

Unified Video Editing as Temporal Reasoner

Anonymous CVPR submission

Paper ID 957



Figure 1. VideoCoF’s video editing capabilities emerge from its **seeing, reasoning, then editing framework**. Trained on only **50k** data (33 frames), this teaser shows multi-instance editing and robust $4\times$ length generalization.

Abstract

Existing video editing methods face a critical trade-off: expert models offer precision but rely on task-specific priors like masks, hindering unification; conversely, unified

temporal in-context learning models are mask-free but lack explicit spatial cues, leading to weak instruction-to-region mapping and imprecise localization. To resolve this conflict, we propose **VideoCoF**, a novel **Chain-of-Frames** approach inspired by Chain-of-Thought reasoning. VideoCoF

004
005
006
007
008

enforces a “see → reason → edit” procedure by compelling the video diffusion model to first predict **reasoning tokens** (edit-region latents) before generating the target video tokens. This explicit reasoning step removes the need for user-provided masks while achieving precise instruction-to-region alignment and fine-grained video editing. Furthermore, we introduce a RoPE alignment strategy that leverages these reasoning tokens to ensure motion alignment and enable length extrapolation beyond the training duration. We demonstrate that with a minimal data cost of only 50k video pairs, VideoCoF achieves state-of-the-art performance on VideoCoF-Bench, validating the efficiency and effectiveness of our approach.

1. Introduction

The development of Video Diffusion Models (VDM) [12, 27, 33, 37] has enabled high-fidelity video generation across a wide range of concepts. Building on these advances, video editing methods support users in designing video by adding [26], removing [15, 43], swapping [6, 36] visual concepts, and performing global style transformation [39].

Current video editing methods mainly follow two strategies: (i) **expert models** [1, 15, 26, 36, 41], which use adapter-based modules to feed *external masks* into the video generation model, yielding precise, localized edits but requiring additional inputs and per-task overhead; and (ii) **unified temporal in-context learning models** [9, 16, 38], which concatenate source tokens with noised edit tokens along the temporal dimension and use self-attention mechanism to guide the edit. However, without explicit spatial cues, these models often exhibit weak accuracy, especially in cases that need multi-instance recognition or spatial reasoning (Fig. 2, left). In short, there is a *trade-off*: expert models are accurate but mask-dependent, while unified in-context models are mask-free but less precise; This raises a critical question: **Can we maintain former’s precision and latter’s unification without the mask dependency?**

Inspired by Chain-of-Thought (CoT) multi-step reasoning [30], we *compel* the video diffusion model to first predict the edit region and then perform the edit, enforcing a “**see → reason → edit**” procedure. Accordingly, we propose **VideoCoF**, a Chain-of-Frames approach that predicts **reasoning tokens** (edit-region latents) before generating the target video tokens, thereby removing the need for user-provided masks while achieving precise instruction-to-region alignment. To explicitly model the reasoning process, we leverage visual grounding, which is naturally suited to simulating reasoning about the edit region. Empirically, we find a soft, gradually highlighted grayscale region is the most effective reasoning format. Additionally, we introduce a RoPE alignment strategy. By explicitly accounting for the reasoning latent, we reset the temporal indices

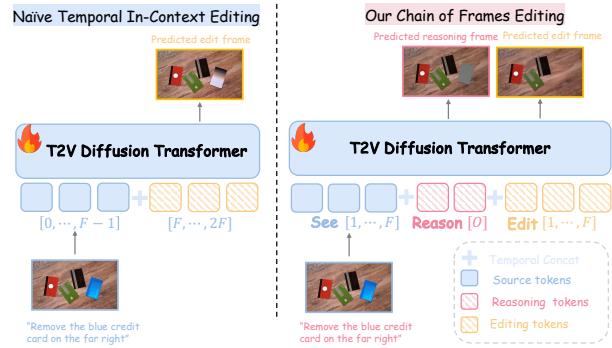


Figure 2. Illustration of the difference between previous methods and our VideoCoF. We enhances the editing accuracy by forcing the video diffusion model to first predict the editing area, and then perform the editing.

of the edited video’s rotary position embeddings to match those of the source segment, ensuring motion alignment and length extrapolation.

To holistically evaluate fine-grained video editing, we further construct VideoCoF-Bench. VideoCoF trained on only **50k** video pairs, outperforms a strong baseline ICVE [16] that uses $\sim 1M$ pretraining videos plus 150k for SFT. Specifically, we improves the instruction-following score by **+15.14%** and the success ratio by **+18.8%**. Our contributions can be summarized as follows:

- We propose VideoCoF, the first framework to introduce a Chain of Frames approach to video editing, enabling temporal reasoning for fine-grained video editing.
- Building on VideoCoF, we explore an effective reasoning format for video diffusion models, and introduce a RoPE alignment strategy that allowing generalization to longer frames exceeding the training duration.
- We demonstrate that with a minimal data cost (only **50k** video pairs), we achieve state-of-the-art quantitative and qualitative performance on VideoCoF-Bench, validating the efficiency and effectiveness of our approach.

2. Related Work

Video Editing Methods. Early training-free video editing methods [21, 33] used inversion and consistency techniques (e.g., attention manipulation [21] or optical flow [5]) but often lack precise control and struggle with complex edits. Data-driven, training-based methods [2, 4] have become the focus, offering higher quality and edit diversity. A concurrent line of research [18, 29, 40] integrates MLLMs to guide the editing process, though this adds significant training and inference cost, which our pure VDM approach avoids.

In-Context Video Editing. Recently, in-context learning (ICL) has emerged as a promising paradigm for unified editing [10, 35, 42]. Methods like UNIC [38] and ICVE [16] concatenate video conditions along the temporal axis

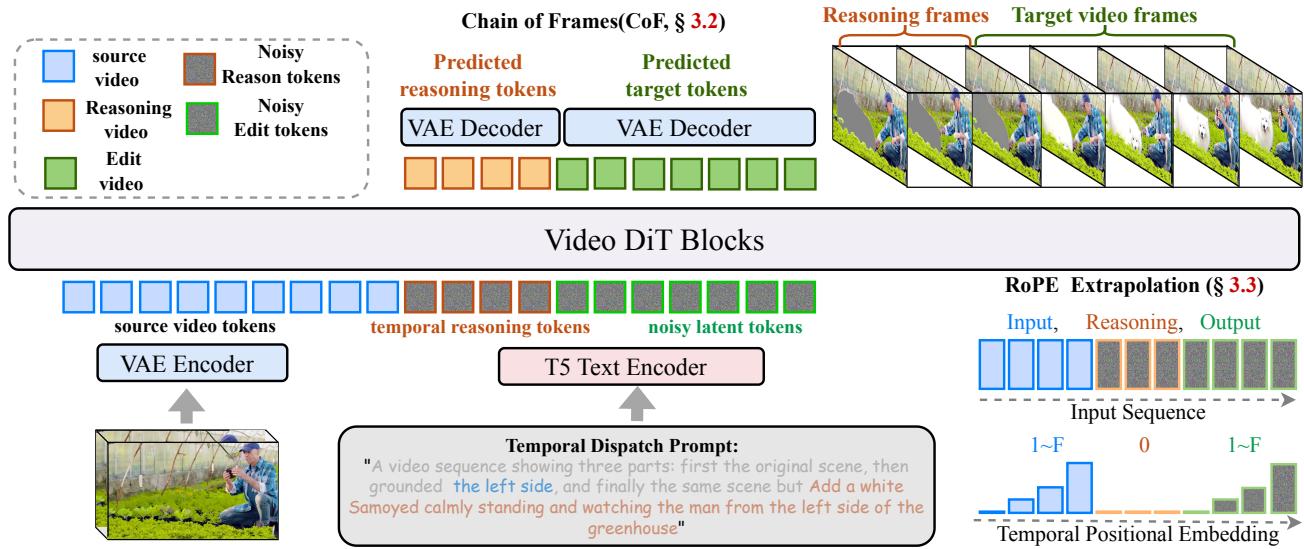


Figure 3. Overview of VideoCoF framework. Our model processes source (blue), reasoning (orange), and target (green) tokens in a unified sequence to “reason” then “edit”. **Bottom right:** Our RoPE design enables length extrapolation.

to perform ICL. However, these methods are often limited by mask requirements [38] or, as we identify, suffer from fundamental issues with editing accuracy and a lack of length extrapolation due to their naive temporal concatenation. While EditVerse [9] also explored unified in-context learning, it was built on a LLaMA-style DiT backbone, whereas our work explores these capabilities within a standard video diffusion transformer.

Chain of Thought in Vision. Chain-of-Thought (CoT) prompting [11, 30] elicits multi-step reasoning in LLMs by having them “think step-by-step.” This concept of emergent reasoning has also been identified in large video generative models [3, 31] that can solve visual puzzles. However, how to leverage visual reasoning for the task of unified video editing remains unexplored. In this work, we investigate whether generative video models can perform a “chain of frames” reasoning to achieve this.

3. Methods

3.1. VideoCoF Framework

As illustrated in Figure 3, VideoCoF employs a VideoDiT [27] for unified video editing. We model editing as a reasoning-then-generation process: the model first reasons where to edit, then generates the intended content in that area. We call this process “**Chain of Frames (CoF)**” (Sec 3.2). All visual inputs (source, reasoning, and target frames) are encoded separately by a Video VAE and then concatenated temporally. The unified frame sequence is then fed into the model, performing unified in-context learning via self-attention and language control via cross-attention. To enable video alignment and variable-length inference, we revisit the design of positional encoding. We adapt the tem-

poral RoPE for source-to-target alignment and reasoning tokens’ RoPE for explicit spatial guidance (Sec 3.3). Subsequent sections detail the insights behind our design choices and data curation pipeline (Sec 3.5).

3.2. Chain of Frames

Seeing, Reasoning, then Editing. Previous video in-context editing methods, such as UNIC [38], ICVE [16], or EditVerse [9], perform in-context learning by temporally concatenating clean source video tokens with noised editing video tokens. However, this approach lacks an explicit constraint mapping the editing instruction to the specific editing region, leading to editing accuracy problems, as shown in Fig 2. Recently, VDM have been shown to possess reasoning capabilities, as demonstrated in [31]. Inspired by this, we explicitly model the reasoning tokens, forcing the model to actively learn the relationship between the editing instruction and the target edit region first. The edit is then executed *after* reasoning, following a “see, reason, then edit” process.

Inspired by Chain of Thought prompting in Large Language Models (LLMs) [30], we argue that a video generative model should also have an analogous chain-reasoning ability. Given the generative priors in video editing, the visual-chain should be progressive, moving from the original video to a visual reference of the editing region, and finally to the edited video. Visual grounding is naturally suitable for this representation. Since video diffusion models are often insensitive to grounding masks (black or white pixels). Therefore, we choose to use a gray highlight to delineate the “grounding region,” which is also evidence in [7]. Finally, the gray-highlighted area as the ground truth for the reasoning frames, teaching the diffusion model to

126
127
128
129

130

131
132
133
134
135
136
137
138
139
140
141
142
143
144

145
146
147
148
149
150
151
152
153
154
155
156
157

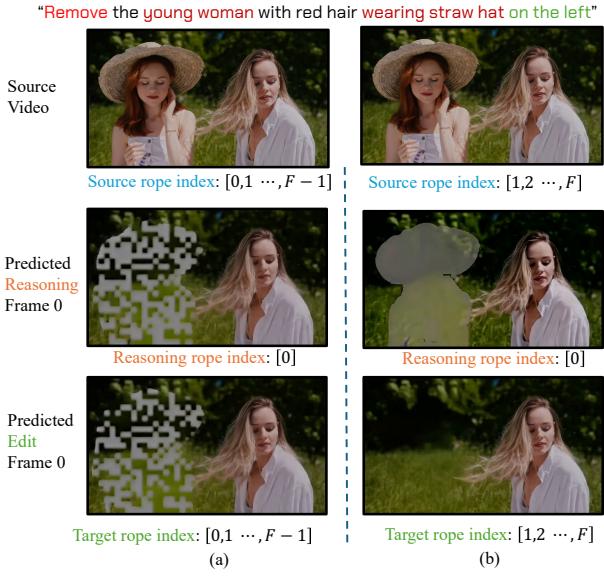


Figure 4. How our RoPE design avoid index collision.

158

reason about where the edit should occur.

159 Consequently, the entire video editing task is reformulated as a chained process: first “seeing” the original video,
 160 then “reasoning” by predicting the grounding region, and
 161 finally “editing” to generate the new video content within
 162 that specified area. We call this **Chain of Frames (CoF)**.

163
 164 Let $\mathcal{E}(\cdot)$ denote the video VAE encoder. We use F
 165 and L for frames in the source/target and reasoning latent
 166 space, respectively, and denote channel, height, and width
 167 by C , H , and W . Given a triplet source-reasoning-target
 168 video pair $\{\mathbf{s}, \mathbf{r}, \mathbf{e}\}$, we first encode them into latent rep-
 169 resentations. The source \mathbf{s} and target video \mathbf{e} yield latent
 170 $z_s = \mathcal{E}(\mathbf{s})$ and $z_e = \mathcal{E}(\mathbf{e})$, both with shape $\mathbb{R}^{F \times C \times H \times W}$.
 171 The reasoning video \mathbf{r} yields a latent $z_r = \mathcal{E}(\mathbf{r})$ with shape
 172 $\mathbb{R}^{L \times C \times H \times W}$. This separate encoding ensures intra-causal
 173 relations and inter-video independence. Then, we perform
 174 temporal concatenation to get the unified latent representa-
 175 tion:

$$\mathbf{z}_{full}^{(t)} = \underbrace{z_s^{(0)}}_{\text{seeing}} \parallel \underbrace{z_r^{(t)}}_{\text{reasoning}} \parallel \underbrace{z_e^{(t)}}_{\text{editing}} \in \mathbb{R}^{(F+L+F) \times C \times H \times W}, \quad (1)$$

176
 177 where the $z_s = \mathbf{z}_{0:F-1}^{(0)}$ denotes anchoring the source
 178 video latent at timestep 0. $z_r = \mathbf{z}_{F:F+L-1}^{(t)}$ and
 179 $z_e = \mathbf{z}_{F+L:2F+L-1}^{(t)}$ mean the reasoning and target noised
 180 video latents at timestep t . At each denoising step, only the
 181 $L + F$ reasoning and target frames are denoised, and the
 182 source video latents are kept clean.

3.3. RoPE Design for Length Extrapolation

183

In VideoDiT, 3D factorized RoPE [23] provides spatio-temporal positions. A naive in-context learning approach applies sequential temporal indices (e.g., 0 to $2F - 1$) across concatenated source and target videos. However, this hinders video length extrapolation, as the model overfits to a static $[0, F - 1] \rightarrow [F, 2F - 1]$ mapping and fails to generalize to videos longer than F frames.

184

A better strategy is to repeat the temporal indices. For our CoF triplet (consider $L = 1$ for reasoning frame), a straightforward reset configuration is to assign temporal indices: $[0, F - 1]$ to the source, “0” to the reasoning frame, and $[0, F - 1]$ to the target.

185

However, as illustrated in Figure 4 (a), this naive reset leads to index collisions at temporal position 0, shared by the source, reasoning, and target frames. This overlap introduces visual artifacts that propagate from the reasoning tokens into the first target frame.

186

To resolve this index collision, we set the temporal indices for both the source video and the target video to the range $[1, F]$, while keeping the reasoning frame’s temporal index at 0. This isolates the reasoning token and prevents artifact leakage while maintaining length generalization.

187

3.4. Training and Inference Paradigm

188

Algorithm 1 Chain of Frame (CoF) Training

Input: Dataset \mathcal{D} with tuples $(\mathbf{z}_s^{(0)}, \mathbf{z}_r^{(0)}, \mathbf{z}_e^{(0)}, \mathbf{c})$

Output: Fine-tuned parameters θ

```

foreach minibatch  $(\mathbf{z}_s^{(0)}, \mathbf{z}_r^{(0)}, \mathbf{z}_e^{(0)}, \mathbf{c}) \sim \mathcal{D}$  do
  foreach sample in minibatch do
     $\mathbf{z}_{full}^{(0)} \leftarrow \mathbf{z}_s^{(0)} \parallel \mathbf{z}_r^{(0)} \parallel \mathbf{z}_e^{(0)}$  Sample  $t \sim \mathcal{U}[0, 1]$ 
    Sample  $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$  with the same shape  $\mathbf{z}_{full}^{(0)}$   $\mathbf{v} \leftarrow (\varepsilon - \mathbf{z}_{full}^{(0)})$ 
     $\mathbf{z}_{r,e}^{(t)} \leftarrow (1 - t)(\mathbf{z}_r^{(0)} \parallel \mathbf{z}_e^{(0)}) + t(\varepsilon_{F:2F+L-1})$   $\mathbf{z}^{(t)} \leftarrow \mathbf{z}_s^{(0)} \parallel \mathbf{z}_{r,e}^{(t)}$ 
     $\hat{\mathbf{v}} \leftarrow \mathbf{F}_\theta(\mathbf{z}^{(t)}, t, \mathbf{c})$ 
     $\mathcal{L} \leftarrow \frac{1}{L+F} \sum_{i=F}^{2F+L-1} \|\mathbf{v}_i - \hat{\mathbf{v}}_i\|_2^2$ 
  Update  $\theta$  using gradients of  $\mathcal{L}$ 

```

Given a concatenated full latent sequence $\mathbf{z}_{full}^{(0)} =$
 TemporalConcat $(\mathbf{z}_s^{(0)}, \mathbf{z}_r^{(0)}, \mathbf{z}_e^{(0)})$, we treat the reasoning+editing block as the generation target during training.

207

Given timestep $t \in [0, 1]$ and Gaussian noise $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, we only progressively noise the reasoning and editing parts, $\mathbf{z}_{r,e}^{(t)} = (1 - t)(\mathbf{z}_r^{(0)} \parallel \mathbf{z}_e^{(0)}) + t\varepsilon_{F:2F+L-1}$, and form the model input $\mathbf{z}^{(t)} = \mathbf{z}_s^{(0)} \parallel \mathbf{z}_{r,e}^{(t)}$. The target velocity field is $\mathbf{v} = \varepsilon - \mathbf{z}_{full}^{(0)}$. Our model $\mathbf{F}_\theta(\cdot)$ predicts this velocity field from the partially noised input, and we train it by minimizing the mean squared error between predicted

208

209

210

211

212

213

214

215

216

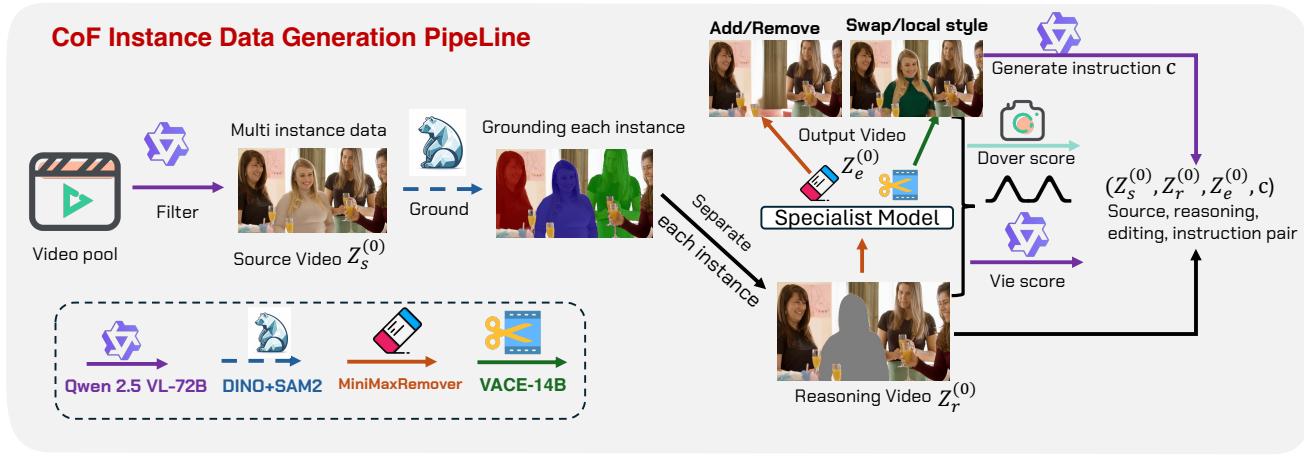


Figure 5. Our data curation pipeline for multi-instance data.

and true velocities. Concretely, we only supervise the reasoning and target frames, so the training loss can be written in per-frame form as

$$\mathcal{L} = \frac{1}{L+F} \sum_{i=F}^{2F+L-1} \left\| \mathbf{v}_i - [\mathbf{F}_\theta(\mathbf{z}^{(t)}, t, \mathbf{c})]_i \right\|_2^2, \quad (2)$$

where $[\mathbf{F}_\theta(\mathbf{z}^{(t)}, t, \mathbf{c})]_i$ denotes the model’s prediction for frame i and \mathbf{c} is the text condition. The model parameters $\mathbf{F}_\theta(\cdot)$ are updated via a gradient step computed from this loss. The full training procedure is summarized in Algorithm 1.

During inference we initialize the reasoning+editing block from Gaussian noise, $\mathbf{z}_{r,e}^{(1)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and form the full latent at $t = 1$ by temporal concatenation with the clean source $\mathbf{z}_{\text{full}}^{(1)} = \text{TemporalConcat}(\mathbf{z}_s^{(0)}, \mathbf{z}_{r,e}^{(1)})$. An ODE solver guided by our model \mathbf{F}_θ evolves $\mathbf{z}_{\text{full}}^{(t)}$ to $\mathbf{z}_{\text{full}}^{(0)}$. The source latents $\mathbf{z}_s^{(0)}$ are held fixed during inference, so only the reasoning/editing parts change. We then extract the edited-target latent using the same slicing index as in training: $\mathbf{z}_{\text{edit}}^{(0)} = (\mathbf{z}_{\text{full}}^{(0)})_{F+L:2F+L-1}$ and decode the final edited video: $\mathbf{x}_{\text{edit}} = \mathcal{D}(\mathbf{z}_{\text{edit}}^{(0)})$.

3.5. Video Data Curation

The training of our VideoCoF requires a large and diverse dataset structured as source, reasoning, and edited video triplets. However, existing video editing datasets and methods predominantly focus on single-instance-level object manipulation. This limitation is a significant barrier, as real-world videos contain complex visual cues, multiple interacting instances, and intricate spatial relationships (e.g., physical left/right, object-to-object interactions). Enabling a generative model to comprehend these complex, instance-level dynamics is a critical step toward true reasoning-based video editing. Therefore, we develop a comprehensive data curation pipeline, illustrated in Figure 5, to specifically generate and process complex, instance-level video data.

Instance-Level Curation Pipeline. Our pipeline begins with a large pool of diverse videos sourced from Pexels [20]. First, we employ the Qwen-VL 72B [28] to perform multi-instance identification, scanning the videos to find scenes that contain multiple, distinct objects. Once these videos are identified, we use Grounding-SAM2 [22] to perform precise segmentation, generating distinct segmentation masks for each individual instance. With these instance-specific masks, we generate triplets for a variety of editing tasks:

- **Object Addition/Removal:** We utilize the Minimaxremover [43] to erase a specific instance from the video. The data for object addition is then created by simply reversing this process.
- **Object Swap and Local Style Transfer:** For these tasks, we leverage the VACE 14B [8] in its inpainting mode to fill the specified masked regions. Critically, the creative prompts for these inpainting edits are generated by GPT-4o [19], as we found Qwen-VL 72B’s imaginative capabilities for this specific task to be limited.

Filtering and Final Dataset. All generated video pairs are rigorously evaluated to ensure quality. We use the Dover Score [32] to assess aesthetic quality and the VIE Score [13] to measure editing fidelity and coherence. A weighted combination of these scores is used to filter for high-quality, successful edits. Finally, we use this pipeline to filter from the large-scale open-source Señorita 2M [44] dataset, and distill a high-quality subset of **50k** videos to supplement our training data. This multi-pronged approach yields our final large-scale dataset, rich in the instance-level complexity required for reasoning-based video editing.

4. Experiments

4.1. Implementation Details.

VideoCoF is trained on WAN-14B [27]. We employ a resolution-bucketing strategy to support multiple aspect ratios, using spatial resolutions of 336×592, 400×704,

250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285

Model	GPT-4o Score (avg.)				Perceptual Quality (avg.)		
	Instruct Follow↑	Preservation↑	Quality↑	Success Ratio↑	CLIP-T↑	CLIP-F↑	DINO↑
InsV2V [4]	3.41	6.15	5.51	6.39%	26.19	0.988	0.978
Señorita [44]	3.26	6.30	5.48	10.35%	26.04	0.994	0.988
VACE [8]	7.47	5.82	7.61	26.60%	27.02	0.994	0.990
ICVE [16]	7.79	8.06	8.14	57.76%	27.49	0.992	0.986
Lucy Edit [25]	5.24	6.50	6.37	29.64%	26.98	0.991	0.986
VideoCoF (Ours)	8.97	8.20	<u>7.77</u>	76.36%	28.00	0.992	0.991

Table 1. We compare VideoCoF with SOTA baselines on VideoCoF-Bench: InsV2V [4], Señorita [44] (an I2V model guided by InsP2P [2]), VACE-14B [8] (using GPT-4o-generated captions), the concurrent ICVE [16] (pretrained on 1M videos and fine-tuned on 150k), and LucyEdit [25]. Despite the extensive training data used by baselines, VideoCoF is fine-tuned on only 50k video pairs and achieves superior instruction-following and success ratio.

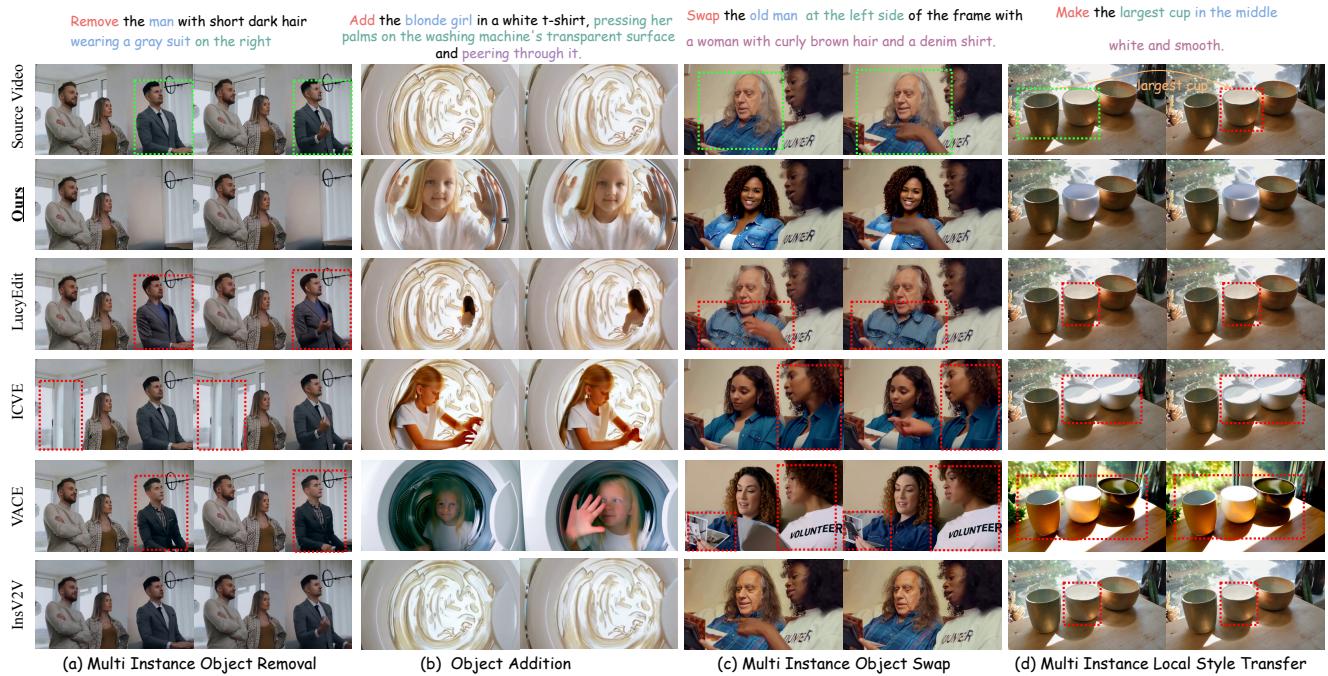


Figure 6. Visual comparsion between our VideoCoF and other methods on diverse video editing tasks.

400×752, and 400×944 (and the corresponding vertical variants, e.g., 592×336). Training videos are sourced from Señorita [44] and are 33 frames long, we only training on **50k** curated video data finally. Thanks to our RoPE alignment design, the model generalizes to longer sequences at inference (e.g., **141 frames and above**). By default we use 33 frames source video, 33 frames edited video, and 4 frames reasoning clip. We train with a global batch size of 16 for approximately 8k iterations, optimizing with AdamW [17] and a base learning rate of 1×10^{-4} .

4.2. VideoCoF-Bench and Experimental Setting

VideoCoF-Bench. Previous video-editing benchmarks such as V2VBench [24], TGVE [34], and FIVE-Bench [14] focus on target-prompt edits and mostly are focused on

class-level object swap. They were mainly designed for training-free methods and are not suitable for instruction-guided or instance-level video editing. Real-world editing requires precise instruction understanding, including instance- and part-level control (e.g., distinguishing multiple people or left vs. right), and complex reasoning. To address these gaps, we introduce VideoCoF-Bench. It contains 200 high-quality videos collected from Pexels [20], covering diverse scenes and both landscape and portrait aspect ratios. VideoCoF-Bench includes four task: Object Removal, Object Addition, Object Swap, and Local Style Transfer, each with 50 samples. Half of these samples per task are instance-level cases with instance-focused editing prompts.

Evaluation Metrics. To evaluate editing performance on

300
301
302
303
304
305
306
307
308
309
310
311
312
313
314

Ablation on Chain of frames and RoPE design			
	Naive Temporal in Context	VideoCoF	
CoF	\times	\times	✓
RoPE Design	0–2F–1	0–F–1, 0–F–1	1–F, 0, 1–F
<i>GPT-4o Score</i>			
Instruct Follow↑	8.109	8.064	8.973
Preservation↑	7.930	7.793	8.203
Quality↑	7.394	7.217	7.765
Success Ratio↑*	72.41%	65.52%	76.36%
<i>Perceptual Quality</i>			
CLIP-T↑	26.880	27.088	28.000
CLIP-F↑	0.9907	0.9905	0.9915
DINO↑	0.9857	0.9826	0.9913

Table 2. Ablation on Chain of frames and RoPE design.

315 VideoCoF-Bench, we employ MLLM-as-a-Judge to pro-
 316 vide a holistic evaluation score. This is achieved by prompt-
 317 ing **GPT-4o** [19] to assess multiple criteria given the origi-
 318 nal video, edited video, and user instruction: (1) Instruction
 319 Following (editing accuracy), (2) Preservation (unedited re-
 320 gions), (3) Video Quality. (4) Success ratio: we prompt
 321 the GPT-4o to provide a binary Success Ratio (Yes/No) to
 322 judge the overall success of the edit. We report three percep-
 323 tual quality metrics quantify low- and high-level visual simi-
 324 larity between source and target frames: CLIP-T for im-
 325 age-text alignment, CLIP-F for temporal consistency, and
 326 DINO for structural consistency.

327 4.3. Comparison on VideoCoF-Bench

328 We show qualitative and quantitative comparisons of
 329 VideoCoF-Bench in this section. As shown in Table 1,
 330 we evaluate VideoCoF against five baseline methods on
 331 the VideoCoF-Bench benchmark, which spans four distinct
 332 video editing tasks: multi-instance removal, object addition,
 333 multi-instance swap, and multi-instance local style transfer.
 334

335 Overall, VideoCoF demonstrates the best performance
 336 in **Instruct Follow** and **Success Ratio** across all categories.
 337 Compared to naive temporal in-context editing approaches
 338 like ICVE [16], our method achieves significantly higher
 339 success rates and better instruction adherence using only
 340 **50k** reasoning pairs, whereas ICVE is pre-trained on 1M
 341 samples and fine-tuned on 150k data.

342 Qualitatively (see Figure 6), our method also shows
 343 clearer, more faithful edits at the instance level: (a) Multi-
 344 instance removal: we precisely remove the right instance
 345 while ICVE[16] incorrectly removes the left instance. (b)
 346 Object addition: the added girl is correctly placed inside
 347 the washing machine, matching the instruction. (c) Object
 348 swap: we replace the elderly person’s face and update cloth-
 349 ing; Lucy Edit [25] changes only clothing, ICVE fails to dis-
 350 ambiguate instances, and VACE often alters non-target peo-
 351 ple. (d) Local style (multi-instance): our model correctly
 352 identifies and edits the largest cup among several similar ob-



Figure 7. Length exploration on frames more than training

jects; other methods either fail to edit or mistakenly edit a
 bowl. These qualitative examples demonstrate VideoCoF’s
 stronger instance-level reasoning and higher editing fidelity.

352 4.4. Ablation Study

353 To verify our novel Chain of Frames (CoF) design, particu-
 354 larly its “reasoning frames” and the RoPE design for length
 355 exploration, we conduct an ablation study on the reasoning
 356 frames, RoPE alignment strategy and reasoning format.

357 **Naive Temporal Incontext VS. CoF.** As shown in Ta-
 358 ble 2, we compare VideoCoF against a “Naive Temporal in-
 359 context” baseline. This applies temporal in-context learning
 360 by using the source video as a condition through temporal
 361 concatenation, an approach similar to ICVE [16].

362 In contrast, our approach introduces **reasoning frames**
 363 as a core component of the (CoF) design. This ensures the
 364 video editing follows a reasoning process, i.e., forcing the
 365 model to predict the editing region first and then execute the
 366 versatile edit within that specific area.

367 The efficacy of this design is evident when comparing
 368 the first ($[0, 2F - 1]$) and third (VideoCoF) columns in Ta-
 369 ble 2. The inclusion of CoF brings substantial gains: the
 370 instruct follow score increases by 10.65% and the success
 371 ratio improves by 5.46%. Furthermore, the 4.16% increase
 372 in CLIP-T confirms that our reasoning frames effectively
 373 enhance the model’s editing accuracy and precision.

374 **Rope Design for length Extrapolation.** As illustrated in
 375 Fig 7, the naive approach ($[0, 2F - 1]$) only learns a fixed
 376 temporal mapping (e.g., mapping frame 0_{th} to frame 33_{th}).
 377 This prevents length extrapolation, causing severe degra-
 378 dation (blurriness, motion misalignment, and artifacts) when a
 379 33-frame trained model is tested on 81 frames (second row).

380 In contrast, our RoPE alignment design ($[1 - F, 0, 1 - F]$)
 381 generalizes to unseen lengths without quality degradation
 382 (third row). As demonstrated in Fig 1, our model extrap-
 383 olates to 141 frames (4x training length) and beyond, sup-
 384 porting theoretically infinite extrapolation.

385 This effectiveness is also quantified in Table 2 (third vs.
 386 first column). We observe a 3.4% relative increase in the
 387 preservation score. Furthermore, the improved DINO score
 388 confirms that our RoPE design better preserves the original
 389 video’s spatio-temporal structure during editing.

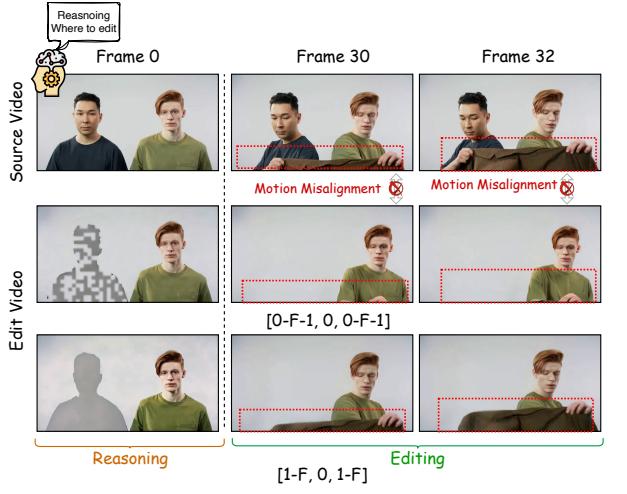


Figure 8. Motion alignment benefit by our rope design

393 **RoPE Design for Motion Alignment.** Setting the temporal
394 index for the reasoning frame latent is a critical design
395 choice. A naive approach is to set its index to 0, aligning it
396 with the first video frame. This causes two severe issues.
397

398 First, it leads to significant motion misalignment (e.g.,
399 the subject fails to perform the "lifting clothes" motion in
400 Fig 8, second row). Second, this "0-index" design causes
401 interference with the first editing video frame (also index
402 0), leading to artifacts where the model incorrectly predicts
403 the first frame as the reasoning frame (Fig 4).

404 Therefore, we fix the reasoning latent's index to 0, while
405 the source and edited video indices range from 1 to F (de-
406 noted as $[1 - F, 0, 1 - F]$). This strategy allows the
407 reasoning frame to provide clear spatial guidance on **where** to
408 edit, without disrupting the video's temporal structure and
409 motion alignment. The improvements across all metrics in
410 Tab 2 (column 3 vs. column 2) validate this design.

411 **Reasoning Frame Format.** First, we explore the most suit-
412 able color for the reasoning frame mask. As shown in Ta-
413 ble 3, we compare three formats: (1) A black mask over the
414 unedit region; (2) A red, 50% transparent highlight, same
415 as veggie [40]; and (3) A pure gray mask (value 127, 0%
416 transparency). The quantitative results show that using a
417 gray mask (column 3) for the edit region yields the best per-
418 formance.

419 Furthermore, we argue that the reasoning frame should
420 act as a gradual transition from the source video to the
421 edited video. Therefore, we test progressive gray mask. In-
422 stead of a single static mask, we interpolate gray mask rea-
423 soning frame and editing frame, with transparency pro-
424 gressively increased (e.g., 0%, 25%, 50%, 75%). As shown
425 by comparing column 4 and column 3 in Table 3, this pro-
426 gressive gray reasoning frame approach works best.

427 Qualitatively, as shown in Figure 9, the mask format is
428 critical. The black mask fails the deletion task, while the

Color Transparency	Ablation on Reasoning Frame Format			
	Black (bg) (0%)	Red (50%)	Gray (0%)	Gray (0-75%)
<i>GPT-4o Score</i>				
Instruct Follow↑	7.512	7.805	8.069	8.973
Preservation↑	7.034	7.350	7.709	8.203
Quality↑	6.155	6.501	6.926	7.765
Success Ratio↑*	52.170%	60.330%	67.980%	76.36%
<i>Perceptual Quality</i>				
CLIP-T↑	26.550	26.810	27.143	28.000
CLIP-F↑	0.9810	0.9855	0.9890	0.9915
DINO↑	0.9750	0.9790	0.9826	0.9913

Table 3. Ablation on transparency mask settings.

red mask incorrectly deletes content on the right side. In
428 contrast, our progressive gray mask accurately performs the
429 intended deletion on the left. We conclude from these ex-
430 periments that the optimal reasoning format is a gray mask
431 with progressive transparency.

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

"Remove the woman with tattoos wearing beige bra on the left"

Figure 9. Ablation on reasoning frame format

5. Conclusion

450 In this paper, we introduced VideoCoF, a unified model for
451 universal video editing via temporal reasoning. We iden-
452 tified that existing temporal in-context learning approaches
453 often fail due to a lack of explicit spatial cues, leading to
454 weak instruction-to-region mapping and imprecise localiza-
455 tion. To address these issues, we proposed the innovative
456 Chain of Frames. CoF compels the video diffusion model to
457 follow a "see, reason, then edit" process by first predicting
458 the editing region before executing the versatile edit. Fur-
459 thermore, to solve the length generalization challenge, we
460 developed a novel RoPE alignment paradigm that accounts
461 for the reasoning latent. This design enables 4 times ex-
462 ploration in the inference. Experimental results show that
463 VideoCoF achieves SOTA performance using a mere 50k
464 video pairs, validating the efficiency and effectiveness of
465 our temporal reasoning design.

450 **References**

- [1] Yuxuan Bian, Zhaoyang Zhang, Xuan Ju, Mingdeng Cao, Liangbin Xie, Ying Shan, and Qiang Xu. Videopainter: Any-length video inpainting and editing with plug-and-play context control. *arXiv preprint arXiv:2503.05639*, 2025.
- [2] 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.
- [3] Lan Chen, Yuchao Gu, and Qi Mao. Univid: Unifying vision tasks with pre-trained video generation models. *arXiv preprint arXiv:2509.21760*, 2025.
- [4] Jiaxin Cheng, Tianjun Xiao, and Tong He. Consistent video-to-video transfer using synthetic dataset. *arXiv preprint arXiv:2311.00213*, 2023.
- [5] Yuren Cong, Mengmeng Xu, Christian Simon, Shoufa Chen, Jiawei Ren, Yanping Xie, Juan-Manuel Perez-Rua, Bodo Rosenhahn, Tao Xiang, and Sen He. Flatten: Optical flow-guided attention for consistent text-to-video editing. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.
- [6] Yuchao Gu, Yipin Zhou, Bichen Wu, Licheng Yu, Jia-Wei Liu, Rui Zhao, Jay Zhangjie Wu, David Junhao Zhang, Mike Zheng Shou, and Kevin Tang. Videoswap: Customized video subject swapping with interactive semantic point correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7621–7630, 2024.
- [7] Nicholas Guttenberg. Diffusion with offset noise. <https://www.crosslabs.org/blog/diffusion-with-offset-noise>, 2023.
- [8] 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.
- [9] 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.
- [10] Xuan Ju, Weicai Ye, Quande Liu, Qiulin Wang, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, and Qiang Xu. Fulldit: Multi-task video generative foundation model with full attention. *arXiv preprint arXiv:2503.19907*, 2025.
- [11] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.
- [12] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanyvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.
- [13] Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhui Chen. Viescore: Towards explainable metrics for conditional image synthesis evaluation, 2023.
- [14] Minghan Li, Chenxi Xie, Yichen Wu, Lei Zhang, and Mengyu Wang. Five: A fine-grained video editing benchmark for evaluating emerging diffusion and rectified flow models. *arXiv preprint arXiv:2503.13684*, 2025.
- [15] Xiaowen Li, Haolan Xue, Peiran Ren, and Liefeng Bo. Difueraser: A diffusion model for video inpainting. *arXiv preprint arXiv:2501.10018*, 2025.
- [16] 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.
- [17] Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5(5):5, 2017.
- [18] 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.
- [19] OpenAI. Hello gpt-4o. Blog post, 2024.
- [20] Pexels. Pexels: Free stock photos, royalty free stock images & videos. <https://www.pexels.com/>, 2025. Accessed: 2025-11-06.
- [21] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15932–15942, 2023.
- [22] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kun-chang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024.
- [23] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.
- [24] Wenhao Sun, Rong-Cheng Tu, Jingyi Liao, and Dacheng Tao. Diffusion model-based video editing: A survey. *arXiv preprint arXiv:2407.07111*, 2024.
- [25] DecartAI Team. Lucy edit: Open-weight text-guided video editing, 2025.
- [26] Yuanpeng Tu, Hao Luo, Xi Chen, Sihui Ji, Xiang Bai, and Hengshuang Zhao. Videoanydoor: High-fidelity video object insertion with precise motion control. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–11, 2025.
- [27] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.
- [28] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

- 563 [29] Cong Wei, Quande Liu, Zixuan Ye, Qiulin Wang, Xintao
564 Wang, Pengfei Wan, Kun Gai, and Wenhua Chen. Univideo:
565 Unified understanding, generation, and editing for videos.
566 *arXiv preprint arXiv:2510.08377*, 2025. 621
- 567 [30] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten
568 Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al.
569 Chain-of-thought prompting elicits reasoning in large lan-
570 guage models. *Advances in neural information processing
571 systems*, 35:24824–24837, 2022. 622
- 572 [31] Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane
573 Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank
574 Jaini, and Robert Geirhos. Video models are zero-shot learn-
575 ers and reasoners. *arXiv preprint arXiv:2509.20328*, 2025. 623
- 576 [32] Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jing-
577 wen Hou Hou, Annan Wang, Wenxiu Sun Sun, Qiong Yan,
578 and Weisi Lin. Exploring video quality assessment on user
579 generated contents from aesthetic and technical perspectives.
580 In *International Conference on Computer Vision (ICCV)*,
581 2023. 624
- 582 [33] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian
583 Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu
584 Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning
585 of image diffusion models for text-to-video generation. In
586 *Proceedings of the IEEE/CVF international conference on
587 computer vision*, pages 7623–7633, 2023. 625
- 588 [34] Jay Zhangjie Wu, Xiuyu Li, Difei Gao, Zhen Dong, Jin-
589 bin Bai, Aishani Singh, Xiaoyu Xiang, Youzeng Li, Zuwei
590 Huang, Yuanxi Sun, et al. Cvpr 2023 text guided video edit-
591 ing competition. *arXiv preprint arXiv:2310.16003*, 2023. 626
- 592 [35] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xin-
593 grun Xing, Ruiran Yan, Chaofan Li, Shuteng Wang, Tiejun
594 Huang, and Zheng Liu. Omnigen: Unified image genera-
595 tion. In *Proceedings of the Computer Vision and Pattern
596 Recognition Conference*, pages 13294–13304, 2025. 627
- 597 [36] Xiangpeng Yang, Linchao Zhu, Hehe Fan, and Yi Yang.
598 Videograins: Modulating space-time attention for multi-
599 grained video editing. In *The Thirteenth International Con-
600 ference on Learning Representations*, 2025. 628
- 601 [37] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu
602 Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiao-
603 han Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video
604 diffusion models with an expert transformer. *arXiv preprint
605 arXiv:2408.06072*, 2024. 629
- 606 [38] Xixuan Ye, Xuanhua He, Quande Liu, Qiulin Wang, Xintao
607 Wang, Pengfei Wan, Di Zhang, Kun Gai, Qifeng Chen, and
608 Wenhua Luo. Unic: Unified in-context video editing. *arXiv
609 preprint arXiv:2506.04216*, 2025. 630
- 610 [39] Xixuan Ye, Huijuan Huang, Xintao Wang, Pengfei Wan, Di
611 Zhang, and Wenhua Luo. Stylemaster: Stylize your video
612 with artistic generation and translation. In *Proceedings of
613 the Computer Vision and Pattern Recognition Conference*,
614 pages 2630–2640, 2025. 631
- 615 [40] Shoubin Yu, Difan Liu, Ziqiao Ma, Yicong Hong, Yang
616 Zhou, Hao Tan, Joyce Chai, and Mohit Bansal. Veg-
617 gie: Instructional editing and reasoning video concepts with
618 grounded generation. In *Proceedings of the IEEE/CVF In-
619 ternational Conference on Computer Vision*, pages 15147–
620 15158, 2025. 632