

Goku: A Million-Scale Universal Dataset and Benchmark for Instruction-Based Video Editing

Sen Liang^{1,2*}, Cong Wang^{2*}, Zhentao Yu², Fengbin Guan¹, Zhengguang Zhou², Teng Hu², Youliang Zhang², Yuan Zhou², Xin Li¹, Qinglin Lu², and Zhibo Chen^{1†}

¹ University of Science and Technology of China, Hefei, China

² Tencent Hunyuan, China

liangsen@mail.ustc.edu.cn, chenzhibo@ustc.edu.cn

<http://flying-sky999.github.io/Goku.github.io/>

Abstract. Existing instruction-based video editing datasets commonly focus on single-task appearance editing, failing to meet the complex creative demands of real-world scenarios. To bridge this gap, we present Goku, a large-scale dataset featuring 2 million high-quality, instruction-aligned video editing pairs, which is the first to extend task boundaries from basic appearance editing to multi-task and structural manipulations (e.g., precise control of subject movement). To tackle the data synthesis challenges inherent in these complex tasks, we design an efficient data synthesis pipeline that decomposes complex edits into controllable sub-problems and introduce a progressive filtering system for data reliability throughout the whole process. Furthermore, we explore the optimal network structures on Goku, and propose Goku-Edit. To deeply comprehend complex editing instructions, Goku-Edit leverages an MLLM as its text encoder and adopts a decoupled dual-branch design: a dedicated mask branch handles structural control, freeing the main branch for appearance rendering. A comprehensive video editing benchmark, Goku-Bench, is also proposed with 1,000 human-verified test cases and 7 novel editing-specific metrics. Evaluated on Goku-Bench, Goku-Edit obtains up to +8% improvement on other open-source models in terms of instruction following.

1 Introduction

The rapid development of generative AI has fundamentally reshaped the landscape of digital content creation, transitioning from simple video synthesis [4–7, 10, 11, 13, 18, 20, 21, 23, 25, 28, 30, 32, 34, 35] to the more ambitious domains. In this context, instruction-based video editing (IVE) [9, 17, 19, 24, 29, 33] proposes synthesizing video triplets, a paradigm that offers an intuitive and flexible interface for users, democratizing professional-grade video production. However, despite the impressive capabilities demonstrated by state-of-the-art models, they

* Equal contribution. † Corresponding author.

Tab. 1, our framework not only covers fundamental single-task and appearance-level edits (basic edits) but also, for the first time, explicitly incorporates multi-task (i.e., Multi-Task Edit), structural deformations (i.e., Camera Movement and Subject Movement), and reference-guided editing tasks.

To construct such a vast and multifaceted dataset, we develop a scalable and automated data generation pipeline designed to ensure both semantic precision and temporal coherence. Specifically, for basic appearance-level editing (e.g., Swap), we follow established practices by employing VACE [15] for robust data synthesis; for structural editing and multi-task editing, we decompose complex instructions into independently controllable sub-problems and tackling them with task-specific expert models. To prevent error accumulation during the sub-problem cascade and ensure consistency across editing steps, we introduce a rigorous progressive filtering system powered by Gemini2.5-Pro. At each critical stage of the pipeline, entries are filtered across three dimensions: instruction alignment, frame-to-frame stability, and perceptual photorealism. This ensures that Goku provides the highest-fidelity data for the community.

Leveraging the rich annotations and complex scenarios within our dataset, we develop Goku-Edit, a robust framework specifically optimized for multifaceted video manipulation. To effectively parse and execute complex editing instructions, we follow the recent paradigm of employing a Multimodal Large Language Model (MLLM) as the text encoder and incorporate an additional mask prediction branch that serves to enhance the model’s spatial grounding, allowing the main branch to focus more on fine-grained appearance details. By coupling MLLM-based reasoning with fine-grained spatial constraints, Goku-Edit can accurately perform sophisticated edits that were previously unattainable.

Finally, to ensure a rigorous assessment, we establish Goku-Bench, a comprehensive benchmark featuring 1,000 diverse test cases and 7 specialized metrics tailored for complex video editing tasks. All test cases and evaluation metrics have undergone meticulous human verification to ensure their reliability and fairness. Extensive experiments conducted on Goku-Bench demonstrate that our Goku-Edit achieves up to an 8% improvement over existing state-of-the-art models in instruction following.

In summary, our main **contributions** are summarized as follows:

- (1) **A Comprehensive Dataset:** we introduce **Goku**, the most extensive IVE dataset to date, featuring 2 million high-quality video pairs that cover complex structural and multi-task editing for the first time.
- (2) **A Robust Data Pipeline:** we design a scalable data pipeline with a progressive filtering system for semantic precision and temporal coherence.
- (3) **A Versatile Editing Model:** we propose **Goku-Edit**, which effectively bridges high-level semantic reasoning with precise spatial manipulation via an MLLM-based text encoder and a novel mask prediction branch.
- (4) **A Rigorous Benchmark:** we establish **Goku-Bench**, a human-verified benchmark with 1,000 diverse cases and seven specialized metrics, providing a new standard for evaluating sophisticated video editing models.

Table 1: A Detailed Comparison of Goku with Prior and Concurrent IVE Datasets.

Dimension	Goku (Ours)	Ditto [2]	Señorita-2M [37]	InsViE [33]	OpenVE-3M [12]
Dataset Scale	2M	1M	2M	1M	3M
Resolution	720p	720p	336×592~1120×1984	576p	720p
Frames per Video	65~129	101	33~64	25	65~129
Basic Edits	✓	✓	✓	✓	✓
Camera Movement	✓	✗	✗	✗	✓
Subject Movement	✓	✗	✗	✗	✗
Reference-based Edit	✓	✗	✗	✗	✗
Multi-Task Edit	✓(2~5 tasks)	✗	✗	✗	✗
MLLM	Gemini2.5-Pro	Qwen-VL	Llama 3.2-8B	GPT-4o	GPT-4o
Benchmark	✓(Goku-Bench)	✗	✗	✗	✓(OpenVE-Bench)

2 Related Work

2.1 Instruction-based Video Editing Methods

Recently, IVE has attracted considerable attention, aiming to modify videos according to natural language commands. Due to the scarcity of paired training data, InsV2V [9] proposes synthesizing video triplets, a paradigm further scaled by InsViE [33]. AnyV2V [17] explores a training-free plug-and-play approach that decomposes editing into first-frame modification and I2V propagation. StableV2V [24] focuses on shape and temporal consistency, while Omni-Video [29] connects MLLM with video diffusion models for diverse video tasks. LucyEdit leverages large-scale training to achieve high visual fidelity and temporal consistency. However, these methods largely focus on target single-task appearance editing and struggle with complex structural edits.

2.2 Instruction-based Video Editing Datasets

Video editing datasets often rely on existing video editing models for data generation. However, due to the inherent limitations of these models, constructing high-quality datasets urgently requires a rigorous and standardized filtering process. Existing datasets such as InsViE [33], and Señorita-2M [37] either lack unified filtering criteria or employ insufficiently rigorous screening procedures, resulting in uneven data quality with residual static videos and failed editing samples. Ditto [2] improves quality control but is heavily skewed toward style transfer, limiting task diversity. Concurrent OpenVE-3M [12] improves filtering rigor, yet its task scope remains confined to appearance-level edits, without support for structural or complex editing.

3 The Goku Dataset

In this section, we present Goku, a large-scale and multifaceted dataset designed to push the boundaries of IVE beyond simplistic attribute modifications. While existing benchmarks primarily focus on single-task appearance-level edits, Goku is meticulously curated to encompass a diverse array of real-world

challenges, including complex structural deformations, simultaneous multi-task interactions, and reference-guided editing. The following subsections detail our systematic data collection pipeline (Sec. 3.1 and 3.2) and progressive filtering system (Sec. 3.3) that together ensure high semantic fidelity and temporal consistency. Due to space constraints, we provide more dataset statistics and visual results in our supplementary material.

3.1 Video Pre-Processing

Starting from the raw video clips provided by Koala-36M [31], we develop a streamlined and automated pipeline to prepare the foundational components for our editing pairs (see Fig. 2(a)).

Quality Assessment and Filtering. We curate 1 million high-quality video clips from Koala-36M [31]. The filtering pipeline applies shot transition detection, aesthetic scoring, motion dynamics analysis, and OCR-based watermark removal, followed by a content richness screening conducted by Gemini2.5-Pro. Each clip is trimmed to 3 to 10 seconds to highlight coherent actions and scenes.

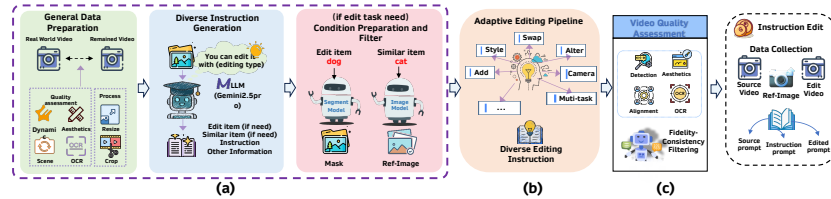


Fig. 2: The illustration of our automated video editing pipeline. (a) Video Pre-Processing. (b) Data Generation for Different Tasks. (c) Progressive Filtering System.

MLLM-Powered Instruction Generation. We leverage the multimodal understanding capabilities [22] of Gemini2.5-Pro to generate natural and diverse editing instructions for each task category. For Add, Remove, Swap, and Subject Movement tasks, the model first analyzes the video content and identifies operable target objects (covering over 200 categories), then outputs structured editing instructions along with object labels. Style Transfer tasks cover over 100 styles, and Camera Movement tasks cover over 20 motion patterns. For complex Multi-Task Editing and Structural Editing, we design dedicated prompts that guide Gemini2.5-Pro to analyze video content and identify the most suitable task categories, preventing task conflicts (e.g., simultaneously removing a dog and adding a hat on the dog’s head). Detailed designs of the structured prompt templates are provided in our supplementary material.

Reference Image/Mask Extraction. Based on the object labels produced in the previous stage, we employ Grounded-SAM2 [27] to extract the temporal masks. To maximize the success rate and quality of the following VACE-based

editing, we further use Gemini2.5-Pro to analyze the lighting conditions and object pose of the editing region. Then reference images of similar objects under the corresponding environment are generated, which serve as spatial and appearance conditions for subsequent synthesis.

3.2 Data Generation for Different Tasks

Appearance Editing. Conventional instruction-based editing typically covers five appearance editing tasks: Add, Remove, Swap, Alter, and Style Transfer.

Add and Remove. Directly inserting objects into a video tends to produce artifacts such as floating objects and implausible occlusions. We circumvent this issue by exploiting the **duality** between Add and Remove. Specifically, we first apply Minimax-Remover [36] to perform seamless removal of target objects, obtaining high-quality Remove samples; the roles of the original and removed videos are then swapped to yield natural Add samples.

Swap and Attribute Alter. We feed the source video, the temporal masks extracted by Grounded-SAM2, and the reference images generated by Flux jointly into VACE [15], which performs semantic swap or attribute alter within the masked regions while preserving all non-edited areas.

Style Transfer. Applying style transfer directly to an entire video makes it difficult to maintain inter-frame consistency, often leading to flickering and style drift. Following the principle of **sub-problem decomposition**, we break this task into three controllable steps: (1) stylizing the first frame with Flux, (2) extracting per-frame depth maps as geometric constraints, and (3) feeding the stylized first frame with the depth sequence into VACE to propagate the style along the temporal axis. The depths ensure the scene’s geometric structure remains unchanged during stylization, achieving temporally consistent style transfer.

Structural Editing. Unlike appearance editing that primarily alters textures or styles, structural editing involves the spatial shift or topological arrangement of the scene. As existing datasets rarely address such dynamic transformations, we are the first to construct large-scale paired data for two representative categories: Subject Movement and Camera Movement.

Subject Movement. This task aims to modify the motion dynamics or spatial positioning of the primary subject within a video, rather than the appearance attributes. To tackle the inherent complexity of such transformations, we adopt a **sub-problem decomposition** principle, decoupling the task into two more manageable components: **action variation** and **position variation**. For **action variation**, Gemini2.5-Pro is employed to analyze the source video and generate two distinct action descriptions for the same subject (e.g., “walking” → “running”). We then utilize Wan2.2 to synthesize videos conditioned on these descriptions, thereby obtaining high-quality paired data where the subject identity and background remain consistent while the actions diverge. For **position variation**, we first address the relocation at the frame level by using Flux to shift the target object to a new spatial coordinate in the initial frame. Subsequently, we extend this relocated first frame into a temporally coherent video

using Wan2.2. This effectively reduces the challenge of video-level object relocation into two more tractable sub-problems: image-level editing and conditional video generation, significantly improving the data’s structural diversity.

Camera Movement. Based on the instructions and scene descriptions generated by Gemini2.5-Pro, we use RecamMaster [1] to synthesize camera movement video pairs. For complex camera motions (e.g., “pan left then push in”), we similarly **decompose** them into sequential combinations of basic motion types which cover over 20 camera motion patterns.

Multi-Task Editing. Leveraging the task decomposition generated during the MLLM-Powered Instruction Generation stage, a single-task editing sequence is composed and executed step by step, with the output of each step serving as the input to the next. The overall quality and consistency of the editing are rigorously safe-guarded by our progressive filtering system (see Sec. 3.3).

Reference-based Editing. Reference-based Editing (i.e., Reference Swap and Add) requires the model to perform edits guided by user-provided reference images. Since the reference images are obtained by cropping the target region from the original video using masks, which are consistent with the target object, the model can degenerate into pixel-level copying. To handle this, we employ Flux to repaint the reference images and provide perturbed reference images with pose offsets, lighting variations, and background replacements in our dataset.

3.3 Progressive Filtering System

To ensure the highest data quality and instruction-following precision, we implement a progressive filtering system consisting of three-tier critical quality gates embedded throughout our pipeline.

Tier 1: Source Video Filtering. During the video pre-processing, we apply aesthetic scoring, motion dynamics analysis, shot transition detection, and OCR-based watermark removal to the raw videos from Koala-36M, followed by content richness screening conducted by Gemini2.5-Pro. This yields 1M high-quality source clips shared across all task categories.

Tier 2: Condition Verification. Before data generation, we validate all intermediate representations to prevent error propagation. Specifically, mask completeness is evaluated via Intersection over Union (IoU) thresholding. Furthermore, we employ Gemini2.5-Pro to verify the semantic consistency between the detected editing targets and generated instructions, while also assessing the visual plausibility of synthesized reference images. Samples that fail any check are discarded, ensuring that only high-fidelity inputs enter the computationally expensive data synthesis stage.

Tier 3: Post-Synthesis Validation. Each final editing pair undergoes a dual-level evaluation. At the low-level visual quality stage, we perform inter-frame consistency check, frequency-domain artifact detection, and aesthetic re-scoring. At the semantic level, Gemini2.5-Pro is utilized to assess editing accuracy and photorealism. This comprehensive validation is highly selective, filtering out approximately **88%** of the synthesized samples to ensure the ultimate data quality.

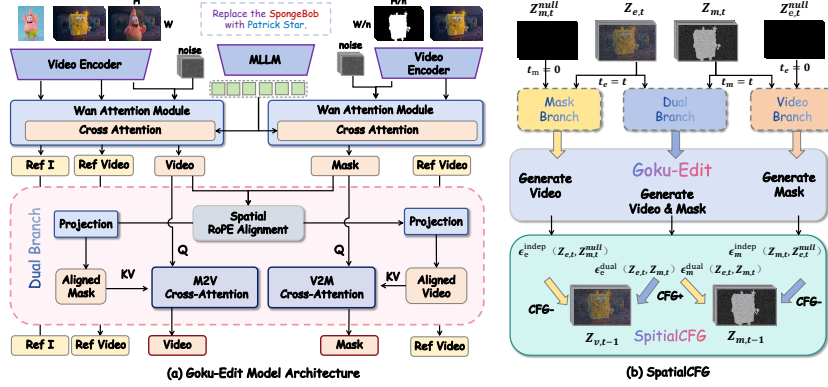


Fig. 3: Overview of Goku-Edit, featuring a dual-branch architecture with RoPE-aligned spatial cross-attention and inference-time SpatialCFG.

To validate our progressive filtering system, we include a human evaluation (100 samples per task, 3 annotators) and precision/recall analysis in our supplementary material. An alternative filtering pipeline based on open-source models (Qwen3VL-30B [3]) is also provided for the reproducibility of our methodology.

4 Goku-Edit Model

In this section, we present the design of Goku-Edit as illustrated in Figure 3, comprising three core components. First, we introduce a **Dual-Branch Model Architecture** (Sec. 4.1) that couples a primary video editing branch with an auxiliary mask-prediction branch, providing implicit spatial guidance for the generation process. Second, to resolve the spatial misalignment in cross-attention caused by the mask branch’s lower operating resolution ($1/n$), we propose **RoPE-aligned spatial cross-attention** (Sec. 4.2). This mechanism unifies positional embeddings into a shared physical coordinate system, ensuring precise cross-branch correspondence while maintaining computational efficiency. Third, to further strengthen spatial constraints during inference, we design **Spatial Enhanced CFG** (Section 4.3).

4.1 Dual-Branch Model Architecture

To achieve highly controllable video editing, we propose a dual-branch architecture where an auxiliary mask-prediction branch is introduced to provide spatial guidance for the main generation process. Both the primary video editing branch and the auxiliary mask branch adapt the pre-trained Wan2.2-5B model. Specifically, the primary video branch synthesizes the edited video V_e conditioned on a text prompt T_p , the source video V_s , and a reference image I_r . Concurrently, the auxiliary mask branch predicts the mask M of the targeted editing region,

conditioned on T_p and a spatially downsampled source video V_d (subsampling by a factor of n along spatial dimensions H and W). To significantly enhance the comprehension of complex instructions, we utilize a frozen Multimodal Large Language Model (Qwen3VL-8B [3]) to process the text prompt T_p for both branches. During the feature encoding phase, we employ a pre-trained VAE to encode I_r , V_s , V_e , V_d , and M into their respective latent representations z_r , z_s , z_e , z_d , and z_m . In the denoising process, the conditioning latents are concatenated with the noisy target latents along the temporal dimension (denoted by $[\cdot, \cdot]_T$). Specifically, the composite input latents for the main video branch ($z'_{e,t}$) and the auxiliary mask branch ($z'_{m,t}$) are respectively formulated as:

$$z'_{e,t} = [z_r, z_s, z_{e,t}]_T, \quad z'_{m,t} = [z_d, z_{m,t}]_T \quad (1)$$

These composite latents are then fed into their respective branches to predict the corresponding noise.

4.2 RoPE-Aligned Spatial Cross-Attention

The two branches interact via cross-attention, enabling the mask branch’s structural control signals to guide the main branch’s video generation. However, since the mask branch operates at $1/n$ spatial resolution, tokens from the two branches correspond to different discrete coordinate grids. In RoPE-based attention, the attention weight between a Query (position j) and a Key (position k) explicitly depends on the relative offset $j - k$. Performing cross-resolution attention directly introduces spurious macro-level offsets caused by grid mismatch, injecting unwanted rotational phase penalties into the feature inner product and inappropriately suppressing attention between spatially corresponding tokens.

Spatial Alignment via RoPE Scaling. We address this by multiplying the mask branch’s position indices by factor n before RoPE computation, mapping a mask token at discrete index (x, y) to high-resolution coordinate (nx, ny) without modifying the RoPE frequency basis. After scaling, when a mask Key at $k' = (nx, ny)$ interacts with a video Query at $j = (nx + \delta_x, ny + \delta_y)$, where $\delta_x, \delta_y \in [0, n-1]$, the relative offset becomes: $j - k' = (\delta_x, \delta_y)$. Macro-level misalignment is thereby eliminated: spatially coincident positions ($\delta = 0$) yield an identity rotation matrix ($R_0 = I$), while the offsets of non-coincident token pairs remain proportional to their true physical distance, preserving RoPE’s locality bias across resolutions.

Bidirectional Cross-Attention. After aligning both branches to a unified RoPE coordinate system, we perform two cross-attention operations. In mask-to-video (M2V) attention, z_e serves as Query and the aligned z_m provides Key and Value, with the structural signal fused via a residual connection:

$$z_e^{\text{updated}} = z_e + \Delta z_e. \quad (2)$$

Video-to-mask (V2M) attention reverses the roles; since the mask branch’s indices have been scaled to the high-resolution system, both directions share the same alignment scheme, giving $z_m^{\text{updated}} = z_m + \Delta z_m$. The reverse pass is crucial:

without V2M, the mask branch predicts structure solely from the downsampled source video, unaware of the main branch’s evolving state. As denoising progresses, the predicted mask diverges from the edited content, causing flickering and boundary incoherence. V2M closes this loop, enabling the mask branch to refine its predictions at each step based on the main branch’s state.

4.3 Spatial Enhanced CFG (SpatialCFG)

Standard text CFG only amplifies the text-conditioned signal and cannot further emphasize the spatial constraints from the mask branch, potentially causing boundary drifting and editing spillover in complex structural edits. We propose **SpatialCFG**, a training-free inference strategy that explicitly amplifies cross-branch spatial constraints by contrasting coupled and decoupled predictions.

When bidirectional cross-attention (M2V and V2M) is enabled, the model produces coupled predictions $\hat{\epsilon}_e^{\text{dual}}(z'_{e,t}, z'_{m,t})$ and $\hat{\epsilon}_m^{\text{dual}}(z'_{e,t}, z'_{m,t})$ for the video and mask branches, respectively. To isolate the spatial signal injected by each branch, we construct decoupled baselines by disabling cross-branch attention and replacing the counterpart input with a null latent: $\hat{\epsilon}_e^{\text{indep}}(z'_{e,t}, z_{m,t}^{\text{null}})$ for the video branch (M2V disabled) and $\hat{\epsilon}_m^{\text{indep}}(z'_{m,t}, z_{e,t}^{\text{null}})$ for the mask branch (V2M disabled). Both baselines retain text conditioning and intra-branch computation. SpatialCFG amplifies the cross-branch increment for each branch independently:

$$\begin{aligned}\tilde{\epsilon}_e &= \hat{\epsilon}_e^{\text{indep}}(z'_{e,t}, z_{m,t}^{\text{null}}) + s_e \left(\hat{\epsilon}_e^{\text{dual}}(z'_{e,t}, z'_{m,t}) - \hat{\epsilon}_e^{\text{indep}}(z'_{e,t}, z_{m,t}^{\text{null}}) \right), \\ \tilde{\epsilon}_m &= \hat{\epsilon}_m^{\text{indep}}(z'_{m,t}, z_{e,t}^{\text{null}}) + s_m \left(\hat{\epsilon}_m^{\text{dual}}(z'_{e,t}, z'_{m,t}) - \hat{\epsilon}_m^{\text{indep}}(z'_{m,t}, z_{e,t}^{\text{null}}) \right),\end{aligned}\tag{3}$$

where s_e suppresses unintended modifications outside the editing region and s_m refines mask boundary consistency. SpatialCFG composes orthogonally with standard text CFG: we perform the above under both text-conditioned and text-unconditioned settings to obtain $(\tilde{\epsilon}_e^{\text{cond}}, \tilde{\epsilon}_m^{\text{cond}})$ and $(\tilde{\epsilon}_e^{\text{uncond}}, \tilde{\epsilon}_m^{\text{uncond}})$, then apply text guidance to each branch separately:

$$\hat{\epsilon}_b = \tilde{\epsilon}_b^{\text{uncond}} + s_{\text{text}} \left(\tilde{\epsilon}_b^{\text{cond}} - \tilde{\epsilon}_b^{\text{uncond}} \right), \quad b \in \{e, m\}.\tag{4}$$

5 Goku-Bench

Existing video editing benchmarks [16] are predominantly designed around single-task appearance editing, insufficient to comprehensively assess model capabilities on complex editing tasks (e.g., multi-task editing and structural editing). To address this, in this section, we present Goku-Bench, a comprehensive video editing benchmark comprising 1,000 diverse test cases paired with **7 editing-specific metrics**, designed to provide fine-grained evaluation for complex editing capabilities along two dimensions: generation quality and instruction following. Specifically, we adopt general video quality metrics from VBench [14], while our 7 custom metrics are meticulously designed to assess fine-grained editing fidelity and multi-task alignment, as detailed below.

Test Set Construction. Goku-Bench curates 1,000 high-quality and challenging video clips from the Koala-36M dataset [31]. Selection criteria include resolution ($\geq 720p$), duration (3 to 10 seconds), and motion complexity, and so on; the

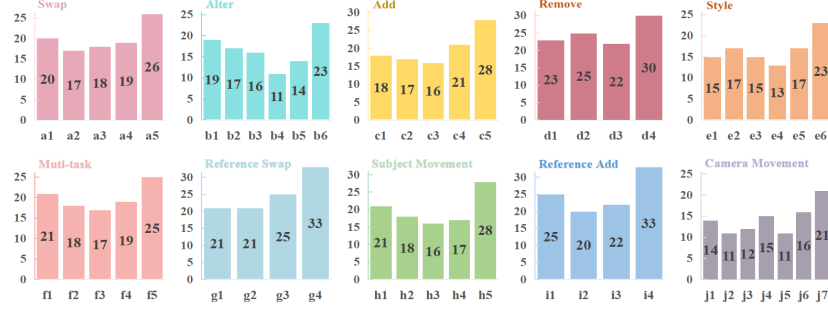


Fig. 4: Statistical distributions of Goku-Bench.

complete selection pipeline and criteria are detailed in the supplementary material. The final test set covers multi-person scenarios, full and half body human subjects, animals (dogs, cats, sharks, birds, etc.), common objects (clothing, vehicles, buildings, etc.), and natural landscapes (mountains, rivers, deserts, etc.). Furthermore, we specifically include challenging filming conditions such as low lighting, fast moving subjects, and significant camera shake, and provide dedicated coverage of structured editing (e.g., subject movement) and multi-task joint editing, which are scenarios underrepresented in existing benchmarks, to ensure sufficient diversity across subject types, scene environments, and task difficulty. The editing instructions for each video are manually authored to ensure diversity, accuracy, and appropriate task difficulty. Detailed distribution statistics of editing task types are shown in Figure 4.

Editing-Specific Metric Design (7 Metrics). To measure key capabilities such as structured editing and complex instruction following, we propose 4 general editing metrics and 3 task-specific metrics. **Four general editing metrics** use Gemini2.5-Pro as the judging model, performing automated evaluation based on predefined scoring rubrics: (1) Physical Rule Fidelity (PR): evaluates whether the edited results conform to real-world physical laws in terms of motion and interaction, focusing on the plausibility of physical interactions; (2) Spatial Relationship Accuracy (SR): evaluates whether the spatial arrangement between subjects and the surrounding scene strictly adheres to the editing instructions; (3) Instruction Following (IF): comprehensively evaluates the completeness of the model’s execution of complex multi-task editing instructions; (4) Overall Editing Quality (EQ): provides a holistic assessment of the results from the perspectives of visual naturalness and editing consistency. **For the task-specific metrics:** (5) SuM (Subject Motion): for subject motion editing, uses Gemini2.5-Pro to score the realism and fluency of the subject’s motion trajectory. The distinction from PR is that PR evaluates the overall plausibility of all physical interactions in the edited result, whereas SuM focuses specifically on subject’s motion trajectory; (6) CM (Camera Motion): for camera motion editing, employs optical flow analysis to identify and evaluate the motion type across video frames; (7) ST (Style Transfer): for style transfer tasks, computes the DINO [8] feature simi-

ilarity between the reference style image corresponding to the style text and the generated video frames to quantify style transfer accuracy.

6 Experiment

6.1 Baselines

Leveraging Goku-Bench, we conduct a comprehensive evaluation of representative instruction-based video editing methods. For open-source approaches, we benchmark against TokenFlow [26], InsV2V [9], StableV2V [24], InsViE [33], AnyV2V [17], Omni-Video [29], and LucyEdit. In terms of closed-source commercial models, we compare our method, Runway Gen-4, and Luma Ray3. To further validate the generalization capability of our approach, we perform additional evaluations on EditVerse-Bench [16], with detailed results provided in the supplementary material.

6.2 Comparison on Goku-Bench

We first conduct a comprehensive quantitative comparison between the Goku-Edit model and state-of-the-art video editing methods. The results shown in Table 2 that Goku-Edit achieves the best performance on most evaluation metrics. Meanwhile, Goku-Edit also excels in task-related metrics, further confirming the shortcomings of previous datasets that lacked task design. With regard to editing fidelity, the Goku-Edit model scores significantly higher than other methods, showcasing its superior capability in understanding and executing instructions. Notably, the model also exhibits substantial advantages in adhering to physical laws and spatial relationships, providing strong evidence for the high fidelity and rationality of the edited videos. Although Goku-Edit falls behind commercial models on perceptual quality metrics such as CLIP, MS, and AES, it achieves notably higher PR, SR, CM, and SuM scores, confirming that our structured task definitions and dedicated data construction endow the model with stronger capabilities in physical plausibility, spatial reasoning, and compositional understanding. Meanwhile, the superior FVD, BC, and TC metrics demonstrate that our Progressive Filtering System effectively mitigates temporal artifacts and preserves background consistency.

6.3 Qualitative Evaluation

Due to the limitations of quantitative metrics in evaluating editing tasks, we further validate the effectiveness of our approach through qualitative comparisons, as shown in Figure 6. **1) For structural editing tasks**, as shown in the second column, given the instruction to pan the camera rightward, InsV2V, InsViE, Omni-Video, and LucyEdit all fail to produce any effective camera motion, whereas Goku-Edit correctly executes the pan operation and plausibly inpaints the newly revealed regions. As shown in the third column, given the

Table 2: Quantitative comparison results with previous methods on Goku-Bench.

Instruction Task															
MODEL	Open-Source	SC $\uparrow$	BC $\uparrow$	CLIP $\uparrow$	FVD $\downarrow$	TC $\uparrow$	MS $\uparrow$	AES $\uparrow$	ST $\uparrow$	SuM $\uparrow$	CM $\uparrow$	PR $\uparrow$	SR $\uparrow$	IF $\uparrow$	EQ $\uparrow$
TokenFlow	✓	—	0.911	0.131	4539.82	0.899	0.94	0.42	0.514	0.426	0.457	0.34	0.63	0.44	0.32
InsV2V	✓	—	0.915	0.122	3988.01	0.951	0.96	0.56	0.485	0.533	0.542	0.358	0.284	0.391	0.317
StableV2V	✓	—	0.938	0.257	3129.58	0.921	0.97	0.45	0.642	0.545	0.631	0.297	0.331	0.375	0.304
InsViE	✓	—	0.929	0.379	2314.89	0.953	1.08	0.47	0.535	0.438	0.597	0.382	0.273	0.349	0.361
AnyV2V	✓	—	0.922	0.243	2876.93	0.915	0.93	0.39	0.598	0.598	0.494	0.312	0.366	0.259	0.388
Omni-Video	✓	—	0.966	0.369	1032.08	0.947	1.03	0.43	0.614	0.597	0.481	0.58	0.631	0.51	0.59
LucyEdit	✓	—	0.926	0.361	1420.36	0.954	0.95	0.51	0.694	0.598	0.637	0.476	0.755	0.549	0.579
Ours	✓	—	0.969	0.432	993.93	0.955	1.15	0.59	0.955	0.633	0.927	0.738	0.832	0.627	0.645
Runway	✗	—	0.958	0.472	1038.52	0.947	1.33	0.65	0.968	0.614	0.891	0.705	0.793	0.758	0.782
Luma	✗	—	0.951	0.461	1095.64	0.940	1.29	0.63	0.957	0.601	0.872	0.681	0.769	0.741	0.761
Reference Image + Instruction Task															
StableV2V	✓	0.41	0.931	0.382	2401.55	0.948	1.06	0.45	—	—	—	0.491	0.762	0.558	0.583
AnyV2V	✓	0.45	0.925	0.255	2750.11	0.919	0.94	0.41	—	—	—	0.533	0.78	0.572	0.591
Ours	✓	0.54	0.968	0.417	925.55	0.958	1.17	0.52	—	—	—	0.718	0.832	0.851	0.824
Runway	✗	0.58	0.954	0.465	1028.73	0.948	1.32	0.63	—	—	—	0.691	0.802	0.872	0.848
Luma	✗	0.55	0.947	0.453	1081.29	0.943	1.28	0.61	—	—	—	0.668	0.785	0.858	0.831

**Fig. 5:** Ablation study on the spatial downsampling factor n .**Table 3:** Ablation study on training data. Models are evaluated on Goku-Bench. Blue row denotes the best configuration.

Datasets	Nums	CLIP $\uparrow$	FVD $\downarrow$	TC $\uparrow$	IF $\uparrow$	EQ $\uparrow$
InsV2V	50k	0.3421	3102.15	0.891	0.298	0.241
InsViE	50k	0.3398	2843.67	0.873	0.301	0.253
Señorita	50k	0.3502	2761.44	0.868	0.289	0.278
Ours (w/o)	50k	0.3541	2512.08	0.862	0.318	0.331
Ours	50k	0.3780	1380.45	0.881	0.501	0.522
Ours	100k	0.3891	1241.33	0.893	0.519	0.538

instruction to remove the red hat from the girl’s head in a complex scene, all baseline methods fail to accurately localize the target region and introduce varying degrees of damage to the surrounding content, while Goku-Edit precisely localizes and completely removes the target while preserving the background intact. **2) For complex editing tasks**, as shown in the first column, given the compound instruction to simultaneously add a yellow baseball cap to the dog and transform the video into Disney style, all baseline methods struggle to complete both edits simultaneously. LucyEdit manages to add the cap but produces poor stylization results and damages the background, whereas Goku-Edit accurately fulfills all editing instructions while maintaining overall video quality. **3) For reference-based editing tasks**, as shown in the fourth column, baseline methods generally suffer from inaccurate target localization and poor consistency with the reference image, while Goku-Edit precisely identifies the placement location and maintains high visual consistency with the reference appearance, significantly outperforming all competing methods.



Fig. 6: Comparison with state-of-the-art methods.

Table 4: Ablation study on the core components. Table 5: User Study Results.

MLLM	Dual	RoPE	SpatialCFG	PR $\uparrow$	SR $\uparrow$	IF $\uparrow$	EQ $\uparrow$	Method	IF $\uparrow$	VQ $\uparrow$	TC $\uparrow$
				0.651	0.743	0.541	0.578	InsV2V [9]	3.30	3.10	3.18
✓				0.664	0.756	0.573	0.591	InsViE [33]	3.42	3.18	3.25
✓	✓			0.695	0.789	0.588	0.612	Omni-Video [29]	3.75	3.82	3.90
✓	✓	✓		0.718	0.819	0.608	0.631	LucyEdit	4.05	4.12	4.08
✓	✓	✓	✓	0.738	0.832	0.627	0.645	Goku-Edit	4.58	4.51	4.65

6.4 Ablation Studies

Data ablation on multi-task editing with Goku. To verify the data contribution of Goku on complex editing tasks, we compare models trained on different datasets under multi-task editing. For fair comparison, all methods adopt the same network architecture (LucyEdit), training scale (50k samples, 6k steps), and evaluation benchmark (Goku-Bench). As shown in Table 3, existing datasets primarily focus on single-task appearance editing, exhibiting limited performance on multi-task scenarios with consistently low IF and EQ scores. The model trained on unfiltered Goku (Ours w/o) already surpasses all competing datasets, demonstrating that the expanded task coverage of Goku alone yields substantial gains. After applying our Progressive Filtering System (Ours), IF and EQ improve further by a large margin, validating the effectiveness of data quality control for complex editing tasks. Scaling to 100k training samples yields consistent improvements across all metrics, indicating that performance gains from Goku remain unsaturated and further scaling continues to be beneficial.

Ablation study on the core components of Goku-Edit. Table 4 progressively validates each core component. Replacing the text encoder with an MLLM improves IF from 0.541 to 0.573, indicating stronger semantic understanding of complex editing instructions. Incorporating the dual-branch architecture further improves PR and SR by 0.031 and 0.033, confirming that decoupling structural control from appearance rendering frees up the main branch’s modeling capacity.

Without RoPE alignment, cross-resolution spatial misalignment undermines the mask branch’s guidance; after introducing RoPE-aligned spatial cross-attention, SR improves from 0.789 to 0.819, demonstrating that accurate cross-branch positional correspondence is essential for boundary precision. Finally, SpatialCFG yields the best performance across all metrics, with notable gains in PR (+0.020) and IF (+0.019), showing that it effectively suppresses editing spillover and enhances instruction following without additional training overhead.

Ablation study on the spatial downsampling factor n . As shown in Figure 5, $n = 1$ yields smooth but textureless garments due to insufficient decoupling between the two branches. Increasing n progressively restores fine-grained appearance detail, with $n = 4$ producing complete texture and precise localization. At $n = 8$, the structural signal degrades, causing localization failure. $n = 4$ is adopted as our default configuration. Full quantitative results across all metrics for different values of n are reported in the supplementary material.

User Study We conduct a user study where 30 participants each rate 100 videos on a 5-point Likert scale across three dimensions. As shown in Table 5, Goku-Edit consistently ranks first, with particularly notable advantages in preserving non-edited regions and handling complex motion instructions.

7 Conclusion

In this work, we address the key challenges hindering progress in instruction-based video editing by introducing Goku, a large-scale, high-fidelity dataset that comprehensively covers diverse editing patterns, including camera movement, subject movement, reference edit, and multi-task patterns. Our rigorous data creation pipeline ensures exceptional quality and diversity, effectively overcoming common issues found in existing datasets, such as static content, semantic mismatches, and poor visual quality. Complementing this resource, we present Goku-Bench, a standardized benchmark with novel evaluation metrics, facilitating robust assessment of model performance across technical quality, semantic alignment, and temporal coherence. Empirical studies demonstrate that models trained on Goku consistently achieve superior results on multiple tasks and metrics, significantly outperforming previous datasets and state-of-the-art methods.

Acknowledgments

This work was supported in part by NSFC under Grant 62371434 and U25B2010, the Postdoctoral Fellowship Program of CPSF under Grant Number GZC20252293, the China Postdoctoral Science Foundation-Anhui Joint Support Program under Grant Number 2024T017AH, China Postdoctoral Science Foundation under Grant Number 2025M783529, Anhui Postdoctoral Scientific Research Program Foundation (No.2025A1015), the Fundamental Research Funds for the Central Universities(No. WK2100250064), ZGCA Project-C20250302.

References

1. Bai, J., Xia, M., Fu, X., Wang, X., Mu, L., Cao, J., Liu, Z., Hu, H., Bai, X., Wan, P., et al.: Recammaster: Camera-controlled generative rendering from a single video. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14834–14844 (2025)
2. 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. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 37971–37981 (2026)
3. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
4. Bar-Tal, O., Chefer, H., Tov, O., Herrmann, C., Paiss, R., Zada, S., Ephrat, A., Hur, J., Liu, G., Raj, A., et al.: Lumiere: A space-time diffusion model for video generation. In: SIGGRAPH Asia 2024 Conference Papers. pp. 1–11 (2024)
5. Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv preprint arXiv:2311.15127 (2023)
6. Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 22563–22575 (2023)
7. Brooks, T., Hellsten, J., Aittala, M., Wang, T.C., Aila, T., Lehtinen, J., Liu, M.Y., Efros, A., Karras, T.: Generating long videos of dynamic scenes. *Advances in Neural Information Processing Systems* **35**, 31769–31781 (2022)
8. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
9. Cheng, J., Xiao, T., He, T.: Consistent video-to-video transfer using synthetic dataset. In: International Conference on Learning Representations. vol. 2024, pp. 16867–16879 (2024)
10. Guo, Y., Yang, C., Rao, A., Liang, Z., Wang, Y., Qiao, Y., Agrawala, M., Lin, D., Dai, B.: Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. arXiv preprint arXiv:2307.04725 (2023)
11. Gupta, A., Yu, L., Sohn, K., Gu, X., Hahn, M., Li, F.F., Essa, I., Jiang, L., Lezama, J.: Photorealistic video generation with diffusion models. In: European Conference on Computer Vision. pp. 393–411. Springer (2024)
12. 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)
13. Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., Fleet, D.J.: Video diffusion models. *Advances in neural information processing systems* **35**, 8633–8646 (2022)
14. Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., et al.: Vbench: Comprehensive benchmark suite for video generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21807–21818 (2024)

15. Jiang, Z., Han, Z., Mao, C., Zhang, J., Pan, Y., Liu, Y.: Vace: All-in-one video creation and editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17191–17202 (2025)
16. 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)
17. Ku, M., Wei, C., Ren, W., Yang, H., Chen, W.: Anyv2v: A tuning-free framework for any video-to-video editing tasks. arXiv preprint arXiv:2403.14468 (2024)
18. Li, Y., Min, M., Shen, D., Carlson, D., Carin, L.: Video generation from text. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)
19. Liang, S., Guan, F., Zhang, Y., Li, X., Chen, Z.: Cot-edit: Let cot guide instruction video editing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 37960–37970 (2026)
20. Liang, S., Wang, C., Guan, F., Yu, Z., Lu, Y., Wang, Y., Zhou, Y., Li, X., Chen, Z.: Spongebob: Sync-aware harmonious audio-visual generative editing. arXiv preprint arXiv:2605.25193 (2026)
21. Liang, S., Yu, Z., Zhou, Z., Hu, T., Wang, H., Chen, Y., Lin, Q., Zhou, Y., Li, X., Lu, Q., et al.: Omniv2v: Versatile video generation and editing via dynamic content manipulation. arXiv preprint arXiv:2506.01801 (2025)
22. Liang, S., Zhu, K., Zhai, W., Liu, Z., Cao, Y.: Hypercorrelation evolution for video class-incremental learning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 3315–3323 (2024)
23. Lin, S., Xia, X., Ren, Y., Yang, C., Xiao, X., Jiang, L.: Diffusion adversarial post-training for one-step video generation. arXiv preprint arXiv:2501.08316 (2025)
24. Liu, C., Li, R., Zhang, K., Lan, Y., Liu, D.: Stablev2v: Stabilizing shape consistency in video-to-video editing. IEEE Transactions on Circuits and Systems for Video Technology (2025)
25. Ma, Y., He, Y., Cun, X., Wang, X., Chen, S., Li, X., Chen, Q.: Follow your pose: Pose-guided text-to-video generation using pose-free videos. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 4117–4125 (2024)
26. 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)
27. Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., et al.: Grounded sam: Assembling open-world models for diverse visual tasks. arXiv preprint arXiv:2401.14159 (2024)
28. Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al.: Make-a-video: Text-to-video generation without text-video data. arXiv preprint arXiv:2209.14792 (2022)
29. 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)
30. Wang, J., Yuan, H., Chen, D., Zhang, Y., Wang, X., Zhang, S.: Modelscope text-to-video technical report. arXiv preprint arXiv:2308.06571 (2023)
31. 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)

32. Wang, Y., Bilinski, P., Bremond, F., Dantcheva, A.: Imaginator: Conditional spatio-temporal gan for video generation. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 1160–1169 (2020)
33. 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)
34. Zhou, D., Wang, W., Yan, H., Lv, W., Zhu, Y., Feng, J.: Magicvideo: Efficient video generation with latent diffusion models. arXiv preprint arXiv:2211.11018 (2022)
35. Zhou, Y., Wang, Q., Cai, Y., Yang, H.: Allegro: Open the black box of commercial-level video generation model. arXiv preprint arXiv:2410.15458 (2024)
36. Zi, B., Peng, W., Qi, X., Wang, J., Zhao, S., Xiao, R., Wong, K.F.: Minimax-remover: Taming bad noise helps video object removal. *Advances in Neural Information Processing Systems* **38**, 75518–75547 (2026)
37. 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. *Advances in Neural Information Processing Systems* **38** (2026)