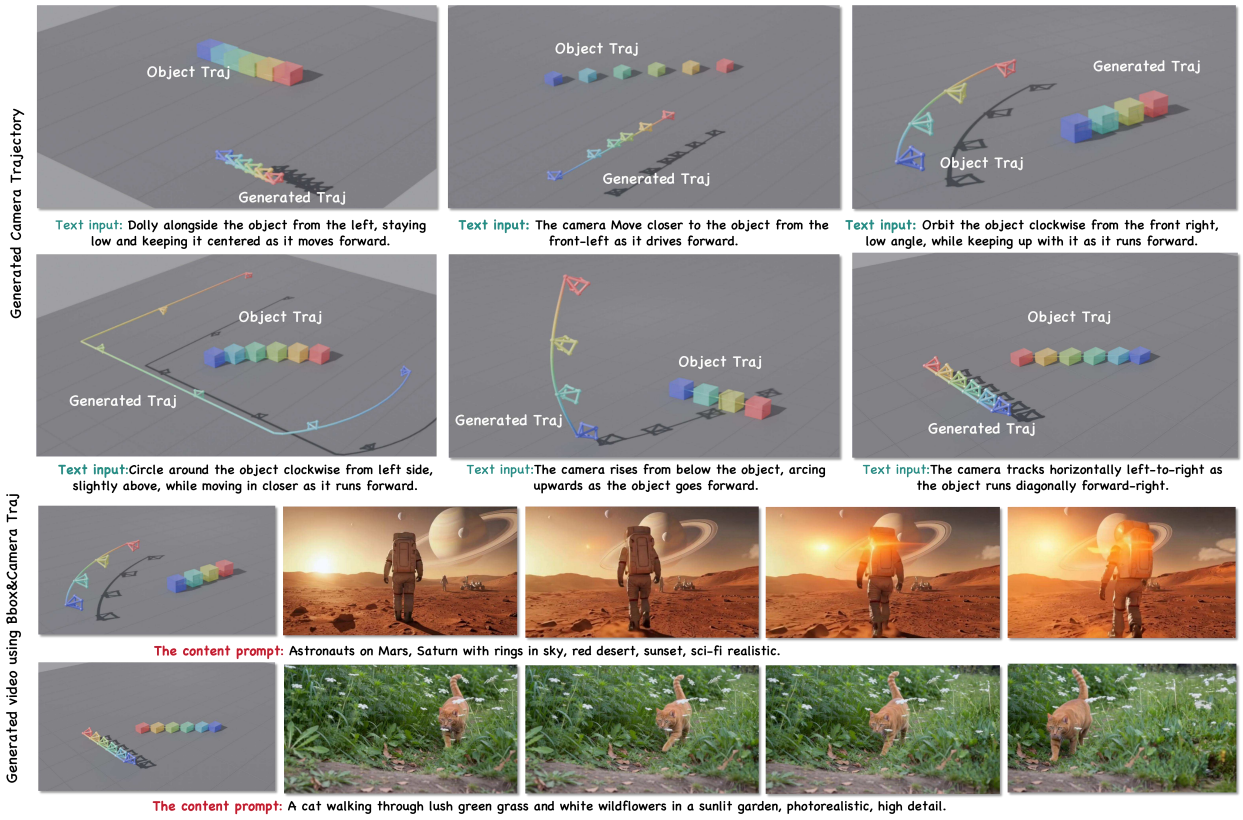


Figure 1: Overview. Top: Given a time-varying 3D target-box sequence and a text prompt, our method predicts a 3D camera trajectory. Bottom: The predicted camera trajectory can condition a camera-controlled video model such as ReCamMaster [1]; projecting the target's 3D boxes along this trajectory yields 2D box tracks that can guide an object-trajectory model such as MagicMotion [20]. The colors from blue to red represent the progression from the first frame to the last frame for both the camera trajectory and the object trajectory.

generation methods provide only limited grounding for object-aware planning. CCD [18] models camera motion in a character-centric coordinate system, whereas E.T. [8] explicitly conditions on a 3D character-center trajectory. These approaches provide target-relative or center-level grounding, but neither encodes the target's full time-varying extent for preserving framing, projected scale, and target-relative viewing direction. Director3D [22] generates text-conditioned camera trajectories within a joint scene-generation pipeline, while GenDoP [50] uses directorial text with optional initial-frame RGB-D; neither explicitly models the dynamic 3D geometry of a designated target object throughout the sequence. Thus, existing methods can produce plausible motion, but they still struggle to maintain a stable camera-object relation over time.

These limitations point to two key challenges. First, relation-preserving control: the model must generate a trajectory compatible with time-varying visibility, composition, and viewing geometry rather than merely follow coarse target motion. Second, data support: learning such behavior requires aligned language, 3D target state, and camera trajectories. Our key insight is that per-frame 3D boxes compactly encode not only location but also target extent and orientation, providing cues for framing and look-at behavior as both camera and target move.

To address these limitations, we establish target-object camera trajectory generation from text and 3D target state as a new benchmark task and introduce BlockCam, a curated 41K-sequence benchmark of aligned textual motion descriptions, target-object 3D bounding box sequences, and camera trajectories collected from real and synthetic videos. We focus on single-shot, single-object clips with continuous target visibility and no shot transition, isolating camera-bbox relation preservation from occlusion, target switching, and target re-acquisition. Building on this benchmark, we propose Camera Operator, a geometry-conditioned flow-matching model [24] for object-aware camera trajectory generation. Language provides sequence-level motion intent and camera-object interaction cues, while the 3D bounding box sequence provides frame-wise target state. Its intermediate clean trajectory estimate enables projection-space supervision of the camera-bbox relation at each sampled flow time, reducing object drift, composition drift, and gaze drift. We benchmark methods using trajectory-alignment

and distributional metrics together with relation-preservation measures, including In-Frame Ratio, Look-at Error, and a ranking-based user study.

Our contributions are:

- We establish target-object camera trajectory generation from text and 3D target state as a new benchmark task, and introduce BlockCam, a curated 41K-sequence benchmark of single-shot clips with continuous target visibility and aligned text, target-object 3D bounding boxes, and camera trajectories.
- We present Camera Operator, a geometry-conditioned flow-matching model that combines language with frame-wise 3D target state and applies geometric supervision to preserve the camera-bbox relation and reduce object drift, composition drift, and gaze drift.
- We introduce In-Frame Ratio and Look-at Error to measure target visibility and camera-target alignment, and complement them with a ranking-based user study.

2 Related Work

2.1 Camera Trajectory Dataset

Existing camera trajectory datasets [18, 23, 49, 51] expose only part of the supervision needed for object-aware camera planning. Large pose collections such as RealEstate10K [51], MVImgNet [49], and DL3DV-10K [23] provide abundant camera motion and scene geometry [4, 9], but they are largely scene-centric and do not align trajectories with a designated target object or object-aware language. Datasets closer to trajectory generation also remain incomplete: CCD [18] provides synthetic tracking data with a tracking-oriented motion taxonomy, E.T. [8] pairs film clips with text and 3D character-center trajectories but does not encode object extent, and GenDoP [50] provides directorial captions for free-moving camera generation without explicit target-object geometry over time. As a result, existing datasets either lack semantic grounding or reduce the target to location-only cues, leaving limited support for studying camera planning that must preserve a stable camera–box relation for a designated moving target.

2.2 Camera Trajectory Generation

Camera trajectory generation has evolved from manual rules and numerical optimization [7, 43] to learned generative models, but existing methods still split between free-moving generation and weakly grounded object-aware control. CCD [18] models tracking-style motion in a local target-relative coordinate system. E.T. [8] moves to text-conditioned diffusion with global-frame 3D character-center trajectories, but center paths mainly indicate where the object is and provide only weak support for shot scale and framing control, which are fundamental concepts in cinematography [3]. Director3D [22] and GenDoP [50] further improve realism and language richness for open-domain camera generation, yet neither explicitly receives target-object geometry that can anchor the camera to a designated moving object over time.

Adjacent visual-content research spans pose- and motion-guided synthesis and animation [5, 29–31, 35, 38, 40, 47], editing and inpainting [6, 25, 26, 28, 42], motion transfer [32, 34], and sequence modeling and understanding [33, 39].

3 Dataset

Existing data sources remain insufficient for object-grounded camera planning. As summarized in Table 1, existing datasets either lack designated target grounding or do not pair camera trajectories with aligned target-object 3D bounding box sequences and corresponding text annotations. To address this gap, we construct a hybrid dataset with 12K real-world sequences and 29K synthetic sequences generated in Unreal Engine. The real branch captures diverse camera behaviors from real videos, while the synthetic branch provides clean and controllable geometric annotations. Both branches use branch-specific processing for camera–box consistency: the real branch constructs camera and box annotations in a shared recovered frame and retains clips labeled with continuous target visibility, whereas the synthetic branch additionally requires all eight projected box corners to remain inside the image throughout the sequence. This yields a unified benchmark for studying camera-bbox relation preservation.

3.1 Real Data Collection Pipeline

Data Filtering and Object Anchoring. As shown in Figure 2, we first segment raw internet videos into single-shot clips using TransNetV2 [41] to remove shot transitions that can introduce abrupt camera changes or target switches. We then use Qwen3-VL [2] to extract specific sequence-level tags, including the target's semantic category, camera ego-motion, global target visibility, and the object's screen-space proportion. Based on these tags, we retain clips labeled as having active camera motion, continuous target visibility, and an appropriate screen-space footprint. This step constructs a curated dataset specifically for object-grounded planning, rather than a fully unconstrained video collection.

Camera Curation. For each filtered video, we use ViPE [17] to estimate the full camera parameters—comprising the extrinsic trajectory $\mathcal{T}_c = \{\mathbf{T}_{c2w}^{(t)}\}_{t=1}^{T}$ and the per-frame intrinsics $\mathbf{K}_t$—alongside pixel-aligned depth maps $\mathcal{D} = \{D_t\}_{t=1}^{T}$. We use the recovered per-frame object masks $\mathcal{M} = \{M_t\}$ with category labels in the subsequent target-matching stage. Following E.T. [8], we filter anomalous jumps in the raw poses and apply a Kalman smoother to improve temporal continuity. We further manually inspect the recovered camera motion against the source video and discard clips whose estimated camera trajectory is semantically inconsistent with the observed shot motion. Although reconstructed monocularly, the retained trajectories are processed, smoothed, and manually verified for consistency with the observed shot motion.

3D Bounding Box Extraction. We first map the target object identified by Qwen3-VL to the corresponding label in the recovered masks $\{M_t\}$. Within the matched mask, dense surface points are tracked using SpatialTrackerV2 [46], with the ViPE-recovered camera parameters and depth supplied as fixed inputs and its internal camera optimization disabled. This keeps the target points and camera trajectory in a shared recovered coordinate frame. With these camera-aligned point trajectories established, we hold the reconstructed box dimensions and relative orientation fixed to regularize temporal scale. We select the reference frame t_{opt} with the largest target-mask area, back-project its masked pixels using $D_{t_{opt}}$ and $\mathbf{T}_{c2w}^{(t_{opt})}$, and filter depth outliers before fitting the box. Fitting an

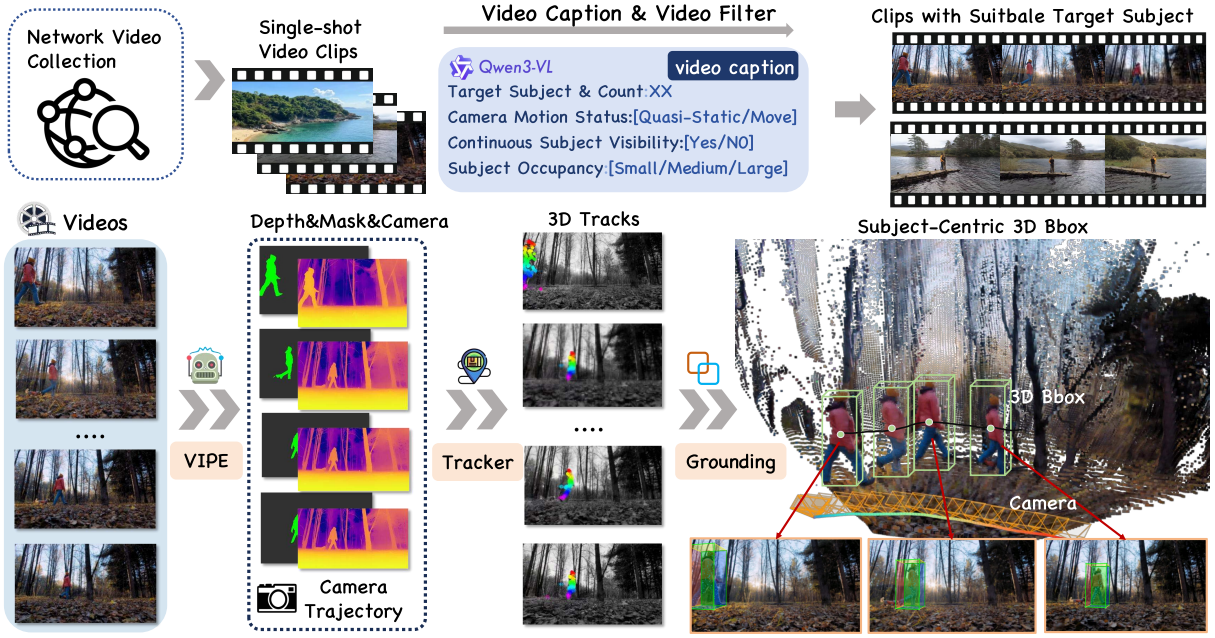


Figure 2: Real Data Collection Pipeline. (1) We employ TransNetV2 [41] for shot segmentation and use Qwen3-VL [2] to filter clips with active camera motion and continuous target visibility. (2) We utilize ViPE [17] to estimate camera trajectories and pixel-aligned depth maps, followed by Kalman smoothing for temporal continuity. (3) We integrate SpatialTrackerV2 [46] and back-projection to extract coordinate-aligned 3D bounding boxes for the target subjects.

Dataset	Domain	Subject Grounding		Text Annotations		Statistics	
		3D BBox	State	Cam.	Subj.	#Sample	#Frame
MVImgNet [49]	Captured	✗	Static	✗	✗	219K	6.5M
RealEstate10k [51]	Youtube	✗	Static	✗	✗	79K	10M
DL3DV-10K [23]	Captured	✗	Static	✗	✗	10K	51M
CCD [18]	Synthetic	✗	Dynamic	✓	✗	25K	4.5M
E.T. [8]	Film	✗	Dynamic	✓	✓	115K	11M
GenDoP [50]	Film	✗	-	✓	~	29K	11M
BlockCam (Ours)	**Real+Synthetic**	✓	**Both**	✓	✓	**41K**	**11.2M**

Table 1: Comparison of camera trajectory datasets. Existing datasets provide different subsets of camera poses, text descriptions, and object cues, but none pairs camera trajectories with aligned target-object 3D bounding box sequences. BlockCam combines static and dynamic objects with paired camera-object language across both real and synthetic data. In the text-annotation columns, "Cam." denotes camera-motion descriptions and "Subj." denotes target-object descriptions; ~ denotes captions that may mention subjects without an explicit tracked-target annotation.

oriented bounding box (OBB) to this cleaned data determines the fixed reconstructed box dimensions S and reference center C_{opt}. With the box geometry S locked, we propagate it to all other frames. Consequently, the box center at frame t, denoted as C_t, is updated purely by adding the median 3D displacement of the tracked points: $C_t = C_{opt} + \text{median}_i(x_t^{(i)} - x_{opt}^{(i)})$, where $x_t^{(i)}$ and $x_{opt}^{(i)}$ are the unprojected 3D coordinates of the i-th tracked point at frame t and reference frame t_{opt}, respectively. This procedure yields a temporally regularized approximation of the target's reconstructed extent in that recovered frame.

3.2 Synthetic Data Collection

We use Unreal Engine [10] to generate sequences with exact synthetic geometry and camera annotations. The environment covers camera–subject motion patterns including orbiting, following, and dollying. Scenes use diverse Fab assets in indoor and outdoor settings. Object paths are obtained by stratified sampling of spline control points, with ground-level height and uniform arc-length frame sampling for approximately constant speed; each path is paired with one of ten cinematographic camera-motion patterns. The strict eight-corner frustum check guarantees that the complete

target box remains inside the image in every retained frame: a sequence is rejected if any projected box corner falls outside the image. This process yields approximately 29K sequences with exact engine annotations under the synthetic branch's strict visibility condition.

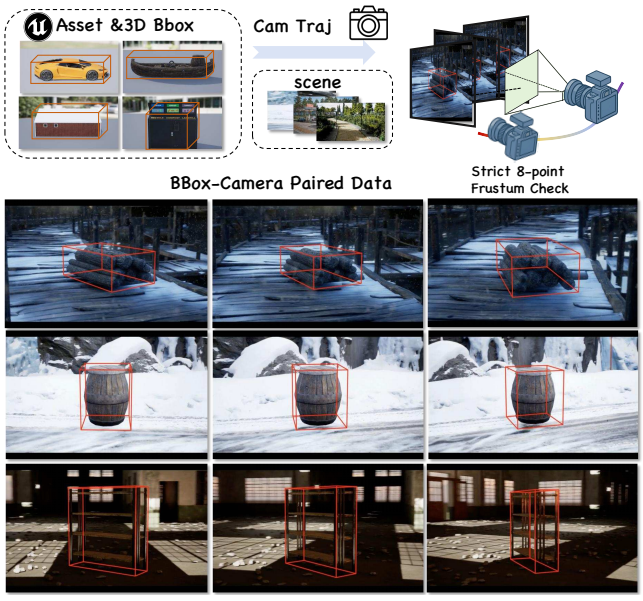


Figure 3: Synthetic Data Generation. (Top) We pair 3D assets with diverse trajectories and apply an eight-corner frustum check to enforce full-box visibility. (Bottom) Rendered examples with exact synthetic 3D box annotations.

3.3 Motion Tagging

For caption construction, given the camera trajectory and object bounding box sequence, we express both in a common reference frame by aligning the first-frame camera pose to a canonical transform. Based on this representation, we compute camera, object, and relative-motion signals. These signals are discretized into 27 translational and 7 rotational states following E.T. [8] and GenDoP [50], with temporal majority voting suppressing isolated frame-level label changes. The resulting structured motion primitives are provided to Qwen3-VL [2] to generate natural-language descriptions. Because the descriptions are generated from processed camera, object, and relative-motion signals, they provide geometry-grounded motion-level annotations rather than unrestricted free-form captions.

Taken together, the three annotations in BlockCam play complementary roles in the dataset. Text provides high-level motion intent and camera-object interaction cues, the 3D bounding box sequence provides the target object's frame-wise spatial state, and the camera trajectory provides the supervision signal to be generated. This alignment makes the dataset suitable for studying object-grounded camera planning through explicit camera-bbox relation preservation, rather than trajectory generation in isolation.

4 Camera Trajectory Generation

4.1 Camera-BBox Relation Preservation

Given a textual motion description τ and a sequence of per-frame 3D bounding boxes $\mathcal{B} = \{B_t\}_{t=1}^{T}$, our goal is to generate a camera trajectory $\mathcal{T} = \{(\mathbf{R}_t, \mathbf{p}_t)\}_{t=1}^{T}$, where $\mathbf{R}_t \in \mathrm{SO}(3)$ is the camera orientation and $\mathbf{p}_t \in \mathbb{R}^3$ is the camera position. Each pose is represented by a 6D rotation representation [52] and a 3D translation, resulting in a 9D pose vector. Following benchmark preprocessing, camera poses and boxes are expressed in the same sample-specific canonical frame defined by the first target box.

A trajectory may be smooth in 3D yet still fail cinematographically if the target leaves the frame, changes apparent scale uncontrollably, or is no longer consistently viewed. To model camera–object interaction, we define a projection- and direction-based relation state for each frame. Let $\mathcal{V}_t = \{\mathbf{v}_{t,k}\}_{k=1}^{8}$ denote the eight vertices of the target-object bounding box at frame t. Given a camera pose and known intrinsics, projecting $\mathcal{V}_t$ and evaluating the camera-to-target viewing direction yield

$$\mathbf{r}_t = (\mathbf{c}_t, \mathbf{s}_t, \rho_t, a_t), \tag{1}$$

Here $\mathbf{c}_t \in \mathbb{R}^2$ is the normalized projected box center, $\mathbf{s}_t \in \mathbb{R}^2_{\geq 0}$ is its normalized image-plane extent, ρ_t is the visible-vertex ratio, and $a_t = \mathbf{f}_t \cdot \mathbf{d}_t$ is the alignment between the camera forward direction $\mathbf{f}_t$ and the unit vector $\mathbf{d}_t$ from the camera center to the box center. Object drift mainly appears as deteriorated $\mathbf{c}_t$ or ρ_t, composition drift as unstable $\mathbf{s}_t$, and gaze drift as reduced a_t.

Compared to point trajectories, 3D bounding boxes provide richer cues for maintaining consistent framing because they encode target extent and orientation.

4.2 Language and Object-State Conditioning

The two inputs play complementary roles: text provides sequence-level motion intent, while the bounding box sequence provides frame-wise object state.

We encode text using a CLIP encoder [37], producing a pooled global feature with token-level features for cross-attention. Text provides global motion intent, while frame-wise relation details are grounded by object geometry. For object representation, each frame is encoded as a structured state derived from the 3D bounding box, including position, approximate orientation, scale, and motion features. These features are processed by a Transformer encoder to produce object-state tokens $\mathbf{H}_{\mathcal{B}} \in \mathbb{R}^{T \times d}$.

4.3 Flow-Based Camera Trajectory Generation

We adopt flow matching [24] because its velocity formulation exposes an estimated clean trajectory at each sampled flow time, enabling projection- and direction-based geometric supervision before final rollout. The overall architecture is shown in Figure 4.

During training, we sample a flow time $s \sim \sigma(\mathcal{N}(0,1))$ and construct an interpolated trajectory

$$\mathbf{y}_s = (1-s)\,\mathbf{y}_0 + s\,\mathbf{y}_1, \tag{2}$$

where $\mathbf{y}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is Gaussian noise and $\mathbf{y}_1$ is the reference trajectory. The target velocity is the straight-path displacement

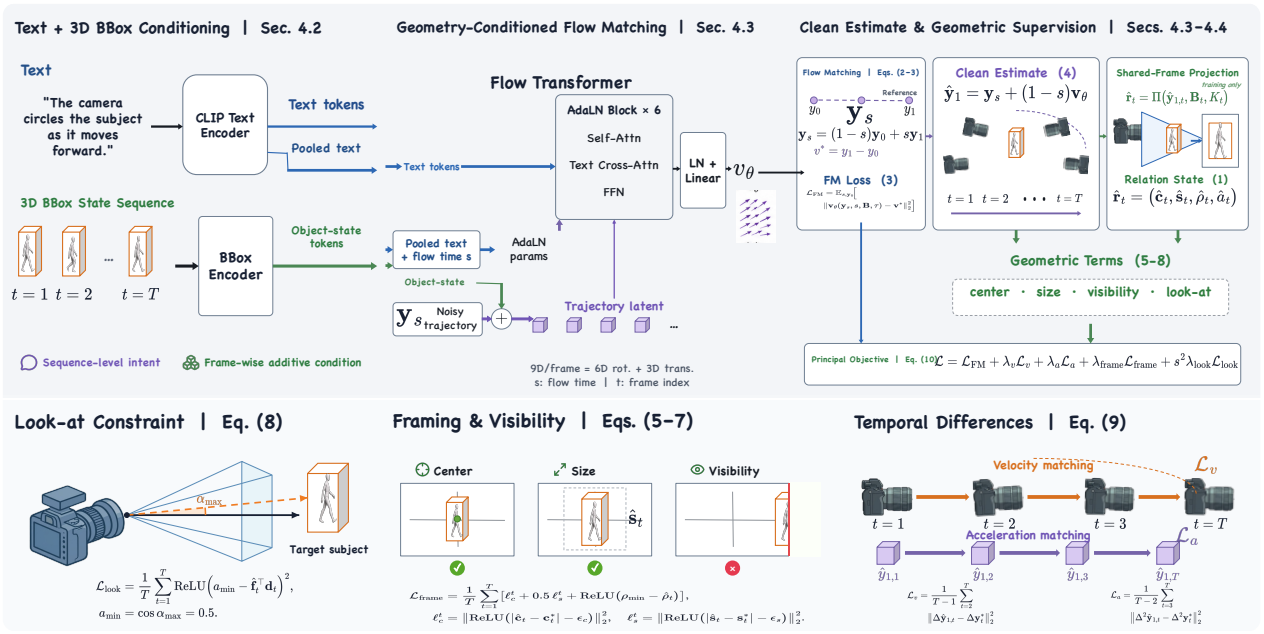


Figure 4: Overview of Camera Operator. Text supplies sequence-level camera-motion intent, while time-varying 3D target boxes provide frame-wise target geometry. The geometry-conditioned flow Transformer predicts a camera-trajectory velocity field. Intermediate clean-trajectory estimates derived from this field support projection-space framing and visibility supervision, shared-frame look-at supervision, and trajectory-space temporal-difference supervision.

$\mathbf{v}^* = \mathbf{y}_1 - \mathbf{y}_0$, and the base objective is

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{s,\mathbf{y}_0}\left[\left\|\mathbf{v}_\theta(\mathbf{y}_s, s, \mathcal{B}, \tau) - \mathbf{v}^*\right\|^2\right]. \tag{3}$$

The trajectory generator is implemented as a Transformer conditioned on both language features and object-state tokens. The noisy trajectory is projected into the hidden space, object-state tokens are injected by frame-wise additive conditioning, and text is incorporated through global conditioning and cross-attention. The model predicts a 9D velocity field, from which the corresponding clean trajectory estimate is

$$\hat{\mathbf{y}}_1 = \mathbf{y}_s + (1-s)\mathbf{v}_\theta. \tag{4}$$

Relation supervision is evaluated on this clean estimate at each sampled flow time. Importantly, we retain trajectory supervision, since the flow-matching objective provides stable learning signals while the geometric terms regularize camera–object relations such as framing and viewing direction. Before evaluating these losses, the normalized trajectory is decoded to the sample-specific geometric frame in which camera poses and target boxes are jointly expressed for projection.

Inference. At inference time, we integrate the learned velocity field with a fixed-step midpoint solver under classifier-free guidance [15] to obtain the final trajectory.

4.4 Geometric Camera-BBox Supervision

Trajectory-space supervision alone does not ensure valid camera–object interaction. We therefore introduce projection- and direction-based geometric constraints that regularize framing and viewing direction while retaining trajectory supervision.

For each frame, let $\mathbf{r}_t^* = (\mathbf{c}_t^*, \mathbf{s}_t^*, \rho_t^*, a_t^*)$ denote the relation state induced by the reference camera trajectory and the target bounding box sequence. We define an admissible region

$$\mathcal{A}_t = \left\{(\mathbf{c}, \mathbf{s}, \rho, a) \;\middle|\; \begin{array}{l} \|\mathbf{c} - \mathbf{c}_t^*\|_\infty \le \epsilon_c,\ \|\mathbf{s} - \mathbf{s}_t^*\|_\infty \le \epsilon_s, \\ \rho \ge \rho_{\min},\ a \ge \cos\alpha_{\max} \end{array}\right\}. \tag{5}$$

Here ϵ_c, ϵ_s, and $\rho_{\min}$ denote admissible tolerances for framing, scale, and visibility. We set the maximum admissible viewing-angle deviation to $\alpha_{\max} = 60^\circ$, so $\cos\alpha_{\max} = 0.5$; we use $\epsilon_c = 0.15$, $\epsilon_s = 0.2$, and $\rho_{\min} = 0.3$.

We first constrain image-plane framing. Given the predicted clean trajectory $\hat{\mathbf{y}}_1$, we project the target object's 3D bounding-box vertices under known intrinsics and compute the normalized center $\hat{\mathbf{c}}_t \in \mathbb{R}^2$, image-plane extent $\hat{\mathbf{s}}_t \in \mathbb{R}^2_{\ge 0}$, and visibility ratio $\hat{\rho}_t$. The framing loss is

$$\mathcal{L}_{\mathrm{frame}} = \frac{1}{T}\sum_{t=1}^{T}\left[\ell_c^t + 0.5\,\ell_s^t + \mathrm{ReLU}(\rho_{\min} - \hat{\rho}_t)\right], \tag{6}$$

where

$$\ell_c^t = \|\,\mathrm{ReLU}(|\hat{\mathbf{c}}_t - \mathbf{c}_t^*| - \epsilon_c)\|^2, \quad \ell_s^t = \|\,\mathrm{ReLU}(|\hat{\mathbf{s}}_t - \mathbf{s}_t^*| - \epsilon_s)\|^2. \tag{7}$$

We next constrain viewing direction. Let $\hat{\mathbf{f}}_t$ denote the predicted camera forward direction and let $\mathbf{d}_t$ be the unit vector from the

Method	Dataset	Text-Trajectory Align.		Trajectory Quality		Relation Pres.		User Study Top-1	
		F1↑	CLaTr↑	Cov.↑	FD_{CLaTr}↓	IFR↑	LE↓	Alignment↑	Framing↑
CCD [18]	Pre-trained	0.31	4.25	0.33	358.0	34.2	39.8	–	–
E.T. [8]	Pre-trained	0.32	2.46	0.05	610.0	41.7	31.5	–	–
Director3D [22]	Pre-trained	0.13	0.00	0.17	542.4	24.8	44.3	–	–
GenDoP [50]	Pre-trained	0.40	5.68	0.21	222	34.2	32.5	–	–
E.T. [8]	BlockCam	0.36	27.4	0.76	41.2	61.8	18.9	10.0	10.0
GenDoP [50]	BlockCam	0.39	31.6	0.82	34.8	68.7	15.2	30.0	22.0
Camera Operator (Ours)	BlockCam	**0.42**	**35.7**	**0.86**	**12.3**	**89.4**	**9.7**	**60.0**	**68.0**

Table 2: Main comparison on BlockCam. Metrics cover text alignment (F1/CLaTr), distributional quality (Cov./FD_{CLaTr}), relation preservation (IFR/LE), and blinded Top-1 preference (%). Pre-trained rows are cross-dataset references; BlockCam-trained rows form the same-data, native-interface comparison.

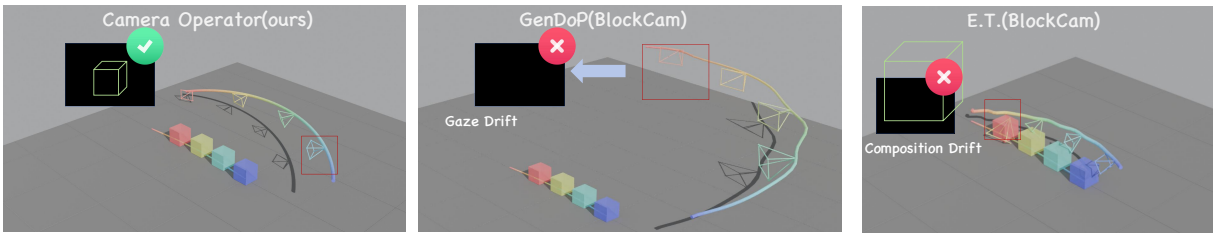


Figure 5: Representative relation failures: GenDoP exhibits gaze drift and E.T. composition drift, while ours maintains a stable camera–object relation.

camera center to the target center. The look-at loss is

$$\mathcal{L}_{\text{look}} = \frac{1}{T}\sum_{t=1}^{T} \text{ReLU}(\cos\alpha_{\max} - \hat{\mathbf{f}}_t \cdot \mathbf{d}_t)^2, \tag{8}$$

This term penalizes trajectories whose optical axis deviates too far from the target.

Condition	Text-Traj		Quality		Relation	
	F1↑	CLaTr↑	Cov.↑	FD_{CLaTr} ↓	IFR↑	LE↓
Text only	0.19	29.0	0.78	44.8	52.1	21.7
Text + Point Traj.	0.22	31.3	0.83	34.6	72.4	13.6
Text + 3D BBox	**0.42**	**35.7**	**0.86**	**12.3**	**89.4**	**9.7**

Table 3: Progressively richer conditioning configurations; applicable geometric supervision varies with the available condition.

Configuration	Text-Traj		Quality		Relation	
	F1↑	CLaTr↑	Cov.↑	FD_{CLaTr} ↓	IFR↑	LE↓
Full Model	**0.42**	**35.7**	**0.86**	**12.3**	**89.4**	**9.7**
w/o Framing Loss	0.21	28.3	0.84	46.0	73.6	10.9
w/o Look-at Loss	0.21	28.4	0.85	40.2	84.1	17.6
w/o Both Geo. Losses	0.23	28.2	0.81	42.4	65.3	22.4

Table 4: Ablation on geometric supervision.

We write compact first- and second-order temporal terms against the reference trajectory $\mathbf{y}^* \equiv \mathbf{y}_1$:

$$\mathcal{L}_v = \frac{1}{T-1}\sum_{t=2}^{T}\|\Delta\hat{\mathbf{y}}_{1,t} - \Delta\mathbf{y}_t^*\|^2, \quad \mathcal{L}_a = \frac{1}{T-2}\sum_{t=3}^{T}\|\Delta^2\hat{\mathbf{y}}_{1,t} - \Delta^2\mathbf{y}_t^*\|^2, \tag{9}$$

where $\Delta\mathbf{y}_t = \mathbf{y}_t - \mathbf{y}_{t-1}$ and $\Delta^2\mathbf{y}_t = \Delta\mathbf{y}_t - \Delta\mathbf{y}_{t-1}$. We use s^2 to down-weight look-at supervision at noisy flow states. The principal objective is

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + \lambda_v\mathcal{L}_v + \lambda_a\mathcal{L}_a + \lambda_{\text{frame}}\mathcal{L}_{\text{frame}} + s^2\lambda_{\text{look}}\mathcal{L}_{\text{look}}, \tag{10}$$

The implementation also uses a training-only interaction-state auxiliary and an absolute screen-space barrier.

Within these relation losses, the reference projection defines an admissible region rather than an exact relation target, preserving tolerance for multiple valid camera trajectories.

5 Experiment

5.1 Experimental Setting

Given text and a per-frame 3D target-box sequence, the model predicts a 9D camera pose (6D rotation and 3D translation) per frame. We use the fixed 940-sequence evaluation protocol reported in Table 2.

We compare with CCD [18], E.T. [8], Director3D [22], and GenDoP [50]. Official checkpoints provide cross-dataset references. For the primary same-data comparison, we train E.T. and GenDoP on BlockCam using the same partitions while retaining their native interfaces: E.T. receives the per-frame 3D-box centers, GenDoP receives text only, and our method receives text and full 3D boxes.

Figure 6: Downstream video generation using predicted camera trajectories. Each row pairs a trajectory visualization (left) with sampled generated frames (right); the content prompt controls appearance, while the predicted trajectory supplies camera motion.

5.2 Quantitative Results

We report text-trajectory alignment, distributional quality, and camera-bbox relation preservation.

Metrics. Following E.T. [8] and GenDoP [50], we use Caption F1 and CLaTr for text–trajectory alignment, and Coverage and FD_{CLaTr} for trajectory support and distributional quality [36].

These borrowed metrics do not test camera–box relations directly. We therefore add In-Frame Ratio (IFR), the percentage of frames with all eight projected box vertices inside the image, and Look-at Error (LE), the mean angle between the optical axis and the camera-to-target-center vector. A blinded user study reports Top-1 preference for text *Alignment* and target *Framing*. All metrics are evaluated post hoc; IFR and LE summarize relation preservation rather than duplicate the differentiable training losses.

Main Results. The same-data, native-interface block in Table 2 is the main end-to-end comparison. Training E.T. and GenDoP on BlockCam improves their in-domain performance on most metrics, but neither closes the IFR/LE gap. GenDoP's strong CLaTr but weaker IFR/LE is consistent with text alignment alone not ensuring target retention or aiming. E.T.'s center path supplies target location but omits extent and orientation, leaving extent-dependent framing underconstrained as projected scale changes. Our full-box system additionally conditions on these frame-wise geometric cues.

Under the reported protocol, our full system leads all four trajectory metrics among the BlockCam-trained rows, improves IFR from 68.7 to 89.4, and reduces LE from 15.2 to 9.7 over the strongest baseline. Its larger preference gain on Framing than Alignment agrees with the relation metrics.

5.3 Qualitative Results

Figure 6 illustrates downstream video generation from our trajectories. Figure 5 shows representative gaze and composition drift for GenDoP and E.T., respectively; in these examples, ours preserves visibility, framing, and viewing direction.

5.4 Ablation Studies

Table 3 compares progressively richer geometry-conditioned configurations. Text supplies sequence-level camera intent; a point trajectory adds the target-center path; and the full box additionally supplies target extent and orientation. The point-to-full-box change improves IFR and LE, while the text-alignment and distributional metrics also improve. This pattern is consistent with full-box state constraining frame-wise camera–target relation feasibility without replacing the role of language.

Table 4 examines the geometric losses. Among single-term ablations, removing framing supervision causes the larger IFR drop, whereas removing look-at supervision more strongly degrades LE, consistent with their screen-space and viewing-direction roles. Removing the complete geometric configuration weakens IFR and LE together and also worsens the CLaTr-based metrics. This is consistent with the relation terms shaping the generated trajectory rather than merely post-processing its projection.

6 Conclusion

We introduce object-grounded camera trajectory generation and BlockCam. Camera Operator separates sequence-level linguistic intent from frame-wise target geometry, while flow matching exposes a clean trajectory estimate for projection-space relation supervision. Results show stronger camera–object relation preservation while retaining text alignment and trajectory quality.

Limitations. We study offline, single-object shots with continuous visibility, geometry-grounded captions, and paired coordinate normalization to isolate camera–object planning from online perception and target re-acquisition. Partial/noisy observations, multi-target composition, shot transitions, broader language, camera-independent normalization, and online planning remain open.

References

[1] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, and Di Zhang. 2025. ReCamMaster: Camera-Controlled Generative Rendering from a Single Video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14834–14844. doi:10.1109/ICCV51701.2025.01376

[2] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. 2025. Qwen3-VL Technical Report. arXiv:2511.21631 [cs.CV] https://arxiv.org/abs/2511.21631

[3] David Bordwell, Kristin Thompson, and Jeff Smith. 2019. *Film Art: An Introduction*. McGraw-Hill Education.

[4] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3D: Learning from RGB-D Data in Indoor Environments. arXiv:1709.06158 [cs.CV] https://arxiv.org/abs/1709.06158

[5] Xuewei Chen, Zhimin Chen, and Yiren Song. 2025. TransAnimate: Taming Layer Diffusion to Generate RGBA Video. arXiv:2503.17934 [cs.CV] https://arxiv.org/abs/2503.17934

[6] Yiyang Chen, Xuanhua He, Xiujun Ma, and Jack Ma. 2026. ContextFlow: Training-Free Video Object Editing via Adaptive Context Enrichment. *Proceedings of the AAAI Conference on Artificial Intelligence* 40, 4 (2026), 3129–3137. doi:10.1609/aaai.v40i4.37306

[7] Marc Christie, Patrick Olivier, and Jean-Marie Normand. 2008. Camera Control in Computer Graphics. *Computer Graphics Forum* 27, 8 (2008), 2197–2218. doi:10.1111/j.1467-8659.2008.01181.x

[8] Robin Courant, Nicolas Dufour, Xi Wang, Marc Christie, and Vicky Kalogeiton. 2024. E.T. the Exceptional Trajectories: Text-to-Camera-Trajectory Generation with Character Awareness. In *ECCV (4) (Lecture Notes in Computer Science, Vol. 15062)*. Springer, 464–480.

[9] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. 2017. ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. arXiv:1702.04405 [cs.CV] https://arxiv.org/abs/1702.04405

[10] Epic Games. 2024. *Unreal Engine*. https://www.unrealengine.com

[11] Wanquan Feng, Jiawei Liu, Pengqi Tu, Tianhao Qi, Mingzhen Sun, Tianxiang Ma, Songtao Zhao, Siyu Zhou, and Qian He. 2025. I2VControl-Camera: Precise Video Camera Control with Adjustable Motion Strength. In *International Conference on Learning Representations*. https://openreview.net/forum?id=6K0UdeOY3f

[12] Haonan Han, Xiangzuo Wu, Huan Liao, Zunnan Xu, Zhongyuan Hu, Ronghui Li, Yachao Zhang, and Xiu Li. 2024. AToM: Aligning Text-to-Motion Model at Event-Level with GPT-4Vision Reward. arXiv:2411.18654 [cs.CV] https://arxiv.org/abs/2411.18654

[13] 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).

[14] Hao He, Ceyuan Yang, Shanchuan Lin, Yinghao Xu, Meng Wei, Liangke Gui, Qi Zhao, Gordon Wetzstein, Lu Jiang, and Hongsheng Li. 2025. CameraCtrl II: Dynamic Scene Exploration via Camera-Controlled Video Diffusion Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13416–13426. doi:10.1109/ICCV51701.2025.01246

[15] Jonathan Ho and Tim Salimans. 2022. Classifier-free diffusion guidance. arXiv:2207.12598 [cs.LG] https://arxiv.org/abs/2207.12598

[16] Chen Hou and Zhibo Chen. 2024. Training-free Camera Control for Video Generation. *CoRR* abs/2406.10126 (2024). https://arxiv.org/abs/2406.10126

[17] Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Korovko, Huan Ling, Xuanchi Ren, Tianchang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, et al. 2025. ViPE: Video Pose Engine for 3D Geometric Perception. arXiv:2508.10934 [cs.CV]

[18] Hongda Jiang, Xi Wang, Marc Christie, Libin Liu, and Baoquan Chen. 2024. Cinematographic Camera Diffusion Model. *Comput. Graph. Forum* 43, 2 (2024), e15055. doi:10.1111/cgf.15055

[19] Zhengfei Kuang, Shengqu Cai, Hao He, Yinghao Xu, Hongsheng Li, Leonidas Guibas, and Gordon Wetzstein. 2024. Collaborative Video Diffusion: Consistent Multi-Video Generation with Camera Control. In *Advances in Neural Information Processing Systems*, Vol. 37. 16240–16271.

[20] Quanhao Li, Zhen Xing, Rui Wang, Hui Zhang, Qi Dai, and Zuxuan Wu. 2025. MagicMotion: Controllable Video Generation with Dense-to-Sparse Trajectory Guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 12112–12123. doi:10.1109/ICCV51701.2025.01126

[21] Ronghui Li, Zhongyuan Hu, Li Siyao, Youliang Zhang, Haozhe Xie, Mingyuan Zhang, Jie Guo, Xiu Li, and Ziwei Liu. 2026. InfiniteDance: Scalable 3D Dance Generation Towards In-the-Wild Generalization. arXiv:2603.13375 [cs.CV]

[22] Xinyang Li, Zhangyu Lai, Linning Xu, Yansong Qu, Liujuan Cao, Shengchuan Zhang, Bo Dai, and Rongrong Ji. 2024. Director3D: Real-world Camera Trajectory and 3D Scene Generation from Text. In *Advances in Neural Information Processing Systems*, Vol. 37. 75125–75151.

[23] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, Xuanmao Li, Xingpeng Sun, Rohan Ashok, Aniruddha Mukherjee, Hao Kang, Xiangrui Kong, Gang Hua, Tianyi Zhang, Bedrich Benes, and Aniket Bera. 2024. DL3DV-10K: A Large-Scale Scene Dataset for Deep Learning-based 3D Vision. In *CVPR*. IEEE, 22160–22169.

[24] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2023. Flow matching for generative modeling. *International Conference on Learning Representations (ICLR)* (2023).

[25] Zeqian Long, Mingzhe Zheng, Kunyu Feng, Xinhua Zhang, Hongyu Liu, Harry Yang, Linfeng Zhang, Qifeng Chen, and Yue Ma. 2026. Follow-Your-Shape: Shape-Aware Image Editing via Trajectory-Guided Region Control. In *The Fourteenth International Conference on Learning Representations*. https://openreview.net/forum?id=uGaR7L3Z1E

[26] Yue Ma, Xiaodong Cun, Sen Liang, Jinbo Xing, Yingqing He, Chenyang Qi, Siran Chen, and Qifeng Chen. 2025. MagicStick: Controllable Video Editing via Control Handle Transformations. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, Piscataway, NJ, USA, 9385–9395. doi:10.1109/WACV61041.2025.00909

[27] Yue Ma, Kunyu Feng, Zhongyuan Hu, Xinyu Wang, Yucheng Wang, Mingzhe Zheng, Bingyuan Wang, Qinghe Wang, Xuanhua He, Hongfa Wang, Chenyang Zhu, Hongyu Liu, Yingqing He, Zeyu Wang, Zhifeng Li, Xiu Li, Sirui Han, Yike Guo, Wei Liu, Dan Xu, Linfeng Zhang, and Qifeng Chen. 2025. Controllable Video Generation: A Survey. arXiv:2507.16869 [cs.GR] https://arxiv.org/abs/2507.16869

[28] Yue Ma, Kunyu Feng, Xinhua Zhang, Hongyu Liu, David Junhao Zhang, Jinbo Xing, Yinhan Zhang, Ayden Yang, Zeyu Wang, and Qifeng Chen. 2025. Follow-Your-Creation: Empowering 4D Creation through Video Inpainting. arXiv:2506.04590 [cs.CV] https://arxiv.org/abs/2506.04590

[29] Yue Ma, Yingqing He, Xiaodong Cun, Xintao Wang, Siran Chen, Xiu Li, and Qifeng Chen. 2024. Follow Your Pose: Pose-Guided Text-to-Video Generation Using Pose-Free Videos. *Proceedings of the AAAI Conference on Artificial Intelligence* 38, 5 (2024), 4117–4125. doi:10.1609/aaai.v38i5.28206

[30] Yue Ma, Yingqing He, Hongfa Wang, Andong Wang, Leqi Shen, Chenyang Qi, Jixuan Ying, Chengfei Cai, Zhifeng Li, Heung-Yeung Shum, Wei Liu, and Qifeng Chen. 2025. Follow-Your-Click: Open-domain Regional Image Animation via Motion Prompts. *Proceedings of the AAAI Conference on Artificial Intelligence* 39, 6 (2025), 6018–6026. doi:10.1609/aaai.v39i6.32643

[31] Yue Ma, Hongyu Liu, Hongfa Wang, Heng Pan, Yingqing He, Junkun Yuan, Ailing Zeng, Chengfei Cai, Heung-Yeung Shum, Wei Liu, and Qifeng Chen. 2024. Follow-Your-Emoji: Fine-Controllable and Expressive Freestyle Portrait Animation. In *SIGGRAPH Asia 2024 Conference Papers*. Association for Computing Machinery, New York, NY, USA, Article 110, 12 pages. doi:10.1145/3680528.3687587

[32] Yue Ma, Yulong Liu, Qiyuan Zhu, Ayden Yang, Kunyu Feng, Xinhua Zhang, Zexuan Yan, Zhifeng Li, Sirui Han, Chenyang Qi, and Qifeng Chen. 2026. Follow-Your-Motion: Video Motion Transfer via Efficient Spatial-Temporal Decoupled Finetuning. In *The Fourteenth International Conference on Learning Representations*. arXiv:2506.05207 [cs.CV]

[33] Yue Ma, Yali Wang, Yue Wu, Ziyu Lyu, Siran Chen, Xiu Li, and Yu Qiao. 2022. Visual Knowledge Graph for Human Action Reasoning in Videos. In *Proceedings of the 30th ACM International Conference on Multimedia*. Association for Computing Machinery, New York, NY, USA, 4132–4141. doi:10.1145/3503161.3548257

[34] Yue Ma, Zhikai Wang, Tianhao Ren, Mingzhe Zheng, Hongyu Liu, Jiayi Guo, Kunyu Feng, Yuxuan Xue, Zixiang Zhao, Konrad Schindler, Qifeng Chen, and Linfeng Zhang. 2026. FastVMT: Eliminating Redundancy in Video Motion Transfer. In *The Fourteenth International Conference on Learning Representations*. arXiv:2602.05551 [cs.CV]

[35] Yue Ma, Zexuan Yan, Hongyu Liu, Hongfa Wang, Heng Pan, Yingqing He, Junkun Yuan, Ailing Zeng, Chengfei Cai, Heung-Yeung Shum, Zhifeng Li, Wei Liu, Linfeng Zhang, and Qifeng Chen. 2026. Follow-Your-Emoji-Faster: Towards Efficient, Fine-Controllable, and Expressive Freestyle Portrait Animation. *International Journal of Computer Vision* 134, 3, Article 130 (2026). doi:10.1007/s11263-025-02685-z

[36] Mathis Petrovich, Michael J. Black, and Gül Varol. 2023. TMR: Text-to-motion retrieval using contrastive 3D human motion synthesis. In *IEEE/CVF International Conference on Computer Vision (ICCV)*. 9488–9497.

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*. 8748–8763.

[38] Quanjian Song, Yiren Song, Kelly Peng, Yuan Gao, and Mike Zheng Shou. 2025. WorldWander: Bridging Egocentric and Exocentric Worlds in Video Generation. arXiv:2511.22098 [cs.CV] https://arxiv.org/abs/2511.22098

[39] Yiren Song, Shijie Huang, Chen Yao, Xiaojun Ye, Hai Ci, Jiaming Liu, Yuxuan Zhang, and Mike Zheng Shou. 2024. ProcessPainter: Learn Painting Process from Sequence Data. arXiv:2406.06062 [cs.CV] https://arxiv.org/abs/2406.06062

[40] Yiren Song, Cheng Liu, Weijia Mao, and Mike Zheng Shou. 2025. Mitty: Diffusion-based Human-to-Robot Video Generation. arXiv:2512.17253 [cs.CV] https://arxiv.org/abs/2512.17253

[41] Tomáš Souček and Jakub Lokoč. 2020. TransNet V2: An effective deep network architecture for fast shot transition detection. arXiv:2008.04838 [cs.CV]

[42] Zhen Wan, Chenyang Qi, Zhiheng Liu, Tao Gui, and Yue Ma. 2025. UniPaint: Unified Space-Time Video Inpainting via Mixture-of-Experts. In *2025 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. IEEE, Piscataway, NJ, USA, 1882–1892. doi:10.1109/ICCVW69036.2025.00198

[43] Xi Wang, Robin Courant, Jinglei Shi, Eric Marchand, and Marc Christie. 2023. JAWS: Just a Wild Shot for Cinematic Transfer in Neural Radiance Fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 16933–16942.

[44] Yuelei Wang, Jian Zhang, Pengtao Jiang, Hao Zhang, Jinwei Chen, and Bo Li. 2024. CPA: Camera-pose-awareness Diffusion Transformer for Video Generation. *arXiv:2412.01429* (2024).

[45] 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.

[46] Yuxi Xiao, Jianyuan Wang, Nan Xue, Nikita Karaev, Yuri Makarov, Bingyi Kang, Xing Zhu, Hujun Bao, Yujun Shen, and Xiaowei Zhou. 2025. SpatialTrackerV2: Advancing 3D Point Tracking with Explicit Camera Motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6726–6737. doi:10.1109/ICCV51701.2025.00633

[47] Pei Yang, Hai Ci, Yiren Song, and Mike Zheng Shou. 2025. X-Humanoid: Robotize Human Videos to Generate Humanoid Videos at Scale. arXiv:2512.04537 [cs.CV] https://arxiv.org/abs/2512.04537

[48] Qijun Ying, Zhongyuan Hu, Rui Zhang, Ronghui Li, Yu Lu, and Zijiao Zeng. 2025. WATCH: World-Aware Allied Trajectory and Pose Reconstruction for Camera and Human. arXiv:2509.04600 [cs.CV]

[49] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Chenming Zhu, Zhangyang Xiong, Tianyou Liang, Guanying Chen, Shuguang Cui, and Xiaoguang Han. 2023. MVImgNet: A Large-scale Dataset of Multi-view Images. In *CVPR*. IEEE, 9150–9161.

[50] Mengchen Zhang, Tong Wu, Jing Tan, Ziwei Liu, Gordon Wetzstein, and Dahua Lin. 2025. GenDoP: Auto-Regressive Camera Trajectory Generation as a Director of Photography. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 18229–18239. doi:10.1109/ICCV51701.2025.01694

[51] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. 2018. Stereo magnification: learning view synthesis using multiplane images. *ACM Trans. Graph.* 37, 4, Article 65 (2018), 12 pages. doi:10.1145/3197517.3201323

[52] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. 2019. On the continuity of rotation representations in neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5745–5753.