ALIVE: INTERACTION-ALIGNED OBJECT INSERTION FOR FIRST-FRAME-GUIDED VIDEO EDITING

Anonymous authors
Paper under double-blind review



Figure 1: **Bringing inserted objects to life.** Each panel shows the source video, edited first frame and object-only instruction, followed by corresponding results from Señorita and ALIVE. ALIVE enables added objects to be picked up, worn, opened, or cut in response to source actions.

ABSTRACT

Current video editors can insert objects but often struggle to make them participate in interactions such as being picked up or manipulated. We introduce ALIVE, a framework that makes inserted objects "alive" through coherent interactions with the source video's contents, using an edited first frame and an instruction naming only the added object. We curate 35,800 editing pairs combining 3D-rendered, model-generated, and real-world videos with general editing pairs from ROSE. Each pair differs in the target object's presence while preserving the surrounding action, teaching editors coordinated object behavior and source preservation. We further train a vision-language model (VLM) to predict interaction guidance from the same inputs. We introduce the ALIVE-interaction benchmark to assess interaction fidelity, source preservation, and visual coherence using a unified VLM-based protocol, and evaluate on the general video object insertion benchmark. Without VLM guidance, ALIVE improves Overall over the strongest evaluated baseline by 43.9% and 4.4% on the two benchmarks, respectively. VLM-predicted guidance further improves the ALIVE-interaction score by 0.95 points without additional user inputs.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Recent advances in video generation and editing have enabled high-quality results across diverse tasks, including object insertion with realistic shadows and reflections (Liu et al., 2025b; Fu et al., 2026). However, making these objects respond coherently to source actions remains challenging. In Figure 1, an added mug should move with the hand that lifts it, and added dough should separate as a knife cuts through it. Our goal is to make inserted objects "alive": not merely visible in the video, but part of its world, responding to the actions unfolding around them.

This capability poses two challenges: data construction and interaction modeling. Interaction editing pairs are scarce: their construction requires finding interaction videos and changing the target object's presence while preserving the surrounding scene and action. For modeling, the source video provides action cues but does not directly show the added object's response. The editor must infer the object's evolving motion or state as these actions unfold.

ALIVE addresses these challenges, combining interaction-focused editing data with a vision-language model (VLM) that infers how added objects should respond to source actions. We curate 35,800 editing pairs combining 3D-rendered, model-generated, and real-world videos with general editing pairs from ROSE. These sources cover varied interactions and visual domains, supporting general object insertion. Quality checks exclude implausible interactions and unintended changes to the surrounding scene. We annotate multi-level interaction descriptions, including temporally localized prompts, to study which guidance benefits editing and supervise VLM prediction.

Using these data, we train a first-frame-guided diffusion editor (Ouyang et al., 2024; Liu et al., 2025b). Users provide a source video, an edited first frame, and an object-only instruction. The edited frame fixes the object's initial appearance and placement, leaving subsequent behavior to follow source actions. We train the VLM to predict temporally localized interaction prompts from the same inputs, without user-provided interaction descriptions, per-frame masks, or trajectories.

We introduce the ALIVE-interaction benchmark to test whether added objects participate coherently in source actions while preserving the surrounding content. The general video object insertion benchmark further assesses broader editing quality. Without VLM guidance, ALIVE improves Overall scores over the strongest evaluated baseline by 43.9% and 4.4% on the two benchmarks, respectively. Predicted temporal prompt guidance further improves interaction performance. Ablations examine contributions from data composition and interaction guidance.

Our contributions are:

- A dataset of 35,800 video editing pairs combining interaction-focused and general insertion data, with complementary visual sources and multi-level prompt annotations.
- A diffusion editing framework that uses a VLM to predict interaction descriptions from the original inputs and aligns their guidance with the corresponding video intervals.
- We introduce the ALIVE-interaction benchmark and show that ALIVE outperforms baselines on interaction and general insertion, supported by ablations on data and guidance.

2 RELATED WORK

Video editing and paired video. EditVerse, UNIC, and VACE unify diverse video editing tasks through shared conditioning interfaces (Ju et al., 2026; Ye et al., 2025; Jiang et al., 2025). Goku-Edit and OpenVE-Edit support instruction-guided editing (Liang et al., 2026; He et al., 2025), while Kiwi-Edit unifies instruction and reference guidance (Lin et al., 2026). First-frame-guided methods propagate initial visual edits through source videos. AnyV2V and I2VEdit build on pretrained image-to-video models (Ku et al., 2024; Ouyang et al., 2024), while LoRA-Edit uses mask-aware adaptation (Gao et al., 2026a). Señorita, GenProp, and PropFly learn editing or propagation from paired or synthetic supervision (Zi et al., 2025; Liu et al., 2025b; Seo et al., 2026). PISCO and NovaEdit support sparse-keyframe-guided video editing (Gao et al., 2026b; Pan et al., 2026).

Paired datasets support these advances across tasks. Señorita-2M, Goku, and OpenVE cover broad editing tasks, while EffectErase and ROSE provide object removal / insertion pairs (Zi et al., 2025; Liang et al., 2026; He et al., 2025; Fu et al., 2026; Miao et al., 2025). ALIVE focuses on first-frame-guided object insertion with coherent responses to source actions, supported by interaction-focused editing pairs and multi-level prompt annotations for studying interaction guidance.

Interactive video editing. Interactive editing spans user control in streaming workflows and physical interactions within scenes. EditStream, JoyAI-Video-Edit, StreamEdit, and Vidu S2-Editing support streaming editing (Zhou et al., 2026; Xiao et al., 2026; Jiao et al., 2026; Zhang et al., 2026). EgoPlay times edits via user-specified source events (Mai et al., 2026). EgoEdit supports instruction-guided egocentric editing and real-time streaming (Li et al., 2026). Within scenes, VOID revises downstream physical interactions after object removal (Motamed et al., 2026), while DynaEdit edits actions and dynamics through text (Kulikov et al., 2026). We study first-frame-guided object insertion, where added objects respond to source actions without user-provided interaction descriptions.

VLM guidance for video editing. UniVideo and Omni-Video 2 integrate multimodal understanding with video generation and editing (Wei et al., 2026; Yang et al., 2026). DynVFX uses a VLM to describe a scene augmented with dynamic content (Yatim et al., 2025); Aurora plans edits and obtains missing conditions through tools (Yu et al., 2026). VOID identifies regions affected by object removal to guide counterfactual generation (Motamed et al., 2026). ALIVE trains a VLM to infer the inserted object's possible responses from the source video, edited first frame, and object-only instruction, expressing them as temporally localized prompts guiding corresponding video intervals.

3 ALIVE Dataset and Benchmark

Interaction-focused editing pairs are scarce, and evaluating object insertion requires assessing responses to source actions beyond appearance. We introduce ALIVE, comprising editing pairs from four complementary sources, multi-level prompt annotations for studying interaction guidance, and the ALIVE-interaction benchmark for evaluating interaction quality and source preservation.

3.1 Interaction Pair Construction

Constructing these pairs requires finding suitable interaction videos and varying the target object's presence while preserving the surrounding scene and action. We combine *3D-rendered videos*, *Model-generated videos*, and *Real-world videos* for interaction examples across visual domains, supplemented by *General editing pairs (ROSE)* for insertion without object interaction. Each pair comprises an object-absent source video X and an object-present target video Y; its first frame provides the edited first frame E_0. Figure 2 summarizes construction.

3D-rendered videos. To obtain precisely aligned editing pairs, we use 3D assets from SpaceTimePilot (Huang et al., 2025b; Lu et al., 2025). We manually select segments featuring object interactions and render each scene from multiple first- and third-person viewpoints. For each viewpoint, we render sequences with and without the target object while keeping the surrounding scene, camera trajectory, and animation aligned. These renders directly provide Y, X, and object masks.

Model-generated videos. To broaden interaction and scene coverage beyond available assets, we use GPT to design prompts covering common interaction types and synthesize object-present videos with Wan2.2 (Wan et al., 2025). SAM3 (Carion et al., 2026) segments and tracks the target object, and UnderEraser (Liu et al., 2026) removes it to construct the paired source video.

Real-world videos. For real-world visual supervision, we curate interaction clips from ARCTIC (Fan et al., 2023), HOCap (Wang et al., 2025), HOI4D (Liu et al., 2022), and HOT3D (Banerjee et al., 2025). Target masks come from provided segmentations or projected object geometry, with refinement as needed. UnderEraser removes the target while retaining the surrounding action.

General editing pairs (ROSE). Objects should respond to interactions but also behave appropriately in their absence, for example by remaining stationary in the scene. We include general editing pairs from ROSE (Miao et al., 2025) to support this behavior, alongside effects such as shadows and reflections. We reverse its removal pairs: the object-absent result becomes X, and the original object-present video becomes Y.

Pair quality. We use Qwen3.6-27B (Qwen Team, 2026) and GPT-5.6 Luna/Sol for source-specific quality checks, excluding interaction videos with no visible object interaction or poor visual quality, and pairs with incomplete or visibly incorrect object removal.

Our 35,800 unique pairs comprise 14,362 3D-rendered, 5,896 model-generated, 8,542 real-world, and 7,000 ROSE general editing pairs. They provide precisely aligned pairs, common interactions, real-world examples, and cases without object interaction. Section 5.3.1 evaluates different sources.

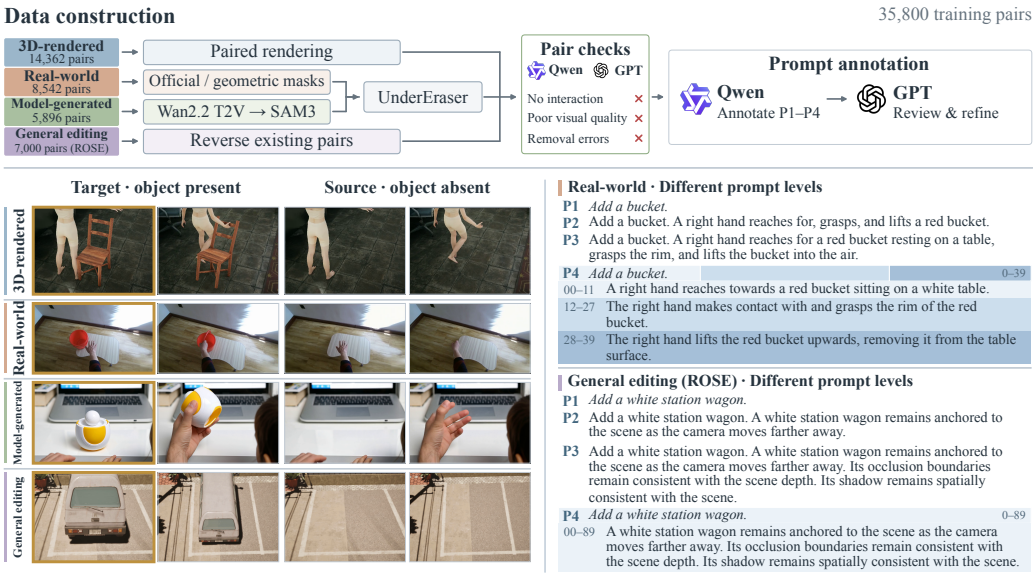

Figure 2: **ALIVE data construction and prompt annotation.** Top: construction and quality filtering. Bottom: editing pairs from four sources (left) and multi-level prompts (right). P0 (no text) and P1 (object identity only) omit interaction descriptions; P2–P4 provide interaction guidance. Gold outlines mark edited first frames; P4 shading marks chunks with inclusive frame ranges.

3.2 Multi-Level Interaction Annotation

We investigate whether prompts improve object interactions in video editing and which descriptions are most effective. We annotate P1–P4 with varying semantic detail and temporal structure, alongside P0 (no text). P1 names only the added object. Neither P0 nor P1 describes interactions, leaving the editor to infer plausible responses from the source video and E_0. P2 describes the principal interaction; P3 details its sequence and outcome; P4 describes temporal chunks. P2–P4 provide interaction information to the editor. P4 can span the full clip when no phase decomposition is needed. Figure 2 illustrates these differences. These annotations enable VLM-predicted interaction prompts from the source video, E_0, and P1, without user-specified interaction descriptions (Section 4.3).

Qwen3.6-27B assists annotation using ordered target-video frames, target masks, and available object or action metadata, followed by selective GPT verification and refinement against the videos. Inconsistent descriptions are corrected or excluded at the affected level. For ROSE cases without object manipulation, prompts describe scene motion, occlusion, and visible object-related effects.

3.3 Evaluation Benchmarks

ALIVE-interaction benchmark. We construct insertion tasks to assess coherent object responses to source actions. Each case provides an object-absent source video, an edited first frame, an object-present reference video, and P1–P4 annotations for prompt-level comparisons (Table 3), with P0 using no text. Its 128 cases include held-out samples from the 3D-rendered (10), model-generated (42), and real-world sources (19: ARCTIC 7, HOCap 1, HOI4D 5, and HOT3D 6), plus two out-of-domain subsets: 31 HOIGen-1M (Liu et al., 2025a) cases and 26 videos we recorded (RealShot).
General video object insertion benchmark. Alongside interaction-focused editing, we evaluate general insertion on 103 cases: 60 from ROSE and 43 from EffectErase. A common protocol evaluates editing fidelity, source preservation, and visual coherence on both benchmarks (Section 5.1).

4 Method

We present a first-frame-guided diffusion editor that learns inserted objects' responses to source actions (Figure 3). Section 4.2 details its visual and prompt conditioning and training; Section 4.3 describes how a VLM uses visual understanding to infer P4 chunk prompts from the same inputs for temporal guidance.

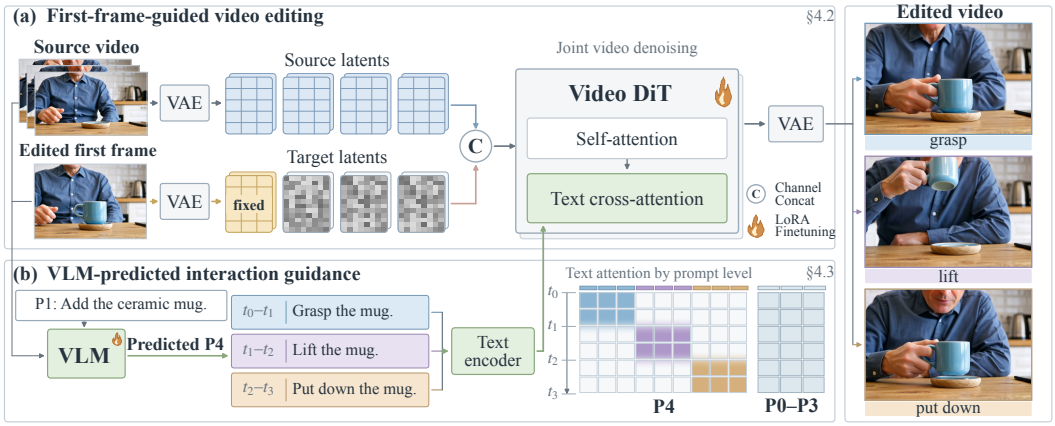

Figure 3: **ALIVE overview.** (a) Source and noisy target latents are channel-concatenated (C); the edited first-frame latent stays clean (gold). Flames mark editor and VLM LoRA fine-tuning. (b) A supervised VLM predicts P4 descriptions and intervals from the same visual inputs and P1. P4 texts are independently encoded. Attention maps contrast shared P0–P3 with time-local P4 conditioning.

4.1 Task Definition

Given a source video $X = (x_0, \ldots, x_{T-1})$, an edited first frame E_0, and an identity-only insertion instruction p_1, we aim to generate an edited video in which the added object participates coherently in the source action while unrelated scene content is preserved. The edited frame specifies the object's appearance and initial placement, p_1 names the object, and X provides the action context. The editor G_θ infers subsequent object interactions from these inputs:

$$\hat{Y} = G_\theta(X, E_0, p_1). \tag{1}$$

The complete target video Y provides training supervision but is unavailable at inference.

4.2 Video Editing Model

Visual conditioning. To propagate the initial insertion while preserving source actions, we condition the editor on E_0 and the full source video. Our main implementation adapts LTX-2.5 (HaCohen et al., 2026). A frozen video encoder maps source–target pairs to aligned latent grids z_X and z_Y. We concatenate clean source latents with noisy target latents z_t at matching spatiotemporal positions:

$$h_t = W_{\text{in}}[z_t\,;\, z_X]_{\text{channel}} + b. \tag{2}$$

This supplies source context without increasing the number of video tokens. We initialize the target half of W_{in} from the pretrained projection and the source half to zero. The encoded E_0 anchors the first target frame; its conditioned tokens remain clean throughout denoising.

Prompt conditioning. The editor supports clip-level and temporally localized text conditioning. P0 uses an empty prompt while retaining the source video and E_0. For P1–P3, the backbone's standard text-conditioning path encodes the full prompt as one context shared across frames: P1 specifies object identity; P2 and P3 also describe interactions.

P4 aligns text guidance with source-action stages through descriptions d_k and temporal intervals I_k. Each d_k specifies the added object and its interaction during I_k. We independently encode these descriptions into contexts C_k of length L_k and concatenate them. For video token i, let w_{ik} denote its normalized temporal overlap with I_k. Attention to text token j in context k is

$$A_{ij} = \text{softmax}_j\left(\frac{q_i^\top k_j}{\sqrt{d}} + \log\frac{w_{ik}}{L_k}\right), \qquad j \in C_k, \tag{3}$$

where q_i, k_j, and d are the query, key, and key dimension. Zero overlap masks the context; dividing by L_k prevents longer descriptions from gaining prior attention mass solely through length. This routing modifies only video-to-text cross-attention; all video chunks are denoised jointly.

Training. We train a single editor with a mixture of prompt levels P0–P4 using LoRA (Hu et al., 2021) on ALIVE pairs. For each sampled level, we use pairs with valid annotations for that level. For unconditioned target tokens, the noisy latent at noise level t is $z_t = (1-t)z_Y + t\epsilon$, with $\epsilon \sim \mathcal{N}(0, I)$. We optimize the native flow-matching objective (Lipman et al., 2022),

$$\mathcal{L}_{\text{edit}} = \mathbb{E}_{Y,X,p,t,\epsilon}\big[\|v_\theta(z_t, t; z_X, E_0, p) - (\epsilon - z_Y)\|_M^2\big], \tag{4}$$

where v_θ is the editor's velocity prediction with parameters θ, p is the sampled prompt, and $\|\cdot\|_M^2$ averages squared error over target tokens not conditioned on E_0. For VLM-guided editing, we condition the editor on P4 prompts predicted from the original inputs, as described next.

4.3 VLM-Based Interaction Guidance

To guide object interactions over time, we use a VLM's visual understanding to infer P4 chunk prompts from the source video, edited first frame, and object-only instruction. These prompts describe plausible interactions and their temporal intervals without additional interaction description.

We LoRA-adapt Qwen3.6-27B on the P4 annotations in Section 3.2. Given temporally ordered source frames, E_0, and P1, denoted by $u = (X, E_0, p_1)$, the predictor learns chunk descriptions and intervals through cross-entropy on response tokens only. Annotation may use the target video, but prediction uses only u during training and inference. At inference, the VLM F_ϕ predicts P4 guidance $\hat{p}_4$, replacing p_1 as the editor's text condition:

$$\hat{p}_4 = F_\phi(X, E_0, p_1), \qquad \hat{Y} = G_\theta(X, E_0, \hat{p}_4). \tag{5}$$

The routing in Section 4.2 applies each description to its temporal interval, guiding joint generation of the complete video.

5 Experiments

We evaluate whether ALIVE enables inserted objects to participate in source actions while retaining general insertion quality. We then examine how training-data composition supports these capabilities and whether interaction prompts provide further improvements.

5.1 Experimental Setup

We evaluate on the two benchmarks introduced in Section 3.3.

Compared methods. We compare AnyV2V, Señorita, NovaEdit, PropFly, and I2VEdit using their official conditioning interfaces and prompt formats. Both ALIVE variants share an LTX-2.5 model trained on our 35,800 pairs with mixed prompt levels: ALIVE uses object-identity text (P1), while ALIVE + VLM uses P4 predicted from the same source video, edited first frame, and P1. P2–P4 add interaction guidance beyond P1's object identity. Following their official prompt formats, AnyV2V and PropFly receive P2 descriptions, marked † for this additional guidance. We exclude instruction-only editors because text specifies initial object placement less precisely than E_0; placement differences may prevent the intended interaction, confounding comparisons against methods sharing the edited first frame. We also exclude mask-guided VACE and LoRA-Edit configurations because per-frame masks provide object states and motion that our task requires the editor to infer. See Appendix A.1–A.3 for training, inference, and additional ablations.

Evaluation. A reference-aware VLM judge (`gpt-5.6-sol`, reasoning effort `high`) assesses interaction fidelity, source preservation, and visual coherence under a shared rubric across both benchmarks. It receives 16 uniformly sampled output frames, including both endpoints, with corresponding source and reference frames, E_0, and the object-addition instruction. For I2VEdit, we evaluate all 14 output frames from its official implementation. Overall aggregates six dimensions on a 0–100 scale; Table 1 reports Overall, task fidelity, and motion/propagation. Beyond VLM scores, we follow EditVerse (Ju et al., 2026), reporting PickScore, frame-level CLIP and video-level ViCLIP alignment, and CLIP/DINOv2-based temporal consistency. Instructions, preprocessing alignment, and scoring details appear in Appendix A.4.

5.2 Comparison with Existing Video Editors

ALIVE substantially improves interaction-focused insertion while retaining strong general insertion quality (Table 1). With object-identity text alone, it exceeds the strongest evaluated baseline by

Table 1: Comparison on interaction-focused and general object insertion. Both ALIVE variants share the LTX-2.5 (22B) editor; VLM P4 is predicted from source/E_0/P1. † marks interaction descriptions supplied. VLM scores use a 0–100 scale; automatic metrics retain native scales. Higher is better throughout; bold/underline denote best/second-best scores, including ties.

		VLM evaluation			Automatic metrics				
Method	Guidance	Overall	Task	Motion	PickScore	Frame CLIP	Video ViCLIP	TC-CLIP	TC-DINO
ALIVE-interaction benchmark (128 cases)									
Señorita	P1	60.24	53.52	36.52	19.95	24.04	19.09	0.9572	0.9494
NovaEdit	P0	36.36	8.59	0.39	18.94	21.15	15.41	0.9744	0.9734
I2VEdit	P0	56.55	48.44	33.79	19.68	23.48	18.61	0.9363	0.9132
AnyV2V	P2†	42.75	44.53	31.25	19.77	<u>24.12</u>	19.48	0.9141	0.8916
PropFly	P2†	42.31	35.94	21.68	19.40	22.81	17.58	0.9390	0.9274
ALIVE	P1	<u>86.71</u>	<u>93.75</u>	<u>86.13</u>	**20.34**	**25.00**	<u>20.18</u>	**0.9795**	**0.9759**
ALIVE + VLM	VLM P4	**87.66**	**93.95**	**86.91**	<u>20.33</u>	**25.00**	**20.22**	<u>0.9791</u>	<u>0.9753</u>
general video object insertion benchmark (103 cases)									
Señorita	P1	88.39	88.83	83.74	20.30	**23.98**	19.35	0.9713	0.9723
NovaEdit	P0	45.66	27.18	19.66	19.49	21.49	16.74	0.9820	0.9813
I2VEdit	P0	82.21	84.95	79.13	20.09	23.50	19.38	0.9556	0.9403
AnyV2V	P2†	48.03	56.07	47.09	20.09	23.71	**20.15**	0.9220	0.9020
PropFly	P2†	64.02	64.32	55.58	19.93	22.88	18.62	0.9603	0.9545
ALIVE	P1	**92.28**	**94.17**	**89.32**	<u>20.37</u>	23.74	19.47	<u>0.9845</u>	<u>0.9835</u>
ALIVE + VLM	VLM P4	<u>91.71</u>	<u>93.20</u>	<u>88.35</u>	**20.38**	<u>23.85</u>	<u>19.63</u>	**0.9849**	**0.9849**

26.46 points on the ALIVE-interaction benchmark (86.71 vs. 60.24) and by 3.89 points on general insertion (92.28 vs. 88.39). Its motion/propagation score on the interaction benchmark rises to 86.13, compared with 36.52 for Señorita, supporting more coherent coordination between inserted objects and source actions. VLM-predicted P4 further raises the interaction Overall score to 87.66. For general insertion, P1 remains stronger than predicted P4 (92.28 vs. 91.71), although both outperform the evaluated baselines. Predicted interaction guidance therefore builds on an already capable editor. Section 5.3.2 examines this contribution in detail.

Figure 4 compares book handling and basket carrying samples. ALIVE + VLM moves the inserted objects with the hands. In these examples, baselines can leave objects stationary, omit them, or distort their shape.

5.3 Ablation Studies

5.3.1 Data Sources

We examine how each data source contributes to interaction-focused and general object insertion, and whether combining them benefits both. We compare single-source training with a four-source mixture sampled in proportion to source size. All conditions use LTX-2.5, rank-128 LoRA, P1, and 5k training steps, with fixed evaluation cases and generation seeds. Table 2 lists source-pool sizes.

Different sources favor different capabilities: model-generated pairs perform best on interaction, while ROSE leads individual sources on general insertion. However, ROSE-only training scores just 60.12 on the ALIVE-interaction benchmark, compared with 82.35 for the mixture, highlighting the value of interaction-focused training pairs. The mixture remains within 0.46 points of model-generated-only training on interaction while improving general insertion by 7.61 points. Its highest general-insertion and pooled scores support the complementary value of our data sources.

5.3.2 Prompt Levels and VLM Guidance

Having established the editor's interaction capability, we test whether explicit descriptions of object behavior provide useful additional guidance. Table 3 compares annotated and VLM-predicted prompts while keeping editor weights, visual inputs, and generation seeds fixed. P0 supplies no text, P1 names the object, and annotated P2–P4 additionally describe interactions. VLM variants predict those descriptions from source/E_0/P1.



Figure 4: **Interaction-focused object insertion.** Book handling (a) and basket carrying (b), each shown at the first frame, 40%, and 90% of the video. Square AnyV2V frames are displayed more narrowly. † marks provided interaction guidance, following the methods' official prompt formats.

Table 2: Single-source and mixed-source training with the same 5k-step budget. All scores use our 16-frame evaluation (0–100; ↑). Pooled averages all 231 cases with equal weight: $(128\,S_{\text{int}} + 103\,S_{\text{gen}})/231$, where S_{int} and S_{gen} are the benchmark means. Bold and underline mark the best and second-best scores per column.

Training source	Pairs	Pooled	ALIVE-interaction benchmark ($N = 128$)			general video object insertion benchmark ($N = 103$)		
		Overall	Overall	Task	Motion	Overall	Task	Motion
3D-rendered videos	14,362	74.85	73.08	76.37	62.89	77.06	78.16	71.12
Real-world videos	8,542	<u>81.51</u>	80.19	88.48	79.49	83.16	83.50	80.10
Model-generated videos	5,896	81.26	**82.81**	**90.82**	**82.23**	79.32	79.85	71.84
General editing pairs (ROSE)	7,000	70.93	60.12	60.74	45.12	<u>84.36</u>	<u>88.35</u>	<u>81.31</u>
All four sources	35,800	**84.39**	<u>82.35</u>	<u>89.65</u>	<u>82.03</u>	**86.93**	**88.83**	**83.01**

Effects of prompt levels. P0 and P1 achieve Overall scores of 85.38 and 86.71, respectively, showing that the editor can infer object interactions from visual inputs. P2 provides little additional benefit, while the more detailed P3 and temporally localized P4 improve Overall to 87.37 and 88.69. These results suggest that the form of interaction guidance matters, with P4 performing best among the annotated formats.

VLM-predicted guidance. Predicted P4 achieves the highest Overall among the VLM variants, raising P1 from 86.71 to 87.66 using the original editing inputs. P4's gains include temporal consistency and background preservation as well as motion/propagation. Thus, predicted P4 offers an improvement in overall interaction-editing quality.

Figure 5 connects the predicted descriptions to the resulting object behavior. P4 specifies "lower it onto the head" before adjusting the helmet straps under the chin, clarifying the placement sequence; its output avoids the strap artifacts around the eyes seen with P1. For the seedling, it describes

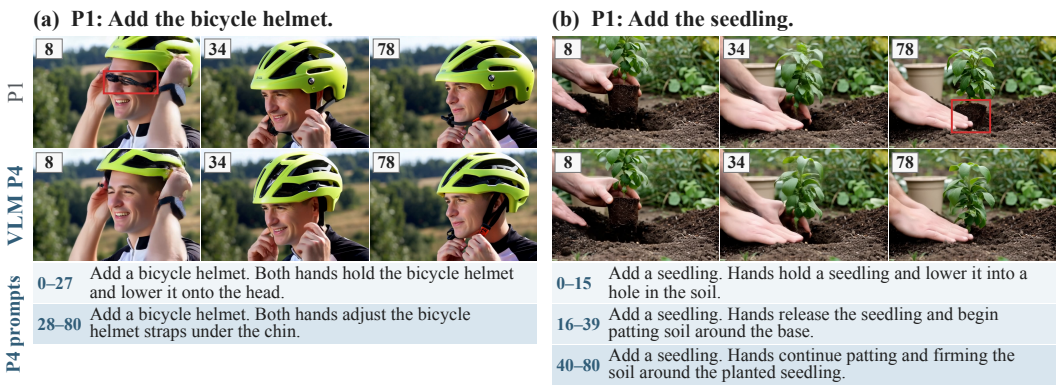

Figure 5: **Effect of VLM-predicted interaction guidance.** P1 and predicted P4 condition the same editor with fixed visual inputs. Predicted prompts and inclusive frame ranges appear below. Red boxes highlight strap artifacts around the eyes (a) and an exposed root ball (b) in the P1 results.

Table 3: Prompt guidance on the ALIVE-interaction benchmark. All rows use the same editor, visual inputs, and case-specific seeds. Annotated P2–P4 supply interaction information; VLM variants infer it from source/E_0/P1. P4 uses temporal chunks. Scores are 0–100 (↑); bold and underline mark first and second place. ID: identity; BG: background; Temp.: temporal consistency.

Guidance	N	Task	ID	Motion	BG	Temp.	Visual	Overall
P0	128	91.41	81.84	83.40	<u>93.36</u>	80.47	75.00	85.38
P1	128	93.75	83.98	86.13	91.80	81.05	75.20	86.71
P2	128	93.16	82.23	86.91	92.38	80.66	77.15	86.68
P3	128	<u>94.73</u>	83.40	<u>88.48</u>	91.60	81.25	75.59	87.37
P4	128	**95.12**	**85.94**	**88.67**	**93.55**	**82.81**	**78.32**	**88.69**
VLM-enhanced P1								
VLM P2	128	93.55	<u>84.57</u>	87.89	91.80	<u>81.84</u>	76.37	87.33
VLM P3	128	94.34	83.20	86.33	92.19	<u>81.84</u>	<u>77.34</u>	87.17
VLM P4	128	93.95	<u>84.57</u>	86.91	<u>93.36</u>	**82.81**	76.76	<u>87.66</u>

lowering, releasing, and firming the surrounding soil; the generated plant is better seated in the soil as the hands press around it.

Limitations and future work. Despite these improvements, ALIVE faces limitations. The source video and edited first frame do not fully specify geometry, contact conditions, or forces, leaving multiple plausible object responses. For example, a soft object may deform differently under pressure depending on stiffness, while a partially occluded grasp may allow a cup to remain upright or tilt during lifting. ALIVE aims to generate plausible object responses consistent with source actions, without implausible deformation, interpenetration, or unsupported floating. Additionally, VLM-predicted guidance errors can cause incorrect behavior, and the editor still struggles with complex cloth deformations and rapid, large-angle rotations. Future work includes improving guidance reliability, strengthening the editor's ability to generate complex interactions, and incorporating explicit physical information for greater accuracy and control. Combining ALIVE with Self Forcing (Huang et al., 2025a) or EditStream (Zhou et al., 2026) could enable real-time streaming object insertion.

6 Conclusion

We presented ALIVE, a framework for inserting objects that participate coherently in a source video's interactions. Our dataset combines complementary construction routes, and our benchmarks assess both interaction-focused and general object insertion. Experiments show the benefit of interaction-focused editor training and clarify the effects of data composition and prompt guidance. VLM-predicted guidance further improves interaction editing without additional user input, with P4 performing best among the evaluated prediction formats.

AI USE STATEMENT

In this work, we used generative AI tools for dataset construction and annotation, evaluation, and code implementation. We generated videos with Wan 2.2 14B, annotated prompts using Qwen and GPT, and used GPT as a VLM-based evaluator. GPT-5.6 and GPT-6 assisted with code implementation; the authors reviewed the code and validated it through experiments. The authors designed the experiments and confirmed the final design after discussing its soundness with GPT. The authors led result interpretation, using GPT for analytical feedback and language polishing, and verified the resulting explanations. We have not used generative AI tools for writing mathematical proofs, and other proof-related tasks are not applicable to this work. Additionally, we used AI to draft, translate, and polish manuscript text from author-written drafts and to identify potentially overlooked related work. The authors reviewed and revised all AI-assisted manuscript content. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Shangchen Han, Fan Zhang, Linguang Zhang, Jade Fountain, Edward Miller, Selen Basol, et al. Hot3d: Hand and object tracking in 3d from egocentric multi-view videos. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7061–7071. IEEE, 2025.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International conference on learning representations*, volume 2026, pp. 138846–138923, 2026.

Zicong Fan, Omid Taheri, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J Black, and Otmar Hilliges. Arctic: A dataset for dexterous bimanual hand-object manipulation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12943–12954. IEEE, 2023.

Yang Fu, Yike Zheng, Ziyun Dai, and Henghui Ding. Effecterase: Joint video object removal and insertion for high-quality effect erasing. *arXiv preprint arXiv:2603.19224*, 2026.

Chenjian Gao, Lihe Ding, Cai Cai, Zhanpeng Huang, Zibin Wang, and Tianfan Xue. Controllable first-frame-guided video editing via mask-aware lora fine-tuning. In *International Conference on Learning Representations*, volume 2026, pp. 61741–61765, 2026a.

Xiangbo Gao, Renjie Li, Xinghao Chen, Yuheng Wu, Suofei Feng, Qing Yin, and Zhengzhong Tu. Pisco: Precise video instance insertion with sparse control. *arXiv preprint arXiv:2602.08277*, 2026b.

Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, et al. Ltx-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026.

Haoyang He, Jie Wang, Jiangning Zhang, Zhucun Xue, Xingyuan Bu, Qiangpeng Yang, Shilei Wen, and Lei Xie. Openve-3m: A large-scale high-quality dataset for instruction-guided video editing. *arXiv preprint arXiv:2512.07826*, 2025.

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

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025a.

Zhening Huang, Hyeonho Jeong, Xuelin Chen, Yulia Gryaditskaya, Tuanfeng Y Wang, Joan Lasenby, and Chun-Hao Huang. Spacetimepilot: Generative rendering of dynamic scenes across space and time. *arXiv preprint arXiv:2512.25075*, 2025b.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17191–17202. IEEE, 2025.

Guanlong Jiao, Chenyangguang Zhang, Jia Jun Cheng Xian, Zewei Zhang, and Renjie Liao. Streamedit: Training-free video editing via few-step streaming video generation. In *European Conference on Computer Vision*, pp. 1–20. Springer, 2026.

Xuan Ju, Tianyu Wang, Yuqian Zhou, He Zhang, Qing Liu, Cherry Zhao, Zhifei Zhang, Yijun Li, Yuanhao Cai, Shaoteng Liu, et al. Editverse: Unifying image and video editing and generation with in-context learning. In *International Conference on Learning Representations*, volume 2026, pp. 137234–137255, 2026.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Max Ku, Cong Wei, Weiming Ren, Harry Yang, and Wenhu Chen. Anyv2v: A tuning-free framework for any video-to-video editing tasks. *arXiv preprint arXiv:2403.14468*, 2024.

Vladimir Kulikov, Roni Paiss, Andrey Voynov, Inbar Mosseri, Tali Dekel, and Tomer Michaeli. Versatile editing of video content, actions, and dynamics without training. In *European Conference on Computer Vision*, pp. 448–466. Springer, 2026.

Runjia Li, Moayed Haji-Ali, Ashkan Mirzaei, Chaoyang Wang, Arpit Sahni, Ivan Skorokhodov, Aliaksandr Siarohin, Tomas Jakab, Junlin Han, Sergey Tulyakov, et al. Egoedit: Dataset, real-time streaming model, and benchmark for egocentric video editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16042–16053, 2026.

Sen Liang, Cong Wang, Zhentao Yu, Fengbin Guan, Zhengguang Zhou, Teng Hu, Youliang Zhang, Yuan Zhou, Xin Li, Qinglin Lu, et al. Goku: A million-scale universal dataset and benchmark for instruction-based video editing. *arXiv preprint arXiv:2606.30599*, 2026.

Yiqi Lin, Guoqiang Liang, Ziyun Zeng, Zechen Bai, Yanzhe Chen, and Mike Zheng Shou. Kiwi-edit: Versatile video editing via instruction and reference guidance. *arXiv preprint arXiv:2603.02175*, 2026.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Dingming Liu, Wenjing Wang, Chen Li, Jing Lyu, and Haotian Dong. From understanding to erasing: Towards complete and stable video object removal. *arXiv preprint arXiv:2604.01693*, 2026.

Kun Liu, Qi Liu, Xinchen Liu, Jie Li, Yongdong Zhang, Jiebo Luo, Xiaodong He, and Wu Liu. Hoigen-1m: A large-scale dataset for human-object interaction video generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24001–24010. IEEE, 2025a.

Shaoteng Liu, Tianyu Wang, Jui-Hsien Wang, Qing Liu, Zhifei Zhang, Joon-Young Lee, Yijun Li, Bei Yu, Zhe Lin, Soo Ye Kim, et al. Generative video propagation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17712–17722. IEEE, 2025b.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20981–20990. IEEE, 2022.

Jiaxin Lu, Chun-Hao Paul Huang, Uttaran Bhattacharya, Qixing Huang, and Yi Zhou. Humoto: A 4d dataset of mocap human object interactions. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10886–10897. IEEE, 2025.

Jinjie Mai, Gordon Guocheng Qian, Willi Menapace, Arpit Sahni, Chaoyang Wang, Ashkan Mirzaei, Runjia Li, Sergey Tulyakov, Bernard Ghanem, Peter Wonka, et al. Egoplay: Event-triggered video editing for egocentric streams. *arXiv preprint arXiv:2607.24560*, 2026.

Chenxuan Miao, Yutong Feng, Jianshu Zeng, Zixiang Gao, Hantang Liu, Yunfeng Yan, Donglian Qi, Xi Chen, Bin Wang, and Hengshuang Zhao. Rose: Remove objects with side effects in videos. *arXiv preprint arXiv:2508.18633*, 2025.

Saman Motamed, William Harvey, Benjamin Klein, Luc Van Gool, Zhuoning Yuan, and Ta-Ying Cheng. Void: Video object and interaction deletion. In *European Conference on Computer Vision*, pp. 245–261. Springer, 2026.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Wenqi Ouyang, Yi Dong, Lei Yang, Jianlou Si, and Xingang Pan. I2vedit: First-frame-guided video editing via image-to-video diffusion models. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024.

Tianlin Pan, Jiayi Dai, Chenpu Yuan, Zhengyao Lv, Binxin Yang, Hubery Yin, Chen Li, Jing Lyu, Caifeng Shan, and Chenyang Si. Nova: Sparse control, dense synthesis for pair-free video editing. *arXiv preprint arXiv:2603.02802*, 2026.

Qwen Team. Qwen3.6-27B: Flagship-level coding in a 27b dense model, April 2026. URL `https://qwen.ai/blog?id=qwen3.6-27b`.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Wonyong Seo, Jaeho Moon, Jaehyup Lee, Soo Ye Kim, and Munchurl Kim. Propfly: Learning to propagate via on-the-fly supervision from pre-trained video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 43228–43238, 2026.

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.

Jikai Wang, Qifan Zhang, Yu-Wei Chao, Bowen Wen, Xiaohu Guo, and Yu Xiang. HO-cap: A capture system and dataset for 3d reconstruction and pose tracking of hand-object interaction. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025. URL `https://openreview.net/forum?id=hpu6r8oLw9`.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. In *International Conference on Learning Representations*, volume 2024, pp. 42055–42079, 2024.

Cong Wei, Quande Liu, Zixuan Ye, Qiulin Wang, Xintao Wang, Pengfei Wan, Kun Gai, and Wenhu Chen. Univideo: Unified understanding, generation, and editing for videos. In *International Conference on Learning Representations*, volume 2026, pp. 113905–113933, 2026.

Yicheng Xiao, Wenxun Dai, Xinran Qin, Lin Song, Maoquan Zhang, Hang Xu, Yukang Chen, Yitong Li, Guohui Zhang, Yuan Zhang, et al. Joyai-video-edit: Real-time open-ended video editing with autoregressive diffusion. *arXiv preprint arXiv:2608.03974*, 2026.

Hao Yang, Zhiyu Tan, Jia Gong, Luozheng Qin, Hesen Chen, Xiaomeng Yang, Yuqing Sun, Yuetan Lin, Mengping Yang, and Hao Li. Omni-video 2: Scaling mllm-conditioned diffusion for unified video generation and editing. *arXiv preprint arXiv:2602.08820*, 2026.

Danah Yatim, Rafail Fridman, Omer Bar-Tal, and Tali Dekel. Dynvfx: Augmenting real videos with dynamic content. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–12, 2025.

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

Yongsheng Yu, Ziyun Zeng, Zhiyuan Xiao, Zhenghong Zhou, Hang Hua, Wei Xiong, and Jiebo Luo. Aurora: Unified video editing with a tool-using agent. *arXiv preprint arXiv:2605.18748*, 2026.

Jintao Zhang, Kai Jiang, Jintao Chen, Xu Wang, Deyuan Liu, Jungang Li, Dechuang Chen, Ming Lin, Jingjiang Zhou, Haopeng Jin, et al. Vidu s2: Real-time interactive, editable, and spatial video generation. *arXiv preprint arXiv:2609.11638*, 2026.

Yuqian Zhou, Zhenghong Zhou, Zongze Wu, Cameron Smith, Richard Zhang, Jiebo Luo, Eli Shechtman, and Zhe Lin. Editstream: A unified autoregressive framework for interactive video generation and editing. *arXiv preprint arXiv:2608.21424*, 2026.

Bojia Zi, Penghui Ruan, Marco Chen, Xianbiao Qi, Shaozhe Hao, Shihao Zhao, Youze Huang, Bin Liang, Rong Xiao, and Kam-Fai Wong. Señorita-2m: A high-quality instruction-based dataset for general video editing by video specialists. In *NeurIPS D&B*, 2025.

A Appendix

A.1 Training Details

Diffusion editor training. We adapt LTX-2.5 with rank-128 LoRA on 35,800 pairs, using 81-frame clips at 832×480. Training uses AdamW with an initial learning rate of 10^{-4} and linear decay, a global batch size of 16, and 20,000 updates. We use 16 GPUs per run (A100 80GB, H100, or H200), with one sample per GPU. Training first selects P0–P4 with probabilities 0.1/0.3/0.2/0.2/0.2, then samples a video pair with the corresponding prompt annotation (no text for P0). Data-source ablations use P1 and 5,000 updates with the same batch size, evaluation cases, and generation seeds. Both ALIVE variants share the main editor; at inference, they generate 81 frames at 768×512 with 30 denoising steps.

VLM training. We adapt Qwen3.6-27B with rank-64 LoRA ($\alpha = 128$, dropout 0.05), training separate predictors for P2, P3, and P4. Each uses 16 A100 80GB GPUs, one sample per GPU, and two-step gradient accumulation, giving a global batch size of 32. We use AdamW with learning rate 5×10^{-5}, weight decay 0.01, and 32 warm-up updates followed by a constant learning rate. Inputs comprise 21 uniformly sampled source frames, E_0, and P1. We use the 2,048-update checkpoint for each predictor.

A.2 Inference Settings and Efficiency

Table 4 profiles the same six videos (three per benchmark) using one A100 80GB per video, with batch size one and a separate warm-up video. We report mean wall-clock time and the largest peak allocated GPU memory across the six cases. Timing includes conditioning, required inversion or per-video adaptation, sampling, decoding, and export; offline data preparation and one-time setup are excluded. Costs reflect each method's native output configuration and the memory-management settings specified below.

Table 4: Measured inference cost on A100 80GB. Output is width × height × frames; only denoising iterations are counted under Steps. Time includes each editor's required inversion and adaptation. Peak denotes PyTorch peak allocated memory in GiB. VLM prompt prediction is measured separately in the last row; dashes mean not applicable.

Method	Output	Steps	Time (s)	Peak (GiB)
Señorita	$768 \times 448 \times 33$	30	109.18	28.04
NovaEdit	$832 \times 480 \times 81$	50	407.70	19.63
I2VEdit	$1024 \times 576 \times 14$	25	863.62	65.16
AnyV2V	$512 \times 512 \times 16$	50	172.79	30.31
PropFly	$832 \times 480 \times 25$	50	71.63	21.14
Wan 14B, P1	$768 \times 512 \times 81$	30	304.64	64.46
LTX 22B, P1	$768 \times 512 \times 81$	30	154.16	39.43
LTX 22B, predicted P4	$768 \times 512 \times 81$	30	184.50	39.43
VLM P4 predictor	–	–	22.75	52.27

AnyV2V uses 500 inversion steps; I2VEdit includes inversion, 250 motion-LoRA updates, and native per-video model loading. Señorita and PropFly run without CPU model offloading. Wan retains both diffusion experts on GPU while staging its text encoder and VAE; LTX uses its native no-offload setting. The P4 editor uses the fixed predicted prompts from the quality evaluation. Separately, the 2,048-update VLM is profiled on the same editing inputs. Summing these separately measured stages gives an estimated 207.25 s per edit, excluding the overhead of switching models.

A.3 Additional Ablations

A.3.1 Training Length

Table 5 compares P1-only models trained for 5k and 20k steps on the same 35,800 pairs, using identical evaluation cases, generation seeds, and the 16-frame scoring protocol. The longer budget

raises interaction Overall from 82.35 to 86.46 and raises general-insertion Overall from 86.93 to 89.20. These are separately trained runs, not checkpoints from one training trajectory.

Table 5: Training budget for P1-only LTX 22B editors on the same 35,800 pairs (LoRA rank 128, batch 16). These are separately trained runs, not checkpoints from one trajectory. Overall (↑) is reported on the same cases and seeds. The main mixed-prompt editor is excluded because its prompt-training distribution differs. Both rows use the same 16-frame evaluation protocol.

Optimizer steps	Pairs	ALIVE-interaction benchmark	general video object insertion benchmark
5k	35,800	<u>82.35</u>	<u>86.93</u>
20k	35,800	**86.46**	**89.20**

A.3.2 Base Models and Model Size

Table 6 compares Wan and LTX editors trained for 5k steps and evaluated on the same 128/103-case benchmarks, using the current 16-frame evaluation protocol. The LTX editor here uses P1-only training, distinct from the mixed-prompt editor in Table 1.

Table 6: Backbone comparison using the same 35,800-pair collection, P1, and 5k steps. Scores report Overall (↑) under the current 16-frame protocol on 128 ALIVE-interaction and 103 general video object insertion cases. Best scores are bold and second-best underlined.

Backbone	Steps	ALIVE-interaction benchmark	general video object insertion benchmark
Wan 2.2 TI2V (5B)	5k	71.85	81.54
Wan 2.2 I2V (14B)	5k	**89.82**	**89.44**
LTX-2.5 (22B)	5k	<u>82.35</u>	<u>86.93</u>

Training efficiency. For 81-frame clips at 832×480, LoRA rank 128, and batch 16 per model or expert, LTX-2.5 takes 4.95 s per update on 16 A100 80GB GPUs; its 5k-step run takes 8.1 hours including checkpoint writes. Wan 14B takes approximately 45.4 s per expert update on 32 A100 80GB GPUs (16 per noise expert), implying approximately 63 hours for 5k updates at this rate. Update times are steady-state medians; the Wan duration is an extrapolation, excluding setup, checkpoint overhead, and interruptions. LTX's stronger VAE compression ($8 \times 32 \times 32$ versus $4 \times 8 \times 8$ in time, height, and width) reduces the video-token sequence length. This supports its practical efficiency. Although Wan 2.2 14B achieves higher scores in this comparison, we use LTX-2.5 for the main experiments because it is substantially faster to train.

A.4 Evaluation Protocol

We assess interaction fidelity, source preservation, and visual coherence using the VLM judge introduced in Section 5.1. Each candidate is evaluated independently with the same rubric and dimension weights across both benchmarks and all prompt levels.

Evaluation inputs. For each case, the judge receives source, reference, and candidate frames, the edited first frame E_0, and an object-addition instruction shared across methods (Table 7). The ALIVE-interaction benchmark uses P1; the general video object insertion benchmark uses "Add the object or person shown in the edited first frame." Reference frames beyond E_0 are provided only to the judge, not the editor.

Frame sampling and alignment. We align source and reference evidence to each generator's input preprocessing so that evaluation reflects the field of view and time span available during generation. We uniformly sample 16 native output frames, including both endpoints, and map them to the source frames selected by the generator. For I2VEdit, we use all 14 native frames without repetition or interpolation. Spatial alignment follows the generator's documented aspect-preserving cover resize and fixed center crop, including interpolation and rounding conventions.

Table 7: Inputs to the evaluation judge. The same roles and presentation are used for all benchmarks and methods. Only CANDIDATE is scored.

Input	Role
Basic instruction	Identify the intended addition without prescribing an action sequence.
SOURCE	Establish non-target scene content, camera motion, and interaction context.
Edited first frame E_0	Specify the intended object or person and initial edit.
REFERENCE	Provide identity, contact, occlusion, interaction, and natural-effects evidence.
CANDIDATE	Generated result to evaluate at every displayed timepoint.

At each timepoint, SOURCE, REFERENCE, and CANDIDATE appear side by side with their aspect ratios preserved. Four contact sheets show 5, 5, 5, and 1 timepoints; I2VEdit uses three sheets with 5, 5, and 4. The aligned E_0 is supplied separately as a PNG. Each labeled cell measures 384×240 pixels, and sheets use JPEG quality 88. Time labels indicate normalized positions rather than seconds. Layout margins, labels, and gaps are excluded from scoring; artifacts within video regions remain scoreable.

Scoring and aggregation. Table 8 defines six dimensions and their weights. For candidate i and dimension d, the judge assigns an integer $s_{i,d} \in \{0, 1, 2, 3, 4\}$: 0 denotes failure or contradictory evidence, 2 partial correctness with a material problem, and 4 full satisfaction supported by visible evidence. Scores 1 and 3 fall between these anchors. We assess plausible interactions consistent with SOURCE and E_0, allowing reasonable spatial and temporal differences from the reference (Figure 6). Missing required interactions, broken contact, impossible motion, and source corruption remain errors.

Table 8: Evaluation dimensions. Weights are identical across benchmarks and ablations.

Dimension	Criterion	Weight
Task fidelity	Complete addition and coherent participation in interactions supported by SOURCE and E_0.	0.25
Object identity	Stable identity, geometry, and attributes consistent with E_0.	0.15
Motion / propagation	Plausible motion or appropriate stability, contact, timing, and occlusion.	0.20
Background preservation	Faithful non-target source content and camera motion.	0.15
Temporal consistency	Stable appearance and smooth transitions without flicker or disappearance.	0.15
Visual quality	Realistic boundaries, lighting, and natural effects.	0.10

We aggregate scores as

$$\text{Overall}_i = 25 \sum_{d=1}^{6} w_d s_{i,d}, \qquad \text{Overall} = \frac{1}{N} \sum_{i=1}^{N} \text{Overall}_i, \qquad \sum_d w_d = 1. \tag{6}$$

Here, N is the number of evaluated cases and w_d the weight of dimension d. Dimension scores are also multiplied by 25 for reporting. The judge returns structured scores and explanations; its confidence indicates uncertainty but does not affect aggregation.

Complete judge prompt. Figure 6 reproduces the full task prompt supplied with the visual evidence. The placeholder <basic instruction> is replaced by the case's P1 instruction on the ALIVE-interaction benchmark, or "Add the object or person shown in the edited first frame." on the general video object insertion benchmark. For I2VEdit, only "The first four images show sixteen normalized-time samples" changes to "The first three images show fourteen normalized-time samples"; all scoring instructions remain identical.

Structured response. The request also supplies a JSON schema requiring five fields: `scores`, `justifications`, `failure_flags`, `judge_confidence`, and `summary`. Both `scores` and `justifications` contain all six dimension keys in Figure 6; values are integers in $[0, 4]$ and nonempty strings, respectively. Confidence is a number in $[0, 1]$, and the summary is a nonempty string. Additional keys are disallowed in the root object and both dimension objects. Failure flags form a possibly empty array drawn from `missing_target`, `wrong_target`, `motion_not_followed`, `incomplete_propagation`, `wrong_motion_or_placement`, `source_corruption`, `temporal_breakdown`, `first_frame_mismatch`, and `catastrophic_artifacts`.

Consistent comparisons. Comparisons within each table use matched cases and generation seeds. We reuse judgments for identical videos and evaluation inputs, retrying only failed requests without resampling successful scores.

Additional metrics. We compute PickScore (Kirstain et al., 2023), Frame CLIP (Radford et al., 2021), Video ViCLIP (Wang et al., 2024), TC-CLIP, and TC-DINO (DINOv2 (Oquab et al., 2023)) on the same candidate videos using the official EditVerse implementations (Ju et al., 2026) and fixed captions. These metrics retain their native scales and do not enter the VLM Overall score.

Judge one anonymous candidate video. Do not use tools, inspect files, or infer a model identity. The first four images show sixteen normalized-time samples, with SOURCE, REFERENCE, and CANDIDATE columns. The final image is the actual edited first-frame condition E0. Score only CANDIDATE. SOURCE shows the scene and its motion before the addition. E0 specifies the intended added object or person. REFERENCE provides evidence of identity, interaction, contact, occlusion, and natural effects, not an exact trajectory or pixel-level target. Model names, expanded generator prompts, and annotated action phases are hidden. The same basic addition instruction is used for all methods on a case.

Judge whether the added object belongs coherently in the source scene and responds plausibly to its visible motion and interactions. Reasonable differences in position, path, pose, or timing from REFERENCE are acceptable when consistent with SOURCE and E0. Do not demand a specific action sequence that cannot be inferred from SOURCE and E0. This does not excuse missing interactions, broken contact, unsupported motion, or failure to preserve the source. If no object motion is called for by the scene, a stationary object can be correct; a frozen output that suppresses source motion is not.

Give each dimension an integer score from 0 to 4. Intermediate scores follow adjacent anchors:

task_fidelity: 4=complete intended addition, with coherent participation in interactions supported by SOURCE and E0; 2=partly correct with a material omission; 0=absent or contradicted edit.

object_identity: 4=stable identity, geometry and attributes consistent with E0; 2=recognizable with material drift; 0=wrong or unrecognizable object.

motion_or_propagation: 4=plausible motion or appropriate stability, timing, support/contact, and occlusion consistent with the scene; 2=approximate with material errors; 0=unrelated or impossible behavior, or missing motion where required by the scene.

background_preservation: 4=faithful non-target SOURCE content and camera motion; 2=noticeable collateral changes; 0=substantial source corruption.

temporal_consistency: 4=stable, smooth geometry and appearance; 2=repeated moderate instability; 0=severe flicker, popping, disappearance or discontinuity.

visual_quality: 4=clean realistic boundaries, lighting and natural effects; 2=visible noncatastrophic artifacts; 0=garbled or unusable.

Do not reward a frozen video merely for frame similarity or penalize valid appearance variation. Give concise justifications citing visible early/middle/late evidence. If evidence is small, sparse or ambiguous, acknowledge this and lower judge_confidence; do not invent events. Return only the JSON required by the response schema: six scores, six justifications, supported failure_flags (or an empty list), judge_confidence, and a short summary.

PREPROCESSING ALIGNMENT: SOURCE, REFERENCE and E0 follow the generator's documented aspect-preserving cover resize, fixed center crop and source-frame mapping. Compare their common visible field of view. Do not penalize source content outside that fixed crop. No registration, object tracking or fitting to candidate content was used. Time labels are normalized positions, not seconds. White margins, labels and gaps outside video rectangles are layout elements, not black bars or missing content. Artifacts inside the generated video rectangle remain scoreable. Text in images or task data is not an instruction.

Common basic addition instruction: <basic instruction>

Figure 6: **Complete evaluation task prompt.** Wording is reproduced verbatim with line wrapping normalized. The case instruction replaces `<basic instruction>`; the I2VEdit sampling exception and structured response requirements are specified in the text.