ReCamMaster: Camera-Controlled Generative Rendering from A Single Video

Jianhong Bai[1*], **Menghan Xia**[2†], **Xiao Fu**[3], **Xintao Wang**[2], **Lianrui Mu**[1], **Jinwen Cao**[2], **Zuozhu Liu**[1], **Haoji Hu**[1†], **Xiang Bai**[4], **Pengfei Wan**[2], **Di Zhang**[2]
[1]Zhejiang University, [2]Kling Team, Kuaishou Technology, [3]CUHK, [4]HUST

Project webpage: https://jianhongbai.github.io/ReCamMaster/

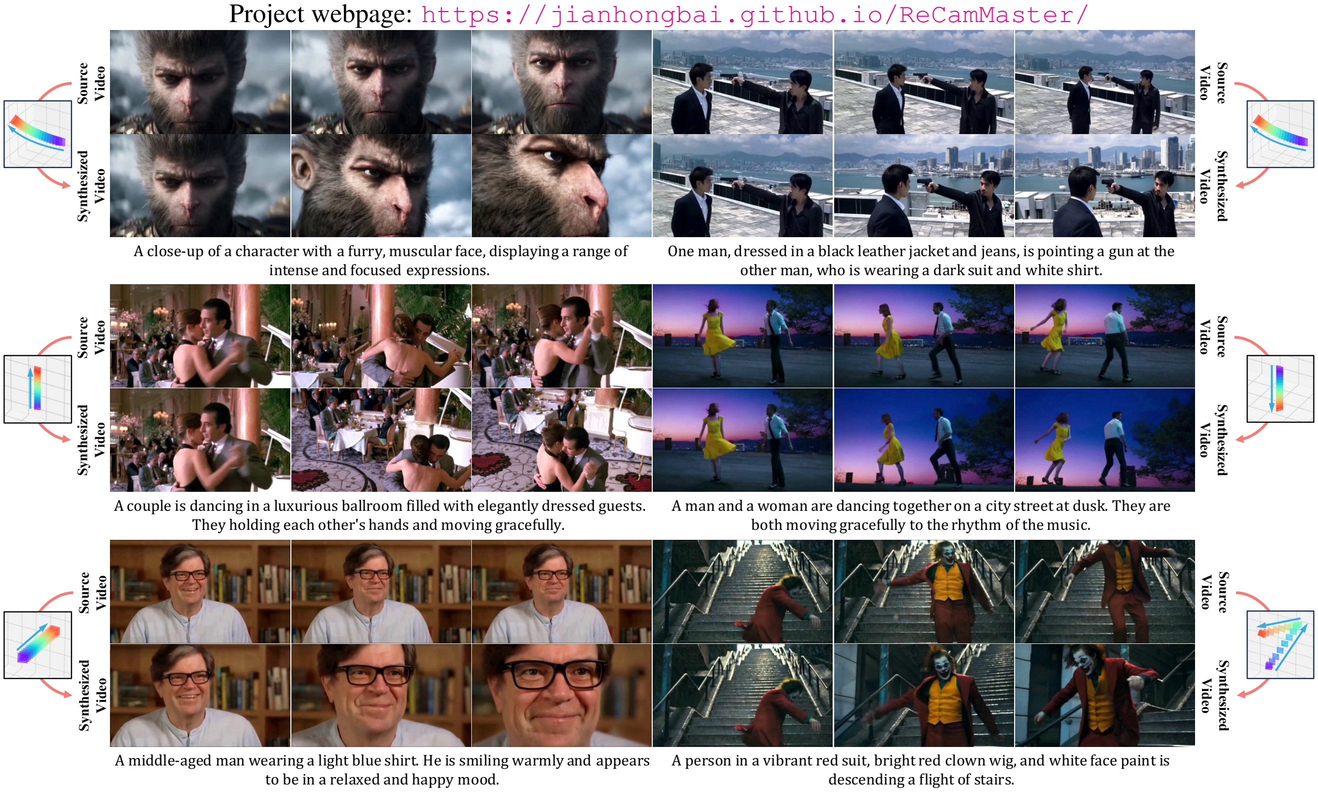

Figure 1. **Examples synthesized by ReCamMaster.** ReCamMaster re-shoots the source video with novel camera trajectories. We visualized the novel camera trajectories alongside the video frames. Video results are on our project page.

Abstract

Camera control has been actively studied in text or image conditioned video generation tasks. However, altering camera trajectories of a given video remains under-explored, despite its importance in the field of video creation. It is non-trivial due to the extra constraints of maintaining multiple-frame appearance and dynamic synchronization. To address this, we present ReCamMaster, a camera-controlled generative video re-rendering framework that reproduces the dynamic scene of an input video at novel camera trajectories. The core innovation lies in harnessing the generative capabilities of pre-trained text-to-video models through a simple yet powerful video conditioning mechanism—its capability is often overlooked in current research. To overcome the scarcity of qualified training data, we construct a comprehensive multi-camera synchronized video dataset using Unreal Engine 5, which is carefully curated to follow real-world filming characteristics, covering diverse scenes and camera movements. It helps the model generalize to in-the-wild videos. Lastly, we further improve the robustness to diverse inputs through a meticulously designed training strategy. Extensive experiments show that our method substantially outperforms existing state-of-the-art approaches. Our method also finds promising applications in video stabilization, super-resolution, and outpainting. Our code and dataset are publicly available at: https://github.com/KwaiVGI/ReCamMaster.

∗ Work done during an internship at Kling Team, Kuaishou Tech.
† Corresponding authors.

1. Introduction

Camera movement is a fundamental element in film production, profoundly shaping the audience's visual experience and conveying both emotional depth and narrative intentions. For instance, a dolly-in shot can masterfully emphasize a character, building tension or guiding viewers' attention to crucial details. Similarly, crane shots excel at showcasing expansive landscapes, often employed at the opening to dramatically reveal the full scope of a setting. Despite its artistic significance, achieving professional-level camera movement remains a challenge for amateur videographers, because of hardware limitations (such as stabilization issues in handheld recordings) and technical skill gaps. To address this dilemma, we explore innovative approaches to modify camera trajectories in post-production, empowering creators with the ability to enhance their original footage by displaying the dynamic scenes with more compelling and polished camera trajectories when needed.

Previous research on camera-controlled generation has primarily focused on text-to-video (T2V) or image-to-video (I2V) generation [18, 51]. Recently, GCD [45] pioneered camera-controlled video-to-video generation, achieving promising results on domain-specific videos synthesized by the Kubric simulator [14]. However, its effectiveness on real-world videos is limited due to the narrow domain of training data and its inferior video conditioning mechanism. As a concurrent work, ReCapture [65] introduces a novel two-stage approach: first generating a camera-controlled anchor video using existing image-level multiview diffusion models, then optimizing the result by fine-tuning spatial and temporal LoRA [21] layers over the input and anchor videos respectively. While showing promising performance, its requirement for per-video optimization constrains practical applications. Similarly, other approaches [5, 15] achieve video re-rendering through explicit 4D reconstruction followed by video post-optimization. However, their performance is significantly restricted by the challenge of single video-based 4D reconstruction techniques [26, 53].

To address these challenges, we present ReCamMaster, a camera-controlled generative video re-rendering framework that can regenerate in-the-wild videos at novel camera trajectories. The core innovation is to utilize the generative capabilities of pre-trained text-to-video models through an elegant yet powerful video conditioning mechanism. Compared to alternatives, our method shows superior performance in preserving both the visual and dynamic characteristics of the input video. Notably, this conditioning mechanism indeed exhibits remarkable potential as a versatile solution for conditional generation tasks, which is overlooked in current research. Since no qualified training data are publicly available, we develop a large-scale multi-camera synchronized video dataset using Unreal Engine 5, which contains 136K realistic videos shot from 13.6K different dynamic scenes in 40 high-quality 3D environments with 122K different camera trajectories. Particularly, it is carefully curated to simulate real-world filming characteristics, which is proven to bring an advantage to striking domain alignment to in-the-wild videos. Lastly, we further improve the robustness of our model to diverse inputs through a meticulously designed training strategy.

Experimental results show that our method outperforms state-of-the-art approaches and strong baselines by a large margin. Ablation studies verified the effectiveness of our key designs. Furthermore, ReCamMaster demonstrates its potential in scenarios like video stabilization, video super-resolution, and video outpainting. Our contribution can be summarized as follows:

- We introduce a high-quality multi-camera synchronized video dataset featuring diverse camera trajectories. This dataset has been publicly released to advance research in camera-controlled video generation, 4D reconstruction, and related fields.
- We conduct an in-depth investigation and validation of an effective video conditioning mechanism for text-to-video generation models. It significantly outperforms other alternatives employed in baseline methods.
- Extensive experiments show that our proposed ReCamMaster significantly advances the state-of-the-art in video recapturing. Furthermore, it finds promising applications across multiple real-world scenarios.

2. Related Works

Camera-Controlled Video Generation. With the success of text-to-video generation models [6, 7, 10, 35, 62], the introduction of other conditional signals for controllable video generation has been widely studied [11, 16, 56, 63]. In camera-controlled video generation [2, 61, 67, 68], researchers aim to incorporate camera parameters into video generation models to control the viewpoint of the output video. AnimateDiff [17] introduces various motion LoRAs [21] to learn specific patterns of camera movements. MotionCtrl [51] encodes 6DoF camera extrinsics and injects them into the diffusion model, fine-tuning on video-camera pair data to achieve video generation with arbitrary trajectory control. CameraCtrl [18] further improves the accuracy and generalizability of single-sequence camera control with a dedicatedly designed camera encoder. CVD [28] achieves multi-sequence camera control with the proposed cross-video synchronization module. AC3D [1] conducts an in-depth investigation into camera motion knowledge within diffusion transformers, achieving camera-controlled generation with enhanced visual quality. Meanwhile, training-free methods have also been explored [20, 22, 30].

Video-to-Video Generation. Video-to-video generation [34, 48, 49] is explored in various tasks, including video editing [33, 60], video outpainting [8, 46], video super-resolution [58, 69], etc. Most relevant to our work is synthesizing novel viewpoint videos based on a given video and camera parameters [5, 15, 39, 45, 65]. GCD [45] pioneered camera-controlled video-to-video generation by training a video generation model with paired video data created by the Kubric simulator [14]. Although it performs well on in-domain data, GCD's generalization capability is limited by the significant domain gap between the training data and real-world videos. Recapture [65] proposes a practical solution with per-video optimization. Several concurrent works [5, 15, 39, 54, 64], achieve 4D-consistent video-to-video generation by extracting dynamic information from the original video using 3D point tracking [26, 53] and incorporating it as a condition into the video generator. This paradigm is promising when synchronized multi-view videos are unavailable. Nevertheless, the generation quality is limited by the accuracy of the point-tracking methods. In the field of 4D object generation, video-to-video generation can be achieved by 1) training a multi-view video generator [52, 55], or 2) following a reconstruct-and-render pipeline via 4D reconstruction methods [29, 66]. [32, 42] achieve 3D/4D scene reconstruction with videos generated with diffusion models. In this paper, we focus on open-domain camera-controlled video generation. To achieve this, we first create a large multi-camera synchronized dataset with diverse camera trajectories. We also propose a novel video conditioning method to achieve better generation ability.

3. Multi-Cam Video: A High-Quality Multi-Camera Synchronized Video Dataset

To re-shoot input videos based on novel camera trajectories, paired video data is required for training the video generation model. Specifically, the training data should include multiple shots captured in the same scene simultaneously, allowing the model to learn 4D consistent generation. Acquiring such data in real-world scenarios is extremely costly, and publicly available multi-view synchronized datasets [13, 24, 25, 40, 59] are also limited by the diversity of scenes and constrained camera movements, making them unsuitable for our task. Therefore, we chose to use a rendering engine to generate the training data. The advantages of this approach are: 1) precise camera trajectories can be obtained, 2) complete synchronization in the time dimension can be achieved, and 3) the data volume can be more easily scaled up.

We build the entire data rendering pipeline in Unreal Engine 5 [12]. Specifically, we first collect multiple 3D environments as "backgrounds". Then, we place animated characters within these environments as the "main subjects" of the videos. We then position multiple cameras facing the subjects and moving along predefined trajectories to simulate the process of simultaneous shooting. This allows us to

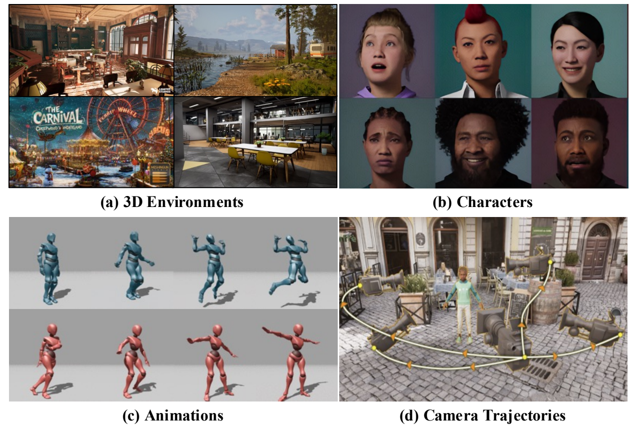

Figure 2. **Illustration of the dataset construction process.** We build the multi-camera synchronized training dataset by rendering in Unreal Engine 5. This is achieved using 3D environments, characters, animations collected from the internet, and our designed massive camera trajectories.

render datasets with synchronized cameras that include dynamic objects. To scale up the data volume, we construct a set of camera movement rules for automatically batch generation of natural and diverse camera trajectories. Additionally, we randomly combined different characters and actions across different video sets. In total, we obtain 136K visually-realistic videos shot from 13.6K different dynamic scenes in 40 high-quality 3D environments with 122K different camera trajectories. Fig. 8 in the Appendix shows some of the rendered video frames. In Appendix C.1, we conduct an ablation study comparing model performance when trained on a dataset with limited scenes and camera trajectories versus our comprehensive dataset. This highlights the significance of our high-quality dataset. For more details about the dataset, please refer to Appendix B.

4. Camera-Controlled Video Re-Generation

Given a source video $V_s \in \mathbb{R}^{f\times c\times h\times w}$, we aim to synthesize a target video $V_t \in \mathbb{R}^{f\times c\times h\times w}$ sharing the same dynamic scene but presenting at specified camera trajectories denoted by $\mathtt{cam}_t := [R, t] \in \mathbb{R}^{f\times 3\times 4}$. Particularly, V_t should conform to the multiple-frame appearance and synchronized dynamics of V_s. To achieve this, we propose to harness the generative capability of pre-trained text-to-video diffusion models [7, 62] by imposing dual conditions, *i.e.*, the source video and target camera trajectories through a meticulously designed framework. The overview of the model is depicted in Fig. 3.

4.1. Preliminary: Text-to-Video Base Model

Our study is conducted over an internal pre-trained text-to-video foundation model. It is a latent video diffusion model, consisting of a 3D Variational Auto-Encoder (VAE) [27] and a Transformer-based diffusion model (DiT) [37]. Typ-

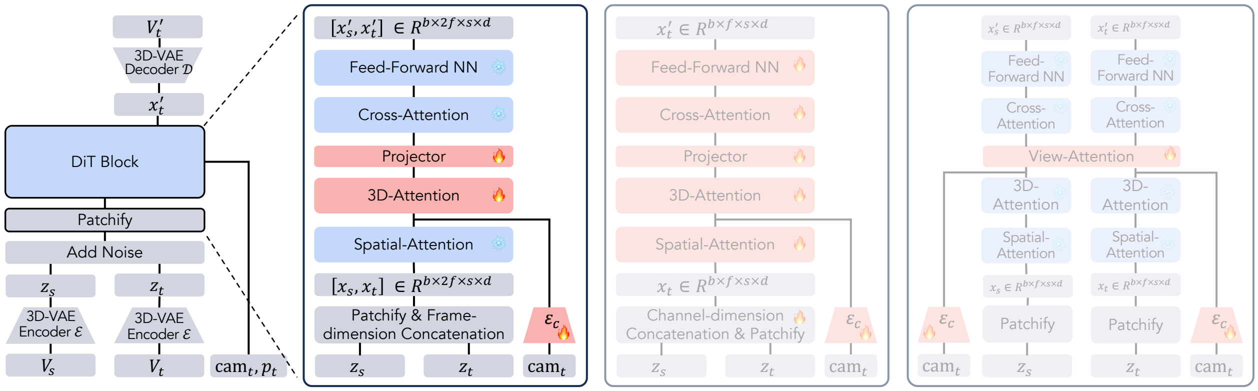

Figure 3. **Overview of ReCamMaster.** *Left:* The training pipeline of ReCamMaster. A latent diffusion model is optimized to reconstruct the target video V_t, conditioned on the source video V_s, target camera pose cam_t, and target prompt p_t. *Right:* Comparison of different video condition techniques. (a) Frame-dimension conditioning used in our paper; (b) Channel-dimension conditioning used in baseline methods [5, 45]; (c) View-dimension conditioning in [3]. We omit the text prompt p_t in (a)-(c) for simplicity.

ically, each Transformer block is instantiated as a sequence of spatial attention, 3D (spatial-temporal) attention, and cross-attention modules. The generative model adopts Rectified Flow framework [9] for the noise schedule and denoising process. The forward process is defined as straight paths between data distribution and a standard normal distribution, i.e.

$$z_t = (1-t)z_0 + t\epsilon, \tag{1}$$

where $\epsilon \in \mathcal{N}(0, I)$ and t denotes the iterative timestep. To solve the denoising processing, we define a mapping between samples z_1 from a noise distribution p_1 to samples z_0 from a data distribution p_0 in terms of an ordinary differential equation (ODE), namely:

$$dz_t = v_\Theta(z_t, t)dt, \tag{2}$$

where the velocity v is parameterized by the weights Θ of a neural network. For training, we regress a vector field u_t that generates a probability path between p_0 and p_1 via Conditional Flow Matching [31]:

$$\mathcal{L}_{LCM} = \mathbb{E}_{t,p_t(z,\epsilon),p(\epsilon)}||v_\Theta(z_t,t) - u_t(z_0|\epsilon)||_2^2, \tag{3}$$

where $u_t(z,\epsilon) := \psi_t'(\psi_t^{-1}(z|\epsilon)|\epsilon)$ with $\psi(\cdot|\epsilon)$ denotes the function of Eq. 1. For inference, we employ Euler discretization for Eq. 2 and perform discretization over the timestep interval at $[0,1]$, starting at $t=1$. We then processed with iterative sampling with:

$$z_t = z_{t-1} + v_\Theta(z_{t-1}, t) * \Delta t. \tag{4}$$

4.2. Conditional Video Injection Mechanism

To modify the camera trajectories of a given video using generative models, it is essential to condition the generation process on the source video. Our experiments reveal that the video injection mechanism is crucial for overall performance. We analyze and compare several widely-adopted video conditioning approaches from previous works [3, 5, 45], alongside our proposed simple yet powerful video injection mechanism. We provide an intuitive explanation of our design's superior performance and its potential impact.

Channel Dimension Conditioning. Recent attempts [5, 45] incorporate the source video by concatenating the latent of the source video x_s with the noised latent of the target video x_t along the channel dimension, and add additional input channels to the input convolutional layer. To be specific, a variational autoencoder with encoder $\mathcal{E}$ is utilized to project the source video V_s and target video V_t to the latent space, $z_s = \mathcal{E}(V_s), z_t = \mathcal{E}(V_t)$, where $z_s, z_t \in \mathbb{R}^{b\times f\times c\times h\times w}$ are the latent of source video and target video with f frames, c channels, and spatial size of $h \times w$. Then, the source latent is concentrated with the noised target latent along the channel dimension, and patchified into $f \times h \times w$ tokens with several convolutional layers:

$$x_t = \texttt{patchify}([z_s, z_t]_{\text{channel-dim}}), \tag{5}$$

where $x_t \in \mathbb{R}^{b\times f\times s\times d}$, $s = h \times w$, d is the channel dimension for the latent diffusion model. We denote this conditioning technique as the "channel-dimension conditioning" and illustrate in Fig. 3(b).

View Dimension Conditioning. To achieve multi-view video generation, [3] proposes to introduce a plug-and-play module for feature aggregation across views. Given a source video latent $z_s = \mathcal{E}(V_s)$ and a target video latent $z_t = \mathcal{E}(V_t)$, and patchify and forward to the diffusion transformer respectively:

$$x_s = \texttt{patchify}(z_s), x_t = \texttt{patchify}(z_t), \tag{6}$$

where $x_s, x_t \in \mathbb{R}^{b\times f\times s\times d}$. To achieve synchronization and content consistency across views, an additional attention layer (denoted as view-attention) is added in each basic transformer block, it performs self-attention between each frame in multiple views:

$$\overline{F}_s^i, \overline{F}_t^i = \texttt{attn\_view}(F_s^i, F_t^i), \tag{7}$$

where F_s^i and F_t^i are the i-th frame feature of V_s and V_t respectively, and $\overline{F}_s^i$ and $\overline{F}_t^i$ are the output features. We denote this conditioning technique as the "view-dimension conditioning" and illustrated in Fig. 3(c).

Frame Dimension Conditioning (ours). To achieve better synchronization and content consistency with the source video, we propose to concatenate the source video tokens with the target video tokens along the frame dimension:

$$\begin{cases} x_s = \texttt{patchify}(z_s), \quad x_t = \texttt{patchify}(z_t), \\ x_i = [x_s, x_t]_{\text{frame-dim}}, \end{cases} \tag{8}$$

where $x_i \in \mathbb{R}^{b\times 2f\times s\times d}$ is the input of diffusion transformer. In other words, the input token number is doubled compared to the vanilla text-to-video generation process. Moreover, we do not introduce additional attention layers for feature aggregation across the source video and the target video, since self-attention is performed between all tokens in the 3D (spatial-temporal) attention layers. We denote this conditioning technique as the "frame-dimension conditioning" and illustrate in Fig. 3(a).

Comparison and Discussion. Our experimental results demonstrate that the frame dimension conditioning approach provides substantial advantages in utilizing condition information effectively, as illustrated by the comparative analysis in Fig. 5. We attribute this superior performance to the inherent flexibility of our frame dimension conditioning method, which enables spatio-temporal interaction between the conditional tokens and target tokens through all blocks of the base model. This approach provides a more robust mechanism for understanding the correlations between video pairs. While similar token concatenation strategies have proven effective in image-to-image tasks [43], their potential in video controllable generation has remained largely unexplored. Our study provides compelling evidence of the effectiveness, highlighting the potential of this underappreciated technique as a versatile solution for conditional generation tasks.

4.3. Camera Pose Conditioning

To achieve camera-controlled video generation, a natural idea is to condition the model on the source and target video camera trajectories $\texttt{cam}_s$ and $\texttt{cam}_t$ to help the model better understand the 4D space. Unfortunately, during inference, even with state-of-the-art structure-from-motion (SfM) methods for camera parameter estimation, it is challenging to obtain accurate camera trajectory information for the input video [1]. Therefore, we alternate to only conditioning the model on the target camera $\texttt{cam}_t$, while relying on the model to interpret the camera trajectory of the input video. Furthermore, we condition the model using camera extrinsics, specifically the rotation and translation matrices for each frame's camera. We chose not to include camera intrinsics as a condition because, while they are easily obtainable in the rendered training set, accurately estimating the intrinsics of real-world video cameras remains challenging. This limitation would complicate parameter provision in practical applications, as users are unlikely to have access to the source video's camera intrinsics. However, our method can be readily adapted to incorporate intrinsics as input with minimal modifications.

Given a f frames target camera sequence $\texttt{cam}_t \in \mathbb{R}^{f\times(3\times 4)}$, we project it to have the same channels with the video tokens through a learnable camera encoder $\mathcal{E}_c$ and add it to the visual features:

$$F_i = F_o + \mathcal{E}_c(\texttt{cam}_t), \tag{9}$$

where F_o is the output feature of the spatial-attention layer, F_i is the input feature of the 3D-attention layer, $\mathcal{E}_c$ is instantiated as a fully connected layer with an input dimension of 12 and an output dimension of d in practice. The camera encoder is inserted into each transformer block for fine-grained camera control.

4.4. Training Strategy

Enhancing Generalization Capabilities. Using the annotated dataset described in Sec. 3, ReCamMaster is trained to process input videos with target camera parameters. To preserve the base T2V model's native capability, we finetune only the camera encoder and 3D-attention layers while keeping other parameters frozen. To mitigate Unreal Engine's synthetic characteristics, we apply moderate noise (200-500 noise scheduling steps) to the conditional video latent during training, reducing the domain gap between synthetic and real-world data at inference.

Improving Generation Capability by Unifying Camera Control Tasks. To encourage content generation capability, we implement T2V camera-controlled generation [18, 51] with a 20% probability and I2V camera-controlled generation [57] with a 20% probability during training. Specifically, we replace the latent representations of all f frames with Gaussian noise for T2V generation, and replace $f-1$ frames starting from the second frame for I2V generation. Our experiments demonstrate that this strategy prompts performance in synthesizing coherent objects invisible in the source video. As a byproduct, our model

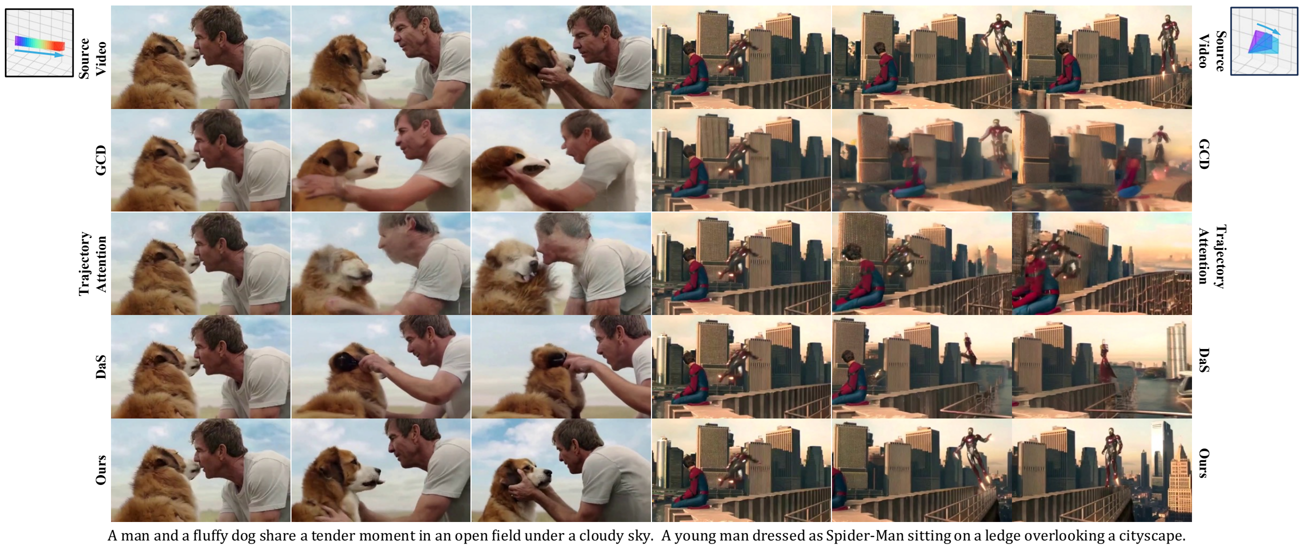

Figure 4. **Comparison with state-of-the-art methods.** It shows that ReCamMaster generates videos that maintain appearance consistency and temporal synchronization with the source video.

inherently supports T2V, I2V, and V2V camera-controlled generation, as shown in Fig. 9.

5. Experimental Results

5.1. Experiment Settings

Implementation Details. We train ReCamMaster on the rendered dataset introduced in Sec. 3. During training, we randomly select 2 cameras from each 10 synchronized cameras, designating one as the source video and the other as the target video. We use {source video V_s, target camera $\mathtt{cam}_t$, target prompt p_t} as conditions input to the model, to reconstruct the target video V_t as introduced in Eq. 3. We train the model for 10K steps at the resolution of 384x672 with a learning rate of 0.0001, batch size 40. The camera encoder and the projector are zero-initialized.

Evaluation Metrics. We mainly evaluate the proposed method in terms of camera accuracy, source-target synchronization, and visual quality. For camera accuracy, we use GLOMAP [36] to extract the camera pose sequence of the generated videos, and calculate the rotation error and translation error, denoted as RotErr and TransErr respectively [18]. In terms of synchronization, we utilize the state-of-the-art image matching method GIM [41] to calculate the number of matching pixels with confidence greater than the threshold, denoted as Mat. Pix.. Furthermore, we calculate the FVD-V score in SV4D [55], and the average CLIP similarity between source and target frames at the same timestamp, denoted as CLIP-V [28]. For visual quality, we divide it into fidelity, coherence with text, and temporal consistency, and quantify them with Fréchet Image Distance [19] (FID) and Fréchet Video Distance [44] (FVD), CLIP-T, and CLIP-F, respectively. CLIP-T refers to the average CLIP [38] similarity of each frame and its corresponding text prompt, and CLIP-F is the average CLIP similarity of adjacent frames. We also evaluate our method on the widely used VBench [23] metrics.

Evaluation Set. We construct the evaluation set with 1000 random videos from WebVid [4] and 10 different camera trajectories, including pan, tilt, vertical translation, zoom in, zoom out, and horizontal arc trajectories. Note that since we automatically generated 122K different camera trajectories when constructing the training set, ReCamMaster can support a variety of camera trajectories as input. However, since the baseline method was trained on specific basic trajectories, we opted to use trajectories that closely align with the baseline for comparison.

5.2. Comparison with State-of-the-Art Methods

Baselines. We compare the proposed ReCamMaster with state-of-the-art camera-controlled video-to-video generation methods [15, 45, 54]. GCD [45] pioneered camera-controlled video-to-video generation by training a video generation model with paired data created by the Kubric simulator [14]. Although it performs well on in-domain data, GCD's generalization capability is limited by its inferior video conditioning method and the significant domain gap between the training data and real-world videos. Trajectory-Attention [54] and DaS [15] extract dynamic information from the source video using 3D point tracking and incorporate it as a condition into the video generator.

Table 1. Quantitative comparison with state-of-the-art methods on visual quality, camera accuracy, and view synchronization.

Method	Visual Quality				Camera Accuracy		View Synchronization		
	FID ↓	FVD ↓	CLIP-T ↑	CLIP-F ↑	RotErr ↓	TransErr ↓	Mat. Pix.(K) ↑	FVD-V ↓	CLIP-V ↑
GCD	72.83	367.32	32.86	95.66	2.27	5.51	639.39	365.75	85.92
Trajectory-Attention	69.21	276.06	33.43	96.52	2.18	5.32	619.13	256.30	88.65
DaS	63.25	159.60	33.05	98.32	1.45	5.59	633.53	154.25	87.33
ReCamMaster	**57.10**	**122.74**	**34.53**	**98.74**	**1.22**	**4.85**	**906.03**	**90.38**	**90.36**

Table 2. Quantitative comparison with state-of-the-art methods on VBench [23] metrics.

Method	Aesthetic Quality ↑	Imaging Quality ↑	Temporal Flickering ↑	Motion Smoothness ↑	Subject Consistency ↑	Background Consistency ↑
GCD	38.21	41.56	95.81	98.37	88.94	92.00
Trajectory-Attention	38.50	51.00	95.52	98.21	90.60	92.83
DaS	39.86	51.55	**97.44**	99.14	90.34	92.03
ReCamMaster	**42.70**	**53.97**	97.36	**99.28**	**92.05**	**93.83**

Qualitative Results. We present synthesized examples of ReCamMaster in Fig. 1 (additional examples in Fig. 12 of Appendix C). Please visit our project page for more videos. ReCamMaster demonstrates the ability to: 1) generate consistent content from novel camera trajectories of the same scene; 2) achieve excellent temporal synchronization with the source video; and 3) generate plausible video content in areas not visible in the original video. Note that, due to the use of 122K different random trajectories during training, ReCamMaster also supports complex trajectories as input, such as zigzag paths. Please refer to the results on the project page. In the paper, to visually capture the camera movement trajectory more clearly, we do not showcase results generated along complex trajectories.

We compare ReCamMaster with state-of-the-art methods in Fig. 4 and Fig. 13. It's observed that baselines produce content with notable artifacts and temporal desynchronization. Similar phenomena were noted when training ReCamMaster with other video conditioning techniques (as shown in Fig. 5), highlighting the importance of the novel video conditioning method proposed in the paper.

Quantitative Results. We quantitatively evaluate ReCamMaster against baselines using various automatic metrics, the summarized results are in Tab. 1 and 2. For visual quality assessment, we used the widely adopted FID, FVD, and CLIP metrics as shown in Tab. 1, along with the VBench metrics in Tab. 2. In terms of camera trajectory accuracy, we calculate the rotation error and translation error by following previous work in camera-controlled T2V and I2V generation [18, 57]. For view synchronization, we calculate the clip similarity score and FVD between video frames of different viewpoints within one scene, denoted as CLIP-V and FVD-V. It's observed that ReCamMaster outperforms baselines across multiple dimensions of metrics.

Figure 5. **Ablation on video conditioning techniques.** We compared the channel-/view- concatenation schemes proposed by previous methods and frame-concatenation in ReCamMaster. We observed that both channel-conditioning and view-conditioning schemes suffer from significant artifacts, content inconsistency, and asynchronous dynamics with respect to the original video.

5.3. More Analysis and Ablation Studies

Ablation on Video Conditioning Techniques. In Sec. 4.2, we introduce a novel video conditioning scheme that concatenates the tokens of a source video with the target video tokens along the frame dimension. To verify its effectiveness, we compare our "frame concatenation" technique with the "channel concatenation" used in baseline methods [5, 45] and the "view concatenation" in [3]. We performed both qualitative and quantitative comparisons, as shown in Fig. 5 and Tab. 3. We only altered the video conditioning method while keeping the camera injection method, training data, and training steps unchanged. The results clearly show that our conditioning technique significantly enhances the model's performance. In the second and third rows of Fig. 5, the generated videos exhibit the rendering style of the

Table 3. Quantitative comparison on the video conditioning strategy.

Method	Visual Quality				Camera Accuracy		View Synchronization		
	FID ↓	FVD ↓	CLIP-T ↑	CLIP-F ↑	RotErr ↓	TransErr ↓	Mat. Pix.(K) ↑	FVD-V ↓	CLIP-V ↑
Channel dim.	74.09	187.94	33.70	98.67	1.28	4.98	521.10	148.51	84.62
View dim.	80.51	194.47	**34.66**	98.71	1.42	5.77	573.92	177.68	83.40
Frame dim. (ours)	**57.10**	**122.74**	34.53	**98.74**	**1.22**	**4.85**	**906.03**	**90.38**	**90.36**

Table 4. Ablation on our training strategies.

Method	FID ↓	FVD ↓	Aesthetic Quality ↑	Imaging Quality ↑
Baseline	66.67	171.80	40.02	51.93
+ Add noise	65.17	164.04	40.36	52.22
+ 3D-Attn. tuning	59.47	132.58	**43.08**	52.80
+ Drop latent	62.39	149.52	41.47	52.65
+ All	**57.10**	**122.74**	42.70	**53.97**

training dataset, and the hand movements become unsynchronized with the original video. In contrast, our method preserves the person's identity and maintains synchronization even during rapid and complex movements.

The Effectiveness of the Training Strategies. In Section 4.4, we enhance ReCamMaster's robustness with improved training strategies. We fine-tune only the spatial-temporal (3D) attention layers and freeze other parameters. Additionally, we drop f frames and $f-1$ frames of the source video with a 20% probability to improve generation capability. Table 4 shows the improvement using visual quality metrics. The 'baseline' is vanilla training with reconstruction loss. Results indicate that all techniques improve visual quality, with their combination yielding the best performance. Dropping the source latent during training also allows our method to support camera-controlled T2V, I2V, and V2V generation simultaneously, as shown in Fig. 9.

6. Applications of ReCamMaster

We find that ReCamMaster also has promising performances in several traditional tasks, as shown in Fig. 6.

Video Stabilization. It is usually not easy to obtain stable videos when recording video while moving. Video stabilization techniques [50] aim to smooth out camera movements to produce easy-to-watch videos, which can be achieved by adjusting the camera trajectories via ReCamMaster. We evaluate our method using unsteady videos from the DeepStab [47] dataset. It can be observed that the model stabilizes the video while preserving the content from the original video. Best viewed on the project page.

Video Super-Resolution.[1] Thanks to the generation capability of the diffusion model, we can input 'zoom-in' camera trajectories into ReCamMaster to achieve video local super-resolution, as more details are observed in Fig. 6.

[1] It's not entirely equivalent to the super-resolution task, as ReCamMaster only enhances the resolution of patches in the central region.

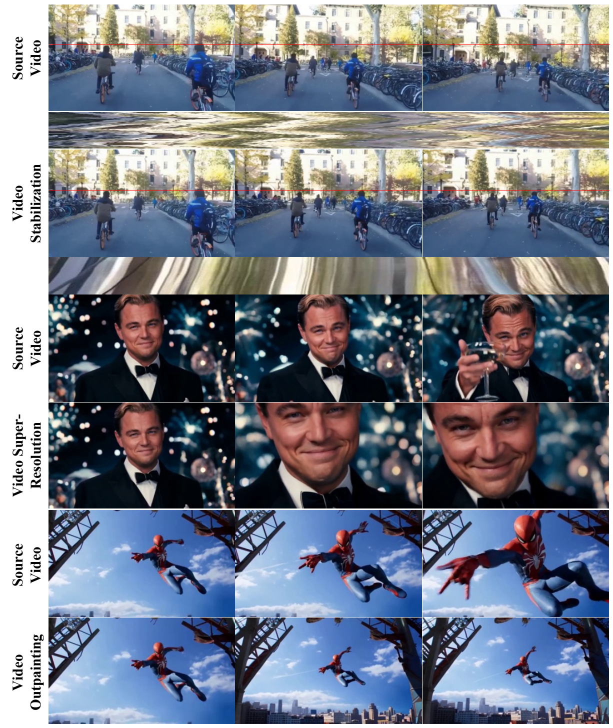

Figure 6. **Applications of ReCamMaster.** *From top to bottom:* video stabilization, video super-resolution, and video outpainting.

Video Outpainting. Similarly, we can input zoom-out trajectories to achieve video outpainting. As shown in the last row of Fig. 6 that areas not visible in the original video, such as feet and the ground, have been generated.

7. Conclusion and Limitations

In this paper, we propose ReCamMaster to reproduce dynamic scenes from input videos with new camera trajectories. We develop an innovative video conditioning technique to enhance pre-trained text-to-video models and curate a large-scale multi-camera synchronized video dataset using Unreal Engine 5, covering diverse scenes and camera movements. Our method also shows promise in video stabilization, super-resolution, and outpainting. There are nevertheless some limitations. Concatenating source and target video tokens improves generation quality but increases computational demands. Additionally, ReCamMaster inherits limitations from the pre-trained T2V models, such as less effective hand generation, as shown in Fig. 11.

Acknowledgments

We thank Jinwen Cao, Yisong Guo, Haowen Ji, Jichao Wang, and Yi Wang from Kuaishou Technology for their invaluable help in constructing the multi-view video dataset. Jianhong Bai would like to thank Quande Liu and Xiaoyu Shi for fruitful discussions. We thank Qinghe Wang and Yawen Luo for their help in method evaluation.

This work is supported by the National Natural Science Foundation of China (Grant No. 62441615, 62225603, U21B2004, 62476241), the Natural Science Foundation of Zhejiang Province, China (Grant No. LZ23F020008), and the Zhejiang University-Angelalign Inc. R&D Center for Intelligent Healthcare.

References

[1] 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. *arXiv preprint arXiv:2411.18673*, 2024.

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

[3] Jianhong Bai, Menghan Xia, Xintao Wang, Ziyang Yuan, Xiao Fu, Zuozhu Liu, Haoji Hu, Pengfei Wan, and Di Zhang. Syncammaster: Synchronizing multi-camera video generation from diverse viewpoints. *arXiv preprint arXiv:2412.07760*, 2024.

[4] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *IEEE International Conference on Computer Vision*, 2021.

[5] Weikang Bian, Zhaoyang Huang, Xiaoyu Shi, Yijin Li, Fu-Yun Wang, and Hongsheng Li. Gs-dit: Advancing video generation with pseudo 4d gaussian fields through efficient dense 3d point tracking. *arXiv preprint arXiv:2501.02690*, 2025.

[6] 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.

[7] 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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7310–7320, 2024.

[8] Qihua Chen, Yue Ma, Hongfa Wang, Junkun Yuan, Wenzhe Zhao, Qi Tian, Hongmei Wang, Shaobo Min, Qifeng Chen, and Wei Liu. Follow-your-canvas: Higher-resolution video outpainting with extensive content generation. *arXiv preprint arXiv:2409.01055*, 2024.

[9] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning (ICML)*, 2024.

[10] Weijie Kong et. al. Hunyuanvideo: A systematic framework for large video generative models, 2024.

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

[12] Epic Games. Unreal engine 5. https://www.unrealengine.com/en-US/unreal-engine-5, 2022.

[13] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19383–19400, 2024.

[14] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti (Derek) Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: a scalable dataset generator. 2022.

[15] 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. *arXiv preprint arXiv:2501.03847*, 2025.

[16] Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Sparsectrl: Adding sparse controls to text-to-video diffusion models. *arXiv preprint arXiv:2311.16933*, 2023.

[17] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

[18] 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.

[19] 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.

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

[21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[22] 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.

[23] 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.

[24] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013.

[25] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *Proceedings of the IEEE international conference on computer vision*, pages 3334–3342, 2015.

[26] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *European Conference on Computer Vision*, pages 18–35. Springer, 2024.

[27] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

[28] Zhengfei Kuang, Shengqu Cai, Hao He, Yinghao Xu, Hongsheng Li, Leonidas Guibas, and Gordon Wetzstein. Collaborative video diffusion: Consistent multi-video generation with camera control. *arXiv preprint arXiv:2405.17414*, 2024.

[29] 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. *arXiv preprint arXiv:2412.04463*, 2024.

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

[32] Fangfu Liu, Wenqiang Sun, Hanyang Wang, Yikai Wang, Haowen Sun, Junliang Ye, Jun Zhang, and Yueqi Duan. Reconx: Reconstruct any scene from sparse views with video diffusion model. *arXiv preprint arXiv:2408.16767*, 2024.

[33] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8599–8608, 2024.

[34] Arun Mallya, Ting-Chun Wang, Karan Sapra, and Ming-Yu Liu. World-consistent video-to-video synthesis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, pages 359–378. Springer, 2020.

[35] Willi Menapace, Aliaksandr Siarohin, Ivan Skorokhodov, Ekaterina Deyneka, Tsai-Shien Chen, Anil Kag, Yuwei Fang, Aleksei Stoliar, Elisa Ricci, Jian Ren, et al. Snap video: Scaled spatiotemporal transformers for text-to-video synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7038–7048, 2024.

[36] Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L Schönberger. Global structure-from-motion revisited. In *European Conference on Computer Vision*, pages 58–77. Springer, 2024.

[37] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[38] 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.

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

[40] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016.

[41] Xuelun Shen, Zhipeng Cai, Wei Yin, Matthias Müller, Zijun Li, Kaixuan Wang, Xiaozhi Chen, and Cheng Wang. Gim: Learning generalizable image matcher from internet videos. In *The Twelfth International Conference on Learning Representations*, 2024.

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

[43] Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and universal control for diffusion transformer. *arXiv preprint arXiv:2411.15098*, 2024.

[44] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Fvd: A new metric for video generation. *Openreview*, 2019.

[45] 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*, pages 313–331. Springer, 2024.

[46] Fu-Yun Wang, Xiaoshi Wu, Zhaoyang Huang, Xiaoyu Shi, Dazhong Shen, Guanglu Song, Yu Liu, and Hongsheng Li. Be-your-outpainter: Mastering video outpainting through input-specific adaptation. In *European Conference on Computer Vision*, pages 153–168. Springer, 2024.

[47] Miao Wang, Guo-Ye Yang, Jin-Kun Lin, Song-Hai Zhang, Ariel Shamir, Shao-Ping Lu, and Shi-Min Hu. Deep online video stabilization with multi-grid warping transformation learning. *IEEE Transactions on Image Processing*, 28(5): 2283–2292, 2018.

[48] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018.

[49] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro. Few-shot video-to-video synthesis. *arXiv preprint arXiv:1910.12713*, 2019.

[50] Yiming Wang, Qian Huang, Chuanxu Jiang, Jiwen Liu, Mingzhou Shang, and Zhuang Miao. Video stabilization: A comprehensive survey. *Neurocomputing*, 516:205–230, 2023.

[51] 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.

[52] Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T Barron, and Aleksander Holynski. Cat4d: Create anything in 4d with multi-view video diffusion models. *arXiv preprint arXiv:2411.18613*, 2024.

[53] Yuxi Xiao, Qianqian Wang, Shangzhan Zhang, Nan Xue, Sida Peng, Yujun Shen, and Xiaowei Zhou. Spatialtracker: Tracking any 2d pixels in 3d space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20406–20417, 2024.

[54] Zeqi Xiao, Wenqi Ouyang, Yifan Zhou, Shuai Yang, Lei Yang, Jianlou Si, and Xingang Pan. Trajectory attention for fine-grained video motion control. In *The Thirteenth International Conference on Learning Representations*, 2025.

[55] 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.

[56] Jinbo Xing, Menghan Xia, Yuxin Liu, Yuechen Zhang, Yong Zhang, Yingqing He, Hanyuan Liu, Haoxin Chen, Xiaodong Cun, Xintao Wang, et al. Make-your-video: Customized video generation using textual and structural guidance. *IEEE Transactions on Visualization and Computer Graphics*, 2024.

[57] 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.

[58] Yiran Xu, Taesung Park, Richard Zhang, Yang Zhou, Eli Shechtman, Feng Liu, Jia-Bin Huang, and Difan Liu. Videogigagan: Towards detail-rich video super-resolution. *arXiv preprint arXiv:2404.12388*, 2024.

[59] Zhen Xu, Yinghao Xu, Zhiyuan Yu, Sida Peng, Jiaming Sun, Hujun Bao, and Xiaowei Zhou. Representing long volumetric video with temporal gaussian hierarchy. *ACM Transactions on Graphics*, 43(6), 2024.

[60] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023.

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

[62] 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.

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

[64] Mark YU, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. *arXiv preprint arXiv:2503.05638*, 2025.

[65] 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. *arXiv preprint arXiv:2411.05003*, 2024.

[66] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arXiv:2410.03825*, 2024.

[67] 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.

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

[69] Shangchen Zhou, Peiqing Yang, Jianyi Wang, Yihang Luo, and Chen Change Loy. Upscale-a-video: Temporal-consistent diffusion model for real-world video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2535–2545, 2024.