SAMA: Factorized Semantic Anchoring and Motion Alignment for Instruction-Guided Video Editing

Xinyao Zhang[1,2*], Wenkai Dong[2*], Yuxin Song[2*†], Bo Fang[2,3], Qi Zhang[2], Jing Wang[1,2], Fan Chen[2], Hui Zhang[2], Haocheng Feng[2], Yu Lu[4‡], Hang Zhou[2], Chun Yuan[1], and Jingdong Wang[2]

[1] Tsinghua University, China
[2] Baidu Inc, China
[3] City University of Hong Kong, Hong Kong SAR, China
[4] Zhejiang University, China
songyuxinbb@outlook.com
Web page: https://cynthiazxy123.github.io/SAMA

Abstract. Current instruction-guided video editing models struggle to simultaneously balance precise semantic modifications with faithful motion preservation. While existing approaches rely on injecting explicit external priors (*e.g.*, VLM features or structural conditions) to mitigate these issues, this reliance severely bottlenecks model robustness and generalization. To overcome this limitation, we present **SAMA** (factorized **S**emantic **A**nchoring and **M**otion **A**lignment), a framework that factorizes video editing into semantic anchoring and motion modeling. First, we introduce *Semantic Anchoring* which establish a reliable visual anchor by jointly predicting semantic tokens and video latents at sparse anchor frames, enabling purely instruction-aware structural planning. Second, *Motion Alignment* pre-trains the same backbone on motion-centric video restoration pretext tasks (cube inpainting, speed perturbation, and tube shuffle), enabling the model to internalize temporal dynamics directly from raw videos. SAMA is optimized with a two-stage pipeline: a factorized pre-training stage that learns inherent semantic-motion representations without paired video-instruction editing data, followed by supervised fine-tuning on paired editing data. Remarkably, the factorized pre-training alone already yields strong *zero-shot* video editing ability, validating the proposed factorization. SAMA achieves state-of-the-art performance among open-source models and is competitive with leading commercial systems (e.g. Kling-Omni). Code, models, and datasets are released in https://cynthiazxy123.github.io/SAMA.

Keywords: Video Editing · Diffusion Models · Zero-Shot Generalization · Representation Learning

*Equal contribution. † Project leader. ‡ Corresponding author.

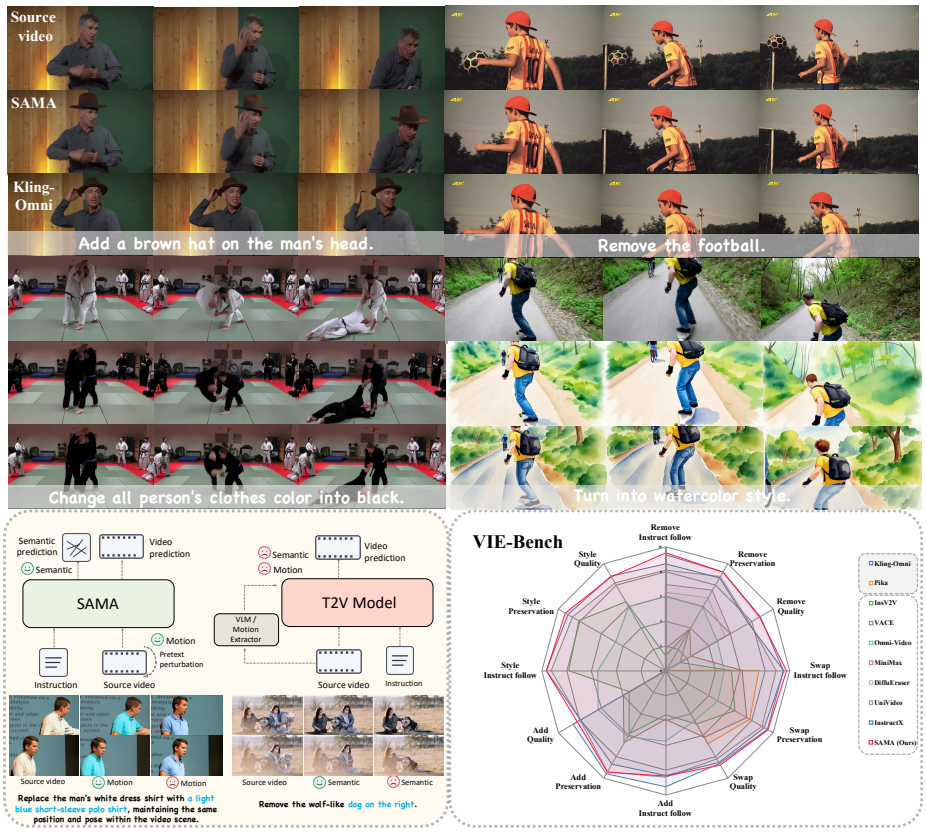


Fig. 1: Teaser and overview. Top: qualitative comparisons on VIE-Bench, comparing SAMA with representative open- and closed-source systems. Bottom left: illustration of SAMA's semantic–motion training objectives. Bottom right: fine-grained VIE-Bench performance comparison.

1 Introduction

Diffusion models have enabled interactive, instruction-guided image editing with impressive fidelity and controllability [6, 14, 33, 53, 64, 67, 71]. Extending this paradigm from single images to videos, however, remains substantially more challenging. A practical instruction-guided video editor must *(i)* apply fine-grained semantic changes that follow the instruction, while *(ii)* preserving temporally coherent motion of the edited subject, background, and camera. In current models, these two requirements often conflict: aggressive semantic changes induce localized artifacts, identity drift, and texture popping, whereas enforcing temporal consistency can dilute the intended edit and reduce instruction fidelity (Fig. 1 top). This tension has been widely observed in diffusion-based video editing and adaptation works [32, 38, 40, 58].

To mitigate these issues, a prevailing trend in existing approaches is to rely on injecting explicit *external* priors, such as VLM-extracted semantic condi-

tions [35, 48] or structural signals like skeletons and depth maps [9, 69]. We argue that this over-reliance reflects a significant bottleneck, which constrains the diffusion backbone from learning *inherent semantic-motion representations* for precise semantic editing and faithful motion alignment with the source video dynamics. Instead, we attribute the core difficulty of instruction-guided video editing to the lack of *factorization* between semantic structure planning and motion modeling [2, 7, 16, 25, 37]. Semantic edits are typically sparse and temporally stable: a small number of anchor frames is often sufficient to determine the desired visual modification. In contrast, motion coherence follows physical and temporal dynamics that can be learned from large-scale raw videos without explicit editing supervision.

Based on this observation, we propose **SAMA** (factorized **S**emantic **A**nchoring and **M**otion **A**lignment), a framework that encourages the model to learn semantic structure planning and motion modeling as two complementary capabilities. First, we introduce *Semantic Anchoring* which predicts semantic tokens together with video latents to support instruction-aware structural planning in the semantic space while retaining high-fidelity rendering in the latent space. Second, *Motion Alignment* strengthens temporal reasoning through motion-centric video restoration tasks, encouraging the backbone to internalize coherent temporal dynamics directly from raw videos.

To realize this factorized learning paradigm, we train SAMA with a two-stage strategy. In the first stage, a *factorized pre-training* process encourages the model to internalize semantic anchoring and motion dynamics as two complementary capabilities, without requiring paired instruction-guided video editing data. Remarkably, we find that this stage alone already induces strong *zero-shot* video editing behavior. This observation suggests that robust instruction-guided video editing can naturally emerge once a model learns to jointly reason about semantic intent and temporal dynamics. In the subsequent *supervised fine-tuning* stage, the model is trained on paired video editing datasets to resolve residual semantic–motion conflicts and improve visual fidelity. Consequently, SAMA achieves state-of-the-art performance among open-source models while delivering results comparable to leading commercial systems (e.g. Kling-Omni [48], Runway [42]).

- We propose a factorized perspective on instruction-guided video editing that separates semantic planning from motion modeling, reducing reliance on brittle external priors.
- We introduce *Semantic Anchoring* and *Motion Alignment* via motion-centric video restoration pre-training, enabling the diffusion backbone to internalize robust semantic and temporal representations.
- SAMA achieves state-of-the-art performance among open-source video editing models and is competitive with leading commercial systems. Code, models, and datasets are released in `https://cynthiazxy123.github.io/SAMA`.

2 Related Work

2.1 Instruction-Guided Video Editing

Instruction-guided video editing aims to edit an input video following a text instruction, with the key challenge of preserving temporal consistency. Early diffusion-based attempts [12,13,15,32,38,58,60] in instruction-guided video editing mainly follow zero-shot or one-/few-shot paradigms, where pretrained text-to-image diffusion models are repurposed for videos with additional temporal modeling to maintain consistency.

With the release of large-scale instruction-guided video editing datasets such as Señorita-2M [73], InsViE-1M [59], Ditto-1M [4], ReCo-Data [70], and OpenVE-3M [17], recent research has shifted toward data-driven video editing models trained end-to-end. Ditto [4] builds its large-scale synthetic data pipeline by combining a strong image editing model with an in-context video generation model, and then trains a model on Ditto-1M to improve instruction-guided and temporal consistency. OpenVE-3M [17] expands supervision across diverse editing categories, while ReCo-Data [70] focuses on region-aware instruction editing to improve local controllability.

Several recent works [10, 22, 23, 29, 30, 46, 61, 63, 70] further explore unified and in-context formulations for video editing. UNIC [63] unifies different video editing tasks by converting the noisy video latents, source video tokens, and multi-modal condition tokens into a single sequence, so a Diffusion Transformer can learn editing behaviors in-context without task-specific adapters or DDIM inversion. VACE [22] explores a unified and controllable editing formulation that supports diverse edit operations, improving the generality and robustness of instruction-guided video editing. ICVE [30] proposes a low-cost pretraining strategy that uses unpaired video clips to learn general editing ability in-context, and then refines the model with a small amount of paired editing data. EditVerse [23] proposes a unified framework for image/video generation and editing by representing text, images, and videos in a shared token space, enabling strong in-context editing and supporting data-driven training with large-scale benchmarks. DiffuEraser [29] studies instruction-guided video object removal by integrating diffusion-based editing with temporal-consistent inpainting, aiming to erase targets while preserving coherent backgrounds across frames. ReCo [70] introduces a joint source-target video diffusion framework and applies region constraints to improve instruction-guided editing. VideoCoF [61] introduces a Chain-of-Frames "see–reason–edit" formulation that predicts where/how to edit across frames before generation, improving instruction-to-region alignment and temporal consistency without requiring user-provided masks.

Beyond editing-centric models, unified video understanding and generation frameworks such as Omni-Video [45], InstructX [35], UniVideo [57], and VINO [8] provide strong representations for video content and motion dynamics.

2.2 Semantic Alignment on Image and Video Generation

Recent progress in image and video generation also benefits from semantic alignment between generative models and strong pretrained encoders. In image generation, REPA [65] aligns intermediate denoising features with clean features from a pretrained image encoder, which stabilizes training and improves generation quality. Following REPA, several works study how to apply representation alignment more effectively, including end-to-end VAE–diffusion training (REPA-E [28]), stage-wise scheduling to avoid late-stage degradation (HASTE [56]), teacher-free self-alignment via self-distillation (SRA [21]).

Similar ideas have recently been extended to video generation. SemanticGen [3] first predicts compact semantic features and then generates VAE latents conditioned on them, which is more efficient for long videos. VideoREPA [68] distills spatio-temporal relational knowledge from video foundation models into text-to-video diffusion models via token-relation alignment. Beyond generation, this relational alignment idea has been adopted for video editing: FFP-300K [20] uses inter-frame relational distillation inspired by VideoREPA to better preserve source motion.

Positioning. Inspired by recent advances in semantic alignment for image/video generation, we apply semantic-alignment regularization to instruction-guided video editing. Our approach improves instruction following and temporal consistency, and accelerates DiT convergence during training, without heavy test-time optimization.

2.3 Self-supervised Learning for Video Representation Learning

Self-supervised learning learns spatiotemporal representations from unlabeled videos via pretext tasks. Motivated by this line of work, we adopt lightweight pretext tasks as motion-centric restoration objectives in our *Motion Alignment* (Sec. 3.3) to better capture coherent temporal dynamics. Prior works mainly fall into three categories: speed-based learning (e.g., SpeedNet [5], PRP [62], Pace Prediction [52]), spatiotemporal puzzles (e.g., Space-Time Cubic Puzzles [24]), and reconstruction-based objectives (e.g., masked video modeling and VideoMAE [49]).

3 Method

Preliminary We adopt a video diffusion transformer framework trained via the flow matching [31] paradigm. The main training objective is to minimize the expected flow matching loss, defined as:

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{t,x_0,x_1} \| v_\theta(x_t, t) - (x_1 - x_0) \|_2^2, \tag{1}$$

where x_1 is the target video and x_0 is the Gaussian prior. The network v_θ learns to regress the vector field $x_1 - x_0$ from the intermediate state $x_t = tx_1 + (1-t)x_0$.

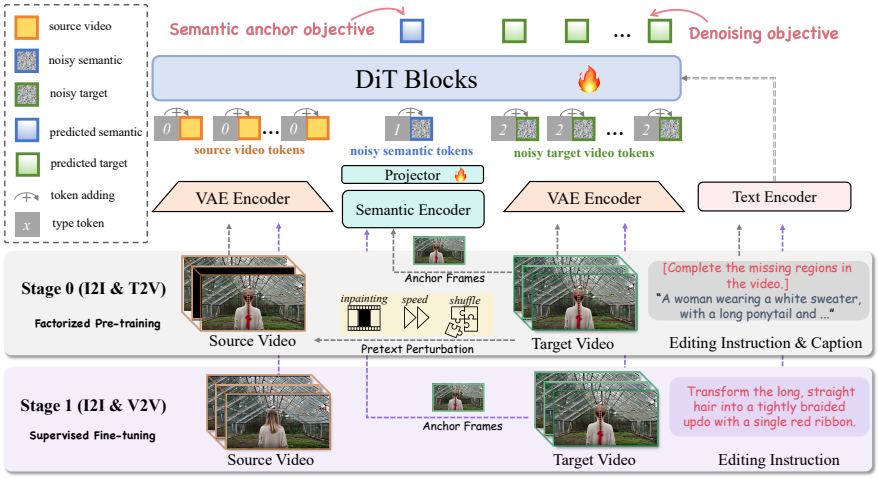


Fig. 2: Overall pipeline. SAMA first performs factorized pre-training (stage 0) on additional perturbed videos by completing a pretext task conditioned on the given captions. It then performs normal supervised fine-tuning (stage 1) on original source videos. Semantic Anchoring is incorporated in both stages to jointly facilitate semantic representation learning and instruction-guided video editing.

This formulation corresponds to the flow ordinary differential equation:

$$\frac{dx}{dt} = v_\theta(x, t). \tag{2}$$

3.1 SAMA

SAMA is built upon the video diffusion model Wan2.1-T2V-14B [50]. Given a source video V_s and an editing instruction y, the goal is to generate an edited target video V_t that follows y while preserving realistic spatiotemporal motion and non-edited content.

Latent tokenization. We encode videos into VAE latents following latent diffusion style formulations [41]. The source and target videos are represented as token sequences $\mathbf{z}_s$ and $\mathbf{z}_t$. We form an in-context V2V input by concatenating the source and (noisy) target token sequences: $\mathbf{z} = [\mathbf{z}_s\,;\,\mathbf{z}_t]$.

Type embeddings. To disambiguate token roles, we add a learned type embedding to each token: type id 0 for source-video latent tokens $\mathbf{z}_s$, type id 2 for target-video latent tokens $\mathbf{z}_t$, and type id 1 for semantic tokens introduced by Semantic Anchoring (Sec. 3.2). This convention is used consistently across all stages. We empirically observe that using type embeddings leads to faster convergence than the commonly used shifted RoPE scheme [43, 44], while minimally perturbing the backbone prior. We provide further discussion and supporting evidence in the Appendix.

SAMA internalizes two complementary capabilities within the diffusion backbone: *Semantic Anchoring (SA)* provides instruction-consistent anchors on sparse

anchor frames to stabilize structural editing (see Sec. 3.2); *Motion Alignment (MA)* aligns the edited video with the source motion dynamics through motion-centric pretext supervision, improving temporal stability and mitigating semantic–motion conflicts (see Sec. 3.3). Building on these two capabilities, we further introduce a two-stage training strategy: we first learn strong inherent semantic–motion representations in a factorized pre-training stage, and then strengthen editing performance with paired supervision in an SFT stage (Sec. 3.4).

3.2 Semantic Anchoring

Semantic Anchoring (SA) is introduced as an auxiliary objective throughout both the *Factorized Pre-training Stage* and the *SFT Stage*. For an image sample, the target image serves as the anchor. For a video sample, we uniformly sample N frames from the target video and treat them as sparse anchor frames. Each anchor frame is encoded by a SigLIP image encoder [66] to obtain patch-level semantic features. We then aggregate these features into a compact token set by pooling, producing M local semantic tokens that capture region-level semantics along with one global token that summarizes the overall content. All semantic tokens are finally projected by a lightweight two-layer MLP into the same embedding space as the VAE latent tokens.

Injecting semantic tokens into the denoising sequence. Let $\hat{\mathbf{s}}$ denote the projected semantic tokens extracted from the N anchor frames. We prepend $\hat{\mathbf{s}}$ to the target latent sequence and treat them as part of the denoising trajectory: we apply the same forward noising process to both semantic tokens and target latents, and feed the concatenated noisy sequence into the DiT. After denoising, we read out the positions corresponding to the semantic tokens and pass them through a semantic prediction head attached to the final DiT layer, yielding predicted semantic tokens $\mathbf{s}$.

Objective. We supervise semantic prediction with an ℓ_1 loss between the predicted tokens and the extracted anchor tokens:

$$\mathcal{L}_{\text{sem}} = \|\hat{\mathbf{s}} - \mathbf{s}\|_1. \tag{3}$$

The overall training objective combines the flow-matching loss and the Semantic Anchoring loss:

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + \lambda \cdot \mathcal{L}_{\text{sem}}. \tag{4}$$

3.3 Motion Alignment

Motion Alignment (MA) is applied on video samples in the *Factorized Pre-training Stage* (Sec. 3.4). Given a source video V_s and instruction y, we apply a motion-centric transformation $\mathcal{T}$ only to the source video to obtain $\tilde{V}_s = \mathcal{T}(V_s)$, while keeping the target side unchanged (i.e., always using the original target video without augmentation). This design forces the model to learn motion recovery and temporal reasoning from the source stream, improving robustness

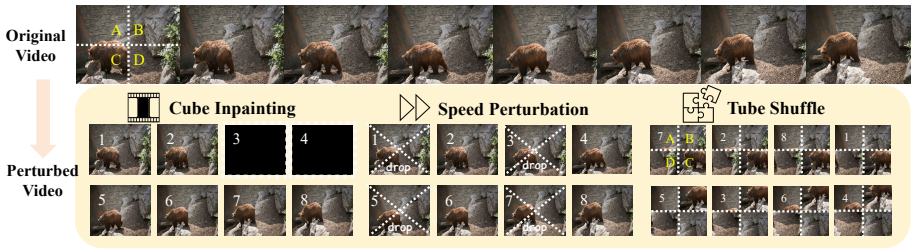


Fig. 3: Illustration of pretext perturbations.

under fast motion and complex camera dynamics. Fig. 3 provides an illustration of the pretext perturbations.

Motion-centric transformations. We adopt three restoration-style perturbations inspired by self-supervised learning for visual sequences [49]: (i) *Cube Inpainting*: mask a continuous temporal block in $\tilde{V}_s$ and recover missing content conditioned on the remaining frames; (ii) *Speed Perturbation*: temporally accelerate $\tilde{V}_s$ and learn to restore normal dynamics, improving robustness to motion-rate changes; (iii) *Tube Shuffle*: partition $\tilde{V}_s$ into a $2\times2\times2$ spatio-temporal tube grid and randomly permute tubes, forcing the model to reason about spatio-temporal structure and restore consistent motion.

Prompting for pretext tasks. To make the objective explicit and unify the formulation across tasks, we prepend a short task token to the editing instruction:

Task token examples for Motion Alignment

- *[Complete the missing regions in the video.]* (Cube Inpainting)
- *[Restore the video to normal playback speed.]* (Speed Perturbation)
- *[Restore the correct spatio-temporal order of the video segments.]* (Tube Shuffle)

Overall, MA encourages the backbone to internalize robust motion dynamics from the source stream while remaining fully compatible with the instruction-conditioned editing formulation.

3.4 Training Strategies

SAMA is optimized with a two-stage training pipeline that mirrors our factorized view of instruction-guided video editing.

Stage 0: Factorized Pre-training. We start from a strong text-to-video prior and pre-train it on a mixture of instruction-based image editing pairs and large-scale text-to-video data [19, 54]. The image editing portion provides broad semantic coverage and improves general instruction grounding, while the text-to-video portion supplies diverse real-world motion patterns. During this stage, we apply SA to both image and video samples, and apply MA only to the video stream: (i) *SA* supervises semantic token prediction on N sparsely sampled anchor frames, encouraging instruction-consistent semantic anchoring while

sharing the same diffusion backbone (Sec. 3.2); (ii) *MA* trains the model to restore temporally perturbed source videos with motion-centric pretext supervision, improving temporal stability and robustness under fast motion (Sec. 3.3). The overall objective at Stage 0 follows Eq. (4),

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + \lambda \cdot \mathcal{L}_{\text{sem}}, \tag{5}$$

where $\mathcal{L}_{\text{FM}}$ is the flow matching loss in Eq. (1) and $\mathcal{L}_{\text{sem}}$ is the SA semantic prediction loss.

Stage 1: Supervised Fine-tuning (SFT). We then perform supervised fine-tuning on paired video editing datasets [4, 17, 70], while mixing a small portion of image editing data to preserve general instruction-following behavior [27, 39]. In this stage, the model is trained on standard instruction-guided video editing triplets (source video, instruction, target video), and we keep SA enabled to maintain stable semantic anchoring on sparse anchor frames. Compared with Stage 0, Stage 1 focuses on aligning generation with paired editing supervision, improving edit fidelity and mitigating remaining semantic–motion conflicts observed in challenging motions and fine-grained edits.

This two-stage design separates the learning of semantic anchoring and motion alignment from scarce paired video-edit data. As a result, Stage 0 already provides strong zero-shot video editing capability, and Stage 1 further improves edit fidelity and benchmark performance with paired supervision.

4 Experiments

4.1 Experimental Settings

Training data. As summarized in Tab. 1, we use NHR-Edit [27], GPT-image-edit [55], X2Edit [34], and Pico-Banana-400K [39] for image editing training. We additionally incorporate text-to-video Koala-36M [54] and MotionBench [19] for pretext motion alignment. Ditto-1M [4], OpenVE-3M [17], and ReCo-Data [70] are employed for video editing. All datasets are additionally subjected to a VLM-based coarse filtering stage to remove low-quality or instruction-inconsistent samples. The detailed filtering criteria are provided in Appendix. Specifically, we only use the Style subset of Ditto-1M [4], and the Local Change, Background, Style, and Subtitles categories from OpenVE-3M [17].

Implementation details. During training, we conduct two-stage training on mixed image and video data. The learning rate is 2×10^{-5} for both stages. The global batch size is 448 for images and 112 for videos, and we train at a resolution of 480p. We support multiple aspect ratios, including $1/2, 2/3, 3/4$, and $1/1$, as well as their reciprocals. We maintain an exponential moving average (EMA [18]) of model parameters with decay 0.9998 and update it every iteration. The loss weight λ (Eq. 4) is set to 0.1. Unless otherwise specified, we uniformly sample N sparse anchor frames for Semantic Anchoring (Sec. 3.2); for efficiency, we set $N = 1$ in all experiments. We use M local semantic tokens per anchor frame (plus one global token), and fix $M = 64$ throughout.

Table 1: Statistics of training data for each stage. ★ denotes we use specific text-to-video data for Motion Alignment by solving pretext transformations.

Training stage	Dataset	# Pairs	Type
Stage 0 *Factorized Pre-training*	NHR-Edit [27]	720,087	image editing
	GPT-Image-Edit [55]	1,015,170	image editing
	X2Edit [34]	768,470	image editing
	Koala-36M [54]	1,532,716	text-to-video★
	MotionBench [19]	53,879	text-to-video★
Stage 1 *Supervised Fine-tuning*	NHR-Edit [27]	720,087	image editing
	Pico-Banana-400K [39]	257,730	image editing
	Ditto-1M [4]	3,936	video editing
	OpenVE-3M [17]	818,232	video editing
	ReCo-Data [70]	206,596	video editing

In the text-to-video data, we use no pretext task as well as three pretext tasks—Cube Inpainting, Speed Perturbation, and Tube Shuffle—with a sampling ratio of 1:2:3:4 (no-pretext : cube inpainting : speed perturbation : tube shuffle). Task-specific settings are deferred to Appendix.

Evaluation details. To evaluate SAMA, we compare it against current state-of-the-art methods, including closed-source and open-source systems. For closed-source models, we include Kling1.6 [26], Kling-Omni [48], Runway [42], MiniMax [72], and Pika [1]. For open-source methods, we compare with InsV2V [10], DiffuEraser [29], VACE [22], InsViE [59], Omni-Video [45], LucyEdit [46], Uni-Video [57], InstructX [35], ICVE [30], Ditto [4], OpenVE-Edit [17], VINO [8], and ReCo [70]. We conduct experiments on three benchmarks: VIE-Bench [35], OpenVE-Bench [17], and ReCo-Bench [70]. We use different VLM judges for scoring across benchmarks: GPT-4o [36] for VIE-Bench [35], Gemini-2.5-Pro [11] for OpenVE-Bench [17], and Gemini-2.5-Flash-Thinking [47] for ReCo-Bench [70].

4.2 Comparisons with State-of-the-Art Methods

Tab. 2 show that our method consistently outperforms existing open-source video-editing models across most metrics, while remaining competitive with state-of-the-art closed-source systems. In particular, SAMA achieves the best overall performance on Swap/Change and Remove, among all compared methods. Similar gains are observed on Tab. 3 on OpenVE-Bench [17] and Tab.4 on ReCo-Bench [70], where SAMA attains the top overall score and delivers strong results across multiple task categories, despite a few metrics where it is not the best-performing method.

Qualitative Comparisons. In the qualitative comparisons on VIE-Bench and ReCo-Bench (see Fig. 4), SAMA demonstrates stronger instruction adherence and temporal consistency across diverse editing types. SAMA follows fine-grained instructions more reliably, correctly handling relative position cues (e.g., "on the left") and attribute constraints (e.g., alternating *light and dark hair*). It also completes replacements (e.g., pigeon→squirrel, seal→crab) with consistent appearance over time. Motion-wise, SAMA better preserves temporal alignment (e.g., keeping the stroller aligned after removal) and maintains identity/details

Table 2: Comparison results on VIE-Bench. The best results are shown in **bold**.

Method	Instruct follow	Preser-vation	Quality	Avg.	Instruct follow	Preser-vation	Quality	Avg.
	Add				**Swap / Change**			
Kling1.6	6.000	8.230	5.576	6.602	9.000	9.060	8.333	8.800
Kling-Omni	9.333	9.589	8.622	9.181	9.495	9.448	8.638	9.194
Runway	8.607	8.913	7.823	8.447	9.580	8.628	9.275	9.161
Pika	-	-	-	-	7.542	7.847	6.837	7.408
InsV2V	3.552	5.891	3.402	4.281	5.304	6.428	4.971	5.567
VACE	3.938	6.696	3.929	4.854	6.171	7.552	6.199	6.640
Omni-Video	5.699	6.135	6.294	6.242	4.733	4.856	4.656	4.748
UniVideo	**8.567**	9.422	7.978	8.656	8.886	8.962	8.200	8.683
InstructX	8.446	8.683	7.919	8.349	9.514	9.171	8.533	9.072
SAMA	8.467	**9.422**	**8.244**	**8.711**	**9.733**	**9.514**	**8.771**	**9.340**
	Remove				**Style / Tone Change**			
Kling1.6	8.440	8.800	7.520	8.253	-	-	-	-
Kling-Omni	9.378	9.233	8.789	9.133	9.867	9.200	8.956	9.341
Runway	8.664	9.145	7.703	8.504	9.583	9.200	8.616	9.133
MiniMax	6.963	7.518	6.037	6.839	-	-	-	-
DiffuEraser	6.346	6.807	5.576	6.243	-	-	-	-
InsV2V	1.209	3.769	1.322	2.098	7.835	8.086	6.437	7.452
VACE	1.812	3.877	2.359	2.682	-	-	-	-
Omni-Video	6.004	5.970	4.807	5.593	5.486	4.655	5.959	5.366
UniVideo	8.133	8.778	7.789	8.233	9.244	8.689	8.200	8.711
InstructX	8.627	8.668	7.672	8.322	**9.650**	9.099	**8.839**	9.196
SAMA	**9.533**	**9.189**	**8.711**	**9.144**	9.644	**9.356**	8.778	**9.259**

Table 3: Comparison results on OpenVE-Bench with Gemini 2.5 Pro. The best results are highlighted in **bold**. Gray shading indicates closed-source models.

Methods	# Params.	Global Style	Background Change	Local Change	Local Remove	Local Add	Subtitle Edit	Creative Edit
Runway	-	3.72	2.62	4.18	4.16	2.78	3.62	3.64
VACE	14B	1.49	1.55	2.07	1.46	1.26	1.48	1.47
Omni-Video	1.3B	1.11	1.18	1.14	1.14	1.36	1.00	2.26
InsViE	2B	2.20	1.06	1.48	1.36	1.17	2.18	2.02
Lucy-Edit	5B	2.27	1.57	3.20	1.75	2.30	1.61	2.86
ICVE	13B	2.22	1.62	2.57	2.51	1.97	2.09	2.41
Ditto	14B	4.01	1.68	2.03	1.53	1.41	2.81	1.23
OpenVE-Edit	5B	3.16	2.36	2.98	1.85	2.15	2.91	2.31
UniVideo	20B	3.64	2.22	3.91	2.70	**2.98**	2.69	2.90
SAMA	14B	**4.05**	**2.59**	**3.93**	**3.32**	2.54	**3.63**	**3.11**

during stylization, while other methods may drift or blur. Overall, SAMA better grounds instruction semantics while maintaining coherent motion, leading to higher-quality and more stable edits.

4.3 Zero-shot Video Editing

We evaluate SAMA in a zero-shot setting, where the model is trained w/o any video editing data and is directly prompted with editing instructions during inference. As show in Fig. 5, SAMA demonstrates strong zero-shot editing capabilities across Replace/Add/Remove/Style/Hybrid tasks, producing consistent edits over multiple frames while largely preserving non-edited content. Despite these encouraging results, we also observe several typical failure modes in the zero-shot setting: (i) attribute edits can be temporally inconsistent, e.g., the edited colors

Table 4: Comparison results on ReCo-Bench with Gemini-2.5-Flash-Thinking. The best results are shown in **bold**. Abbreviations: SA, semantic accuracy; SP, scope precision; CP, content preservation; AN, appearance naturalness; SN, scale naturalness; MN, motion naturalness; VF, visual fidelity; TS, temporal stability; ES, edit stability. $S_{EA}/S_{VN}/S_{VQ}$ are category scores and S is the overall score.

Task	Method	Edit Accuracy			Video Naturalness			Video Quality			Average Score			
		SA	SP	CP	AN	SN	MN	VF	TS	ES	S_{EA}	S_{VN}	S_{VQ}	S
Add	InsViE	2.60	2.79	2.78	2.33	3.98	3.74	3.71	3.91	3.58	2.60	3.10	3.46	3.05
	Lucy-Edit	6.27	6.32	7.75	4.63	7.08	6.08	6.31	6.82	7.57	6.47	5.70	6.77	6.31
	Ditto	7.46	7.24	6.30	6.30	8.85	8.30	8.13	8.55	9.03	6.70	7.57	8.41	7.56
	UniVideo	9.39	**9.27**	9.69	7.27	9.23	8.80	8.44	8.89	9.75	9.40	8.31	8.99	8.90
	ReCo	8.65	8.40	9.22	6.39	8.78	8.28	8.02	8.61	9.61	8.54	7.55	8.61	8.23
	SAMA	**9.51**	9.26	**9.83**	**7.44**	**9.50**	**8.87**	**8.78**	**9.03**	**9.76**	**9.43**	**8.33**	**9.00**	**8.92**
Replace	InsViE	1.89	2.38	2.48	2.58	5.25	5.05	3.76	4.00	3.52	2.10	3.91	3.49	3.17
	Lucy-Edit	6.57	7.49	7.73	5.13	7.46	6.65	6.32	6.64	8.08	7.08	6.21	6.88	6.72
	Ditto	4.95	4.83	4.79	5.81	8.63	8.10	7.55	7.95	8.71	4.56	7.21	7.96	6.58
	UniVideo	9.03	9.68	9.73	7.73	9.30	8.92	**8.57**	**8.91**	**9.80**	9.40	8.39	8.90	8.90
	ReCo	9.38	9.43	9.59	7.07	8.87	8.47	8.19	8.65	9.67	9.43	8.01	8.77	8.74
	SAMA	**9.58**	**9.82**	**9.82**	**7.77**	**9.35**	**8.98**	8.55	8.80	9.72	**9.71**	**8.60**	**8.98**	**9.10**
Remove	InsViE	2.53	2.49	2.44	2.63	4.87	4.72	3.41	3.67	3.40	2.44	3.76	3.29	3.16
	VACE	4.58	4.58	4.56	4.96	6.09	5.89	5.48	5.50	5.57	4.57	5.43	5.56	5.19
	UniVideo	7.37	7.43	7.28	6.06	7.61	7.13	6.28	6.43	7.72	7.33	6.59	6.51	6.81
	ReCo	7.43	7.43	7.17	6.20	7.43	7.30	6.48	6.63	7.68	7.28	6.90	6.82	7.00
	SAMA	**8.76**	**8.71**	**8.43**	**7.16**	**8.73**	**8.42**	**7.31**	**7.52**	**8.73**	**8.61**	**7.94**	**7.73**	**8.09**
Style	InsViE	7.59	8.86	8.49	6.77	9.14	9.28	7.13	6.40	8.99	8.17	8.21	7.35	7.91
	Lucy-Edit	3.73	5.59	5.39	4.20	5.88	5.88	4.44	4.17	5.87	4.65	4.67	5.17	4.83
	Ditto	9.10	9.36	9.26	8.25	9.51	9.58	8.33	8.33	9.77	9.20	9.07	8.77	9.01
	UniVideo	8.10	9.82	9.50	8.56	9.65	**9.84**	**8.91**	8.57	**9.88**	8.95	9.23	9.00	9.06
	ReCo	**9.11**	9.82	9.54	8.43	9.55	9.70	8.61	8.35	9.87	**9.42**	9.19	8.90	9.17
	SAMA	8.46	**9.95**	**9.64**	**8.79**	**9.77**	9.77	8.88	**8.59**	9.83	9.24	**9.42**	**9.07**	**9.25**

may vary across frames; (ii) newly added objects may appear slightly blurry; (iii) removal edits may leave residual ghosting.

4.4 Ablation Study

Semantic Anchoring. We first observe that incorporating the semantic prediction objective accelerates the decrease of the diffusion loss, leading to faster DiT convergence. In addition, SA stabilizes training, as evidenced by a noticeably reduced loss variance (see Fig. 6b). We set the baseline by concatenating the source latent with the video latent, without SA or MA. We use the smaller Wan2.2-T2V-5B [51] for efficiency with type embeddings and train it on a subset of the Ditto-1M [4], obtaining the baseline results. Building on this baseline, adding SA leads to consistent mean score improvements across all tasks on VIE-Bench.

We further provide qualitative comparisons under the same number of training steps in Fig. 6a. As shown, the model with SA produces higher-quality edits at earlier training stages, whereas the baseline without it often yields incomplete or less accurate modifications. These results corroborate that SA facilitates faster convergence in practice.

Fig. 4: Qualitative comparisons with prior methods on VIE-Bench and ReCo-Bench.

Motion Alignment. We conduct a qualitative analysis on the effect of MA. We find that enabling MA improves temporal consistency under fast motion and alleviates motion blur. Representative qualitative results are shown in Fig. 7.

In the tennis case with large camera motion, with MA noticeably improves background sharpness (e.g., clearer on-screen text), while the baseline appears blurred. Similar improvements are observed in the car and the third example, where the baseline often loses background motion. Quantitative ablation results on MA are summarized in Tab. 5. On VIE-Bench, adding MA alone improves the overall score by 0.399 over the baseline. When combining SA and MA, the overall score

Table 5: Ablations of SAMA modules.

method	instruct follow	preservation	quality	Overall
baseline	6.575	6.261	6.100	6.312
w/ SA	7.002	6.744	6.342	6.696
w/ MA	6.969	6.620	6.544	6.711
SAMA	**7.402**	**6.998**	**6.884**	**7.095**

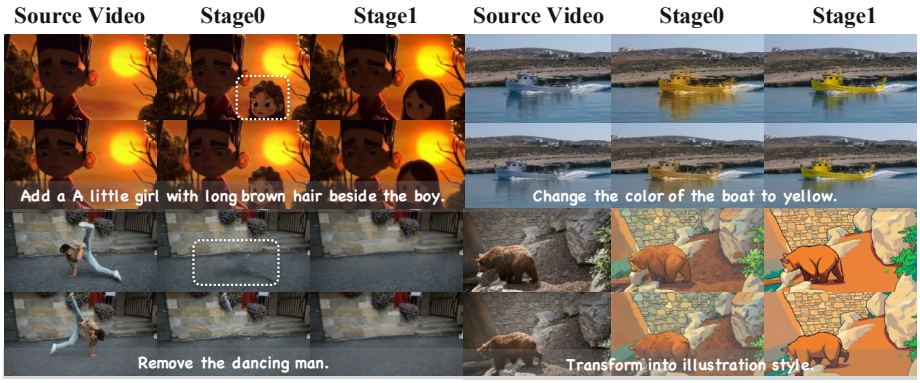
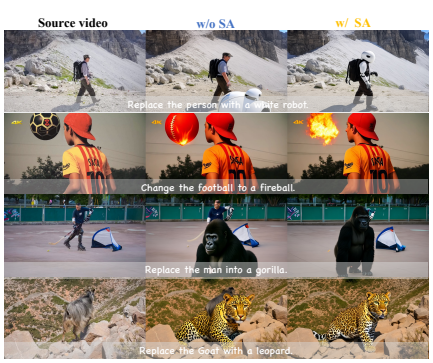
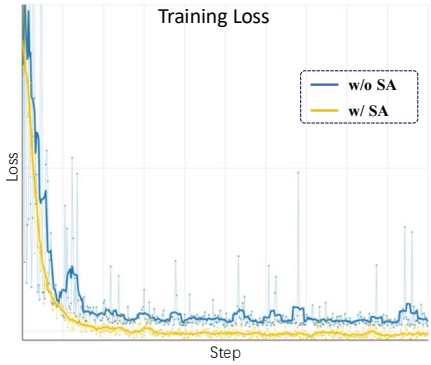


Fig. 5: Zero-shot qualitative results on VIE-Bench at two training stages.



(a) Visual results of SAMA w/ SA (right column) and w/o SA (middle column).



(b) Training loss curves.

Fig. 6: Ablations for Semantic Anchoring (SA).

further increases by 0.783, indicating that the two components are complementary.

5 Conclusion

We presented **SAMA**, a factorized framework for instruction-guided video editing that separates semantic anchoring and motion alignment within a DiT. Semantic anchoring introduces an explict prior via semantic-token prediction at anchor frames, while motion alignment improves temporal coherence through motion-centric restoration pre-training on text-to-video data. Extensive experiments on VIE-Bench, OpenVE-Bench, and ReCo-Bench demonstrate state-of-the-art performance among open-source methods and competitive results against commercial systems. Moreover, SAMA exhibits strong zero-shot editing behavior, suggesting that robust instruction following can emerge from learning disentangled semantic and motion representations. Future work will focus on long-

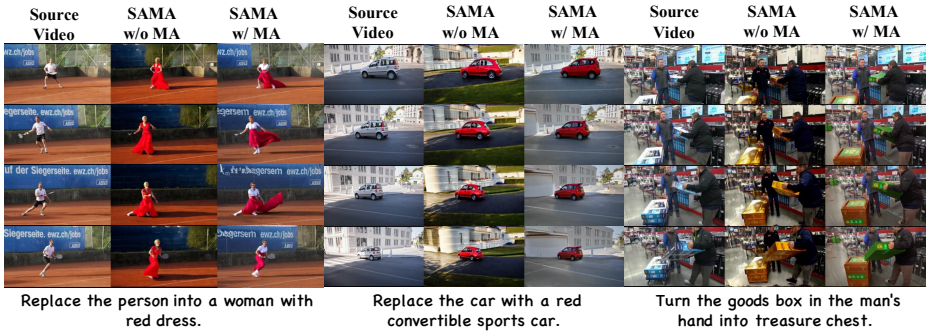


Fig. 7: Qualitative comparison of SAMA w/ and w/o *MA*.

video editing, fast-motion scenarios, and stronger semantic tokenization to further reduce residual artifacts and temporal inconsistencies.

Acknowledgement

This work was supported by the SSTIC Grant (KJZD20230923115106012, KJZD20230923114916032, GJHZ20240218113604008).

References

1. Pika: Idea-to-video platform (web product). `https://pika.art/`
2. Agarwal, N., et al.: Cosmos: World foundation model platform for physical ai. arXiv preprint arXiv:2501.03575 (2025)
3. Bai, J., Wu, X., Wang, X., Fu, X., Zhang, Y., Wang, Q., Shi, X., Xia, M., Liu, Z., Hu, H., et al.: Semanticgen: Video generation in semantic space. arXiv preprint arXiv:2512.20619 (2025)
4. Bai, Q., Wang, Q., Ouyang, H., Yu, Y., Wang, H., Wang, W., Cheng, K.L., Ma, S., Zeng, Y., Liu, Z., et al.: Scaling instruction-based video editing with a high-quality synthetic dataset. arXiv preprint arXiv:2510.15742 (2025)
5. Benaim, S., Ephrat, A., Lang, O., Mosseri, I., Freeman, W.T., Rubinstein, M., Irani, M., Dekel, T.: Speednet: Learning the speediness in videos. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9922–9931 (2020)
6. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18392–18402 (2023)
7. Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., et al.: Video generation models as world simulators. OpenAI Blog **1**(8), 1 (2024)
8. Chen, J., He, T., Fu, Z., Wan, P., Gai, K., Ye, W.: Vino: A unified visual generator with interleaved omnimodal context. arXiv preprint arXiv:2601.02358 (2026)
9. Chen, W., Ji, Y., Wu, J., Wu, H., Xie, P., Li, J., Xia, X., Xiao, X., Lin, L.: Control-a-video: Controllable text-to-video diffusion models with motion prior and reward feedback learning (2024), `https://arxiv.org/abs/2305.13840`

10. Cheng, J., Xiao, T., He, T.: Consistent video-to-video transfer using synthetic dataset. arXiv preprint arXiv:2311.00213 (2023)
11. Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang, D., Rosen, E., et al.: Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261 (2025)
12. Cong, Y., Xu, M., Simon, C., Chen, S., Ren, J., Xie, Y., Perez-Rua, J.M., Rosenhahn, B., Xiang, T., He, S.: Flatten: Optical flow-guided attention for consistent text-to-video editing. In: Proceedings of the International Conference on Learning Representations (ICLR) (2024), https://arxiv.org/abs/2310.05922
13. Couairon, P., Rambour, C., Haugeard, J.E., Thome, N.: Videdit: Zero-shot and spatially aware text-driven video editing. Transactions on Machine Learning Research (2023)
14. Ge, Y., Zhao, S., Li, C., Ge, Y., Shan, Y.: Seed-data-edit technical report: A hybrid dataset for instructional image editing. arXiv preprint arXiv:2405.04007 (2024)
15. Geyer, M., Bar-Tal, O., Bagon, S., Dekel, T.: Tokenflow: Consistent diffusion features for consistent video editing. arXiv preprint arXiv:2307.10373 (2023)
16. Ha, D., Schmidhuber, J.: World models. arXiv preprint arXiv:1803.10122 (2018)
17. He, H., Wang, J., Zhang, J., Xue, Z., Bu, X., Yang, Q., Wen, S., Xie, L.: Openve-3m: A large-scale high-quality dataset for instruction-guided video editing. arXiv preprint arXiv:2512.07826 (2025)
18. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
19. Hong, W., Cheng, Y., Yang, Z., Wang, W., Wang, L., Gu, X., Huang, S., Dong, Y., Tang, J.: Motionbench: Benchmarking and improving fine-grained video motion understanding for vision language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 8450–8460 (2025)
20. Huang, X., Xu, C., Luo, D., Hu, X., Tang, P., Peng, X., Zhang, J., Wang, C., Fu, Y.: Ffp-300k: Scaling first-frame propagation for generalizable video editing. arXiv preprint arXiv:2601.01720 (2026)
21. Jiang, D., Wang, M., Li, L., Zhang, L., Wang, H., Wei, W., Dai, G., Zhang, Y., Wang, J.: No other representation component is needed: Diffusion transformers can provide representation guidance by themselves. arXiv preprint arXiv:2505.02831 (2025)
22. Jiang, Z., Han, Z., Mao, C., Zhang, J., Pan, Y., Liu, Y.: Vace: All-in-one video creation and editing. arXiv preprint arXiv:2503.07598 (2025)
23. Ju, X., Wang, T., Zhou, Y., Zhang, H., Liu, Q., Zhao, N., Zhang, Z., Li, Y., Cai, Y., Liu, S., et al.: Editverse: Unifying image and video editing and generation with in-context learning. arXiv preprint arXiv:2509.20360 (2025)
24. Kim, D., Cho, D., Kweon, I.S.: Self-supervised video representation learning with space-time cubic puzzles. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 8545–8552 (2019)
25. Kong, W., Tian, Q., Zhang, Z., et al.: Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint arXiv:2412.03603 (2024)
26. Kuaishou: Kling1.6. https://app.klingai.com/ (2025)
27. Kuprashevich, M., Alekseenko, G., Tolstykh, I., Fedorov, G., Suleimanov, B., Dokholyan, V., Gordeev, A.: Nohumansrequired: Autonomous high-quality image editing triplet mining. arXiv preprint arXiv:2507.14119 (2025)
28. Leng, X., Singh, J., Hou, Y., Xing, Z., Xie, S., Zheng, L.: Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18262–18272 (2025)

29. Li, X., Xue, H., Ren, P., Bo, L.: Diffueraser: A diffusion model for video inpainting. arXiv preprint arXiv:2501.10018 (2025)
30. Liao, X., Zeng, X., Song, Z., Fu, Z., Yu, G., Lin, G.: In-context learning with unpaired clips for instruction-based video editing. arXiv preprint arXiv:2510.14648 (2025)
31. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=PqvMRDCJT9t
32. Liu, S., Zhang, Y., Li, W., Lin, Z., Jia, J.: Video-p2p: Video editing with cross-attention control. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8599–8608 (2024)
33. Liu, S., Han, Y., Xing, P., Yin, F., Wang, R., Cheng, W., Liao, J., Wang, Y., Fu, H., Han, C., et al.: Step1x-edit: A practical framework for general image editing. arXiv preprint arXiv:2504.17761 (2025)
34. Ma, J., Zhu, X., Pan, Z., Peng, Q., Guo, X., Chen, C., Lu, H.: X2edit: Revisiting arbitrary-instruction image editing through self-constructed data and task-aware representation learning. arXiv preprint arXiv:2508.07607 (2025)
35. Mou, C., Sun, Q., Wu, Y., Zhang, P., Li, X., Ye, F., Zhao, S., He, Q.: Instructx: Towards unified visual editing with mllm guidance. arXiv preprint arXiv:2510.08485 (2025)
36. OpenAI: Hello gpt-4o. Blog post (May 2024), https://openai.com/index/hello-gpt-4o/
37. Polyak, A., Zohar, A., Brown, A., Tjandra, A., Sinha, A., Lee, A., Vyas, A., Shi, B., Ma, C.Y., Chuang, C.Y., et al.: Movie gen: A cast of media foundation models. arXiv preprint arXiv:2410.13720 (2024)
38. Qi, C., Cun, X., Zhang, Y., Lei, C., Wang, X., Shan, Y., Chen, Q.: Fatezero: Fusing attentions for zero-shot text-based video editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15932–15942 (2023)
39. Qian, Y., Bocek-Rivele, E., Song, L., Tong, J., Yang, Y., Lu, J., Hu, W., Gan, Z.: Pico-banana-400k: A large-scale dataset for text-guided image editing. arXiv preprint arXiv:2510.19808 (2025)
40. Qu, L., Zhang, H., Liu, Y., Wang, X., Jiang, Y., Gao, Y., Ye, H., Du, D.K., Yuan, Z., Wu, X.: Tokenflow: Unified image tokenizer for multimodal understanding and generation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 2545–2555 (2025)
41. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
42. Runway: Runway gen-4. https://runwayml.com/ (2025)
43. Song, Y., Dong, W., Wang, S., Zhang, Q., Xue, S., Yuan, T., Yang, H., Feng, H., Zhou, H., Xiao, X., et al.: Query-kontext: An unified multimodal model for image generation and editing. arXiv preprint arXiv:2509.26641 (2025)
44. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. Neurocomputing **568**, 127063 (2024)
45. Tan, Z., Yang, H., Qin, L., Gong, J., Yang, M., Li, H.: Omni-video: Democratizing unified video understanding and generation. arXiv preprint arXiv:2507.06119 (2025)
46. Team, D.: Lucy edit: Open-weight text-guided video editing (2025)
47. Team, G., Anil, R., Borgeaud, S., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., Millican, K., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023)

48. Team, K., Chen, J., Ci, Y., Du, X., Feng, Z., Gai, K., Guo, S., Han, F., He, J., He, K., et al.: Kling-omni technical report. arXiv preprint arXiv:2512.16776 (2025)
49. Tong, Z., Song, Y., Wang, J., Wang, L.: Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. Advances in neural information processing systems **35**, 10078–10093 (2022)
50. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
51. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
52. Wang, J., Jiao, J., Liu, Y.H.: Self-supervised video representation learning by pace prediction. In: European conference on computer vision. pp. 504–521. Springer (2020)
53. Wang, P., Shi, Y., Lian, X., Zhai, Z., Xia, X., Xiao, X., Huang, W., Yang, J.: Seededit 3.0: Fast and high-quality generative image editing. arXiv preprint arXiv:2506.05083 (2025)
54. Wang, Q., Shi, Y., Ou, J., Chen, R., Lin, K., Wang, J., Jiang, B., Yang, H., Zheng, M., Tao, X., et al.: Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 8428–8437 (2025)
55. Wang, Y., Yang, S., Zhao, B., Zhang, L., Liu, Q., Zhou, Y., Xie, C.: Gpt-image-edit-1.5 m: A million-scale, gpt-generated image dataset. arXiv preprint arXiv:2507.21033 (2025)
56. Wang, Z., Zhao, W., Zhou, Y., Li, Z., Liang, Z., Shi, M., Zhao, X., Zhou, P., Zhang, K., Wang, Z., et al.: Repa works until it doesn't: Early-stopped, holistic alignment supercharges diffusion training. arXiv preprint arXiv:2505.16792 (2025)
57. Wei, C., Liu, Q., Ye, Z., Wang, Q., Wang, X., Wan, P., Gai, K., Chen, W.: Univideo: Unified understanding, generation, and editing for videos. arXiv preprint arXiv:2510.08377 (2025)
58. Wu, J.Z., Ge, Y., Wang, X., Lei, S.W., Gu, Y., Shi, Y., Hsu, W., Shan, Y., Qie, X., Shou, M.Z.: 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. pp. 7623–7633 (2023)
59. Wu, Y., Chen, L., Li, R., Wang, S., Xie, C., Zhang, L.: Insvie-1m: Effective instruction-based video editing with elaborate dataset construction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16692–16701 (2025)
60. Yang, S., Zhou, Y., Liu, Z., Loy, C.C.: Rerender a video: Zero-shot text-guided video-to-video translation. In: SIGGRAPH Asia 2023 Conference Papers. pp. 1–11 (2023)
61. Yang, X., Xie, J., Yang, Y., Huang, Y., Xu, M., Wu, Q.: Unified video editing with temporal reasoner. arXiv preprint arXiv:2512.07469 (2025)
62. Yao, Y., Liu, C., Luo, D., Zhou, Y., Ye, Q.: Video playback rate perception for self-supervised spatio-temporal representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6548–6557 (2020)
63. Ye, Z., He, X., Liu, Q., Wang, Q., Wang, X., Wan, P., Zhang, D., Gai, K., Chen, Q., Luo, W.: Unic: Unified in-context video editing. arXiv preprint arXiv:2506.04216 (2025)

64. Yu, Q., Chow, W., Yue, Z., Pan, K., Wu, Y., Wan, X., Li, J., Tang, S., Zhang, H., Zhuang, Y.: Anyedit: Mastering unified high-quality image editing for any idea. arXiv preprint arXiv:2411.15738 (2024)
65. Yu, S., Kwak, S., Jang, H., Jeong, J., Huang, J., Shin, J., Xie, S.: Representation alignment for generation: Training diffusion transformers is easier than you think. arXiv preprint arXiv:2410.06940 (2024)
66. Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid loss for language image pre-training. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11975–11986 (2023)
67. Zhang, K., Mo, L., Chen, W., Sun, H., Su, Y.: Magicbrush: A manually annotated dataset for instruction-guided image editing. Advances in Neural Information Processing Systems **36**, 31428–31449 (2023)
68. Zhang, X., Liao, J., Zhang, S., Meng, F., Wan, X., Yan, J., Cheng, Y.: Videorepa: Learning physics for video generation through relational alignment with foundation models. arXiv preprint arXiv:2505.23656 (2025)
69. Zhang, Y., Wei, Y., Jiang, D., ZHANG, X., Zuo, W., Tian, Q.: Controlvideo: Training-free controllable text-to-video generation. In: The Twelfth International Conference on Learning Representations
70. Zhang, Z., Long, F., Li, W., Qiu, Z., Liu, W., Yao, T., Mei, T.: Region-constraint in-context generation for instructional video editing. arXiv preprint arXiv:2512.17650 (2025)
71. Zhao, H., Ma, X.S., Chen, L., Si, S., Wu, R., An, K., Yu, P., Zhang, M., Li, Q., Chang, B.: Ultraedit: Instruction-based fine-grained image editing at scale. Advances in Neural Information Processing Systems **37**, 3058–3093 (2024)
72. Zi, B., Peng, W., Qi, X., Wang, J., Zhao, S., Xiao, R., Wong, K.F.: Minimax-remover: Taming bad noise helps video object removal. arXiv preprint arXiv:2505.24873 (2025)
73. Zi, B., Ruan, P., Chen, M., Qi, X., Hao, S., Zhao, S., Huang, Y., Liang, B., Xiao, R., Wong, K.F.: Señorita-2m: A high-quality instruction-based dataset for general video editing by video specialists. arXiv preprint arXiv:2502.06734 (2025)