# VideoCoF: Unified Video Editing with Temporal Reasoner

## Supplementary Material

This document provides more details of our approach and additional experimental results, which are organized as follows:

- Discussion on RoPE Design (§1)
- Editing Length Upper Bound (§2)
- Full Comparison (§3)
- TGVE+ and V2VBench (§4)
- More Ablation studies (§5)
- Implementation Details (§6)
- Metrics (§7)
- Future Directions (§8)

## 1. Discussion on RoPE Design

**Difference from UNIC.** VideoCoF targets *length extrapolation* in V2V editing, whereas UNIC focuses on *source-target alignment*. To enable extrapolation, we **pre-position** the ID/reasoning frames so that their indices never overlap with the target video. In contrast, UNIC **post-positions** the ID/reasoning tokens (e.g., shifting them to a later index range). As the video becomes longer and reaches the same index range, overlap becomes unavoidable, leading to temporal index collisions and content perturbations.

**Difference from T2V extrapolation methods.** RIFLEx [28] and UltraViCo [29] mainly address motion or content repetition in *T2V* length extrapolation, whereas our *V2V* extrapolation is primarily limited by source-target alignment and temporal index collision. We apply UltraViCo to the baseline, and Fig. 1 shows that it still fails.

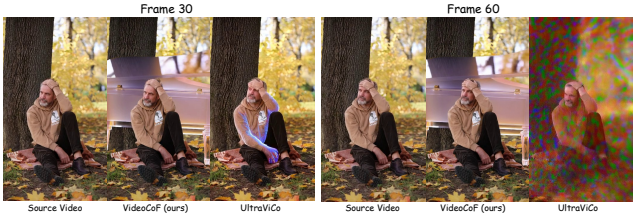


**Figure 1.** Comparison between VideoCoF and UltraViCo

## 2. Editing Length Upper Bound

**Systematic Temporal Extrapolation Failure Cases.** We test 16× extrapolation in both single-shot and multi-shot settings. Single-shot setting supports 16× extrapolation. For multi-shot, as shown in Fig. 2, it remains stable for the first 500 frames but shows slight degradation at 512 frames.

## 3. Full Comparison

As shown in Tab. 6, we provide a detailed breakdown of the results across four distinct tasks: Object Removal, Object Addition, Object Swap, and Local Style Transfer. Our VideoCoF consistently achieves the highest scores in **instruction following** and **success ratio** across all tasks, demonstrating superior capability in understanding and executing editing requests. We note that our scores in Preservation and Quality are slightly lower than the concurrent work ICVE [9]. This performance gap is reasonable given that ICVE benefits from large-scale pre-training on 1M pairs, and its supervised fine-tuning (SFT) dataset scale (150K) is three times larger than ours (50k). Furthermore, in terms of perceptual quality, VideoCoF achieves the highest **CLIP-T** scores across all tasks. This further demonstrates superior video-text alignment, consistent with our leading performance in GPT-4o score.

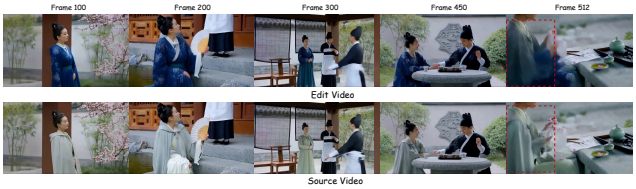


**Figure 2.** Temporal extrapolation failure case.

## 4. TGVE+ and V2VBench

Since TGVE[22] has only 304 samples, we instead evaluate on the comprehensive TGVE+ subset used in EVE [16] (1,417 samples across 7 instruction editing tasks), and report V2VBench [17] results. As shown in the tables, VideoCoF achieves the best performance on both benchmarks.

**Table 1.** Comparison on the TGVE+ benchmark.

| Dataset | Methods | PickScore ↑ | CLIP-F ↑ | ViCLIP$_{dir}$ ↑ | ViCLIP$_{out}$ ↑ |
|---|---|---|---|---|---|
| TGVE+ | Tune-A-Video [21] | 20.47 | 0.933 | 0.131 | 0.242 |
| | SDEdit [10] | 20.35 | 0.899 | 0.131 | 0.241 |
| | STDF [24] | 20.60 | 0.933 | 0.093 | 0.227 |
| | Fairy [20] | 19.81 | 0.933 | 0.140 | 0.197 |
| | InsV2V [3] | 20.37 | 0.925 | 0.174 | 0.236 |
| | EVE [16] | 20.88 | 0.926 | 0.198 | 0.251 |
| | **VideoCoF** | **20.90** | **0.956** | **0.213** | **0.257** |

**Table 2.** Comparison on the V2VBench benchmark.

| V2VBench | Frames Quality ↑ | Semantic Consis. ↑ | Object Consis. ↑ | Frames-Text Align. ↑ | Frames Pick. ↑ | Video-Text Align. ↑ | Motion Align. ↑ |
|---|---|---|---|---|---|---|---|
| Tune-A-Video [21] | 5.001 | 0.934 | 0.917 | 27.513 | 20.701 | 0.254 | -5.599 |
| SimDA [23] | 4.988 | 0.940 | 0.929 | 26.773 | 20.512 | 0.248 | -4.756 |
| VidToMe [8] | 4.988 | 0.949 | 0.945 | 26.813 | 20.546 | 0.240 | -3.203 |
| VideoComposer [19] | 4.429 | 0.914 | 0.905 | 28.001 | 20.272 | 0.262 | -8.095 |
| MotionDirector [30] | 4.984 | 0.949 | 0.951 | 27.845 | 20.923 | 0.262 | -3.088 |
| **VideoCoF (Ours)** | **5.024** | **0.954** | **0.956** | **28.336** | **21.154** | **0.275** | **-2.620** |

## 5. More Ablation Studies

In this section, we validate key design choices of VideoCoF: the length of reasoning frames and the dispatch prompt.

### 5.1. Ablation on Reasoning Frames

**Table 3. Ablation on the number of Reasoning Frames.** We investigate the impact of varying the number of reasoning frames from 1 to 5. Our default setting (4 frames) achieves the best balance.

| Frames | Ablation on Reasoning Frames 1 | 2 | 3 | **4 (Ours)** | 5 |
|---|---|---|---|---|---|
| *GPT-4o Score* | | | | | |
| Instruct Follow↑ | 8.219 | 8.312 | 8.281 | **8.973** | 7.915 |
| Preservation↑ | 8.115 | 8.150 | 8.191 | **8.203** | 6.542 |
| Quality↑ | 7.692 | 7.752 | 7.735 | **7.765** | 5.274 |
| Success Ratio↑ | 68.47% | 69.39% | 68.32% | **76.36%** | 29.06% |
| *Perceptual Quality* | | | | | |
| CLIP-T↑ | 27.092 | 27.148 | 27.136 | **28.000** | 26.997 |
| CLIP-F↑ | 0.9892 | 0.9893 | 0.9899 | **0.9915** | 0.9849 |
| DINO↑ | 0.9815 | 0.9827 | 0.9836 | **0.9913** | 0.9719 |

Tab. 3 investigates the optimal number of reasoning frames ($F$) for spatial guidance. Considering the VideoVAE temporal compression formula $L = (F-1)//4+1$, frames $1 \sim 4$ map to a single latent frame ($L = 1$), while $F = 5$ introduces latent frames $L = 2$. Results show that $F = 4$ achieves the best performance. This indicates maximizing spatial information within a single latent frame is more effective than expanding to a second latent, which introduces unnecessary temporal complexity and degradation.

### 5.2. Ablation on Temporal Triptych Prompt

**Table 4. Ablation on Temporal Triptych Prompt.** We compare the performance of our model with and without the triptych patch prompt mechanism. The inclusion of the triptych prompt significantly enhances instruction following and overall success rates.

| | Ablation on Temporal Triptych Prompt w/o Triptych | w/ Triptych (Ours) |
|---|---|---|
| *GPT-4o Score* | | |
| Instruct Follow↑ | 8.064 | **8.973** |
| Preservation↑ | 8.094 | **8.203** |
| Quality↑ | 7.360 | **7.765** |
| Success Ratio↑ | 71.43% | **76.36%** |
| *Perceptual Quality* | | |
| CLIP-T↑ | 27.07 | **28.00** |
| CLIP-F↑ | 0.989 | **0.992** |
| DINO↑ | 0.980 | **0.991** |

To adapt a standard T2V model for instruction-based editing tasks, we draw inspiration from in-context image editing approaches [4, 15, 27]. Specifically, we implement a **temporal triptych prompt** mechanism in VideoCoF to describe the evolution of video content along the temporal dimension. As illustrated in Fig. 3, our prompt template is structured as follows: *"A video sequence showing three parts: first the original scene, then grounded {ground instruction}, and finally the same scene but {edit instruction}."*


**Figure 3.** Input Prompt Variants for In-Context Video Editing. We evaluate two prompt formats: (a) Temporal Triptych Prompt - instructions embedded in a structure "A video sequence showing three parts: first the original scene, then grounded {ground instruction}, and finally the same scene but {edit instruction}.") (b) Direct Instruction - explicit editing commands provided directly.

As evidenced in Tab. 4, this mechanism brings significant performance gains across all metrics. Crucially, unlike the concurrent work ICVE [9], which requires computationally expensive pre-training on 1M video pairs to align the T2V model with an instruction mode, our "temporal triptych prompt" approach offers a practically zero-cost solution to effectively bridge the gap between generation and editing without the need for massive instruction tuning.

## 6. Implementation Details

### 6.1. Training Dataset

To equip our model with robust instruction-following capabilities, we constructed a unified chain-of-frames video editing dataset comprising 50k video pairs. As detailed in Table 5, the dataset is strategically balanced across four core editing tasks: object addition, removal, swapping, and local stylization. The data construction pipeline integrates both filtered open-source data and high-quality synthetic data. Object Addition and Removal: These subsets (25k samples total) are derived from the Señorita dataset. We employ MiniMax-Remover [31] to synthesize paired data. Specifically, for the removal task (15k), we treat the original video

**Table 5. Statistics of the VideoCoF Training Data.** The dataset consists of 50k samples balanced across four tasks.

| Dataset | #Samples | Information |
|---|---|---|
| **Video Editing Tasks** | | |
| **Obj. Addition** | 10,000 | Derived from filtered Señorita. Source generated by removing objects from target via MiniMax-Remover [31]. (absent → present). |
| **Obj. Removal** | 15,000 | Derived from filtered Señorita. Target generated via MiniMax-Remover [31]. Includes 5k multi-instance samples. (present → absent). |
| **Obj. Swap** | 15000 | Generated via VACE-14B [5] using GPT-4o prompts and Grounding DINO masks. Covers rigid & non-rigid swaps and 5k multi-instance object swap samples. |
| **Local Style** | 10000 | Generated via VACE-14B [5] using GPT-4o prompts and Grounding DINO masks. Focuses on texture & stylization. |
| **Total** | **50,000** | **Unified Dataset** |

as the source and the object-erased version as the target. Conversely, for the addition task (10k), we invert this pair (absent → present). Notably, the removal subset includes 5,000 samples featuring multi-instance objects to enhance model robustness in complex scenes.

Object Swap and Local Style: To capture fine-grained structural and textural changes, we generated 25,000 samples (15k for swap, 10k for style) utilizing VACE-14B [5]. The generation process is guided by GPT-4o for diverse prompt synthesis and Grounding DINO for precise mask extraction. The swap subset encompasses both rigid and non-rigid object replacements, while the local style subset focuses on texture modification and artistic stylization.

### 6.2. VideoCoF-Bench

**Benchmark Construction.** To strictly evaluate the generalization capability, we introduce VideoCoF-Bench, a diverse evaluation set specifically curated to have no overlap with the training domain. The benchmark is constructed from three distinct sources to ensure comprehensive coverage:

- **Pexels [13] Subset:** We manually curated a collection of high-quality videos from Pexels, comprising 50 samples for each editing task. These samples are balanced across the four core editing tasks (Addition, Removal, Swap, and Local Style) to test resolution adaptability and instruction following in varied scenes.
- **Standard Benchmark Integration:** To ensure a fair comparison with existing methods, we incorporated representative samples from established benchmarks, including EditVerse [6] and UNIC-Bench [25].
- **Adaptation for Fairness:** Notably, for samples sourced from UNIC-Bench (which typically involves ID-driven editing), we removed the reference identity images. This adaptation unifies the evaluation protocol, focusing purely on text-driven editing capabilities.

This combination results in a highly diverse benchmark that challenges models with unseen content and complex editing instructions.

## 7. Metrics

**GPT Evaluation.** To comprehensively assess the editing performance, we employ the state-of-the-art Vision-Language Model, GPT-4o [12], serving as an automated judge. Following the protocol of InstructX [11], we sample three frames from each video pair and utilize structured prompts in [11] to evaluate the results across the following dimensions:

- **Instruction Following (Score 1-10):** This metric measures the precision with which the edit adheres to the user's specific command. Higher scores indicate that the editing result strictly follows the prompt instructions without ambiguity.
- **Visual Quality (Score 1-10):** This evaluates whether the edited video is visually seamless, natural-looking, and aesthetically pleasing. It penalizes artifacts, distortions, or unnatural transitions introduced during the editing process.
- **Preservation (Score 1-10):** This assesses the coherence with the original video context. It strictly penalizes unintended changes to non-edited regions, ensuring the background and non-target objects remain intact.
- **Success Rate (Binary Yes/No):** To mitigate scoring variance, we incorporate a stricter discrete metric inspired by [9]. GPT-4o performs a binary judgment based on a rigorous three-step verification logic: (1) *Target Identification* (confirming the target matches the descriptor/position); (2) *Modification Accuracy* (verifying the specific edit is applied); and (3) *Strict Preservation* (ensuring no other instances are altered).

As presented in Table 6, VideoCoF achieves superior performance across all these metrics, validating the effectiveness of our reasoning-driven approach.

**Perception Quality.** In addition to semantic evaluation, we report quantitative metrics to measure the visual alignment and temporal consistency:

- **CLIP-T (Text-Image Alignment):** This metric assesses the semantic alignment between the editing instruction and the output video. We compute the cosine similarity between the CLIP [14] text embedding of the instruction

and the CLIP vision embedding of each output frame, reporting the average score across all frames.

- **CLIP-F (Frame-wise Consistency):** To evaluate temporal stability, we utilize the ViT-L/14 vision encoder from CLIP to extract features for each frame. The consistency score is calculated as the average cosine similarity between feature vectors of adjacent frames.
- **DINO (Structure Consistency):** While CLIP focuses on semantics, we aim to capture more fine-grained structural and textural consistency. We repeat the temporal consistency calculation using features extracted from a pre-trained DINOv2 [2] model. DINO's self-supervised training enables it to capture object-level details that might be overlooked by CLIP.

## 8. Future Directions

**Scaling up Chain-of-Frames.** Currently, VideoCoF achieves SOTA performance in instruction following and success rate using only 50k source-reasoning-editing pairs. This demonstrates remarkable data efficiency compared to existing large-scale baselines. For instance, EditVerse [6] utilizes 4M videos and 8M images, ICVE [9] leverages 2M pre-training data with 150k SFT samples, and InstructX [11] employs 200k SFT samples with joint training. Despite the significant gap in data scale, our method's superior performance suggests that the "reasoning-then-editing" paradigm is highly effective for Video Diffusion Models (VDMs). A promising future direction is to explore the performance ceiling of VideoCoF by scaling the dataset to 200k or even millions of samples. Investigating how the reasoning capabilities evolve with larger-scale data could reveal new upper limits for precise video editing.

**Joint Image-Video Editing and Efficient Architectures.** While our current work focuses on video data, integrating high-quality image editing datasets (e.g., MagicBrush [26], NHR-Edit[7]) presents a valuable opportunity. Many recent studies have shown that joint training can enhance visual quality and concept understanding. Future work could investigate the optimal mixture ratios between image and video datasets to maximize performance. Furthermore, designing unified and efficient attention mechanisms is crucial for handling the varying temporal dimensions of images and videos within a single model. Such advancements would likely improve the model's cross-modal learning capabilities, allowing it to transfer fine-grained editing skills from images to complex video dynamics.

**Generalizing VideoCoF to Broader Tasks.** VideoCoF has demonstrated exceptional performance in local editing tasks. However, the underlying reasoning framework is inherently flexible and can be extended to a wider range of applications. For *Global Editing* (e.g., style transfer), the reasoning frame could employ a full-frame gray mask to guide global transformations. For *ID-Driven Editing*, reference identity images could be integrated as "reasoning frames" to guide specific character insertions or swaps. Unifying these diverse tasks—ranging from local modifications to global stylization and ID injection—under the VideoCoF paradigm represents an exciting avenue for future exploration.

## References


[1] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023. 5

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 4

[3] Jiaxin Cheng, Tianjun Xiao, and Tong He. Consistent video-to-video transfer using synthetic dataset. *arXiv preprint arXiv:2311.00213*, 2023. 1, 5

[4] Lianghua Huang, Wei Wang, Zhi-Fan Wu, Yupeng Shi, Huanzhang Dou, Chen Liang, Yutong Feng, Yu Liu, and Jingren Zhou. In-context lora for diffusion transformers. *arXiv preprint arxiv:2410.23775*, 2024. 2

[5] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025. 3, 5

[6] Xuan Ju, Tianyu Wang, Yuqian Zhou, He Zhang, Qing Liu, Nanxuan Zhao, Zhifei Zhang, Yijun Li, Yuanhao Cai, Shaoteng Liu, et al. Editverse: Unifying image and video editing and generation with in-context learning. *arXiv preprint arXiv:2509.20360*, 2025. 3, 4

[7] Maksim Kuprashevich, Grigorii Alekseenko, Irina Tolstykh, Georgii Fedorov, Bulat Suleimanov, Vladimir Dokholyan, and Aleksandr Gordeev. Nohumansrequired: Autonomous high-quality image editing triplet mining. *arXiv preprint arXiv:2507.14119*, 2025. 4

[8] Xirui Li, Chao Ma, Xiaokang Yang, and Ming-Hsuan Yang. Vidtome: Video token merging for zero-shot video editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7486–7495, 2024. 1

[9] Xinyao Liao, Xianfang Zeng, Ziye Song, Zhoujie Fu, Gang Yu, and Guosheng Lin. In-context learning with unpaired clips for instruction-based video editing. *arXiv preprint arXiv:2510.14648*, 2025. 1, 2, 3, 4, 5

[10] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 1

[11] Chong Mou, Qichao Sun, Yanze Wu, Pengze Zhang, Xinghui Li, Fulong Ye, Songtao Zhao, and Qian He. Instructx: Towards unified visual editing with mllm guidance. *arXiv preprint arXiv:2510.08485*, 2025. 3, 4

[12] OpenAI. Hello gpt-4o. Blog post, 2024. 3

**Table 6. Quantitative full comparison over 4 video editing tasks on VideoCoF-Bench**. We compare VideoCoF with SOTA baselines: InsV2V [3]; Señorita [32] (an I2V model guided by an InstructPix2Pix [1] first frame); VACE-14B [5] (using GPT-4o generated captions); the concurrent work ICVE [9] (pre-trained 1M, fine-tuned 150k); and Lucy Edit Dev [18]. Despite extensive baseline training data, our VideoCoF is fine-tuned on only 50k source-reasoning-editing triplets and shows superior instruction following and success ratio.

| Model | GPT-4o Score | | | | Perceptual Quality | | |
|---|---|---|---|---|---|---|---|
| | Instruct Follow↑ | Preservation↑ | Quality↑ | Success Ratio↑ | CLIP-T↑ | CLIP-F↑ | DINO↑ |
| **Object Removal** | | | | | | | |
| InsV2V [3] | 3.11 | 4.02 | 3.77 | 3.92% | 26.85 | 0.984 | 0.973 |
| Señorita [32] | 3.11 | 4.68 | 4.38 | 9.80% | 26.96 | 0.995 | 0.990 |
| VACE [5] | N/A | N/A | N/A | 0.00% | 25.57 | **0.996** | 0.995 |
| ICVE [9] | 5.38 | 7.30 | **7.68** | 25.49% | 26.64 | 0.994 | 0.989 |
| Lucy Edit [18] | 2.06 | 4.09 | 4.45 | 1.96% | 27.37 | 0.992 | 0.988 |
| **VideoCoF (Ours)** | **9.65** | **7.35** | 6.94 | **86.27%** | **27.50** | 0.988 | **0.996** |
| **Object Addition** | | | | | | | |
| InsV2V [3] | 2.71 | 5.31 | 4.84 | 2.04% | 25.50 | 0.985 | 0.966 |
| Señorita [32] | 2.63 | 5.43 | 4.80 | 6.12% | 25.26 | 0.990 | 0.981 |
| VACE [5] | 7.12 | 5.40 | 7.38 | 30.61% | 28.01 | **0.990** | 0.980 |
| ICVE [9] | 8.95 | 8.65 | **8.33** | 77.55% | 29.13 | 0.987 | 0.974 |
| Lucy Edit [18] | 6.96 | 7.29 | 6.78 | 44.90% | 27.39 | 0.987 | 0.978 |
| **VideoCoF (Ours)** | **9.12** | **8.78** | 8.27 | **79.59%** | **29.60** | 0.988 | **0.982** |
| **Object Swap** | | | | | | | |
| InsV2V [3] | 1.52 | 7.37 | 6.54 | 0.00% | 26.22 | 0.991 | 0.984 |
| Señorita [32] | 1.69 | 7.39 | 6.40 | 0.00% | 25.97 | 0.994 | 0.990 |
| VACE [5] | 8.11 | 6.53 | 7.79 | 34.62% | 26.93 | **0.995** | 0.992 |
| ICVE [9] | 9.08 | **8.40** | **8.57** | 73.08% | 26.54 | 0.993 | 0.989 |
| Lucy Edit [18] | 6.81 | 7.58 | 7.50 | 44.23% | 26.46 | 0.992 | 0.988 |
| **VideoCoF (Ours)** | **9.10** | 8.39 | 8.14 | **80.77%** | **27.10** | 0.993 | **0.996** |
| **Local Style Transfer** | | | | | | | |
| InsV2V [3] | 6.29 | 7.89 | 6.89 | 19.61% | 26.19 | 0.992 | 0.987 |
| Señorita [32] | 5.60 | 7.69 | 6.33 | 25.49% | 25.97 | 0.995 | 0.992 |
| VACE [5] | 7.18 | 5.53 | 7.65 | 41.18% | 27.56 | 0.996 | **0.994** |
| ICVE [9] | 7.75 | 7.89 | **7.98** | 54.90% | 27.64 | 0.994 | 0.991 |
| Lucy Edit [18] | 5.12 | 7.05 | 6.73 | 27.45% | 26.71 | 0.993 | 0.992 |
| **VideoCoF (Ours)** | **8.02** | **8.29** | 7.71 | **58.82%** | **27.80** | **0.997** | 0.991 |


[13] Pexels. Pexels: Free stock photos, royalty free stock images & videos. https://www.pexels.com/, 2025. Accessed: 2025-11-06. 3

[14] 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. 3

[15] Chaehun Shin, Jooyoung Choi, Heeseung Kim, and Sungroh Yoon. Large-scale text-to-image model with inpainting is a zero-shot subject-driven image generator. 2024. 2

[16] Uriel Singer, Amit Zohar, Yuval Kirstain, Shelly Sheynin, Adam Polyak, Devi Parikh, and Yaniv Taigman. Video editing via factorized diffusion distillation. In *European Conference on Computer Vision*, pages 450–466. Springer, 2024. 1

[17] Wenhao Sun, Rong-Cheng Tu, Jingyi Liao, and Dacheng Tao. Diffusion model-based video editing: A survey. *arXiv preprint arXiv:2407.07111*, 2024. 1

[18] DecartAI Team. Lucy edit: Open-weight text-guided video editing. 2025. 5

[19] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability. *Advances in Neural Information Processing Systems*, 36:7594–7611, 2023. 1

[20] Bichen Wu, Ching-Yao Chuang, Xiaoyan Wang, Yichen Jia,

Kapil Krishnakumar, Tong Xiao, Feng Liang, Licheng Yu, and Peter Vajda. Fairy: Fast parallelized instruction-guided video-to-video synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8261–8270, 2024. 1

[21] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7623–7633, 2023. 1

[22] Jay Zhangjie Wu, Xiuyu Li, Difei Gao, Zhen Dong, Jinbin Bai, Aishani Singh, Xiaoyu Xiang, Youzeng Li, Zuwei Huang, Yuanxi Sun, et al. Cvpr 2023 text guided video editing competition. *arXiv preprint arXiv:2310.16003*, 2023. 1

[23] Zhen Xing, Qi Dai, Han Hu, Zuxuan Wu, and Yu-Gang Jiang. Simda: Simple diffusion adapter for efficient video generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7827–7839, 2024. 1

[24] Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. Space-time diffusion features for zero-shot text-driven motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8466–8476, 2024. 1

[25] Zixuan Ye, Xuanhua He, Quande Liu, Qiulin Wang, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, Qifeng Chen, and Wenhan Luo. Unic: Unified in-context video editing. *arXiv preprint arXiv:2506.04216*, 2025. 3

[26] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 36:31428–31449, 2023. 4

[27] Zechuan Zhang, Ji Xie, Yu Lu, Zongxin Yang, and Yi Yang. In-context edit: Enabling instructional image editing with in-context generation in large scale diffusion transformer. *arXiv preprint arXiv:2504.20690*, 2025. 2

[28] Min Zhao, Guande He, Yixiao Chen, Hongzhou Zhu, Chongxuan Li, and Jun Zhu. Riflex: A free lunch for length extrapolation in video diffusion transformers. *arXiv preprint arXiv:2502.15894*, 2025. 1

[29] Min Zhao, Hongzhou Zhu, Yingze Wang, Bokai Yan, Jintao Zhang, Guande He, Ling Yang, Chongxuan Li, and Jun Zhu. Ultravico: Breaking extrapolation limits in video diffusion transformers. *arXiv preprint arXiv:2511.20123*, 2025. 1

[30] Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jia-Wei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. Motiondirector: Motion customization of text-to-video diffusion models. In *European Conference on Computer Vision*, pages 273–290. Springer, 2024. 1

[31] Bojia Zi, Weixuan Peng, Xianbiao Qi, Jianan Wang, Shihao Zhao, Rong Xiao, and Kam-Fai Wong. Minimax-remover: Taming bad noise helps video object removal. *arXiv preprint arXiv:2505.24873*, 2025. 2, 3

[32] Bojia Zi, Penghui Ruan, Marco Chen, Xianbiao Qi, Shaozhe Hao, Shihao Zhao, Youze Huang, Bin Liang, Rong Xiao, and Kam-Fai Wong. Se\~ norita-2m: A high-quality instruction-based dataset for general video editing by video specialists. *arXiv preprint arXiv:2502.06734*, 2025. 5