Video Camera Trajectory Editing with Generative Rendering from Estimated Geometry

Junyoung Seo[1*], Jisang Han[1*], Jaewoo Jung[1*], JoungBin Lee[1], Takuya Narihira[2], Kazumi Fukuda[2], Takashi Shibuya[2], Donghoon Ahn[1], Shoukang Hu[2], Seungryong Kim[1†], Yuki Mitsufuji[2†],

[1]KAIST AI
[2]Sony AI and Sony Group Corporation

{junyoung.seo, onground, jaewoo.jung, seungryong.kim}@kaist.ac.kr, yuhki.mitsufuji@sony.com

Abstract

We introduce **Vid-CamEdit**, a novel framework for video camera trajectory editing, enabling the re-synthesis of monocular videos along user-defined camera paths. This task is challenging due to its ill-posed nature and the limited multi-view video data for training. Traditional reconstruction methods struggle with extreme trajectory changes, and existing generative models for dynamic novel view synthesis cannot handle in-the-wild videos. Our approach consists of two steps: estimating temporally consistent geometry, and generative rendering guided by this geometry. By integrating geometric priors, the generative model focuses on synthesizing realistic details where the estimated geometry is uncertain. We eliminate the need for extensive 4D training data through a factorized fine-tuning framework that separately trains spatial and temporal components using multi-view image and video data. Our method outperforms baselines in producing plausible videos from novel camera trajectories, especially in extreme extrapolation scenarios on real-world footage.

Introduction

When browsing through our camera albums, we often find ourselves wishing to view the videos we've captured from different camera poses. For instance, seeing footage originally shot from the side as if it were filmed from the front, or transforming a moving shot into one that appears as if taken from a stationary camera. *What if we can freely manipulate the camera movement within recorded videos to re-synthesize them from any viewpoint?* This ability will not only revolutionize how we experience our own videos but also impact fields like video editing and 4D content creation.

In this work, we focus on the task of re-synthesizing a given video along a user-defined camera trajectory, a process we refer to as *video camera trajectory editing*. This task is inherently related to the extreme case of dynamic novel view synthesis (NVS) given a monocular video, as it involves generating views from significantly altered or entirely new camera trajectories that were not present in the original footage. Existing approaches encounter two main challenges when tackling this task:

Reconstruction-based methods struggle with unseen areas. The extensive modification to the camera's path makes the problem highly ill-posed, causing existing reconstruction-based methods for dynamic NVS (Zhang et al. 2024b; Zhao et al. 2024) to fail in synthesizing visually realistic novel views, as illustrated in Fig. 2-(b). Because these methods focus on accurately reconstructing observed regions rather than handling unseen areas, they cannot accommodate the significant extrapolation required when the new camera trajectory deviates greatly from the original.

Generation methods require large-scale 4D data. While generative models have shown promising results in synthesizing highly realistic novel views in static scenes by training on large-scale multi-view image datasets (3D data), applying this approach t o dynamic scenes is challenging due to the limited availability of extensive real-world multi-view videos (4D data). Recent work, e.g., Generative Camera Dolly (Van Hoorick et al. 2024), tackles the problem by training on synthetic multi-view video data. However, they often fail to generalize to real-world videos due to domain gaps, as shown in Fig. 2-(c).

In this work, to overcome these challenges, we explore a more practical and data-efficient approach, which sidesteps the need for extensive real 4D training data. Instead of taking the data-driven solution, we decompose the task into two sub-tasks: (1) temporally-consistent geometry estimation and (2) generative video rendering based on the estimated geometry. Specifically, we ground pre-trained video generative models with geometry estimated from off-the-shelf geometry estimation models (Zhang et al. 2024b) (Fig. 2-(b)), allowing it to synthesize realistic novel view videos while relying on the geometry as a scaffold. This geometric prior reduces the burden on the generative model, enabling it to focus primarily on enhancing uncertain regions instead of learning full 4D dynamics from scratch, thereby greatly reducing the need for large-scale 4D training data.

To further reduce the need for 4D data, we incorporate a factorized fine-tuning strategy. By considering the spatio-temporal blocks of our video generative model independently, we train the spatial block with multi-view image (3D)

*These authors contributed equally.

Figure 1: Teaser: We aim to re-synthesize an input monocular video (top) following a desired camera trajectory. Our generated video (bottom) preserves the motion and structure of the input video while demonstrating realistic visual quality.

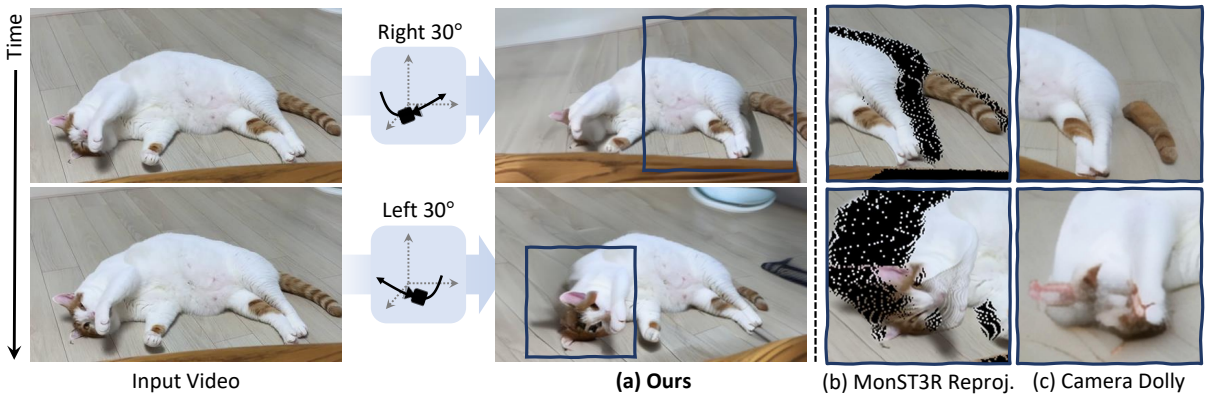

Figure 2: Motivation: To edit camera trajectories in monocular videos, we embed knowledge from video geometry prediction models, e.g., MonST3R (Zhang et al. 2024b), into video generative models (Guo et al. 2023), allowing the model to synthesize realistic novel views by filling occluded regions the geometry model cannot infer. By incorporating geometrical cues for generation, our approach demonstrates superior performance on novel view video synthesis, compared to fully generative approaches e.g., Generative Camera Dolly (Van Hoorick et al. 2024).

data and train the temporal block with video data. As both 3D and video data are accessible up to scale, the training of generative models no longer requires 4D data.

Related Work

Dynamic novel view synthesis via reconstruction. Similar to traditional novel view synthesis, which aims to reconstruct the scene given multi-view observations, dynamic novel view synthesis extends its application to dynamic scenes. Building upon the success of Neural Radiance Fields (NeRF) and 3D Gaussian Splatting for novel view synthesis, existing approaches (Han et al. 2025; Zhao et al. 2024; Wang et al. 2024a) tackle dynamic scenes by introducing an additional time-dimension or learning time-based deformations. While these approaches handle dynamics well, they struggle to extrapolate or estimate unseen areas, limiting novel views to those near the original input. This restricts their use in in-the-wild videos, causing large gaps in unseen regions and accumulating reprojection errors.

Video geometry estimation. Unlike monocular depth estimation (MDE) (Yang et al. 2024b; Ke et al. 2024), which infers depth from a single image, video depth estimation must ensure temporal consistency across frames. Early approaches (Luo et al. 2020; Zhang et al. 2021b; Yang et al. 2024a; Hu et al. 2024; Shao et al. 2024a) achieved this by fine-tuning MDE models or generative models and modeling motions for each input video. In parallel, a novel approach MonST3R (Zhang et al. 2024b) extends DUSt3R's (Wang et al. 2024c) unique pointmap representation, which capitalizes on accurate correspondence between images (An et al. 2025), enables dense 3D scene reconstruction to dynamic scenes.

Generative dynamic novel view synthesis. Extending such camera-controllable video models to video-camera trajectory editing is non-trivial, as it requires both semantic understanding and low-level perception of the user-provided video. Generative Camera Dolly (Van Hoorick et al. 2024) is the first attempt at this, paving the way for future re-

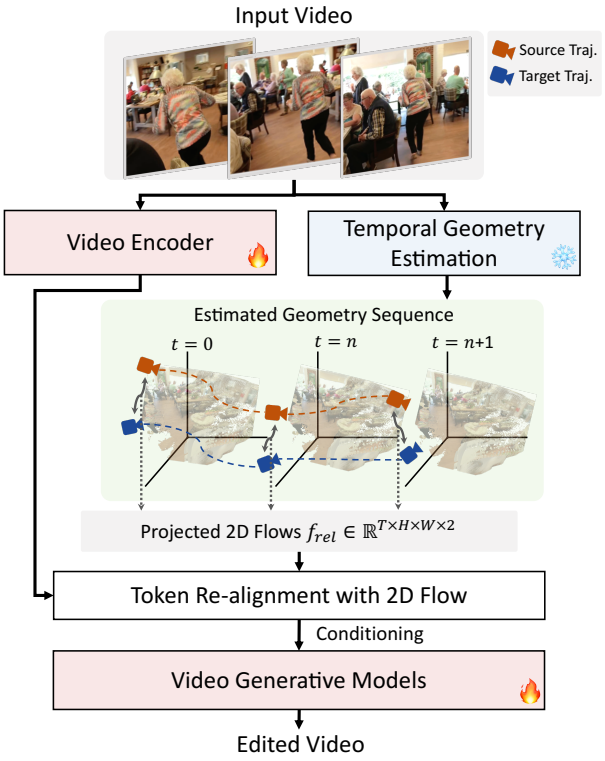

Figure 3: Overview of our framework. Given a video and a target camera trajectory, we first extract video feature tokens and obtain the dynamic scene's geometry. We then ground the video generative model on this estimated geometry by re-aligning the video feature tokens according to the 2D flow between the source and target camera trajectories.

search; however, it still shows clear weaknesses in generalizing to in-the-wild videos, being highly fitted to the 4D synthetic training data. 4DiM (Watson et al. 2024) generates novel view videos conditioned on one or more input images. However, it relies on 4D data from Google Street View and its generalizability to in-the-wild videos has not yet been validated. Recent and concurrent efforts, such as ReCapture (Zhang et al. 2024a), CAT4D (Wu et al. 2024), ReCamMaster (Bai et al. 2025) and TrajectoryCrafter (YU et al. 2025) share goals and motivations akin to ours. For instance, ReCamMaster employs synthetic 4D datasets based on Unreal Engine, while TrajectoryCrafter trains models through self-generated occlusions on monocular videos, whereas our approach addresses this task through lightweight fine-tuning without additional data processing.

Methodology

Problem definition

Given a monocular video as input, which can be captured from either a stationary or a moving camera, our objective of *video camera trajectory editing* is to design a framework that can synthesize a new video from any desired camera trajectory.

We first define the input video with T frames of size $H \times W$ as $X \in \mathbb{R}^{T\times H\times W\times 3}$ and its camera trajectory as $C_X \in \mathbb{R}^{T\times 3\times 4}$, which consists of a series of camera extrinsic matrices. The desired camera trajectory for the novel video $Y \in \mathbb{R}^{T\times H\times W\times 3}$ is defined as C_Y, where C_Y is obtained by applying per-frame relative camera transformations C_{rel} to C_X. Altogether, our framework $\mathcal{F}(\cdot)$ synthesizes a new video Y conditioned on the input video X and relative camera transformations C_{rel} as follows:

$$Y = \mathcal{F}(X, C_{\text{rel}}, K), \tag{1}$$

where we assume both the original and synthesized videos share the same camera intrinsics K.

Overview and motivation

To handle extensive extrapolation inherently required for our task, we design $\mathcal{F}(\cdot)$ as a generative framework, which has shown promising results in large extrapolation in static scenes (Liu et al. 2023; Sargent et al. 2023). However, leveraging generative models for dynamic NVS raises a unique challenge, which is the lack of sufficient real 4D data (multi-view videos).

To address this challenge, we explore a practical and data-efficient solution for the framework $\mathcal{F}(\cdot)$: a hybrid strategy that grounds strong geometry priors into video generative models. Our key intuition is to reduce the burden on the generative model by simplifying its task. Instead of relying solely on the generative model, we decompose the 4D problem into 3D spatial geometry and 1D temporal dynamics. For the 3D spatial geometry, we utilize a temporally consistent geometry estimation model to capture the 3D structure. As illustrated in Fig. 2-(b), this provides geometric cues to the video generation model, where the video generation model can utilize the geometry as a scaffold for realistic generation. To handle the 1D temporal dynamics, we leverage the temporal consistency capabilities inherent in video generative models. By exploiting these capabilities, we ensure that the generated frames are temporally coherent, preserving motion consistency over time. The overview of our pipeline is illustrated in Fig. 3.

Generative rendering from estimated geometry

Temporally-consistent geometry estimation. To effectively reduce the burden on the video generative model of our framework $\mathcal{F}(\cdot)$, the geometric prediction model g serves as a general model that is capable of estimating temporally consistent geometry of the given video. Although various models can be leveraged as g , we build up our framework on the recently proposed MonST3R (Zhang et al. 2024b), as its joint estimation of consistent camera trajectory and pointmaps can be effectively utilized in our framework. Specifically, the geometry of the input video is represented as a series of pointmaps $G \in \mathbb{R}^{T\times H\times W\times 3}$, which are coordinate maps indicating the 3D location of each pixel within the global 3D space. For each frame t, the pointmap G_t provides a dense mapping from 2D pixel coordinates (u, v) to their corresponding 3D world coordinates.

Geometry-grounded video-to-video translation. With the estimated temporally-consistent geometry, we now reformulate the video generative model in our framework as

a geometry-guided video-to-video translation problem. We incorporate the predicted geometry G as a crucial cue alongside the desired camera trajectory. At a high level, this framework can be expressed as:

$$\mathcal{F}(X, C_{\text{rel}}, K) := \text{Sample}\big(p_\theta(Y \mid X, C_{\text{rel}}, K, G)\big), \quad (2)$$

where p_θ is a learned distribution of a diffusion model θ and $\text{Sample}(\cdot)$ is a sampling function for diffusion reverse process (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2020). Although providing the 3D geometry information can facilitate novel view video generation, the model would still need to learn a mechanism that enables NVS that well reflects the input 3D geometry and camera parameters. Assuming that the pre-trained video generative model lacks the ability to explicitly understand 3D representations, we further simplify the task for the video model.

Given the pointmap G_t for frame t, we can obtain 2D flow fields $f_{\text{rel}} \in \mathbb{R}^{T\times H\times W\times 2}$ by projecting these 3D points onto the target viewpoint. Specifically, for each pixel (u, v) in the source frame, we obtain the 2D flow f_{rel}:

$$f_{\text{rel}}(u, v, t) = \Pi(C_{\text{rel}}(t) \cdot G(u, v, t), K) - (u, v), \quad (3)$$

where $\Pi(\cdot)$ is the perspective projection function.

This process maps each source pixel to its corresponding location in the target view for the time t, effectively grounding the translation in geometry without requiring the model to handle and understand complex 3D structures directly. We thus reformulate the generative process as:

$$\mathcal{F}(X, C_{\text{rel}}, K) := \text{Sample}\big(p_\theta(Y \mid X, f_{\text{rel}})\big), \quad (4)$$

where we note that the previous conditions – camera poses C_{rel}, intrinsics K, and 3D geometry G – are all inherently embedded within the 2D flow maps f_{rel}. This reformulation simplifies the task for the generative model while maintaining geometric consistency through the explicitly computed correspondences.

Re-aligning input video tokens. For the reformulated video generative model that takes the input video and 2D flow maps as conditions, we incorporate both conditions into a pretrained video diffusion model (Guo et al. 2023). For video conditioning, the model must preserve the input video's details, such as color and texture. We adopt the architecture of ReferenceNet (Hu 2024), which has been shown to effectively preserve the low-level semantics of input images (Hu 2024; Men et al. 2024). Specifically, our approach is based on a U-Net-based video diffusion model where spatial and temporal blocks are interleaved. On top of this, we define a video encoder $\mathcal{E}_\phi$ that shares the same architecture as the video diffusion model. The feature tokens of $\mathcal{E}_\phi$ are then concatenated into the self-attention map of each spatial block in the diffusion model, which is for spatial interaction within each frame of the novel view video being generated.

For flow conditioning, we align the feature tokens of $\mathcal{E}_\phi$ with the flow condition f_{rel}. To this end, we can either explicitly warp the feature tokens of input video (Müller et al. 2024; Niu et al. 2024) or encourage the model to perform reliable internal re-alignment with flow-conditioning methods (Seo et al. 2024; Zhang et al. 2024c; Cai et al. 2024). In this work, motivated by GenWarp (Seo et al. 2024), we rearrange the positional embeddings for input video according to the flow map f_{rel} and employ them as additional positional embeddings, thereby allowing the model to naturally learn the flow condition. Specifically, for a given position (u, v) in frame t, the re-aligned positional encoding PE' is computed as:

$$\text{PE}'(u, v, t) = \text{PE}\big(u + f_{\text{rel}}(u, v, t)_x, v + f_{\text{rel}}(u, v, t)_y, t\big), \quad (5)$$

where PE denotes the sinusoidal positional encoding for the input video, and $f_{\text{rel}}(u, v)_x$, $f_{\text{rel}}(u, v)_y$ represents the flow vectors in x and y directions respectively.

The re-aligned positional embeddings are additionally incorporated into the video diffusion model alongside the original positional embeddings, enabling the video generative model to take the flow condition with the input video.

Fine-tuning without 4D data

While our geometry-grounded strategy effectively alleviates the computational burden on generative models, naïvely training such models still hinges on real-world 4D data (i.e., multi-view videos), which is prohibitively expensive and impractical to acquire at large scales. We instead adopt a factorized training that capitalizes on more readily available datasets: multi-view images and conventional video data, similarly to (Shao et al. 2024b; Watson et al. 2024). This shift obviates the need for comprehensive 4D data collection, offering a more scalable solution.

Architecture. We employ a video generative model backbone (Guo et al. 2023; Blattmann et al. 2023) composed of interleaved spatial and temporal interaction blocks. We inject conditioning derived from input video tokens solely into the spatial interaction blocks – thereby converting them into multi-view blocks – to concentrate on 3D synthesis. Meanwhile, the temporal interaction blocks remain dedicated to learning temporal priors. For detailed diagrams, please refer to Appendix.

Block-wise supervision. Given our architectural design, we employ an intuitive factorized training strategy. When training on videos, we freeze the multi-view blocks; and when training on multi-view images, we freeze the temporal blocks. Multi-view images are treated as multi-view videos with $T = 1$, updating only the multi-view blocks, whereas video data are treated as multi-view videos with the same input and output cameras, updating only the temporal blocks. Here, the conditioning tokens from the video encoder are replaced with a null condition at a predefined probability, similarly to CFG (Ho and Salimans 2022). By alternately freezing these blocks, we mitigate overfitting to either modality and successfully train our model without relying on 4D data.

Experiments

Implementation details Our framework consists of two key components, geometry prediction model and video generative model. As mentioned in Section , we leverage MonST3R (Zhang et al. 2024b) as our geometry prediction model. For the video generative model, our framework

Figure 4: Qualitative results. Given a user-provided monocular video, our method can synthesize high-quality videos along desired camera trajectories. The frames from the original videos are depicted in the yellow box of the top right of each image.

can leverage any spatio-temporally factorized video diffusion models (Guo et al. 2023; Blattmann et al. 2023; Chen et al. 2023). Among them, we adopt AnimateDiff (Guo et al. 2023) based on Stable Diffusion 1.5 (Rombach et al. 2022) as our base model, generating $T = 12$ frames at once, as it best fits our computational constraints. To condition the diffusion model with only the input video and cameras (2D flow), we replace the original text condition with CLIP (Radford et al. 2021) image features. Code and weights will be publicly available.

Training dataset. For multi-view image data, we utilize RealEstate10K (Zhou et al. 2018), MannequinChallenge (Li et al. 2019), MegaScene (Tung et al. 2025), and ScanNet (Dai et al. 2017). For temporal fine-tuning, we initialize the temporal modules from the pre-trained checkpoint (Guo et al. 2023) trained on WebVid-10M (Bain et al. 2021) and additionally use the TikTok dataset (Jafarian and Park 2021) in fine-tuning.

Baselines As our task demands extensive interpolation and extrapolation, we primarily compare our method with generation and generalizable methods: Generative Camera Dolly (GCD) (Van Hoorick et al. 2024) and Pseudo-DVS (Zhao et al. 2024). We also report performance improvements over our baseline: reprojection using MonST3R (Zhang et al. 2024b). We evaluate two variants with MonST3R: all-frame reprojection and per-frame reprojection. All-frame reprojection leverages all pointmaps from MonST3R's global alignment, projecting all static points across frames $[1, T]$ and combining them with dynamic points from frame t.

Qualitative comparisons. Fig. 4 and Fig. 5 show qualitative results and comparisons on in-the-wild videos with the baseline methods. MonST3R (Zhang et al. 2024b) and Pseudo-DVS (Zhao et al. 2024) show reasonable performance in some regions, however, failing to synthesize occluded regions. GCD (Van Hoorick et al. 2024) generates synthetic artifacts when dealing with in-the-wild video, failing to generalization. Additionally, we present results of reprojection-and-inpainting (Zi et al. 2024) baseline, struggling with refining ill-warped artifacts when conditioned on noisy reprojections. In contrast, our method generates feasible videos from new camera trajectories.

Quantitative comparisons. We perform a quantitative comparison of our method and generalizable reconstruction and generation methods on the multi-view video datasets, Neu3D (Li et al. 2022) and ST-NeRF dataset (Zhang et al. 2021a) in Tab. 1. For frame consistency, we measure CLIP score between each frame of input videos and generated videos, following (Jeong et al. 2024). The results show that our method achieves superior performance across all the datasets. Additionally, we report a user study for human preference in Fig. 6, and VBench (Huang et al. 2024) scores, VLM-based automated benchmarks in Fig. 7.

Application for per-scene 4D reconstruction. While our primary goal is to directly generate video renderings, our approach can be seamlessly integrated into per-scene 4D reconstruction methods that produce 4D representations as output, by leveraging our generated results as additional supervision. As shown in Tab. 2, quantitative evaluations on the DyCheck dataset (Gao et al. 2022) demonstrate that incorporating our method into existing per-scene reconstruction pipelines yields higher reconstruction quality.

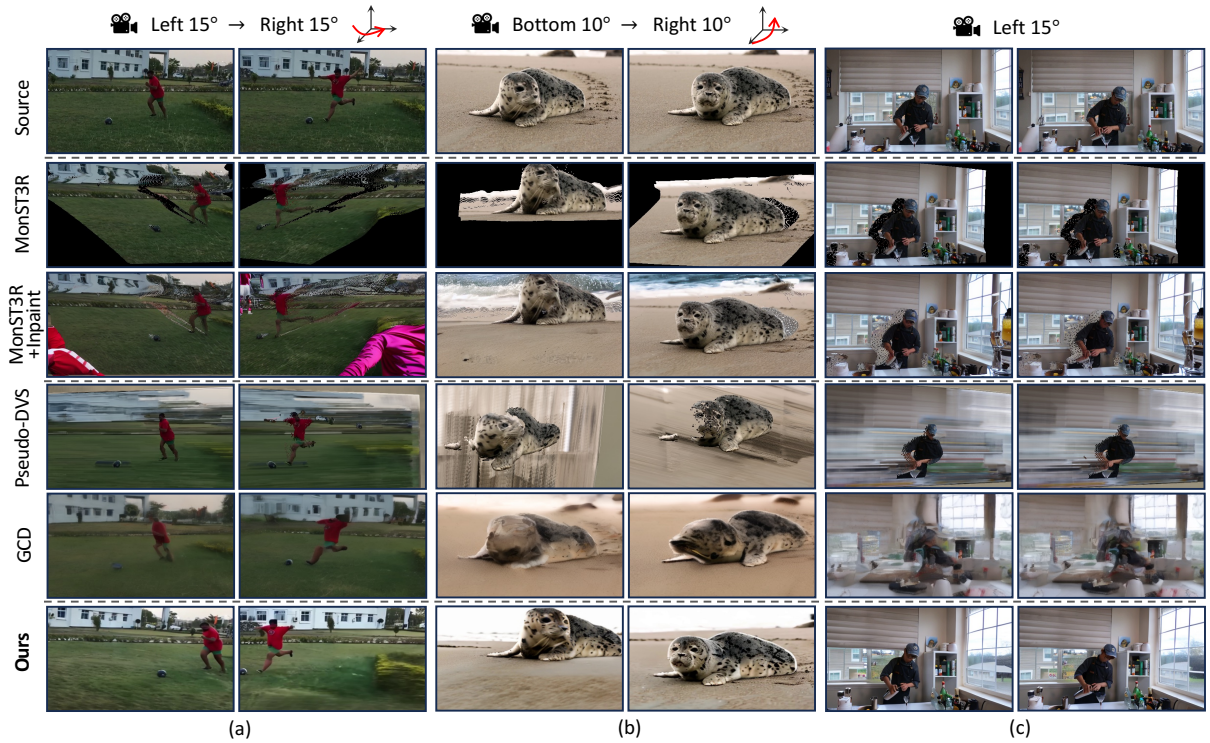

Figure 5: Qualitative comparisons with MonST3R (Zhang et al. 2024b), video inpainting (Zi et al. 2024) w/MonST3R, GCD (Generative Camera Dolly) (Van Hoorick et al. 2024), and Pseudo-DVS (Zhao et al. 2024).

Methods	Neu3D (Li et al. 2022)				ST-NeRF dataset (Zhang et al. 2021a)			
	LPIPS ↓	SSIM ↑	PSNR ↑	Frame-Con. ↑	LPIPS ↓	SSIM ↑	PSNR ↑	Frame-Con. ↑
MonST3R (Zhang et al. 2024b)	0.562	0.206	10.42	0.747	0.649	0.224	8.39	0.757
MonST3R (Zhang et al. 2024b) (Per-frame proj.)	0.453	0.291	11.73	0.800	0.478	0.288	9.91	0.811
Pseudo-DVS (Zhao et al. 2024)	0.564	0.352	14.43	0.655	0.527	**0.415**	**15.33**	0.742
Generative Camera Dolly (Van Hoorick et al. 2024)	0.505	0.249	10.71	0.682	0.425	0.346	13.60	0.748
Ours	**0.414**	**0.358**	**14.91**	**0.858**	**0.386**	0.381	14.89	**0.917**

Table 1: Quantitative results with generalized/generation baselines. We show quantitative comparisons in multi-view dynamic datasets, Neu3D (Li et al. 2022) and ST-NeRF (Zhang et al. 2021a).

Analyses

Performance on varying trajectory difficulties. We analyze how performance changes as the desired camera trajectory becomes more challenging in Fig. 8, where our method achieves the best performance. Specifically, following the evaluation protocol in GeoGPT (Rombach, Esser, and Ommer 2021), we measure the LPIPS between the input video and the target GT video as the generation difficulty due to viewpoint change, and consider the LPIPS between the generated video and the target GT video as a degree of distortion. We then compare the degree of distortion against the generation difficulty.

Ablation on design choices. We provide an ablation study on various design choices in our framework. We test a case where we directly inject camera poses (Plücker coordinates) (Sitzmann et al. 2021) into the model in place of our proposed grounding. Additionally, we compare the performance against a baseline method: reprojection and video inpainting (Zi et al. 2024), which is one possible naïve approach to combining geometry estimation models. As shown in Tab. 3, our full framework is most effective in both cases.

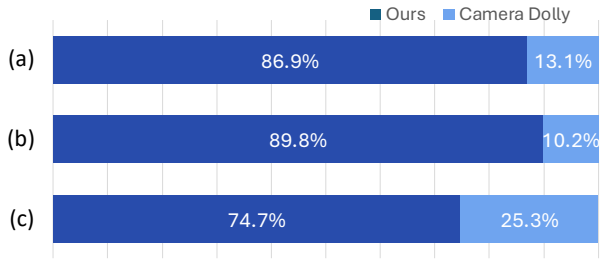

Figure 6: User study is conducted by surveying 59 participants to evaluate (a) consistency to input videos, (b) video realness, and (c) faithfulness on camera trajectories.

Video geometry estimation models. We provide an ablation study of using various video geometry estimation models g in our framework. We evaluate: MonST3R (Zhang et al. 2024b), DepthAnyVideo (Yang et al. 2024a), Depth-

Method	Co-visible			Occluded		
	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑
HyperNeRF (Park et al. 2021)	15.99	0.510	0.590	-	-	-
HyperNeRF*	14.32	0.667	0.552	14.22	0.554	0.834
DyniBar (Li et al. 2023)	13.41	0.550	0.480	-	-	-
Shape-of-Motion (Wang et al. 2024b)	<u>16.72</u>	0.450	<u>0.630</u>	-	-	-
Shape-of-Motion*	16.71	0.394	**0.646**	<u>15.24</u>	<u>0.465</u>	**0.856**
+ Ours	**16.84**	**0.261**	0.573	**15.56**	**0.129**	**0.856**

Table 2: Quantitative results of 4D reconstruction on DyCheck (Gao et al. 2022). Employing our method to the existing per-scene reconstruction method yield better reconstruction quality. * denotes reproduced results for occluded region evaluation.

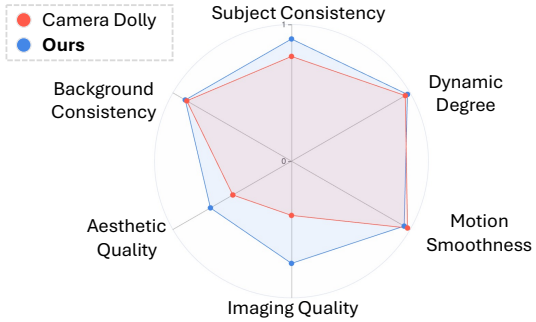

Figure 7: Quantitative comparisons on Vbench (Huang et al. 2024) with Camera Dolly (Van Hoorick et al. 2024) on uncurated in-the-wild videos.

Baseline	LPIPS ↓	SSIM ↑
w/o Geometry grounding	0.498	0.314
Reproj. + Inpainting (Zi et al. 2024)	0.485	0.336
Ours	**0.414**	**0.358**

Table 3: Ablation study on design choices comparing our framework with and without geometry grounding and a reprojection/video-inpainting baseline.

Base Geometry Models	LPIPS ↓	SSIM ↑
DepthCrafter (Hu et al. 2024)	<u>0.416</u>	<u>0.356</u>
Depth-Anything 2 (Yang et al. 2024b)	0.420	0.354
DepthAnyVideo (Yang et al. 2024a)	0.425	0.349
MonST3R (Zhang et al. 2024b)	**0.414**	**0.358**

Table 4: **Ablation on video geometry models.**

Anything2 (Yang et al. 2024b), and DepthCrafter (Hu et al. 2024) on the Neu3D dataset (Li et al. 2022). As shown in Tab. 4, our framework achieves consistent quality regardless of the chosen geometry prediction model.

Conclusion

We have introduced a novel framework **Vid-CamEdit** for video camera trajectory editing, using generative rendering grounded by estimated geometry. By combining temporally consistent geometry estimation and a factorized fine-tuning approach, we achieve robust and visually consistent novel view video synthesis.

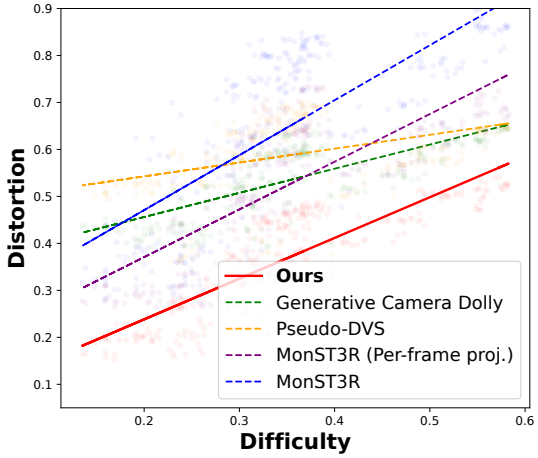

Figure 8: Comparison on various camera trajectory. We measure LPIPS between generated videos and target videos (Distortion) over LPIPS between input videos and target videos (Difficulty). Ours consistently achieves best performance.

Acknowledgments

This research was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2019-II190075, RS-202400509279, RS-2025-II212068, RS-2023-00227592, RS-2025-02214479, RS-2024-00457882, RS2025-25441838, RS-2025-25441838, RS-2025-02214479, RS-2025-02217259) and the Culture, Sports, and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism (RS-2024-00345025, RS-2024-00333068, RS-202300222280, RS-2023-00266509), and National Research Foundation of Korea (RS-2024-00346597).

References

An, H.; Kim, J. H.; Park, S.; Jung, J.; Han, J.; Hong, S.; and Kim, S. 2025. Cross-view completion models are zero-shot correspondence estimators. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 1103–1115.

Bai, J.; Xia, M.; Fu, X.; Wang, X.; Mu, L.; Cao, J.; Liu, Z.; Hu, H.; Bai, X.; Wan, P.; et al. 2025. Recammaster: Camera-controlled generative rendering from a single video. *arXiv preprint arXiv:2503.11647*.

Bain, M.; Nagrani, A.; Varol, G.; and Zisserman, A. 2021. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1728–1738.

Blattmann, A.; Dockhorn, T.; Kulal, S.; Mendelevitch, D.; Kilian, M.; Lorenz, D.; Levi, Y.; English, Z.; Voleti, V.; Letts, A.; et al. 2023. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*.

Cai, S.; Ceylan, D.; Gadelha, M.; Huang, C.-H. P.; Wang, T. Y.; and Wetzstein, G. 2024. Generative rendering: Controllable 4d-guided video generation with 2d diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7611–7620.

Chen, H.; Xia, M.; He, Y.; Zhang, Y.; Cun, X.; Yang, S.; Xing, J.; Liu, Y.; Chen, Q.; Wang, X.; et al. 2023. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*.

Dai, A.; Chang, A. X.; Savva, M.; Halber, M.; Funkhouser, T.; and Nießner, M. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5828–5839.

Gao, H.; Li, R.; Tulsiani, S.; Russell, B.; and Kanazawa, A. 2022. Monocular dynamic view synthesis: A reality check. *Advances in Neural Information Processing Systems*, 35: 33768–33780.

Guo, Y.; Yang, C.; Rao, A.; Liang, Z.; Wang, Y.; Qiao, Y.; Agrawala, M.; Lin, D.; and Dai, B. 2023. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*.

Han, J.; An, H.; Jung, J.; Narihira, T.; Seo, J.; Fukuda, K.; Kim, C.; Hong, S.; Mitsufuji, Y.; and Kim, S. 2025. Dˆ 2USt3R: Enhancing 3D Reconstruction with 4D Pointmaps for Dynamic Scenes. *arXiv preprint arXiv:2504.06264*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Ho, J.; and Salimans, T. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Hu, L. 2024. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8153–8163.

Hu, W.; Gao, X.; Li, X.; Zhao, S.; Cun, X.; Zhang, Y.; Quan, L.; and Shan, Y. 2024. Depthcrafter: Generating consistent long depth sequences for open-world videos. *arXiv preprint arXiv:2409.02095*.

Huang, Z.; He, Y.; Yu, J.; Zhang, F.; Si, C.; Jiang, Y.; Zhang, Y.; Wu, T.; Jin, Q.; Chanpaisit, N.; et al. 2024. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21807–21818.

Jafarian, Y.; and Park, H. S. 2021. Learning high fidelity depths of dressed humans by watching social media dance videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12753–12762.

Jeong, H.; Chang, J.; Park, G. Y.; and Ye, J. C. 2024. DreamMotion: Space-Time Self-Similar Score Distillation for Zero-Shot Video Editing. *arXiv preprint arXiv:2403.12002*.

Ke, B.; Obukhov, A.; Huang, S.; Metzger, N.; Daudt, R. C.; and Schindler, K. 2024. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9492–9502.

Li, T.; Slavcheva, M.; Zollhoefer, M.; Green, S.; Lassner, C.; Kim, C.; Schmidt, T.; Lovegrove, S.; Goesele, M.; Newcombe, R.; et al. 2022. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5521–5531.

Li, Z.; Dekel, T.; Cole, F.; Tucker, R.; Snavely, N.; Liu, C.; and Freeman, W. T. 2019. Learning the depths of moving people by watching frozen people. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4521–4530.

Li, Z.; Wang, Q.; Cole, F.; Tucker, R.; and Snavely, N. 2023. Dynibar: Neural dynamic image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4273–4284.

Liu, R.; Wu, R.; Van Hoorick, B.; Tokmakov, P.; Zakharov, S.; and Vondrick, C. 2023. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9298–9309.

Luo, X.; Huang, J.-B.; Szeliski, R.; Matzen, K.; and Kopf, J. 2020. Consistent video depth estimation. *ACM Transactions on Graphics (ToG)*, 39(4): 71–1.

Men, Y.; Yao, Y.; Cui, M.; and Bo, L. 2024. Mimo: Controllable character video synthesis with spatial decomposed modeling. *arXiv preprint arXiv:2409.16160*.

Müller, N.; Schwarz, K.; Rössle, B.; Porzi, L.; Bulò, S. R.; Nießner, M.; and Kontschieder, P. 2024. MultiDiff: Consistent Novel View Synthesis from a Single Image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10258–10268.

Niu, M.; Cun, X.; Wang, X.; Zhang, Y.; Shan, Y.; and Zheng, Y. 2024. MOFA-Video: Controllable Image Animation via Generative Motion Field Adaptions in Frozen Image-to-Video Diffusion Model. *arXiv preprint arXiv:2405.20222*.

Park, K.; Sinha, U.; Hedman, P.; Barron, J. T.; Bouaziz, S.; Goldman, D. B.; Martin-Brualla, R.; and Seitz, S. M. 2021. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Rombach, R.; Esser, P.; and Ommer, B. 2021. Geometry-free view synthesis: Transformers and no 3d priors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 14356–14366.

Sargent, K.; Li, Z.; Shah, T.; Herrmann, C.; Yu, H.-X.; Zhang, Y.; Chan, E. R.; Lagun, D.; Fei-Fei, L.; Sun, D.; et al. 2023. Zeronvs: Zero-shot 360-degree view synthesis from a single real image. *arXiv preprint arXiv:2310.17994*.

Seo, J.; Fukuda, K.; Shibuya, T.; Narihira, T.; Murata, N.; Hu, S.; Lai, C.-H.; Kim, S.; and Mitsufuji, Y. 2024. GenWarp: Single Image to Novel Views with Semantic-Preserving Generative Warping. *arXiv preprint arXiv:2405.17251*.

Shao, J.; Yang, Y.; Zhou, H.; Zhang, Y.; Shen, Y.; Poggi, M.; and Liao, Y. 2024a. Learning Temporally Consistent Video Depth from Video Diffusion Priors. *arXiv preprint arXiv:2406.01493*.

Shao, R.; Pang, Y.; Zheng, Z.; Sun, J.; and Liu, Y. 2024b. Human4DiT: 360-degree Human Video Generation with 4D Diffusion Transformer. *ACM Transactions on Graphics (TOG)*, 43(6).

Sitzmann, V.; Rezchikov, S.; Freeman, B.; Tenenbaum, J.; and Durand, F. 2021. Light field networks: Neural scene representations with single-evaluation rendering. *Advances in Neural Information Processing Systems*, 34: 19313–19325.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Tung, J.; Chou, G.; Cai, R.; Yang, G.; Zhang, K.; Wetzstein, G.; Hariharan, B.; and Snavely, N. 2025. MegaScenes: Scene-Level View Synthesis at Scale. In *European Conference on Computer Vision*, 197–214. Springer.

Van Hoorick, B.; Wu, R.; Ozguroglu, E.; Sargent, K.; Liu, R.; Tokmakov, P.; Dave, A.; Zheng, C.; and Vondrick, C. 2024. Generative Camera Dolly: Extreme Monocular Dynamic Novel View Synthesis. *arXiv preprint arXiv:2405.14868*.

Wang, C.; Zhuang, P.; Siarohin, A.; Cao, J.; Qian, G.; Lee, H.-Y.; and Tulyakov, S. 2024a. Diffusion priors for dynamic view synthesis from monocular videos. *arXiv preprint arXiv:2401.05583*.

Wang, Q.; Ye, V.; Gao, H.; Austin, J.; Li, Z.; and Kanazawa, A. 2024b. Shape of motion: 4d reconstruction from a single video. *arXiv preprint arXiv:2407.13764*.

Wang, S.; Leroy, V.; Cabon, Y.; Chidlovskii, B.; and Revaud, J. 2024c. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 20697–20709.

Watson, D.; Saxena, S.; Li, L.; Tagliasacchi, A.; and Fleet, D. J. 2024. Controlling space and time with diffusion models. *arXiv preprint arXiv:2407.07860*.

Wu, R.; Gao, R.; Poole, B.; Trevithick, A.; Zheng, C.; Barron, J. T.; and Holynski, A. 2024. Cat4d: Create anything in 4d with multi-view video diffusion models. *arXiv preprint arXiv:2411.18613*.

Yang, H.; Huang, D.; Yin, W.; Shen, C.; Liu, H.; He, X.; Lin, B.; Ouyang, W.; and He, T. 2024a. Depth Any Video with Scalable Synthetic Data. *arXiv preprint arXiv:2410.10815*.

Yang, L.; Kang, B.; Huang, Z.; Zhao, Z.; Xu, X.; Feng, J.; and Zhao, H. 2024b. Depth Anything V2. *arXiv preprint arXiv:2406.09414*.

YU, M.; Hu, W.; Xing, J.; and Shan, Y. 2025. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. *arXiv preprint arXiv:2503.05638*.

Zhang, D. J.; Paiss, R.; Zada, S.; Karnad, N.; Jacobs, D. E.; Pritch, Y.; Mosseri, I.; Shou, M. Z.; Wadhwa, N.; and Ruiz, N. 2024a. ReCapture: Generative Video Camera Controls for User-Provided Videos using Masked Video Fine-Tuning. *arXiv preprint arXiv:2411.05003*.

Zhang, J.; Herrmann, C.; Hur, J.; Jampani, V.; Darrell, T.; Cole, F.; Sun, D.; and Yang, M.-H. 2024b. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arXiv:2410.03825*.

Zhang, J.; Liu, X.; Ye, X.; Zhao, F.; Zhang, Y.; Wu, M.; Zhang, Y.; Yu, J.; and Xu, L. 2021a. Editable free-viewpoint video using a layered neural representation.

Zhang, Z.; Cole, F.; Tucker, R.; Freeman, W. T.; and Dekel, T. 2021b. Consistent depth of moving objects in video. *ACM Transactions on Graphics (ToG)*, 40(4): 1–12.

Zhang, Z.; Liao, J.; Li, M.; Qin, L.; and Wang, W. 2024c. Tora: Trajectory-oriented diffusion transformer for video generation. *arXiv preprint arXiv:2407.21705*.

Zhao, X.; Colburn, R. A.; Ma, F.; Bautista, M. Á.; Susskind, J. M.; and Schwing, A. 2024. Pseudo-Generalized Dynamic View Synthesis from a Video. In *The Twelfth International Conference on Learning Representations*.

Zhou, T.; Tucker, R.; Flynn, J.; Fyffe, G.; and Snavely, N. 2018. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*.

Zi, B.; Zhao, S.; Qi, X.; Wang, J.; Shi, Y.; Chen, Q.; Liang, B.; Wong, K.-F.; and Zhang, L. 2024. CoCoCo: Improving Text-Guided Video Inpainting for Better Consistency, Controllability and Compatibility. *arXiv preprint arXiv:2403.12035*.