DynVFX: Augmenting Real Videos with Dynamic Content

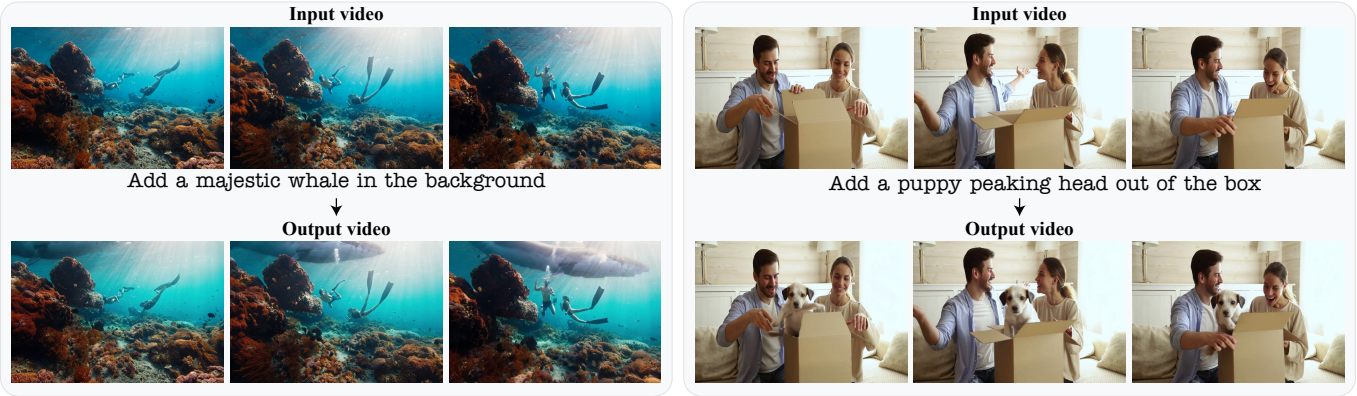

Fig. 1. DynVFX augments real-world videos with new dynamic content described via simple user-provided text instruction.

We present a method for augmenting real-world videos with newly generated dynamic content. Given an input video and a simple user-provided text instruction describing the desired content, our method synthesizes dynamic objects or complex scene effects that naturally interact with the existing scene over time. The position, appearance, and motion of the new content are seamlessly integrated into the original footage while accounting for camera motion, occlusions, and interactions with other dynamic objects in the scene, resulting in a cohesive and realistic output video. We achieve this via a zero-shot, training-free framework that harnesses a pre-trained text-to-video diffusion transformer to synthesize the new content and a pre-trained Vision Language Model to envision the augmented scene in detail. Specifically, we introduce a novel inference-based method that manipulates features within the attention mechanism, enabling accurate localization and seamless integration of the new content while preserving the integrity of the original scene. Our method is fully automated, requiring only a simple user instruction. We demonstrate its effectiveness on a wide range of edits applied to real-world videos, encompassing diverse objects and scenarios involving both camera and object motion[1]. Project page: https://dynvfx.github.io/

Additional Key Words and Phrases: Text-to-video editing, diffusion models

1 Introduction

Incorporating computer-generated imagery (CGI) into real-world footage has been a transformative capability in film production, enabling the creation of visual effects that would be difficult or impossible to achieve otherwise. For instance, the seamless integration of CGI characters, such as *Gollum* in *The Lord of the Rings* or *T-Rex* in *Jurassic Park*, has empowered filmmakers to blend fantastical elements with real-world environments, resulting in immersive storytelling. Inspired by these capabilities, we pose a new creative task: augmenting real-world videos with newly generated dynamic content. Specifically, given an input video and a text prompt describing the desired edit, our goal is to synthesize new dynamic objects or complex scene effects, which naturally interact with the existing scene across time (Fig. 1).

Our task poses several new fundamental challenges. First, the generation must be *content-aware*, such that the position, appearance, and motion of the synthesized dynamic content integrate naturally with the original scene. This entails synthesizing objects that respect occlusions, maintain appropriate relative size and perspective with respect to the camera position, and realistically interact with other dynamic objects. All of this must be achieved while maintaining the integrity of the original video, ensuring that new content enhances the scene without compromising its authenticity.

Our method leverages a pre-trained text-to-video model without any fine-tuning or additional training. Specifically, given an input video along with a short user instruction describing the new content (e.g., "add a massive whale"), our method produces an edited video where the new content is seamlessly integrated into the original video. We achieve it by estimating the residual to the latent representation of the original video. To envision the edited scene and to identify prominent foreground objects and visual elements, we leverage a vision-language model that translates the user's instructions into detailed prompts for the text-to-video diffusion model.

As our task requires careful placement of the new content, we propose to steer the localization of the edit through *Anchor Extended Attention* - incorporate a specific set of keys/values extracted from the original video as additional context to the model. Additionally, to improve the edit harmonization with the original scene, we propose to iteratively update the estimated edit residual.

It is worth noting that our method utilizes a publicly available text-to-video model, which exhibits a significant gap in video generation quality compared to recent state-of-the-art video models. Nevertheless, we observe that within our problem formulation and objective, we can distill from this model surprisingly powerful generative capabilities. We demonstrate the effectiveness of our approach on a variety of edits applied to real-world videos.

To summarize, our work makes the following contributions:

- We introduce a new task of integrating newly generated dynamic content into real-world videos without relying on the

[1]Code will be made publicly available.

Author's Contact Information:

user to provide complex references of the effect (for example, a VFX asset or masks to specify where to locate the VFX).
- We propose a tuning-free, zero-shot method that enables harmonized content integration while maintaining high fidelity to the original scene
- We propose an automatic VLM-based evaluation metric tailored for our task, considering multiple factors, including original content preservation, new content harmonization, overall visual quality, and alignment with the edit prompt.
- We demonstrate state-of-the-art results compared to competing methods, achieving the best trade-off between synthesizing new dynamic elements and maintaining high fidelity to the original content.

2 Related Work

Text-to-Video Models. With the rise of large-scale video-text datasets, there has been significant progress in training text-to-video models using new architectures [Bar-Tal et al. 2024; Polyak et al. 2024]. While the foundational architecture of diffusion models has been commonly linked to inflating text-to-image U-Net-based models with temporal layers [cerspense 2023; Guo et al. 2023; Wang et al. 2023a], recently, a new family of Transformer-based models [HaCohen et al. 2024; Kong et al. 2025; OpenAI 2024; Yang et al. 2024; Zheng et al. 2024], referred to as Diffusion Transformers (DiTs) [Peebles and Xie 2023], have gained popularity, as DiTs enhance spatial coherence and enable arbitrary aspect ratio and video-length training. In this work, we utilize a publicly available DiT-based model [Yang et al. 2024], CogVideoX, for augmenting real-world videos with newly generated dynamic content in a zero-shot manner.

Object Insertion in Images. With the advancement of text-to-image models, techniques for image manipulation leveraging such models have also evolved rapidly. Among these advancements, notable progress has been made in the task of instruction-based image editing. Several works [Brooks et al. 2023; Sheynin et al. 2023; Zhang et al. 2023a,c] have proposed to directly fine-tune generative models on pairs of original and edited images coupled with user-provided instructions. EmuEdit [Sheynin et al. 2023] leverages a diffusion model trained on a large synthetic dataset to perform various editing tasks guided by task embeddings.

The task of object insertion into images belongs to the same category and can be considered a subfield of instruction-based editing methods. For instance, EraseDraw [Canberk et al. 2024], Paint By Inpaint [Wasserman et al. 2024], and ObjectDrop [Winter et al. 2024] leverage inpainting models to create paired-image datasets, which are then used to fine-tune image editing models. However, extending these approaches to videos presents significant challenges. In particular, generating large-scale instruction-paired video datasets can be prohibitively expensive both in time and computational resources, as it requires substantial manual effort to annotate frames and ensure alignment between textual instructions and video content. This cost and complexity make it challenging to adapt existing image-based methodologies directly to the video domain.

Concurrently to our method, Add-It [Tewel et al. 2024] proposes to manipulate the attention features of a pre-trained text-to-image diffusion model to insert objects into images in a training-free manner. While their method relies on weighted global extended attention, we propose to apply extended attention only to specific regions of the source scene, allowing the generation to focus on essential elements.

Controllable Video Generation. Recently, numerous methods have been developed to incorporate various forms of control signals into video generation pipelines. Several video-to-video methods propose to condition the generation on per-frame spatial maps such as depth maps and edge maps [Chen et al. 2023; Wang et al. 2023b]. A line of work [Geyer et al. 2023; Jeong et al. 2023; Park et al. 2024; Yatim et al. 2023; Zhao et al. 2023] proposed to utilize a text-to-video model for the task of motion transfer. Unlike these methods, which are not designed to deviate from the existing structures within a video, our approach focuses on integrating additional dynamic elements into the video.

Recent methods [Bar-Tal et al. 2024; Ma et al. 2024; Zhang et al. 2023b; Zhou et al. 2023; Zi et al. 2024] have explored adapting text-to-video models for video inpainting by conditioning on a masked video and a corresponding binary mask. This setup encourages the model to preserve unmasked information while generating new content in the masked region. All of these methods require user-provided masks — an impractical requirement for integrating complex dynamics as it requires anticipating the placement of dynamic objects (e.g., Fig. 5 jellyfish, tsunami, dinosaurs) ahead of time. In contrast, our method allows for automatic new content localization without any user-provided masks. While a static object can be masked with a simple bounding box, manually defining masks for complex motion or interactions per frame is extremely difficult.

Recently, generative Omnimatte [Lee et al. 2024] proposed a method to automatically decompose a video into object layers and their corresponding effects. However, it is not designed for the task of new content generation, as it allows only for the removal of existing scene elements. VideoDoodles [Yu et al. 2023] combines hand-drawn animations with video footage in a scene-aware manner by tracking a user-provided planar canvas in 3D. However, it does not support the creation of non-planar animations. Our framework includes both localizing dynamic new content and integrating in a scene-aware manner without reliance on user-provided positions.

Language Models for Video Content Creation. Advancements in Vision Language Models (VLMs) have enabled methods to utilize such models in various video-related tasks. Some methods [Chen et al. 2024; Yang et al. 2024] use VLMs to produce detailed video captions from a series of frames, which are then utilized to train text-to-video generative models. Other methods utilize such models for achieving better generation controllability. For instance, VideoDirectorGPT [Lin et al. 2024] utilizes a VLM for multi-scene video generation by training diffusion adapters to incorporate additional conditioning inputs, while LVD [Lian et al. 2023] incorporates layout guidance from the VLM during the sampling process. AutoVFX [Hsu et al. 2024] uses an LLM to generate a video editing program pipeline based on the user instruction. In our work, we employ a VLM as a "VFX assistant" that, based on a short user instruction, provides a comprehensive description of the edited video along with the prominent objects present in the scene.

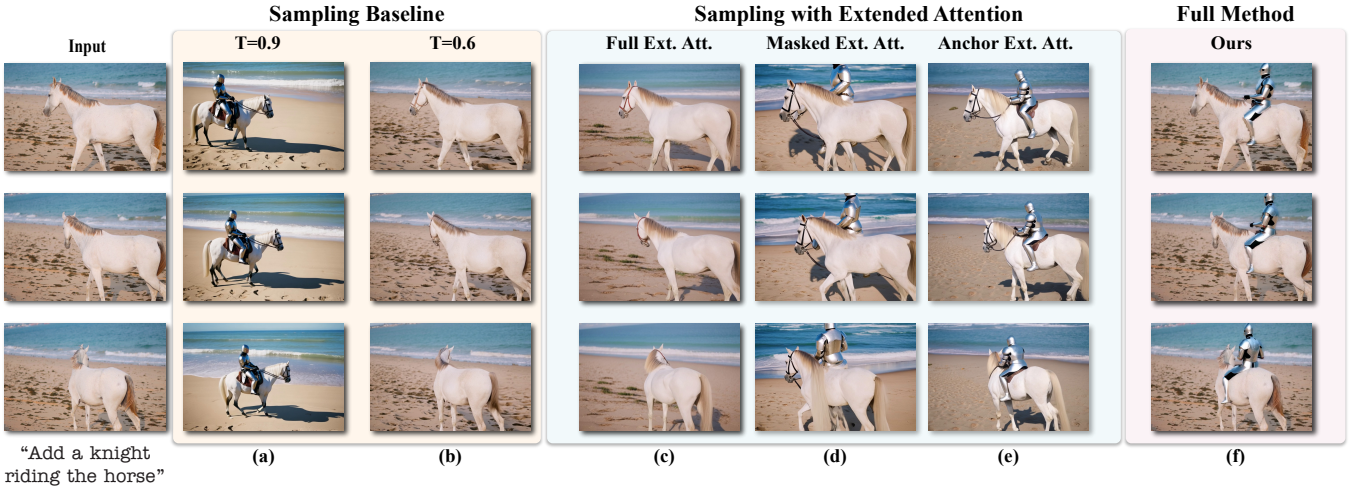

Fig. 2. Controlling fidelity to the original scene using different extended attention mechanisms. (a-b) SDEdit suffers from the original scene preservation/edit fidelity trade-off. (c-e) Three Extended Attention variants during sampling demonstrate different control levels: Full Extended Attention closely reconstructs the input scene, Masked Extended Attention proves too constrained in overlapping regions despite allowing new content emergence, and our Anchor Extended Attn. achieves optimal results by applying dropout – extending attention only at sparse points within selected regions.

Professional Software For Video Animation In professional visual effects production, tools such as Autodesk Maya [Autodesk, INC. [n. d.]], Blender [Community 2018], Unreal Engine [Epic Games [n. d.]], Adobe After Effects [Christiansen 2013] and Houdini [SideFX Houdini FX. SideFX [n. d.]] are widely used for creating and compositing complex visual effects. These tools provide artists with precise control over object modeling, animation, and integration into video footage. While powerful, they require significant expertise, extensive manual intervention, and detailed inputs, such as 3D scene reconstruction or motion tracking. All the aforementioned software count on the user to provide the 3D assets. Even though creating input 3D assets has become an easier task to solve, using new datasets [Deitke et al. 2023; Qiu et al. 2024], or generating assets based on a user prompt (For example, Meshy.ai and Alpha3D), still, 3D physical elements (like fluid or explosion) or global/multi-object effects present a significant challenge. In this work, we take the first steps towards a novice user-friendly workflow.

3 Preliminaries

Diffusion probabilistic models [Ho et al. 2020; Sohl-Dickstein et al. 2015] are a class of generative models that aim to learn a mapping from noise $\boldsymbol{x}_T \sim \mathcal{N}(0, I)$ to a data distribution q. Starting from a Gaussian i.i.d noise $\boldsymbol{x}_T \sim \mathcal{N}(0, I)$, the diffusion model Φ is applied iteratively through a sequence of denoising steps, ultimately producing a clean output sample $\boldsymbol{x}_0$.

Recently, a new class of latent text-to-video (T2V) models, built on Diffusion Transformers (DiTs) [Peebles and Xie 2023], has gained significant popularity. These models comprise multi-modal blocks (MMDiT [Esser et al. 2024]) that allow for joint processing of both text and image modalities, allowing each to inform and refine the other's representations. To process both modalities together, first, a pre-trained encoder compresses an RGB video $\mathcal{V}$ both spatially and temporally to a latent space. Then, the latent is patchified, and the resulting tokens are concatenated to the text tokens produced by the text encoder [Raffel et al. 2023].

In each MMDiT block, text tokens and spatiotemporal tokens are projected into queries, keys and values using separate sets of weights for each modality, and the sequences of the two modalities are concatenated as a joint input for the attention operation: $\mathbf{Q} = [\mathbf{Q}_{\text{text}}, \mathbf{Q}_{\text{spatio}}]$, $\mathbf{K} = [\mathbf{K}_{\text{text}}, \mathbf{K}_{\text{spatio}}]$ and $\mathbf{V} = [\mathbf{V}_{\text{text}}, \mathbf{V}_{\text{spatio}}]$. The attention [Vaswani et al. 2017] operation computes the affinities between the d-dimensional projections $\mathbf{Q}, \mathbf{K}$ to yield the output of the layer:

$$\mathbf{A} \cdot \mathbf{V} \text{ where } \mathbf{A} = \texttt{Attention}(\mathbf{Q}, \mathbf{K}) = \texttt{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right) \quad (1)$$

To capture inter-token relationships, Rotary Position Embeddings (RoPE) [Su et al. 2021] are applied to the input to the attention operation.

4 Method

Given an input video $\mathcal{V}_{\text{orig}}$ and a textual instruction $\mathcal{P}_{\text{VFX}}$, our goal is to synthesize a new video $\mathcal{V}_{\text{VFX}}$, in which new dynamic elements are seamlessly integrated to the existing scene. We address this task by estimating a residual in the latent space of the text-to-video diffusion model, which is added to the latent of the original video. The final edited video is obtained by decoding their sum with the text-to-video diffusion model's VAE decoder

Tackling this task requires ensuring that the generated content, such as new objects or effects, adheres to the dynamics of the existing scene. The location and size of the new content must align with the camera motion and the environment, while its actions and movements must appropriately respond to other dynamic objects present in the scene. Our framework (illustrated in Fig. 3) addresses these challenges by incorporating the following key components:

(1) *VLM as a VFX assistant.* We utilize a pre-trained VLM to interpret the user's instructions, reason about the interactions with the scene's dynamics, and provide a descriptive prompt for the

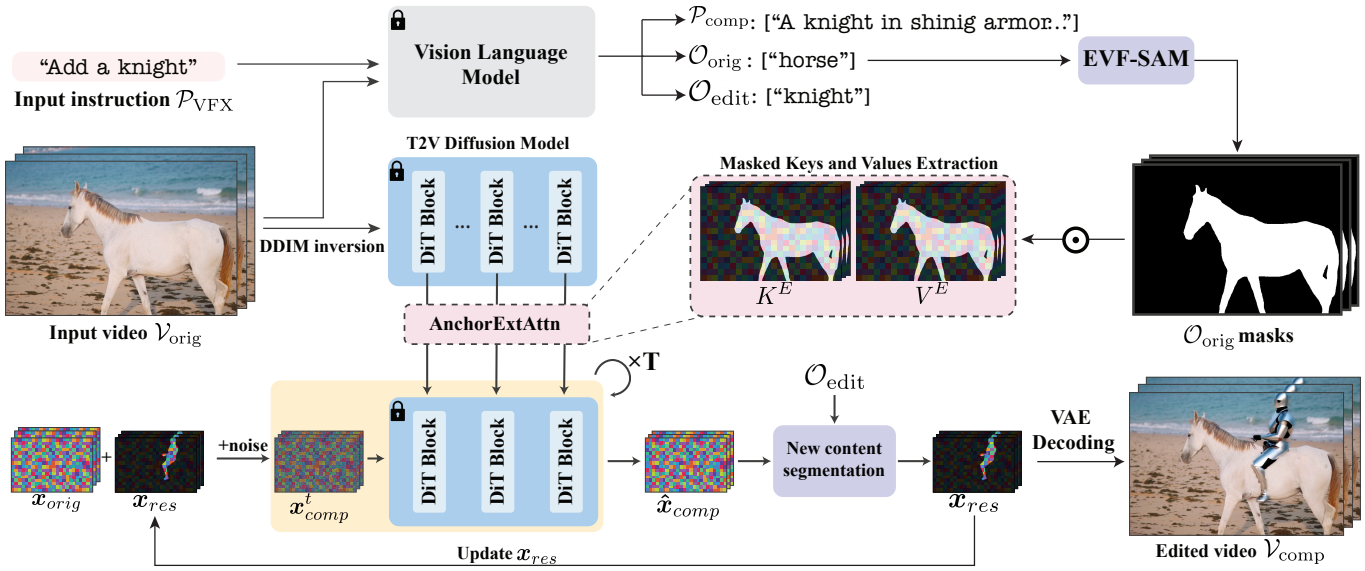

Fig. 3. DynVFX pipeline. Top row: Given an input video $\mathcal{V}_{\text{orig}}$, we apply DDIM inversion (see Sec. 3) and extract spatiotemporal keys and values $[\mathbf{K}_{\text{orig}}, \mathbf{V}_{\text{orig}}]$ from the original noisy latents. Given the user instruction $\mathcal{P}_{\text{VFX}}$ we instruct the VLM to envision the augmented scene and output the text edit prompt $\mathcal{P}_{\text{comp}}$, prominent object descriptions $\mathcal{O}_{\text{orig}}$ that are used to mask out the extracted keys and values and target object descriptions $\mathcal{O}_{\text{edit}}$. Bottom row: We estimate a residual $\boldsymbol{x}_{\text{res}}$ to the original video latent ($\boldsymbol{x}_{\text{orig}}$). This is done by iteratively applying SDEdit with our Anchor Extended Attention, segmenting the target objects ($\mathcal{O}_{\text{edit}}$) from the clean result, and updating $\boldsymbol{x}_{\text{res}}$ accordingly.

T2V diffusion model by guiding it to act as "VFX assistant" via a system prompt containing guidelines in the context of our tasks.

(2) *Localization via Anchor Extended Attention.* To steer the localization of the edit and make it content-aware, we propose to utilize Extended Attention during sampling from the T2V DiT model to a set of keys and values extracted from sparse locations in the original video.

(3) *Content Harmonization via Iterative refinement.* To improve the blending of the generated content with the input video and achieve better harmonization, we iteratively update the estimated residual latent by repeating the sampling process with AnchorExtAttn multiple times, progressively reducing the level of noise added at each step.

4.1 VLM as a VFX assistant

To create a fully automatic framework requiring only a simple user-provided instruction describing the desired content, we incorporate a Vision-Language Model (VLM) into our framework. Specifically, given the user instruction $\mathcal{P}_{\text{VFX}}$, along with evenly spaced keyframes from the original video $\mathcal{V}_{\text{orig}}$, we instruct the VLM [Achiam et al. 2023] to provide a detailed caption - a composition prompt $\mathcal{P}_{\text{comp}}$ describing the new composited scene. While the model gives an accurate, descriptive source scene caption, in some cases, we observed that it fails to give captions suitable for compositing VFX with the scene. To overcome this, we guide the model to reply in an in-context matter by asking it to imagine a conversation with a visual effects (VFX) artist to obtain a caption that would describe the composited scene correctly. In this conversation, VLM will "consult" with a VFX artist about how the new content should be integrated into the scene. Based on their discussion, it provides a caption that describes how the added content fits into the scene. This results in text prompts that encourage the generated output video to include a natural interaction between the new content and the original environment. The guidelines include focusing on (1) spatial and dynamic awareness of existing scene elements, (2) preservation of original scene behaviors, and (3) atmospheric coherence between new and existing content. See SM for more details about how we use the VLM to reason about the new scene integration with in-context reasoning for VFX.

We also use the VLM to obtain a list of prominent foreground objects in the original video: $\mathcal{O}_{\text{orig}}$ and the object that will be added according to the edit prompt: $\mathcal{O}_{\text{edit}}$. See more details about prompting and mask processing in SM.

4.2 Localization via Anchor Extended Attention

A pivotal aspect of our method is the accurate localization of the new content in the existing scene, ensuring alignment with camera motion, occlusions, and scene depth. While the composition prompt can describe the desired location, naive noising-denoising with this prompt introduces a trade-off: As shown in Fig. 2, using SDEdit [Meng et al. 2022] with a high noising timestep fails to retain the original scene, resulting in misaligned new content. In contrast, a low noising timestep, as illustrated in Fig. 2 (b), limits deviations from the original video.

To tackle the localization challenge, we extend the attention module during sampling to include the input video's corresponding attention features. Specifically, we apply DDIM inversion [Song et al. 2020] to the original video $\mathcal{V}_{\text{orig}}$ and extract the spatio-temporal

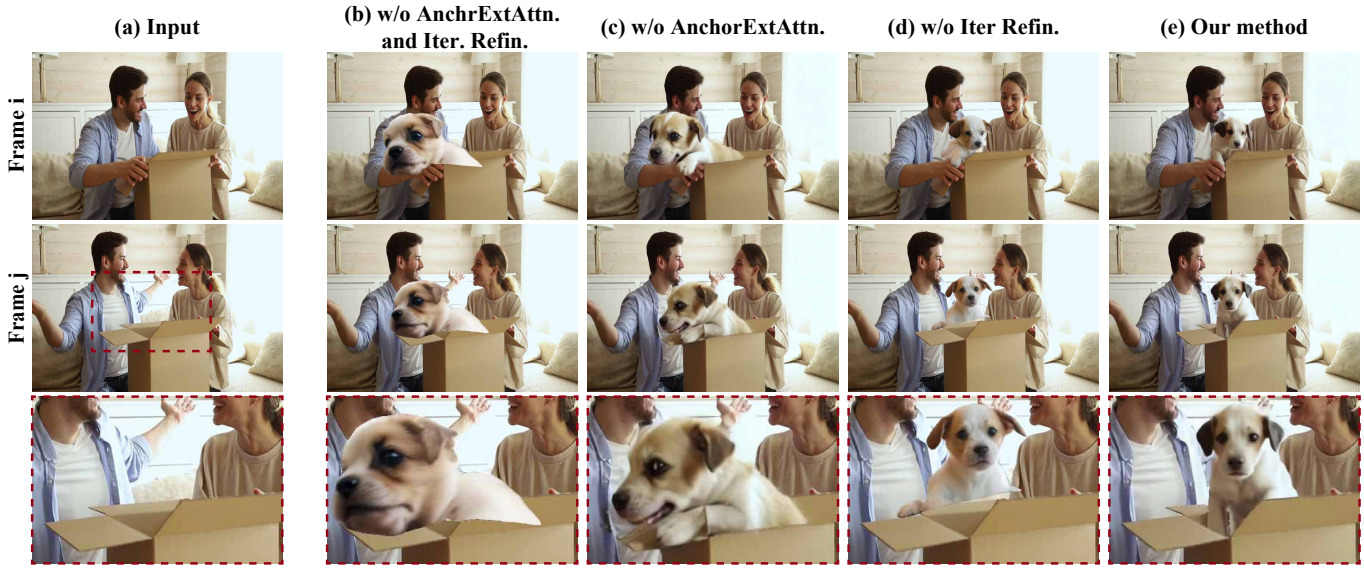

Fig. 4. Ablations. (b) Excluding both AnchorExtAttn and the Iterative refinement process results in significant misalignment with the original scene and poor harmonization (e.g., the size of the puppy relative to the scene and boundary artifacts). (c) Omitting AnchorExtAttn leads to incorrect positioning of the new content. (d) Removing iterative refinement results in poor harmonization. Our full method (e) exhibits good localization and harmonization of the edit

keys and values $K_{\text{orig}}, V_{\text{orig}}$ from the attention module of every block in the network and generation timestep t. These keys and values are then used to extend the attention mechanism during sampling with $\mathcal{P}_{comp}$ to control the localization of the edit.

When using all keys and values, the extended attention operation can be expressed as:

$$\texttt{Attn}(Q_{\text{VFX}}, [K_{\text{VFX}}, K_{\text{orig}}], [V_{\text{VFX}}, V_{\text{orig}}]) \tag{2}$$

Interestingly, extending keys and values provides more than just global information, but each feature locally encodes corresponding patches in the video. As shown in Fig. 2 (c), extending the attention to the full set of keys and values is sufficient to achieve an approximate reconstruction of the original video. We hypothesize that this occurs because the same positional embedding is applied to the K_{orig} as to K_{VFX}, aligning the positional embeddings in spatiotemporal locations. This observation provides strong evidence for the significance of the positional embedding. Due to this, the setup proves to be overly restrictive in adhering to $\mathcal{P}_{\text{comp}}$, as illustrated in Fig. 2(c).

Selective Attention. To overcome this limitation, we propose to restrict the extended attention only to specific positions in the source scene. Specifically, we use a selection function $\mathcal{F}$ to determine which keys and values are retained.

An intuitive strategy is to apply region-based attention with Masked Extended Attention by identifying the most critical regions for preserving scene coherence and extending attention with keys and values within these masked regions M_{orig}, i.e., $\mathcal{F}(\mathcal{A}) = M_{\text{orig}} \circ \mathcal{A}$. To get M_{orig}, we ask a VLM to provide a list of foreground objects in the original video O_{orig} (typically the most spatially prominent elements within a scene) and then we obtain corresponding masks using a text-based segmentation model. For example, Fig. 2(d), where the masked region includes the horse, illustrates the effect of this on the generated content. As shown, this setup preserves high fidelity to the original scene within the mask M_{orig} and allows new content to emerge in the unmasked regions.

However, this approach proves too constrained in overlapping regions. To address this limitation, we propose our Anchor Extended Attention:

$$\begin{aligned} \texttt{AnchorExtAtt} :=& \texttt{Attn}(Q_{\text{VFX}}, [K_{\text{VFX}}, K^E], [V_{\text{VFX}}, V^E]) \\ & \text{s.t. } K^E := \mathcal{F}(K_{\text{orig}}, M_{\text{orig}}) \textit{ and } V^E := \mathcal{F}(V_{\text{orig}}, M_{\text{orig}}) \\ & \mathcal{F} := \begin{cases} \texttt{Drop}_{FG}(M_{\text{orig}}) \cup \texttt{Drop}_{BG}(\sim M_{\text{orig}}) \circ \mathcal{A} \end{cases} \end{aligned} \tag{3}$$

This formulation introduces dropout within the masked regions M_{orig}, generating a sparse set of anchor points that guide the generation while preserving flexibility. This formulation introduces a predominantly content-aware selection of anchors from foreground regions, along with a sparser set of background anchors, to achieve robust and spatially coherent integration of new content (e.g. the horse to enforce the knewly added knight to align with its motion). As demonstrated in Fig. 2(e), this balanced approach offers sufficient flexibility for creative edits while preserving key spatial cues from the original scene.

4.3 Content Harmonization

Our anchor extended attention steers the placement of the new content to align with the original scene. However, it does not guarantee precise pixel-level alignment. As can be seen in Fig. 2(e), the legs of the horse are not perfectly aligned with the original horse. To guarantee pixel-level alignment, a straightforward approach is to extract a mask of the new content $\boldsymbol{M}^{rgb}_{VFX}$ from the sampling with AnchorExtAttn (Eq. 3) output $\hat{\mathcal{V}}_{comp}$. More concretely, by applying

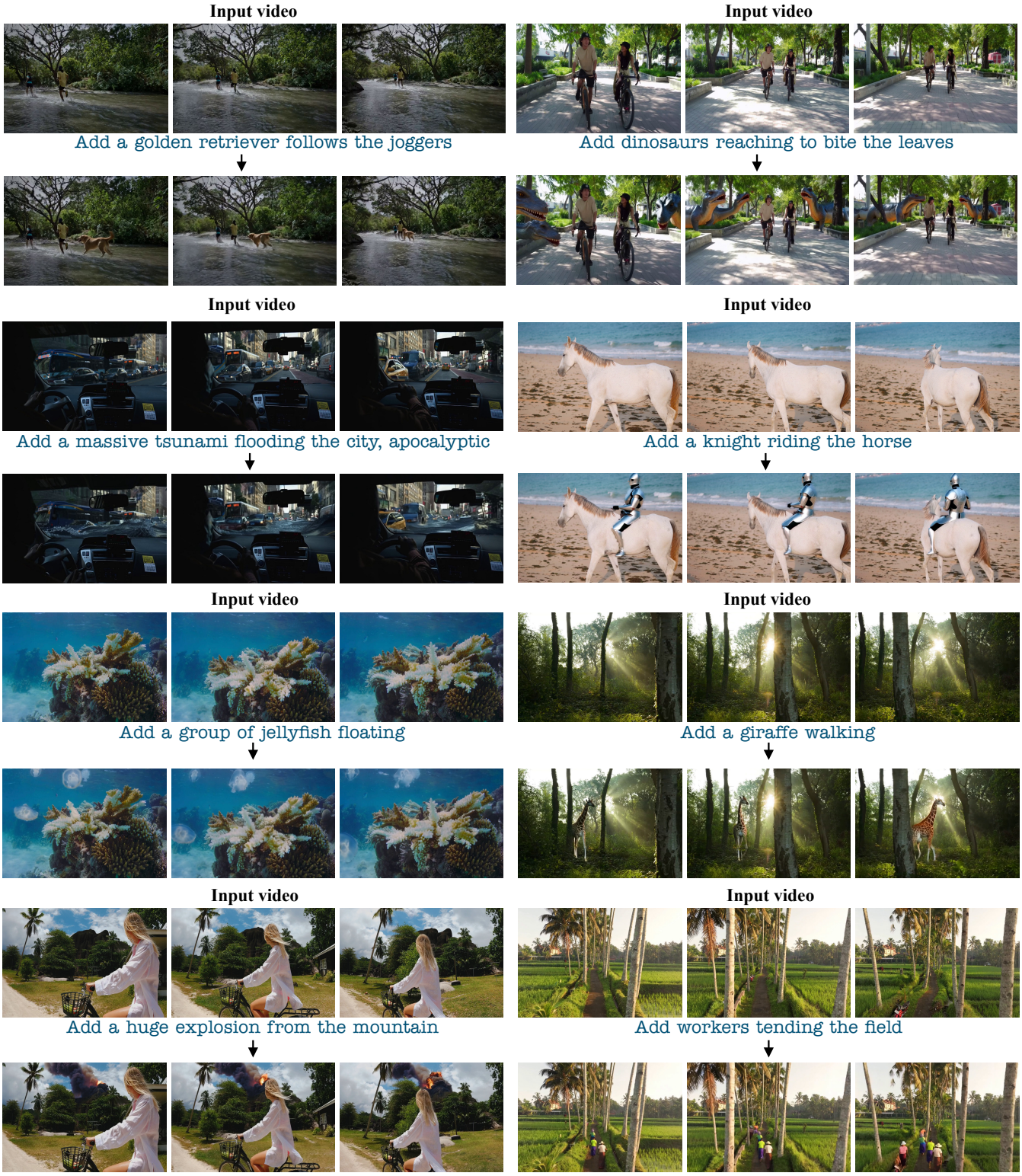

Fig. 5. Sample results of our method. See SM for full vide results.

Algorithm 1 DynVFX Algorithm

Input:
- $\mathcal{V}_{\text{orig}}, \mathcal{P}_{\text{VFX}}$ ▷ Input video & instruction prompt
- τ_{A} ▷ Extended Attention threshold
- Ψ ▷ Video segmentation model
- VLM ▷ Vision Language model

Preprocess:
- $\mathcal{P}_{\text{comp}} \leftarrow \text{VLM}[\mathcal{V}_{\text{orig}}, \mathcal{P}_{\text{VFX}}]$ ▷ Composition Prompt
- $O_{\text{orig}}, O_{\text{edit}} \leftarrow \text{VLM}[\mathcal{V}_{\text{orig}}, \mathcal{P}_{\text{VFX}}]$ ▷ Original objects and VFX object
- $M_{orig} \leftarrow \texttt{Get-Latent-Mask}(\Psi; \mathcal{V}_{\text{orig}}, O_{\text{orig}})$ ▷ Extract source masks
- $x_{orig} \leftarrow \texttt{Encode}[\mathcal{V}_{\text{orig}}]$ ▷ Encode video into latent space
- $\mathbf{K}_{\text{orig}}, \mathbf{V}_{\text{orig}} \leftarrow \text{DDIM-Inv}[x_{orig}] \quad \forall t \in [T]$

For $t = \tilde{T}, \ldots, T_{min}$ **do**
- $\boldsymbol{x}_{res} = \mathbf{0}$ ▷ initialize the residual latent
- $\boldsymbol{x}_{comp} = x_{orig} + x_{res}$
- if $t > \tau_A$ then $K^E, V^E \leftarrow \mathcal{F}(K_{\text{orig}}, M_{\text{orig}}), \mathcal{F}(V_{\text{orig}}, M_{\text{orig}})$
- else $K^E, V^E \leftarrow \emptyset$
- $\hat{\boldsymbol{x}}_{comp} \leftarrow \texttt{Sampling}[x_{comp}, \mathcal{P}_{\text{comp}}, t; \texttt{AnchorExtAttn}[K^E, V^E]]$
- $\hat{\mathcal{V}}_{comp} \leftarrow \texttt{Decode}(\hat{\boldsymbol{x}}_{comp})$ ▷ Decode latent
- $\boldsymbol{M}_{VFX} \leftarrow \texttt{Get-Latent-Mask}(\Psi; \hat{\mathcal{V}}_{comp}, O_{\text{edit}})$ ▷ Extract VFX masks
- $\boldsymbol{x}_{res} = \boldsymbol{M}_{VFX} \cdot (\hat{x}_{comp} - x_{orig})$

$\boldsymbol{x}_{comp} = x_{orig} + x_{res}$

$\mathcal{V}_{\text{comp}} \leftarrow \texttt{Decode}[x_{comp}]$ ▷ Output video

Output: $\mathcal{V}_{\text{comp}}$

a text-based segmentation model using the added object description provided by the VLM.

The mask can then be used to replace the pixels outside it with the corresponding pixels from the original video: $\mathcal{V}\text{comp}[\sim \boldsymbol{M}^{rgb}_{VFX}] = \mathcal{V}\text{orig}[\sim \boldsymbol{M}^{rgb}_{VFX}]$. While this preserves the unaffected regions, it often results in poor harmonization with the input video.

To improve content harmonization, we propose a different approach: repeat the sampling process with AnchorExtAttn (Eq. 3) multiple times, progressively reducing the level of noise added at each step. This iterative approach gradually refines the new content's interaction with the original scene. As shown in Fig.3, we update $\boldsymbol{x}_{res}$ representing the difference between the generated output $\hat{\mathcal{V}}_{comp}$ and the original video $\mathcal{V}_{orig}$ within the regions where new content appears, allowing each iteration to adjust the generated content's high-frequency details to better match the original video. We summarize our method in Alg. 1.

5 Results

We evaluated our method on a dataset of 18 publicly sourced videos, featuring a wide range of complex scenes in terms of camera and object motion, lighting conditions, and physical environments. Additionally, we use some videos from DAVIS [Pont-Tuset et al. 2017]. Our videos and implementation details are available in the SM.

Figures 1, 5 show sample results of our method. As seen, our method facilitates natural integration of broad range of visual effects, ranging from environmental effects (tsunami in Fig. 4 and explosion in Fig. 5) to new object insertion (horse riding knight in Fig. 5 and dancing bear in Fig. 6). In all examples, the new content is naturally localized in the scene, even in challenging scenarios of multiple objects (dinosaurs or workers in Fig. 5) and partial occlusions (puppy in Fig. 1 and giraffe in Fig. 5).

5.1 Qualitative Evaluation

To the best of our knowledge, no existing method has been designed to synthesize dynamic objects in a given real video without user masks or any user information except for the input video itself and a simple instruction. We thus compare our method to the following baselines: *(i)* SDEdit [Meng et al. 2022] using the same T2V model as ours, *(ii)* DDIM inversion [Song et al. 2020] and sampling with the target prompt, *(iii)* LORA fine-tuning [Hu et al. 2021] of the T2V model and sampling with the target prompt and *(iv)* Gen-3 [R Team, Runway [n. d.]] video-to-video, designed for video stylization.

Figure 6 shows a qualitative comparison to the baselines. As can be seen, all baselines exhibit different limitations in maintaining scene fidelity while introducing new content. SDEdit [Meng et al. 2022] manages to fulfill the edit prompt, yet the scene has significantly deviated from the original one in terms of appearance, motion, positioning, or scale (e.g., deer in the creek). DDIM inversion is not suitable for editing. LORA fine-tuning suffers from the trade-off between preserving aspects of the original scene and adding new content to the scene. Either over-fitting original scene appearance (e.g., added dragon-dog hybrid), or under-fitting the original layout (e.g., incorrect scale of deer). Gen-3 is conditioned on depth maps extracted from the input video, hence tends to significantly alter scene appearance and not allow the insertion of new objects that change the scene layout. In each case, these limitations affect the overall scene coherence and realism of the added elements. Our method successfully adds new content to the scene, achieving high fidelity to the user instructions while allowing for natural interactions between original and added elements (e.g. natural interaction between woman and bear).

5.2 Quantitative Evaluation

We numerically evaluate our results using the following metrics:
(i) Edit fidelity. Following previous works (e.g., [Hsu et al. 2024; Tewel et al. 2024]), we measure per-frame Directional CLIP Similarity [Gal et al. 2021; Radford et al. 2021] to assess the alignment between the change in the source and the target prompt, and the change between the source and edited frames.
(ii) Original content preservation. We evaluate how well the edited video preserves the original content outside the modified region. To this end, we segment the new content in the edited video using [Zhang et al. 2024] , and compute the masked Structural Similarity Index (SSIM) over the complementary regions to the edited ones.
(iii) VLM quality evaluation. We employ a Vision-Language Model (VLM) for expanding the per-frame metrics above in the following manner. We input to the VLM several frames from the edited videos and instruct it to evaluate four key aspects: how well the edit follows the text prompt (*Text Alignment*), the overall visual quality of the edited frames (*Visual Quality*), how well the new content is harmonized with the source frames (*Edit Harmonization* and the realism of the added object's dynamics relative to the scene (*Dynamics Score*). For each aspect, the VLM outputs a score between 0 and 1,

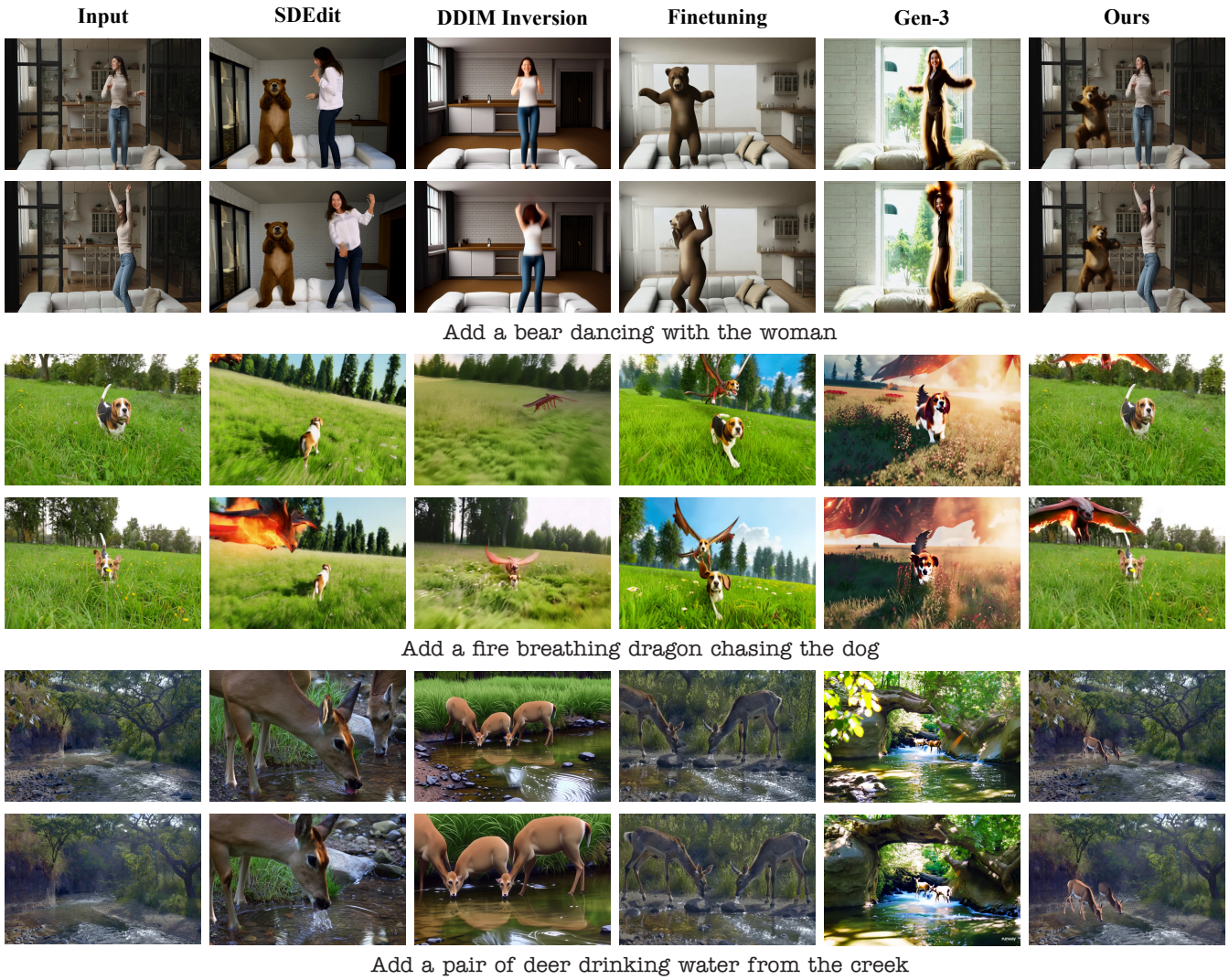

Fig. 6. Qualitative comparison. Sample results of our method, SDEdit [Meng et al. 2022], DDIM inversion [Song et al. 2020], Lora fine-tuning [Hu et al. 2021], and Gen-3 [R Team, Runway [n. d.]]. See SM for videos.

Method	Metrics		VLM-based evaluation				User Study	
	CLIP Directional	SSIM	Text Alignment	Visual Quality	Edit Harmonization	Dynamics Score	Content Integration	Edit Harmonization
Gen-3	0.130	0.285	0.418	0.610	0.374	0.379	97.65	93.33
LORA finetuning	0.277	0.361	0.812	0.787	0.756	0.759	92.22	81.11
DDIM inv. sampling	0.184	0.444	0.535	0.699	0.528	0.529	99.20	98.67
SDEdit (0.9)	0.272	0.332	0.794	0.799	0.754	0.756	98.91	82.13
SDEdit (0.6)	0.111	0.567	0.510	0.704	0.513	0.504	97.69	96.76
w/o AnchorExtAttn	**0.317**	0.697	0.775	0.724	0.683	0.691	89.30	88.89
w/o Iterative Refinement	0.295	0.760	0.817	0.789	0.769	0.760	85.80	86.42
Ours	0.311	**0.775**	**0.860**	**0.803**	**0.796**	**0.785**	-	-

Table 1. Quantitative Evaluation. We assess the quality of our method compared to several baselines.

with higher scores indicating better performance. Our evaluation protocol is included in the SM.

Figure 7 and Table 1 present the results of the described metrics on a set of 27 video-edit text pairs comprising 20 unique videos. As shown, our method outperforms the baselines in both SSIM and

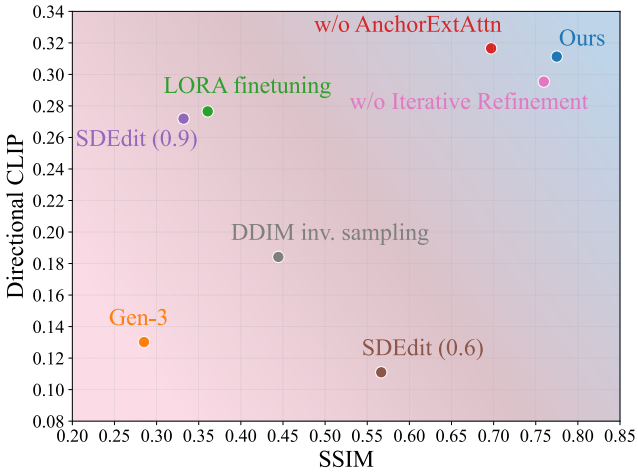

Fig. 7. Metrics. We measure CLIP Directional score (higher is better) and masked SSIM (higher is better). Our method demonstrates a better balance between these two metrics.

Directional CLIP metrics, demonstrating superior edit fidelity and maintaining higher structural similarity in the unedited regions. The VLM-based evaluation aligns with this assessment and additionally shows that our method produces videos that achieve better content integration and greater motion realism.

(iv) User study. We conducted a user study to evaluate the ability to integrate the new content while preserving the original video. Participants were shown the input video, a text description of the new content, our result, and a baseline output. They were asked two questions: "Which video better preserves the original footage while adding new content?" and "Which video better integrates the new content in a realistic and seamless way?". In total, we collected 3240 user judgments from 120 users. As seen in Table 1, our method is consistently preferred over all baselines.

5.3 Ablations

We ablate key design choices of our method: anchor extended attention and iterative updates of the edit, by excluding each component from our framework.

As seen in Fig. 4(c), omitting AnchorExtAttn leads to new content being misaligned relative to the original scene - the added content is poorly integrated into the original scene. Applying only the first iteration of our method (w/o iterative refinement, Fig. 4(d)) results in a better alignment with the input video, but the composition is still unstable, as evident, for example, in the puppy's body hovering over the box in the first scene. Our full method achieves better composition with proper spatial relationships, demonstrating the importance of both components for realistic scene editing Fig. 4(f). We numerically evaluate each ablation with the same set of metrics described in Sec 5.2 and report them in Table 1.

6 Discussion and Conclusions

We introduced the task of augmenting real videos with new dynamic content based on a user-provided instruction. We presented a zero-shot method utilizing the T2V diffusion model in a feature manipulation framework, enabling correct localization and natural blending of new content with existing video elements.

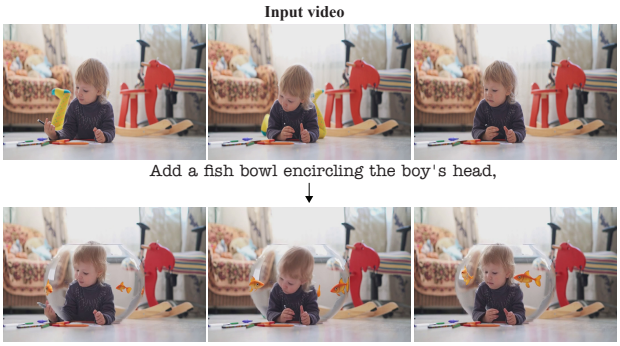

Fig. 8. Limitations. In some cases, the T2V diffusion model struggles to precisely follow the edit prompt

As our method is built upon the pre-trained T2V diffusion model, the quality of the generated edits is inherently tied to the performance and capabilities of the underlying model. As seen in Fig. 8, the T2V model sometimes struggles with generating videos precisely following the edit prompt. Additionally, the text-based localization relies on the capabilities of the segmentation model [Zhang et al. 2024], which can sometimes produce inaccurate masks and fails to account for effects like shadows and reflections if not specified in the text prompt. Despite the limitations, our method significantly improves over baselines, expanding the capabilities of pre-trained text-to-video diffusion models.

7 Acknowledgement

This project received funding from the Israeli Science Foundation (grant 2303/20).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

Autodesk, INC. [n. d.]. *Maya.* https:/autodesk.com/maya

Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, Yuanzhen Li, Michael Rubinstein, Tomer Michaeli, Oliver Wang, Deqing Sun, Tali Dekel, and Inbar Mosseri. 2024. Lumiere: A Space-Time Diffusion Model for Video Generation. arXiv:2401.12945 [cs.CV]

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. 2023. InstructPix2Pix: Learning to Follow Image Editing Instructions. In *CVPR.*

Alper Canberk, Maksym Bondarenko, Ege Ozguroglu, Ruoshi Liu, and Carl Vondrick. 2024. EraseDraw: Learning to Draw Step-by-Step via Erasing Objects from Images. (2024).

cerspense. 2023. https://huggingface.co/cerspense/zeroscope_v2_576w.

Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and S. Tulyakov. 2024. Panda-70M: Captioning 70M Videos with Multiple Cross-Modality Teachers. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 13320–13331. https://api.semanticscholar.org/CorpusID:268091168

Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang-Jin Lin. 2023. Control-A-Video: Controllable Text-to-Video Generation with Diffusion Models. *ArXiv* abs/2305.13840 (2023). https://api.semanticscholar.org/CorpusID:258841645

Mark Christiansen. 2013. *Adobe After Effects CC Visual Effects and Compositing Studio Techniques.* Adobe Press.

Blender Online Community. 2018. *Blender - a 3D modelling and rendering package.* Blender Foundation, Stichting Blender Foundation, Amsterdam. http://www.blender.org

Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. 2023. Objaverse-XL: A Universe of 10M+ 3D Objects. *arXiv preprint arXiv:2307.05663* (2023).

Epic Games. [n. d.]. *Unreal Engine.* https://www.unrealengine.com

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. 2024. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. arXiv:2403.03206 [cs.CV] https://arxiv.org/abs/2403.03206

Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. 2021. StyleGAN-NADA: CLIP-Guided Domain Adaptation of Image Generators. arXiv:2108.00946 [cs.CV]

Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. 2023. TokenFlow: Consistent Diffusion Features for Consistent Video Editing. *arXiv preprint arxiv:2307.10373* (2023).

Jiaqi Guo, Lianli Gao, Junchen Zhu, Jiaxin Zhang, Siyang Li, and Jingkuan Song. 2024. MagicVFX: Visual Effects Synthesis in Just Minutes. In *ACM Multimedia*. https://api.semanticscholar.org/CorpusID:273642722

Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. 2023. AnimateDiff: Animate Your Personalized Text-to-Image Diffusion Models without Specific Tuning.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. 2024. LTX-Video: Realtime Video Latent Diffusion. arXiv:2501.00103 [cs.CV] https://arxiv.org/abs/2501.00103

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* (2020).

Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. 2022. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868* (2022).

Hao-Yu Hsu, Zhi-Hao Lin, Albert Zhai, Hongchi Xia, and Shenlong Wang. 2024. AutoVFX: Physically Realistic Video Editing from Natural Language Instructions. *arXiv preprint arXiv:2411.02394* (2024).

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685 [cs.CL] https://arxiv.org/abs/2106.09685

Hyeonho Jeong, Geon Yeong Park, and Jong Chul Ye. 2023. VMC: Video Motion Customization using Temporal Attention Adaption for Text-to-Video Diffusion Models. *arXiv preprint arXiv:2312.00845* (2023).

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014).

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. 2025. HunyuanVideo: A Systematic Framework For Large Video Generative Models. arXiv:2412.03603 [cs.CV] https://arxiv.org/abs/2412.03603

Yao-Chih Lee, Erika Lu, Sarah Rumbley, Michal Geyer, Jia-Bin Huang, Tali Dekel, and Forrester Cole. 2024. Generative Omnimatte: Learning to Decompose Video into Layers. *arXiv preprint arXiv:2411.16683* (2024).

Long Lian, Baifeng Shi, Adam Yala, Trevor Darrell, and Boyi Li. 2023. LLM-grounded Video Diffusion Models. *arXiv preprint arXiv:2309.17444* (2023).

Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. 2024. VideoDirectorGPT: Consistent Multi-Scene Video Generation via LLM-Guided Planning. In *COLM*.

Jingwei Ma, Erika Lu, Roni Paiss, Shiran Zada, Aleksander Holynski, Tali Dekel, Brian Curless, Michael Rubinstein, and Forrester Cole. 2024. VidPanos: Generative Panoramic Videos from Casual Panning Videos. In *ACM SIGGRAPH Conference and Exhibition on Computer Graphics and Interactive Techniques in Asia*. https://api.semanticscholar.org/CorpusID:273403638

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. 2022. SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations. In *International Conference on Learning Representations*.

OpenAI. 2024. Sora: Creating video from text.

Geon Yeong Park, Hyeonho Jeong, Sang Wan Lee, and Jong Chul Ye. 2024. Spectral Motion Alignment for Video Motion Transfer using Diffusion Models. arXiv:2403.15249 [cs.CV]

William Peebles and Saining Xie. 2023. Scalable Diffusion Models with Transformers. arXiv:2212.09748 [cs.CV] https://arxiv.org/abs/2212.09748

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, David Yan, Dhruv Choudhary, Dingkang Wang, Geet Sethi, Guan Pang, Haoyu Ma, Ishan Misra, Ji Hou, Jialiang Wang, Kiran Jagadeesh, Kunpeng Li, Luxin Zhang, Mannat Singh, Mary Williamson, Matt Le, Matthew Yu, Mitesh Kumar Singh, Peizhao Zhang, Peter Vajda, Quentin Duval, Rohit Girdhar, Roshan Sumbaly, Sai Saketh Rambhatla, Sam Tsai, Samaneh Azadi, Samyak Datta, Sanyuan Chen, Sean Bell, Sharadh Ramaswamy, Shelly Sheynin, Siddharth Bhattacharya, Simran Motwani, Tao Xu, Tianhe Li, Tingbo Hou, Wei-Ning Hsu, Xi Yin, Xiaoliang Dai, Yaniv Taigman, Yaqiao Luo, Yen-Cheng Liu, Yi-Chiao Wu, Yue Zhao, Yuval Kirstain, Zecheng He, Zijian He, Albert Pumarola, Ali Thabet, Artsiom Sanakoyeu, Arun Mallya, Baishan Guo, Boris Araya, Breena Kerr, Carleigh Wood, Ce Liu, Cen Peng, Dimitry Vengertsev, Edgar Schonfeld, Elliot Blanchard, Felix Juefei-Xu, Fraylie Nord, Jeff Liang, John Hoffman, Jonas Kohler, Kaolin Fire, Karthik Sivakumar, Lawrence Chen, Licheng Yu, Luya Gao, Markos Georgopoulos, Rashel Moritz, Sara K. Sampson, Shikai Li, Simone Parmeggiani, Steve Fine, Tara Fowler, Vladan Petrovic, and Yuming Du. 2024. Movie Gen: A Cast of Media Foundation Models. arXiv:2410.13720 [cs.CV]

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. 2017. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675* (2017).

Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi Zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. 2024. Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9914–9925.

R Team, Runway. [n. d.]. *Platform for AI-powered video editing and generative media creation.* https://runwayml.com

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning*. https://api.semanticscholar.org/CorpusID:231591445

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. arXiv:1910.10683 [cs.LG] https://arxiv.org/abs/1910.10683

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya K. Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloé Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross B. Girshick, Piotr Doll'ar, and Christoph Feichtenhofer. 2024. SAM 2: Segment Anything in Images and Videos. *ArXiv* abs/2408.00714 (2024). https://api.semanticscholar.org/CorpusID:271601113

Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. 2023. Emu Edit: Precise Image Editing via Recognition and Generation Tasks. https://api.semanticscholar.org/CorpusID:265221391

SideFX Houdini FX. SideFX. [n. d.]. *Side Effects Software Inc.* https://sidefx.com

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*. PMLR, 2256–2265.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising Diffusion Implicit Models. In *International Conference on Learning Representations*.

Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding. *ArXiv* abs/2104.09864 (2021). https://api.semanticscholar.org/CorpusID:233307138

Yoad Tewel, Rinon Gal, Dvir Samuel Yuval Atzmon, Lior Wolf, and Gal Chechik. 2024. Add-it: Training-Free Object Insertion in Images With Pretrained Diffusion Models. *ArXiv* abs/2411.07232 (2024). https://api.semanticscholar.org/CorpusID:273962996

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. 2023. Plug-and-Play Diffusion Features for Text-Driven Image-to-Image Translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1921–1930.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Neural Information Processing Systems*. https://api.semanticscholar.org/CorpusID:13756489

Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. 2023a. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571* (2023).

Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. 2023b. VideoComposer: Compositional Video Synthesis with Motion Controllability. *ArXiv* abs/2306.02018 (2023). https://api.semanticscholar.org/CorpusID:259075720

Navve Wasserman, Noam Rotstein, Roy Ganz, and Ron Kimmel. 2024. Paint by Inpaint: Learning to Add Image Objects by Removing Them First. *ArXiv* abs/2404.18212 (2024). https://api.semanticscholar.org/CorpusID:269449302

Daniel Winter, Matan Cohen, Shlomi Fruchter, Yael Pritch, Alex Rav-Acha, and Yedid Hoshen. 2024. ObjectDrop: Bootstrapping Counterfactuals for Photorealistic Object

Removal and Insertion. arXiv:2403.18818 [cs.CV]
Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. 2024. CogVideoX: Text-to-Video Diffusion Models with An Expert Transformer. *arXiv preprint arXiv:2408.06072* (2024).
Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. 2023. Space-Time Diffusion Features for Zero-Shot Text-Driven Motion Transfer. *arXiv preprint arxiv:2311.17009* (2023).
Emilie Yu, Kevin Blackburn-Matzen, Cuong Nguyen, Oliver Wang, Rubaiat Habib Kazi, and Adrien Bousseau. 2023. VideoDoodles: Hand-Drawn Animations on Videos with Scene-Aware Canvases. *ACM Trans. Graph.*, Article 54 (2023), 12 pages. https://doi.org/10.1145/3592413
Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. 2023a. MagicBrush: A Manually Annotated Dataset for Instruction-Guided Image Editing. In *Advances in Neural Information Processing Systems*.
Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Haiquan Wang, Silvio Savarese, Stefano Ermon, Caiming Xiong, and Ran Xu. 2023c. HIVE: Harnessing Human Feedback for Instructional Visual Editing. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), 9026–9036. https://api.semanticscholar.org/CorpusID:257622925
Yuxuan Zhang, Tianheng Cheng, Rui Hu, Lei Liu, Heng Liu, Longjin Ran, Xiaoxin Chen, Wenyu Liu, and Xinggang Wang. 2024. EVF-SAM: Early Vision-Language Fusion for Text-Prompted Segment Anything Model. (2024). arXiv:2406.20076 [cs.CV] https://arxiv.org/abs/2406.20076
Zhixing Zhang, Bichen Wu, Xiaoyan Wang, Yaqiao Luo, Luxin Zhang, Yinan Zhao, Péter Vajda, Dimitris N. Metaxas, and Licheng Yu. 2023b. AVID: Any-Length Video Inpainting with Diffusion Model. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), 7162–7172. https://api.semanticscholar.org/CorpusID:266055411
Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jiawei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. 2023. MotionDirector: Motion Customization of Text-to-Video Diffusion Models. *arXiv preprint arXiv:2310.08465* (2023).
Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. 2024. *Open-Sora: Democratizing Efficient Video Production for All.* https://github.com/hpcaitech/Open-Sora
Shangchen Zhou, Chongyi Li, Kelvin C.K Chan, and Chen Change Loy. 2023. ProPainter: Improving Propagation and Transformer for Video Inpainting. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.
Bojia Zi, Shihao Zhao, Xianbiao Qi, Jianan Wang, Yukai Shi, Qianyu Chen, Bin Liang, Kam-Fai Wong, and Lei Zhang. 2024. CoCoCo: Improving Text-Guided Video Inpainting for Better Consistency, Controllability and Compatibility. *ArXiv* abs/2403.12035 (2024). https://api.semanticscholar.org/CorpusID:268532447

A Implementation Details

A.1 Models

Text-to-Video Model. We use a publicly available CogVideoX-5B [Hong et al. 2022; Yang et al. 2024] text-to-video model, which can generate videos with up to 480x720 pixel resolution, 6 seconds in length, 49 frames at 8 fps. This model is a transformer-based model that processes both text and video modalities together.

Segmentation Model. To segment the prominent objects in the video and the newly generated content, we utilize EVF-SAM2 [Zhang et al. 2024] - a text-based video segmentation model based on SAM2 [Ravi et al. 2024].

Visual Language Model.. Our vision-language model of choice is GPT-4o [Achiam et al. 2023], which we use through the official OpenAI API.

A.2 Keys and Values Extraction

Following [Tumanyan et al. 2023; Yatim et al. 2023], to obtain T2V diffusion model intermediate latents, we use DDIM inversion (applying DDIM sampling in reverse order) on the input video, using 250 forward steps, with an empty string as text prompt. During the forward pass in our method, the intermediate latents are used for the extraction of keys and values.

A.3 Latent Mask Extraction

As discussed in Sec. 4.3, we iteratively update the residual latent $\boldsymbol{x}_{res}$ in the regions where the new content appears. This requires calculating the mask of the new content in the latent space. To do this, we first apply the segmentation model [Zhang et al. 2024] to the current output of SDEdit and get the mask of the new content in RGB space. However, the VAE in the T2V diffusion model involves both spatial and temporal downsampling, making it challenging to directly map RGB pixels to their corresponding latent regions. To address this, we encode the RGB masks through the VAE-Encoder and apply clustering to partition the resulting latents into two groups, effectively producing downsampled masks that align with the latent space representation.

A.4 Runtime

Our method's two most computationally intensive parts are - DDIM inversion, which takes ~15 minutes, and iterative updates of the edit residual, which takes ~20 minutes. Importantly, DDIM inversion needs to be performed only once per video and can support multiple subsequent edits, making the process more efficient when applying various modifications to the same video content.

B Baselines comparison details

For LORA fine-tuning baseline, we use the following default hyperparameters: Adam optimizer [Kingma and Ba 2014], $1e-4$ learning rank, LORA rank 128, 800 fine-tuning steps. For comparison with GEN-3 [R Team, Runway [n. d.]], we utilize the Gen-3 Alpha model via the publicly accessible web-based API, setting the "Structure Transformation" hyperparameter to 5.

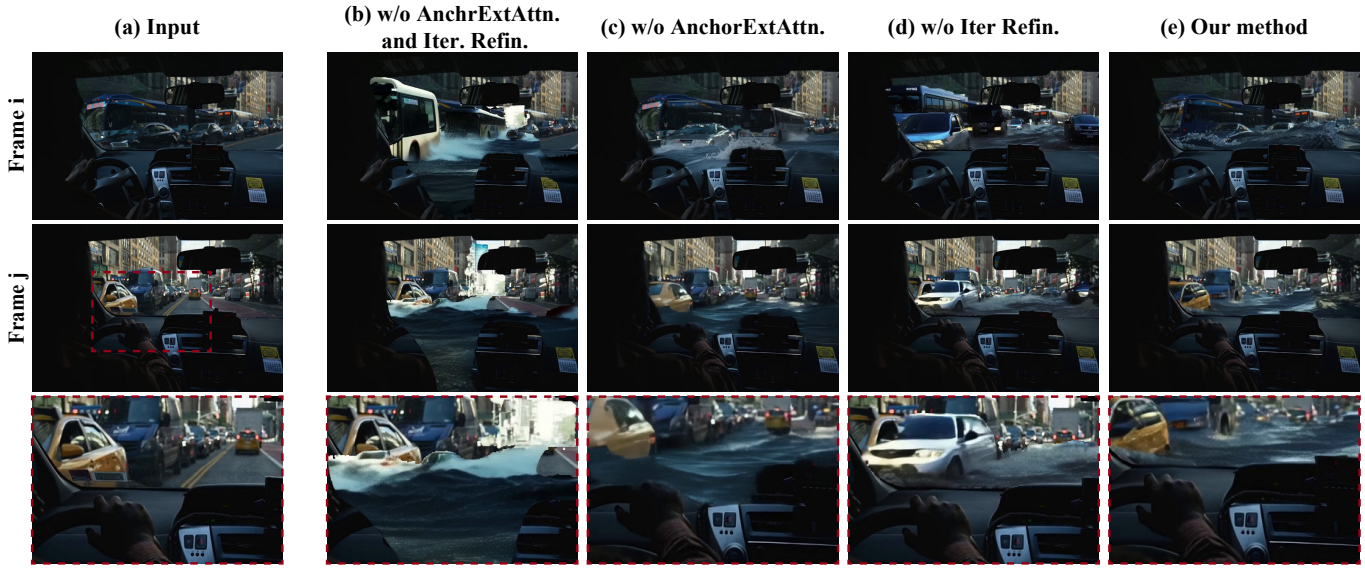

Fig. 9. Additional examples for Ablations.

C Additional comparisons

We perform an additional qualitative comparison to MagicVFX [Guo et al. 2024]. As can be seen in Fig. 10, MagicVFX struggles to remain faithful to the original scene and has lower visual quality compared to our method.

D VLM Prompting

While the model gives an accurate, descriptive source scene caption, in some cases, we observed that it fails to give captions suitable for compositing VFX with the scene. To overcome this, we ask the model to imagine a conversation with a visual effects (VFX) artist to obtain a caption that would describe the composited scene correctly. In this conversation, GPT-4o will "consult" with a VFX artist about how the new content should be integrated into the scene. Based on their discussion, it will be asked to provide a caption that describes how the added content fits into the scene. This results in text prompts that encourage the generated output video to include a natural interaction between the new content and the original environment. In this prompt, we also ask the VLM to provide a list of prominent foreground objects in the original video: O_{orig} and the object that will be added according to the edit prompt: O_{edit}. The full prompt for the VLM is shown in Figure 12.

In addition, as discussed in Sec. 5.2 we utilize the VLM for interpretable quality assessment. The full set of instructions for the VLM can be seen in Fig. 13.

Fig. 10. Comparison to MagicVFX. The result of MagicVFX the output differs significantly from the original video.

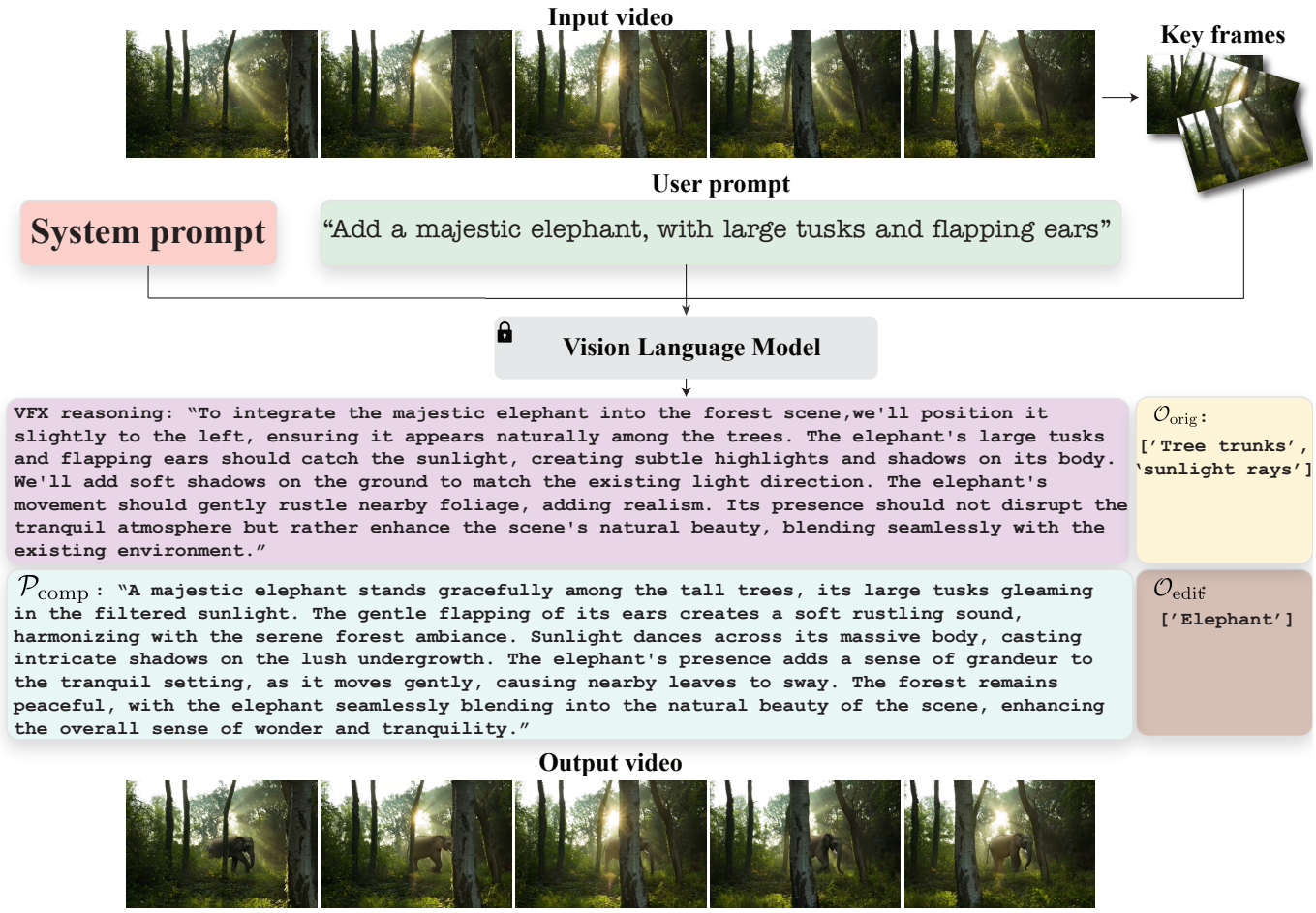

Fig. 11. Output example for protocol

You will receive a few images of the source scene and a description of new content to be added to the scene. It is possible that you will receive a source prompt as well.
Your task is to provide two captions based on the following steps:
Source Scene Caption:
Note: If a source scene prompt is provided, use it as is!
Provide a detailed description of the source scene without considering the added content.
Focus on the existing objects, environment, and actions in the scene.
Ensure the description maintains the original mood and setting.
VFX Conversation:
Imagine a conversation with a Visual Effects (VFX) artist about how the new content should be integrated into the scene.
Remember, the new content can be objects or multiple objects or effect or really anything the user provides. so be clear to explain this to the VFX artist.
The new content should interact naturally with the environment (e.g., shadows, lighting, or physical elements like grass, water, or other objects) but without altering the dynamics of the source scene.
The object must fit into the scene without affecting the original characters' behavior or actions.
The interaction between new content and foreground object must be included (e.g. object A is interacting with object B). in terms of dynamics and motion as well.
Describe how the object interacts and how it blends into the scene.
Composited Scene Caption:
Based on the conversation with the VFX artist, provide a caption that describes how the added content fits into the scene.
The caption must reflect natural interaction between the new content and the environment (e.g., lighting, shadows, physical effects), while ensuring the original dynamics remain unchanged.
The content should be aware of the surroundings, but the behavior, and flow of the original scene should remain consistent.
The overall atmosphere might change of course due to this addition to scene.
Output format* - a dictionary with keys: "source_scene_caption", "vfx_conversation", and "composited_scene_caption".
- **source_scene_caption**: source_scene_caption will be - A detailed caption of the source scene. If provided, use the given caption.
- **vfx_conversation**: A simulated conversation about how the new content should be integrated into the scene.
- **composited_scene_caption**: will be - A detailed caption of the composited scene, integrating the new content.
Note:The composited_scene_caption and source_scene caption must each have between 90-95 words. Extra words will be ignored.
Note:The vfx_conversation could be as long as required in order to succeed.
Note: Don't start the composited_scene_caption with - "The scene now." or "Added to the scene" "Scene has transformed",
the composited_scene_caption should be understandable to anyone that does not have access to the source_scene_caption.
And you should not simply concatenate between the source and composition.
You should have an entirely new caption that describes the essence of the integrated scene with both the source content and new content.
Don't use anything similar to "now the scene"

Fig. 12. VLM instructions used for generating the textual descriptions.

You are a helpful assistant that pays attention to context and estimates the perceptual quality of provided videos, specifically for the task of integrating new content into a given video.

I would like you to help me estimate the quality of an edited videos based on the original frames along with text descriptions.

You will be shown four grids. Each grid will be of the following type: left column will contain three frames from the original video.
The next 2 columns will each contain three frames from different video editing methods. Above each column there will be a caption (original video, 1, 2, ...).
Each method's task is to integrate the new content into the source video according to the edit prompt.

The prompt describing the original video is "{original_prompt}". The edit prompt for all of the methods is "{edit_prompt}".
Now, please conduct a perceptual quality comparison in terms of 1) alignment with the edit prompt; 2) visual quality, 3) new content harmonization and 4) dynamics

For each method provide a score from 0 to 1 for each of the five criteria with higher scores indicating better results.
Your response must include a concise description regarding the perceptual quality of each method and a score to summarize quality for each criterion while well aligning with the given description.
1) When assessing the alignment with the edit prompt, consider how well the method follows the edit prompt and if the frames contain the desired content.
If the method fails to follow the edit prompt, the score should be low.
2) For visual quality, consider how realistic the frames look - are there any visual artifacts, are the lighting and colors realistic, are the objects in the image recognizable.
3) For content harmonization - how well the content is harmonized with the scene, are the proportions of the added content correct, is the depth
and perspective of the added content consistent with the scene. Is placement of the added object physically realistic - does it look like it belongs in the scene or does it look out of place.
Are the occlusions of the added content consistent with the scene.
4) For dynamics assessment - how realistically the added object is moving relatively to the scene. Is its motion aligned with the camera motion of the original video?
If the object, for example floats unrealistically or flickers, the score should be low.

Fig. 13. VLM evaluation protocol